



Configuring IBM Network Media Translation

This chapter describes how to configure the Cisco IOS software for IBM network media translation with Synchronous Logical Data Link Control (SDLLC) or Qualified Logical Link Control (QLLC). For a complete description of the SDLLC and QLLC commands in this chapter, refer to the “IBM Network Media Translation Commands” chapter of the *Cisco IOS Bridging and IBM Networking Command Reference, Volume 1*. To locate documentation of other commands that appear in this chapter, use the command reference master index or search online.

This chapter contains the following sections:

- IBM Network Media Translation Overview, page 417
- SDLLC Configuration Task List, page 424
- Monitoring and Maintaining SDLLC Media Translation, page 431
- QLLC Conversion Configuration Task List, page 431
- Monitoring and Maintaining QLLC Conversion, page 436
- SDLLC Configuration Examples, page 436
- QLLC Conversion Configuration Examples, page 443
- NCP and VTAM Sysgen Parameters, page 447

IBM Network Media Translation Overview

The Cisco IOS software includes the following media translation features that enable network communications across heterogeneous media:

- SDLLC media translation enables a device on a Token Ring to communicate with a device on a serial link.
- QLLC conversion enables an IBM device to communicate with an X.25 network without having to install the X.25 software on local IBM equipment.

SDLLC is a Cisco Systems proprietary software feature that enables a device on a Token Ring to communicate with a device on a serial link by translating between LLC2 and SDLC at the link layer.

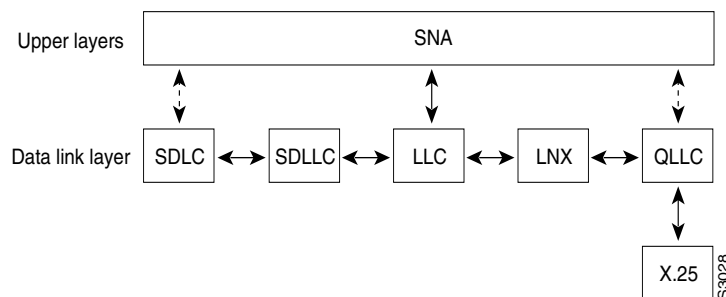
SNA uses SDLC and LLC2 as link layer protocols to provide a reliable connection. The translation function between these industry-standard protocols takes place in the proprietary Cisco software.

This section contains a brief overview of IBM Network Media Translation which is described in the following topics:

- SDLLC Media Translation Features, page 418
- QLLC Conversion, page 420
- Cisco's Implementation of QLLC Conversion, page 421
- Comparing QLLC Conversion to SDLLC, page 423
- Other Implementation Considerations, page 424

Figure 169 illustrates how SDLLC provides data link layer support for SNA communication.

Figure 169 SNA Data Link Layer Support



SDLLC Media Translation Features

The SDLLC feature allows a PU 4, PU 2.1, or PU 2 to communicate with a PU 2 SDLC device as follows:

- SDLLC with direct connection—A 37x5 FEP on a Token Ring and the 3x74 cluster controller connected to a serial line are each connected to an interface on the same router configured with SDLLC.
- SDLLC with RSRB—A 37x5 FEP on a Token Ring and a 3x74 cluster controller connected to a serial line are connected to different routers. Only the device to which the 3x74 is connected is configured with SDLLC. The routers communicate via RSRB using direct encapsulation, RSRB over an FST connection, or RSRB over a TCP connection.
- SDLLC with RSRB and local acknowledgment—A 37x5 FEP on a Token Ring and a 3x74 cluster controller connected to a serial line are connected to different routers. Only the device to which the 3x74 is connected is configured with SDLLC. The routers communicate via RSRB over a TCP connection that has local acknowledgment enabled.

In all these topologies, each IBM end node (the FEP and cluster controller) has no indication that its counterpart is connected to a different medium running a different protocol. The 37x5 FEP responds as if the 3x74 cluster controller were communicating over a Token Ring, whereas the 3x74 responds as though the 37x5 FEP were communicating over a serial line. That is, the SDLLC software makes translation between the two media transparent to the end nodes.

Virtual Token Ring Concept

Central to Cisco's SDLLC feature is the concept of a virtual Token Ring device residing on a virtual Token Ring. Because the Token Ring device expects the node with which it is communicating also to be on a Token Ring, each SDLLC device on a serial line must be assigned an SDLLC virtual Token Ring address (SDLLC VTRA). Like real Token Ring addresses, SDLLC VTRAs must be unique across the network.

In addition to the SDLLC VTRA, an SDLLC virtual ring number must be assigned to each SDLLC device on a serial line. (The SDLLC virtual ring number differs from the virtual ring group numbers that are used to configure RSRB and multiport bridging.)

As part of its virtual telecommunications access method (VTAM) configuration, the IBM node on the Token Ring has knowledge of the SDLLC VTRA of the serial device with which it communicates. The SDLC VTRA and the SDLLC virtual ring number are a part of the SDLLC configuration for the router's serial interface. When the Token Ring host sends out explorer packets with the SDLLC VTRA as the destination address in the MAC headers, the router configured with that SDLLC VTRA intercepts the frame, fills in the SDLLC virtual ring number address and the bridge number in the RIF, then sends the response back to the Token Ring host. A route is then established between the Token Ring host and the router. After the Cisco IOS software performs the appropriate frame conversion, the system uses this route to forward frames to the serial device.

Resolving Differences in LLC2 and SDLC Frame Size

IBM nodes on Token Ring media normally use frame sizes greater than 1 KB, whereas the IBM nodes on serial lines normally limit frame sizes to 265 or 521 bytes. To reduce traffic on backbone networks and provide better performance, Token Ring nodes should send frames that are as large as possible. As part of the SDLLC configuration on the serial interface, the largest frame size the two media can support should be selected. The Cisco IOS software can fragment the frames it receives from the Token Ring device before forwarding them to the SDLC device, but it does not assemble the frames it receives from the serial device before forwarding them to the Token Ring device.

Maintaining a Dynamic RIF Cache

SDLLC maintains a dynamic RIF cache and caches the entire RIF; that is, the RIF from the source station to destination station. The cached entry is based on the best path at the time the session begins. SDLLC uses the RIF cache to maintain the LLC2 session between the router and the host FEP. SDLLC does not age these RIF entries. Instead, SDLLC places an entry in the RIF cache for a session when the session begins and flushes the cache when the session terminates. You cannot flush these RIFs because if you flush the RIF entries randomly, the Cisco IOS software cannot maintain the LLC2 session to the host FEP.

Other Considerations

The following are additional facts regarding SDLC and SDLLC:

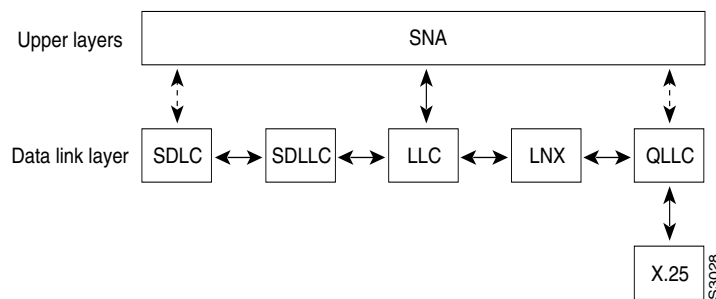
- As part of Cisco's SDLC implementation, only modulus 8 Normal Response Mode (NRM) sessions are maintained for the SDLC session.
- SDLC sessions are always locally acknowledged. LLC2 sessions can be optionally configured for local acknowledgment.
- SDLLC does not apply to SNA subarea networks, such as 37x5 FEP-to 37x5 FEP communication.

- Parameters such as the maximum number of information frames (I-frames) outstanding before acknowledgment, frequency of polls, and response time to poll frames can be modified per interface. If local acknowledgment is not enabled, these parameters are modified on the SDLC interface. If local acknowledgment is enabled, these parameters are modified on the Token Ring interface.
- Local acknowledgment only applies when the remote peer is defined for RSRB using IP encapsulation over a TCP connection. If no local acknowledgment is used, the remote peer can be defined for RSRB using direct encapsulation, RSRB using IP encapsulation over an FST connection, or RSRB using IP encapsulation over a TCP connection.

QLLC Conversion

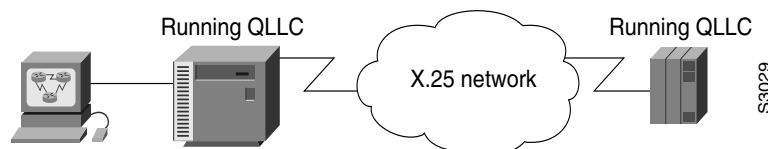
Qualified Logical Link Control (QLLC) is a data link protocol defined by IBM that allows SNA data to be transported across X.25 networks. (Although IBM has defined other protocols for transporting SNA traffic over an X.25 network, QLLC is the most widely used.) Figure 170 illustrates how QLLC conversion provides data link layer support for SNA communication.

Figure 170 SNA Data Link Layer Support



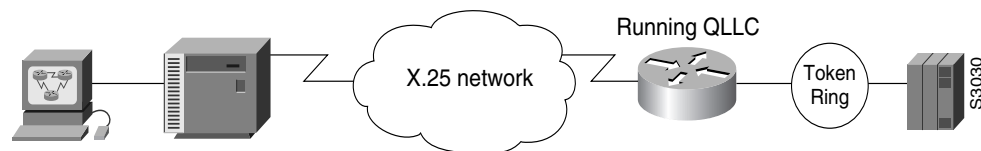
As shown in Figure 171, any devices in the SNA communication path that use X.25, whether end systems or intermediate systems, require a QLLC implementation.

Figure 171 SNA Devices Running QLLC



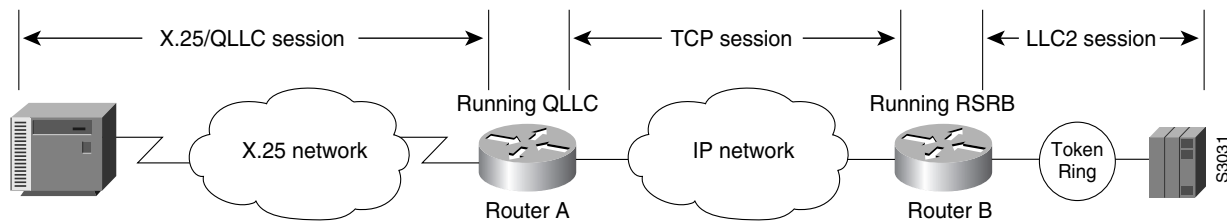
As shown in Figure 172, the QLLC conversion feature eliminates the need to install the X.25 software on local IBM equipment. A device attached locally to a Token Ring network can communicate through a router running the QLLC Conversion feature with a remote device attached to an X.25 network using QLLC. Typically, the locally attached device is an FEP, an AS 400, or a PS/2, and the remote device is a terminal controller or a PS/2. In this case, only the remote device needs an X.25 interface and the FEP can communicate with the terminal controller as if it were directly attached via a Token Ring network.

Figure 172 Router Running QLLC Conversion Feature



More elaborate configurations are possible. The router that implements QLLC conversion need not be on the same Token Ring network as the FEP. As shown in Figure 173, QLLC/LLC2 conversion is possible even when an intermediate IP WAN exists between the router connected to the X.25 network and the router connected to the Token Ring.

Figure 173 QLLC Conversion Running on a Router with an Intermediate IP Network

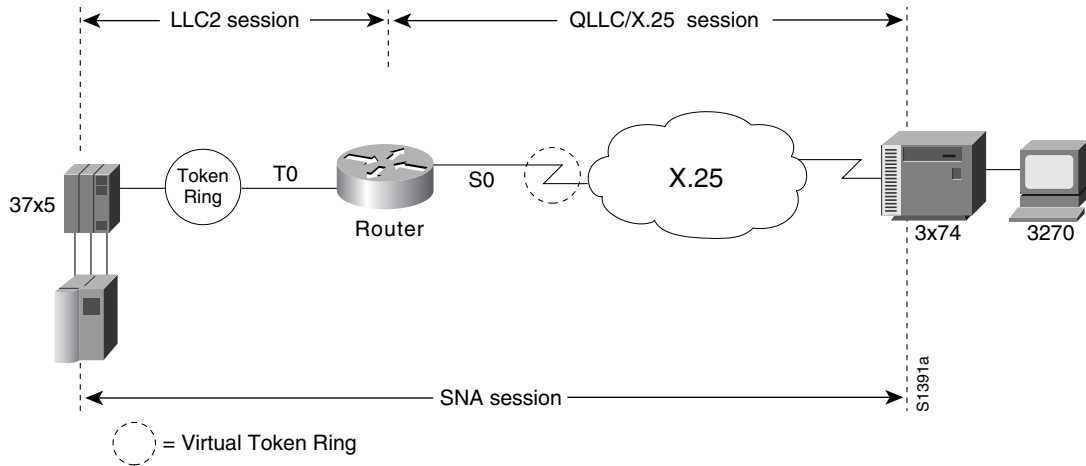


Cisco's Implementation of QLLC Conversion

SNA uses QLLC and X.25 as link layer protocols to provide a reliable connection. QLLC itself processes QLLC control packets. In a Token Ring environment, SNA uses LLC to provide a reliable connection. The LAN-to-X.25 (LNX) software provides a QLLC conversion function to translate between LLC and QLLC.

Figure 174 shows the simplest QLLC conversion topology: a single Token Ring device (for example, a 37x5 FEP) communicates with a single remote X.25 device (in this case a 3x74 cluster controller). In this example, a router connects the Token Ring network to the X.25 network.

Figure 174 QLLC Conversion Between a Single 37x5 and a Single 3x74

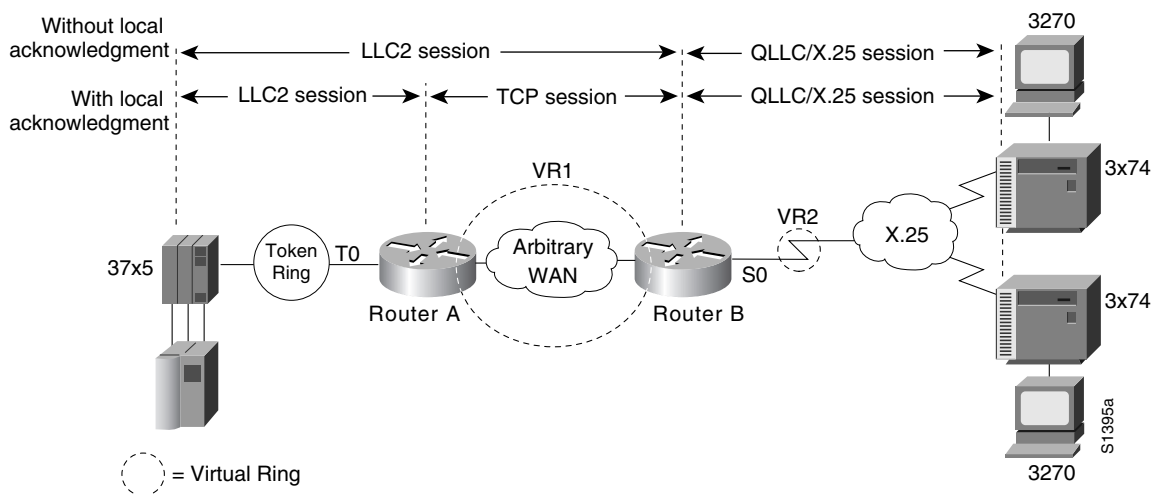


In Figure 174, each IBM end node has no indication that its counterpart is connected to a different medium running a different protocol. The 37x5 FEP responds as if the 3x74 cluster controller were communicating over a Token Ring, whereas the 3x74 responds as though the 37x5 FEP were communicating over an X.25 network. This is accomplished by configuring the router's X.25 interface as a virtual Token Ring, so that the X.25 virtual circuit appears to the Token Ring device (and to the router itself) as if it were a Token Ring to which the remote X.25 device is attached.

Also in this figure, the LLC2 connection extends from the 37x5 FEP across the Token Ring network to the router. The QLLC/X.25 session extends from the router across the X.25 network to the 3x74 cluster controller. Only the SNA session extends across the Token Ring and X.25 networks to provide an end-to-end connection from the 37x5 FEP to the 3x74 cluster controller.

As Figure 175 shows, a router need not directly connect the two IBM end nodes; instead, some type of backbone WAN can connect them. Here, RSRB transports packets between Router A and Router B, while Router B performs all conversion between the LLC2 and X.25 protocols. Only the router attached to the serial line (Router B) needs to be configured for QLLC conversion. Both Router A and Router B are configured for normal RSRB.

Figure 175 QLLC Conversion Between a Single 37x5 and Multiple 3x74s Across an Arbitrary WAN



How communication sessions are established over the communication link varies depending on whether or not LLC2 local acknowledgment has been configured on Router A's Token Ring interface. In both cases, the SNA session extends end-to-end and the QLLC/X.25 session extends from Router B to the 3x74 cluster controller. If LLC2 local acknowledgment has not been configured, the LLC2 session extends from the 37x5 FEP across the Token Ring network and the arbitrary WAN to Router B. In contrast, when LLC2 local acknowledgment has been configured, the LLC2 session extends from the 37x5 FEP Router A, where it is locally terminated. A TCP session is then used across the arbitrary WAN to Router B.

Comparing QLLC Conversion to SDLLC

Although the procedures you use to configure QLLC are similar to those used to configure SDLLC, there are structural and philosophical differences between the point-to-point links that SDLC uses and the multiplexed virtual circuits that X.25 uses.

The most significant structural difference between QLLC conversion and SDLLC is the addressing. To allow a device to use LLC2 to transfer data, both SDLLC and QLLC provide virtual MAC addresses. In SDLLC, the actual MAC address is built by combining the defined virtual MAC (whose last byte is 0x00) with the secondary address used on the SDLC link; in this way, SDLLC supports multidrop. In QLLC conversion, multidrop is meaningless, so the virtual MAC address represents just one session and is defined as part of the X.25 configuration. Because one physical X.25 interface can support many simultaneous connections for many different remote devices, you only need one physical link to the X.25 network. The different connections on different virtual circuits all use the same physical link.

The most significant difference between QLLC conversion and SDLLC is the fact that a typical SDLC/SDLLC operation uses a leased line. In SDLC, dial-up connections are possible, but the maximum data rate is limited. In QLLC, both switched virtual circuits (SVCs) and permanent virtual circuits (PVCs) are available, but the favored use is SVC. While the router maintains a permanent connection to the X.25 network, a remote device can use each SVC for some bounded period of time and then relinquish it for use by another device. Using a PVC is very much like using a leased line.

Table 7 shows how the QLLC commands correspond to the SDLLC commands.

Table 7 QLLC and SDLLC Command Comparison

QLLC Command	Analogous SDLLC Command
<code>qllc largest-packet</code>	<code>sdllc ring-largest-frame, sdllc sdlc-largest-frame</code>
<code>qllc partner</code>	<code>sdllc partner</code>
<code>qllc sap</code>	<code>sdllc sap</code>
<code>qllc srb, x25 map qllc, x25 pvc qllc</code>	<code>sdllc traddr</code>
<code>qllc xid</code>	<code>sdllc xid</code>
<code>source-bridge qllc-local-ack</code>	<code>source-bridge sdllc-local-ack</code>

Other Implementation Considerations

Consider the following when implementing QLLC conversion:

- To use the QLLC conversion feature, a router must have a physical link to an X.25 public data network (PDN). It must also have an SRB/RSRB path to an IBM FEP. This link could be a Token Ring or Ethernet interface, or even FDDI, if RSRB is being used.
- QLLC conversion can run on any router with at least one serial interface configured for X.25 communication and at least one other interface configured for SRB or RSRB.
- QLLC conversion security depends upon access control in SRB/RSRB and X.25 and upon XID validation.

You can configure DLSw+ for QLLC connectivity, which enables the following scenarios:

- Remote LAN-attached devices (physical units) or SDLC-attached devices can access an FEP or an AS/400 over an X.25 network.
- Remote X.25-attached SNA devices can access an FEP or an AS/400 over a Token Ring or over SDLC.

For information on configuring DLSw+ for QLLC conversion, refer to the “Configuring DLSw+” chapter.

You can configure DSPUs for QLLC. For more information on this configuration, refer to the “Configuring DSPU and SNA Service Point” chapter.

SDLLC Configuration Task List

To configure SDLLC, perform the tasks in the following sections:

- Configuring SDLLC with Direct Connection, page 425
- Configuring SDLLC with RSRB, page 426
- Configuring SDLLC with RSRB and Local Acknowledgment, page 428
- Configuring SDLLC with Ethernet and Translational Bridging, page 428
- Customizing SDLLC Media Translation, page 429

**Note**

Because data-link switching plus (DLSw+) contains its own media conversion, SDLLC is not required when using DLSw+.

For more information on configuring SDLLC and QLLC, see the following sections:

- QLLC Conversion Configuration Task List, page 431
- SDLLC Configuration Examples, page 436
- QLLC Conversion Configuration Examples, page 443

Configuring SDLLC with Direct Connection

In the SDLLC configuration with direct connection, a 37x5 front-end processor (FEP) on a Token Ring and a 3x74 cluster controller connected to a serial line are each connected to an interface on the same router configured with SDLLC. In this configuration, the Logical Link Control, type 2 (LLC2) session extends from the 37x5 FEP across the Token Ring to the router. The SDLLC session extends from the router across the serial line to the 3x74 cluster controller. The Systems Network Architecture (SNA) session extends across the Token Ring and the serial line to provide an end-to-end connection. The router is configured with source-route bridging (SRB).

To configure SDLLC with direct connection, you must perform the tasks in the following sections:

- Enabling SDLLC Media Translation, page 425
- Associating a SAP Value, page 425
- Specifying the XID Value, page 426
- Initiating a Connection to the Token Ring Host, page 426

For an example of how to configure SDLLC with direct connection, see the “SDLLC with Direct Connection Example” section on page 437.

Enabling SDLLC Media Translation

The interfaces you will configure for SDLLC media translation are the serial interfaces that connect to the serial lines linking the remote Synchronous Data Link Control (SDLC) devices. To configure them, use the following command in interface configuration mode:

Command	Purpose
<code>sdllc traddr xxxx.xxxx.xx00 lr bn tr</code>	Enables SDLLC media translation on a serial interface.

Associating a SAP Value

You can associate a Service Access Point (SAP) value by using the following command in interface configuration mode:

Command	Purpose
<code>sdllc sap sdlc-address ssap dsap</code>	Associates a SAP value.

Specifying the XID Value

The XID value you define in the Cisco IOS software must match that of the IDBLK and IDNUM system generation parameters defined in VTAM of the Token Ring host to which the SDLC device will be communicating. To define XID, use the following command in interface configuration mode:

Command	Purpose
<code>sdllc xid address xxxxxxxx</code>	Specifies the XID value appropriate for the SDLC station to match VTAM values.

Initiating a Connection to the Token Ring Host

The Token Ring host is always kept in a state ready to accept a connection from the remote serial device. The remote serial device is responsible for initiating connections. The advantage of this scheme is that the serial device can communicate with the Token Ring host whenever it chooses without requiring personnel to be on the host site.

The Cisco IOS software actually initiates the connection on behalf of the serial device. To initiate connections, both the Media Access Control (MAC) address of the Token Ring host and the SDLC line address are required. You must configure the Cisco IOS software to define the Token Ring host as the partner of the serial device. To do so, use the following command in interface configuration mode:

Command	Purpose
<code>sdllc partner mac-address sdlc-address</code>	Enables connections for SDLLC.

Configuring SDLLC with RSRB

A router need not directly connect the two IBM end nodes: a 37x5 FEP on a Token Ring and a 3x74 cluster controller connected to a serial line can be connected to different routers. However, the router to which the 3x74 is connected must be configured with SDLLC. They communicate via remote source-route bridging (RSRB) using direct encapsulation, RSRB over an FST connection, or RSRB over a TCP connection. RSRB transports packets between Router A and Router B, while Router B performs all conversion between the LLC2 and SDLC protocols by means of the SDLLC software.

To configure the router for SDLLC with RSRB, you must perform all the tasks in the “Configuring SDLLC with Direct Connection” section on page 425. In addition, you must perform one of the sets of tasks in the following sections:

- Configuring RSRB Using Direct Encapsulation, page 427
- Configuring RSRB over an FST Connection, page 427
- Configuring RSRB over a TCP Connection, page 427

For more information about configuring RSRB, see the chapter “Configuring Source-Route Bridging” in this publication and “Source-Route Bridging Commands” in the *Cisco IOS Bridging and IBM Networking Command Reference, Volume 1*.

**Note**

When you configure RSRB, you must include a **source-bridge remote peer** command on the router connected to the serial line and another **source-bridge remote peer** command on the one connected to the Token Ring. If you have more than one serial line connected to the same router, then you will have a **source-bridge remote peer** command for each interface in its configuration that will be using SDLLC with RSRB.

For an example of how to configure SDLLC with RSRB, see the “SDLLC with RSRB (Multiple 3x74s) Example” section on page 440.

Configuring RSRB Using Direct Encapsulation

To configure SDLLC with RSRB using direct encapsulation, use the following commands in global configuration mode:

	Command	Purpose
Step 1	source-bridge ring-group <i>ring-group</i> [<i>virtual-mac-address</i>]	Defines a ring group.
Step 2	source-bridge remote-peer <i>ring-group interface</i> <i>interface-name</i> [<i>mac-address</i>]	Defines a remote peer.

Configuring RSRB over an FST Connection

To configure SDLLC with RSRB over an FST connection, use the following commands in global configuration mode:

	Command	Purpose
Step 1	source-bridge ring-group <i>ring-group</i> [<i>virtual-mac-address</i>]	Defines a ring group.
Step 2	source-bridge fst-peername <i>local-interface-address</i>	For FST connection only, sets up an FST peer name.
Step 3	source-bridge remote-peer <i>ring-group fst ip-address</i>	Defines a remote peer.

Configuring RSRB over a TCP Connection

To configure SDLLC with RSRB over a TCP connection, use the following commands in global configuration mode:

	Command	Purpose
Step 1	source-bridge ring-group <i>ring-group</i> [<i>virtual-mac-address</i>]	Defines a ring group.
Step 2	source-bridge remote-peer <i>ring-group tcp ip-address</i>	Defines a remote peer.

Configuring SDLLC with RSRB and Local Acknowledgment

RSRB can be configured for only local acknowledgment with RSRB using IP encapsulation over a TCP connection. Configuring SDLLC local acknowledgment can reduce time-outs and keepalive traffic on the connection.

If LLC2 local acknowledgment is configured, it must be configured on the serial interface of the router on the 3x74 cluster controller side of the connection and on the Token Ring interface of the router on the 37x5 FEP side of the connection. Whether or not local acknowledgment is configured, the SNA session extends end-to-end and the SDLC session extends from the router configured with the serial interface to the 3x74 cluster controller. However, the LLC2 session extends from the 37x5 FEP to the router with the Token Ring interface configured. The LLC2 session is locally terminated at that router. A TCP session is then established across the WAN to a router on the 3x74 side of the connection.

To configure the Cisco IOS software for SDLLC with RSRB and local acknowledgment, you must perform all the tasks in the “Configuring SDLLC with Direct Connection” section on page 425. In addition, you must use the following commands in global configuration mode:

	Command	Purpose
Step 1	<code>source-bridge ring-group ring-group [virtual-mac-address]</code>	Defines a ring group.
Step 2	<code>source-bridge remote-peer ring-group tcp ip-address local-ack</code>	Defines a remote peer with the local acknowledgment feature.
Step 3	<code>source-bridge sdllc-local-ack</code>	Enables local acknowledgment for connections involving SDLLC media translation.

Local acknowledgment is not supported when the LLC2 device is attached to an Ethernet rather than to a Token Ring.

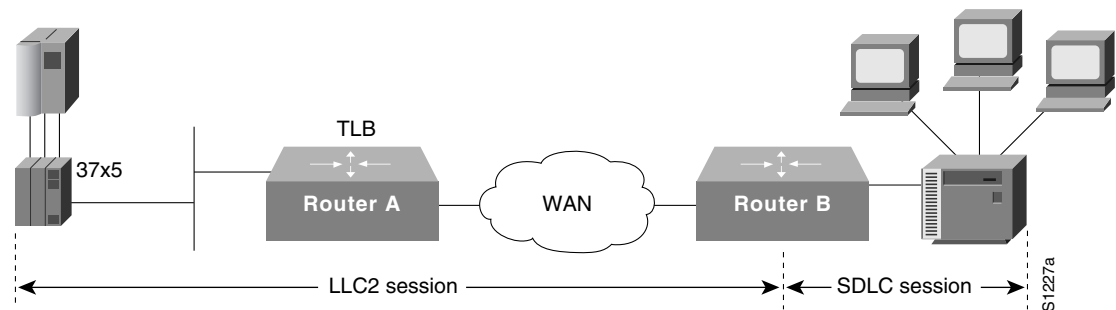
For an example of how to configure SDLLC with RSRB and local acknowledgment, see the “SDLLC with RSRB and Local Acknowledgment Example” section on page 441.

For more information about configuring RSRB and local acknowledgment, see the chapter “Configuring Source-Route Bridging” in this manual and “Source-Route Bridging Commands” in the *Cisco IOS Bridging and IBM Networking Command Reference, Volume I*.

Configuring SDLLC with Ethernet and Translational Bridging

SDLLC support over Ethernet combines translational bridging with Ethernet support of 37x5 FEP connections. Figure 176 shows SDLLC with Ethernet and translational bridging. The 3x75 FEP is attached to Router A through Ethernet. The same router is configured for translational bridging, which translates Ethernet packets into Token Ring packets and passes them across the WAN to Router B connected to the 3x74 cluster controller via a serial line. The LLC2 session terminates at Router B connected to the 3x74 cluster controller. In addition, Router B maintains an SDLC session from itself to the cluster controller.

Figure 176 SDLLC with Ethernet and Translational Bridging



Customizing SDLLC Media Translation

To increase performance on connections involving SDLLC media translation, perform the tasks in the following sections:

- Setting the Largest LLC2 I-Frame Size, page 429
- Setting the Largest SDLC I-Frame Size, page 430
- Increasing the SDLC Line Speed, page 430
- Other Customizing Considerations, page 430

Setting the Largest LLC2 I-Frame Size

Generally, the router and the LLC2 device with which it communicates should support the same maximum SDLC I-frame size. The larger this value, the better the line is used, thus increasing performance.

Faster screen updates to 3278-style terminals often result by configuring the Token Ring FEP to send as large an I-frame as possible and then allowing the Cisco IOS software to segment the frame into multiple SDLC I-frames.

After the Token Ring FEP has been configured to send the largest possible I-frame, it is best to configure the software to support the same maximum I-frame size. The default is 516 bytes. The maximum value the software can support is 8144 bytes.

To set the largest LLC2 I-frame size, use the following command in interface configuration mode:

Command	Purpose
<code>sdllc ring-largest-frame value</code>	Specifies the largest I-frame size that can be sent or received by the designated LLC2 primary station.

Setting the Largest SDLC I-Frame Size

Generally, the router and the SDLC device with which it communicates should support the same maximum SDLC I-frame size. The larger this value, the better the line is utilized, thus increasing performance.

After the SDLC device has been configured to send the largest possible I-frame, you must configure the Cisco IOS software to support the same maximum I-frame size. The default is 265 bytes. The maximum value the software can support must be less than the value of the LLC2 largest frame value defined when setting the largest LLC2 I-frame size.

To set the largest SDLC I-frame size, use the following command in interface configuration mode:

Command	Purpose
<code>sdllc sdlc-largest-frame address value</code>	Sets the largest I-frame size that can be sent or received by the designated SDLC station.

Increasing the SDLC Line Speed

You can increase the data transfer rate by increasing the SDLC line speed on the serial interface. If possible, increase the link speed of the 3x74 to 19.2 kbps on older units, or to 64 kbps on new units.

To increase the SDLC line speed, use the following command in interface configuration mode:

Command	Purpose
<code>clock rate bps</code>	Adjusts the clock rate on the serial interface of the SCI and MCI cards to an acceptable bit rate.

Other Customizing Considerations

In addition to adjusting the SDLLC parameters described in this section, you can improve performance on the connection by adjusting the LLC2 and SDLC parameters described in the chapter “Configuring LLC2 and SDLC Parameters.”

For IBM host configuration consider changing the default MAXOUT (window size) value. Widely used installation guides for IBM equipment show a MAXOUT value of 1 in the VTAM-switched major node for the IBM 3174 PU. Changing this value to 7 improves the performance, because VTAM can send seven frames before requiring an acknowledgment.

Monitoring and Maintaining SDLLC Media Translation

To monitor connections using SDLLC media translation, use the following monitoring commands in privileged EXEC mode:

	Command	Purpose
Step 1	<code>show interfaces</code>	Displays information about SDLC and LLC2 connections involving interfaces on which SDLLC media translation has been enabled.
Step 2	<code>show sdllc local-ack</code>	Displays the current state of any connections using local acknowledgment for LLC2 and SDLLC connections.
Step 3	<code>show llc2</code>	Displays information about LLC2 connections involving interfaces on which SDLLC media translation has been enabled.

In `show llc2` command output, look for the LLC2 connections that correspond to the MAC addresses you assigned to the SDLLC interfaces using the `sdllc traddr` command. For information about these commands, see the chapter “LLC2 and SDLC Commands” and “IBM Network Media Translation Commands” in the *Cisco IOS Bridging and IBM Networking Command Reference, Volume I*.

QLLC Conversion Configuration Task List

Perform the tasks in the following sections to configure QLLC conversion. The first task is required; all others are optional and depend on your specific needs.

- Enabling QLLC Conversion on a Serial Interface, page 431
- Customizing QLLC Conversion, page 434

See the end of this chapter for QLLC configuration examples.

Enabling QLLC Conversion on a Serial Interface

The interfaces you configure for QLLC conversion are the serial interfaces that connect to the X.25 network linking the remote devices with which you plan to communicate.

To enable QLLC conversion, you must perform the first of the following tasks. Perform the remaining tasks as appropriate.

- Enabling QLLC Conversion on the Appropriate Serial Interfaces, page 432
- Defining the XID Value Associated with an X.25 Device, page 433
- Enabling the Opening of a Connection to the Local Token Ring Device, page 433

Enabling QLLC Conversion on the Appropriate Serial Interfaces

You can enable QLLC conversion on a serial interface to support either a switched virtual circuit (SVC) or a permanent virtual circuit (PVC). The tasks you perform differ somewhat depending on the type of virtual circuit you plan to support on the interface. In either case, first verify that RSRB is enabled by using the following command in privileged EXEC mode:

Command	Purpose
<code>show configuration</code>	Ensures that RSRB is enabled on the interfaces.

In the sections for the appropriate serial interfaces of the **show configuration** display, look for one or more **source-bridge remote-peer** entries and a **source-bridge rn** entry. For more information about configuring a serial interface for RSRB, see the chapter “Configuring LLC2 and SDLC Parameters” in this publication.

To enable QLLC conversion to support an SVC, use the following commands in interface configuration mode:

	Command	Purpose
Step 1	<code>x25 map ql1c virtual-mac-addr x121-addr [x25-map-options]</code>	Maps a virtual Token Ring MAC address for the interface to its X.121 address.
Step 2	<code>ql1c srb virtual-mac-addr srn trn</code>	Enables the use of QLLC conversion on the interface.

To enable QLLC conversion to support a PVC, use the following commands in interface configuration mode:

	Command	Purpose
Step 1	<code>x25 pvc circuit ql1c x121-address [x25-map-options]</code>	Sets up a PVC for QLLC conversion.
Step 2	<code>ql1c srb virtual-mac-addr srn trn</code>	Enables the use of QLLC conversion on the interface.

To configure QLLC to accept a call from any remote X.25 device, use the following command in interface configuration mode:

Command	Purpose
<code>ql1c accept-all-calls</code>	Configures QLLC to accept a call from any remote X.25 device.

In a Token Ring or RSRB environment the LAN-attached devices initiate a connection by sending a null XID packet upstream. If the Cisco IOS software forwards this null XID to an X.25-attached FEP, the FEP responds as if it were connecting to an PU2.1 device, and breaks the connection when the PU 2.0 next sends an XID Format 0 Type 2. To resolve this situation and to enable the connection, use the following command in interface configuration mode:

Command	Purpose
<code>qllc npsi-poll virtual-mac-addr</code>	Enables connection between a PU 2.0 on the LAN side and a FEP running NPSI on the X.25 side.

The `qllc npsi-poll` command intercepts any null XID packet that the router receives on the LAN interface, and returns a null XID response to the downstream device. It continues to allow XID Format 3 and XID Format 0 packets through the X.25 device.

Defining the XID Value Associated with an X.25 Device

The exchange identification (XID) serves as a password to ensure that only those devices that should communicate with the Token Ring host have that privilege. If the XID is defined in NCP on the host, you must enable the Cisco IOS software to reply (on behalf of the X.25 device) to the Token Ring host's requests for an XID reply. Although the XID value is used to reply to XID requests received on the LLC2 side of the connection, you apply this command on the serial interface defined for X.25. This XID value must match that of IDBLK and IDNUM defined in the NCP.



Note

For most QLLC installations, you do not need to define the XID value. You only need to do so if the remote X.25 device is not configured to send its own XID. This is only possible for a device that is attached through a PVC, although most devices that are connected through X.25 send their own XIDs.

To define the XID value associated with an X.25 device, use the following command in interface configuration mode:

Command	Purpose
<code>qllc xid virtual-mac-addr xid</code>	Specifies the XID value appropriate for the X.25 device associated with the Token Ring interface.

Enabling the Opening of a Connection to the Local Token Ring Device

If you plan to use SVCs rather than PVCs, you must enable the Cisco IOS software to open a connection to the local Token Ring device on behalf of the remote X.25 device when an incoming call is received. When QLLC conversion is used over an SVC, the remote X.25 device typically initiates the X.25/QLLC session, and the software in turn initiates the LLC2 session.

To enable the software to open a connection to the local Token Ring device, use the following command in interface configuration mode:

Command	Purpose
<code>qllc partner virtual-mac-addr mac-addr</code>	Enables the software to open a connection to the local Token Ring device.

Customizing QLLC Conversion

To customize your configuration of QLLC conversion, you can perform one or more of the following tasks in the following sections:

- Enabling QLLC Local Acknowledgment for Remote Source-Route-Bridged Connections, page 434
- Specifying SAP Values Other Than the Default IBM SAP Values, page 435
- Specifying the Largest Packet That Can Be Sent or Received on the X.25 Interface, page 435

Enabling QLLC Local Acknowledgment for Remote Source-Route-Bridged Connections

Enable local acknowledgment when the round-trip time through the TCP/IP network is as large or larger than the LLC2 timeout period.

To enable QLLC local acknowledgment for RSRB connections, use the following global configuration command on the router connected to the X.25 interface and configure the remote peers for local acknowledgment:

Command	Purpose
<code>source-bridge qllc-local-ack</code>	Enables QLLC local acknowledgment for remote source-route-bridged connections.

If, for example, Router B with X.25 interface has the IP address *ip1*, and the remote peer (Router A) has the address *ip2*, and they use a virtual ring group *vrg*, then both routers use the following configuration commands:

```
source-bridge ring-group vrg
source-bridge remote-peer vrg tcp ip1 local-ack
source-bridge remote-peer vrg tcp ip2
```

The configuration for Router B is as follows:

```
source-bridge ring-group vrg
source-bridge remote-peer vrg tcp ip1
source-bridge remote-peer vrg tcp ip2 local-ack
```

This will not affect Router A.

Specifying SAP Values Other Than the Default IBM SAP Values

To use SAP values other than the default IBM SAP values, use the following command in interface configuration mode:

Command	Purpose
<code>qllc sap virtual-mac-addr ssap dsap</code>	Specifies a SAP value other than the default IBM SAP value.

Specifying the Largest Packet That Can Be Sent or Received on the X.25 Interface

There are two ways for a packet to become segmented:

- The X.25 software performs the segmentation and the other X.25 station re-assembles the packet.
- The QLLC conversion performs SNA header segmentation. In this case, QLLC does not re-assemble, but passes smaller SNA segments to the IBM end station.

If the QLLC software does not perform SNA segmentation, then the X.25 software must be capable of performing X.25 segmentation of the largest packet that it can receive from the LLC2 side. This packet can be several thousand bytes long, whereas the typical size for X.25 packets is 1024 bytes or less. (The default is 128, but that can be overridden with larger values.) The X.25 software, especially in the X.25 attached IBM end station, might not be able to reassemble a very large packet. In this situation, specifying the largest QLLC packet can be useful.

By default, the maximum SNA data unit size established for the virtual circuit is the maximum packet size that can be sent or received on the X.25 interface. If packets received on the LLC2 interface are larger than the largest value allowed on the X.25 connection, they can be segmented by the X.25 software before being sent on the X.25 interface. Moreover, there is no reassembly on receiving packets on the X.25 interface before sending them on the LLC2 interface. Thus, you might need to reconfigure the maximum packet size for the X.25 interface to match that for the LLC2 interface.

When the remote X.25 device has a limit on the maximum total length of recombined X.25 segments it will support, you must ensure the length is not exceeded. For example, a device whose maximum SNA packet size is limited to 265 bytes might not be able to handle a series of X.25 packets that it has to recombine to make a 4, 8, or 17 KB SNA packet, such as one often encounters in an LLC2 environment.

You cannot configure the X.25 interface with a larger packet size than the LLC2 interface.

To specify the largest packet that can be sent or received on the X.25 interface, use the following command in interface configuration mode:

Command	Purpose
<code>qllc largest-packet virtual-mac-address max-size</code>	Specifies the largest packet that can be sent or received on the X.25 interface.

Monitoring and Maintaining QLLC Conversion

To monitor connections using QLLC conversion, use the following commands in privileged EXEC mode:

	Command	Purpose
Step 1	<code>show interfaces serial <i>number</i></code>	Displays information about X.25 and LLC2 connections involving interfaces on which QLLC conversion has been enabled.
Step 2	<code>show qllc</code>	Displays the current state of any connections using QLLC local acknowledgment.
Step 3	<code>show llc2</code>	Displays information about LLC2 connections involving interfaces on which QLLC conversion has been enabled.

SDLLC Configuration Examples

The following sections provide SDLLC configuration examples:

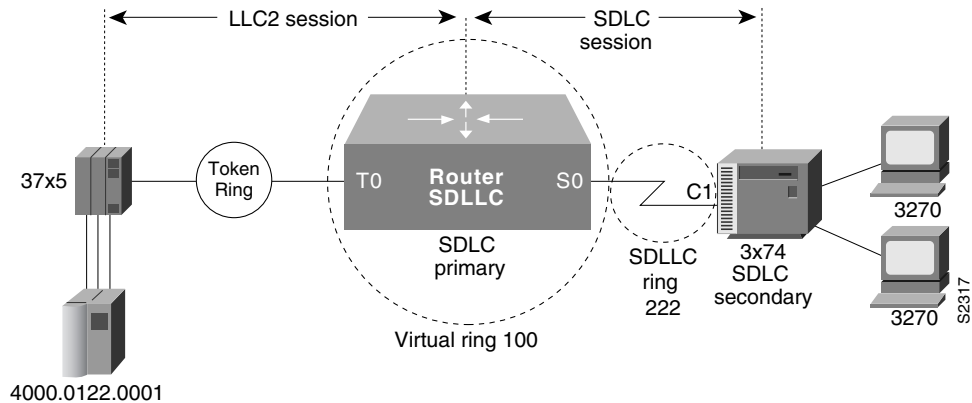
- SDLLC with Direct Connection Example, page 437
- SDLLC with Single Router Using RSRB Example, page 437
- SDLLC with RSRB (Single 3x74) Example, page 438
- SDLLC with RSRB (Multiple 3x74s) Example, page 440
- SDLLC with RSRB and Local Acknowledgment Example, page 441

In the “QLLC Conversion Configuration Examples” section on page 443, refer to the “NCP and VTAM Sysgen Parameters” section on page 447 for sample NCP definitions that the 37x5 FEP in these topologies could use and for sample VTAM definitions that the IBM host in these topologies could use to reflect the routers in the communication path.

SDLLC with Direct Connection Example

Figure 177 shows a router configuration when the router directly connects the Token Ring and the serial line. The Cisco IOS software is configured with SRB.

Figure 177 SDLLC Communication Between a 37x5 and a 3x74 Connected to the Same Router (Direct Connection)



A configuration file that enables direct connection follows:

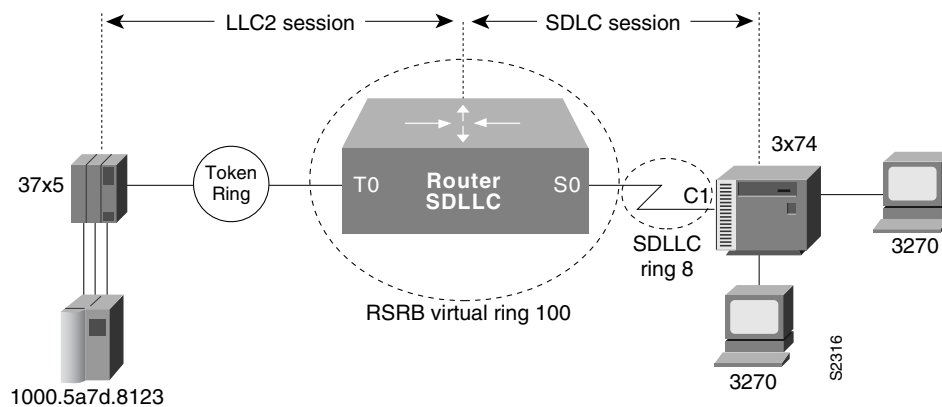
```
source-bridge ring-group 100
!
interface tokenring 0
 source-bridge 111 1 100
!
interface serial 0
 encapsulation sdhc-primary
 sdhc address c1
 sdllc traddr 0110.2222.3300 222 2 100
 sdllc partner 4000.0122.0001 c1
 sdllc xid c1 1720001
```

SDLLC with Single Router Using RSRB Example

Figure 178 shows a software configuration in which the router directly connects the Token Ring and the serial line, but uses RSRB to create a virtual ring 100. This configuration has the following characteristics:

- The FEP (37x5) sees C1 3x74 at MAC address 0110.2222.3300
- The RIF from the FEP to the devices would appear as:
ring 111—bridge 1—ring 100—bridge 1—ring 8

Figure 178 SDLLC with Single Router Using RSRB



The following sample configuration file is for SDLLC with a single router using RSRB:

```
source-bridge ring-group 100
source-bridge remote-peer 100 tcp 172.18.1.1
source-bridge remote-peer 100 tcp 172.18.2.2

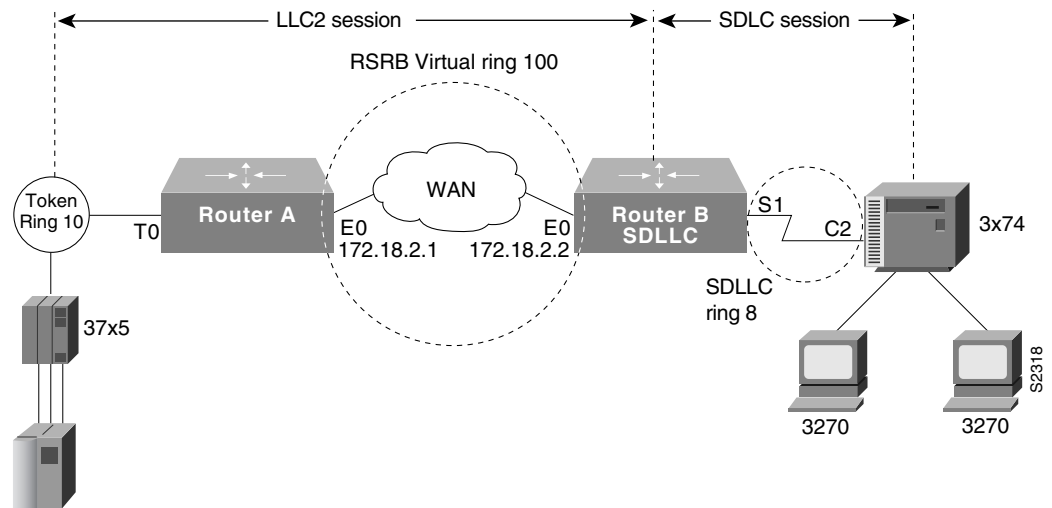
interface tokenring 0
ip address 172.18.2.2 255.255.255.0
source-bridge 111 1 100

interface serial 0
encapsulation sdhc-primary
sdhc address c1
sdllc traddr 0110.2222.3300 8 1 100
sdllc partner 1000.5a7d.8123 c1
sdllc xid c1 17200c1
```

SDLLC with RSRB (Single 3x74) Example

In Figure 179, SDLLC with RSRB connects a FEP (37x5) and a single 3x74 cluster controller. The host wants to communicate with a single 3174 that its FEP sees on a Token Ring. However, the 3x74 seen by the FEP is in fact SDLC device C1 connected by means of a serial link through a remote router.

Figure 179 SDLLC with RSRB with a Single 3x74



The configuration files for the network shown in Figure 179 follow.

Router A

```
source-bridge ring-group 100
source-bridge remote-peer 100 tcp 172.18.1.1
source-bridge remote-peer 100 tcp 172.18.2.2
!
interface tokenring 0
ip address 172.18.1.1 255.255.255.0
source-bridge 10 1 100
!
interface ethernet 0
ip address 172.18.2.1 255.255.255.0
```

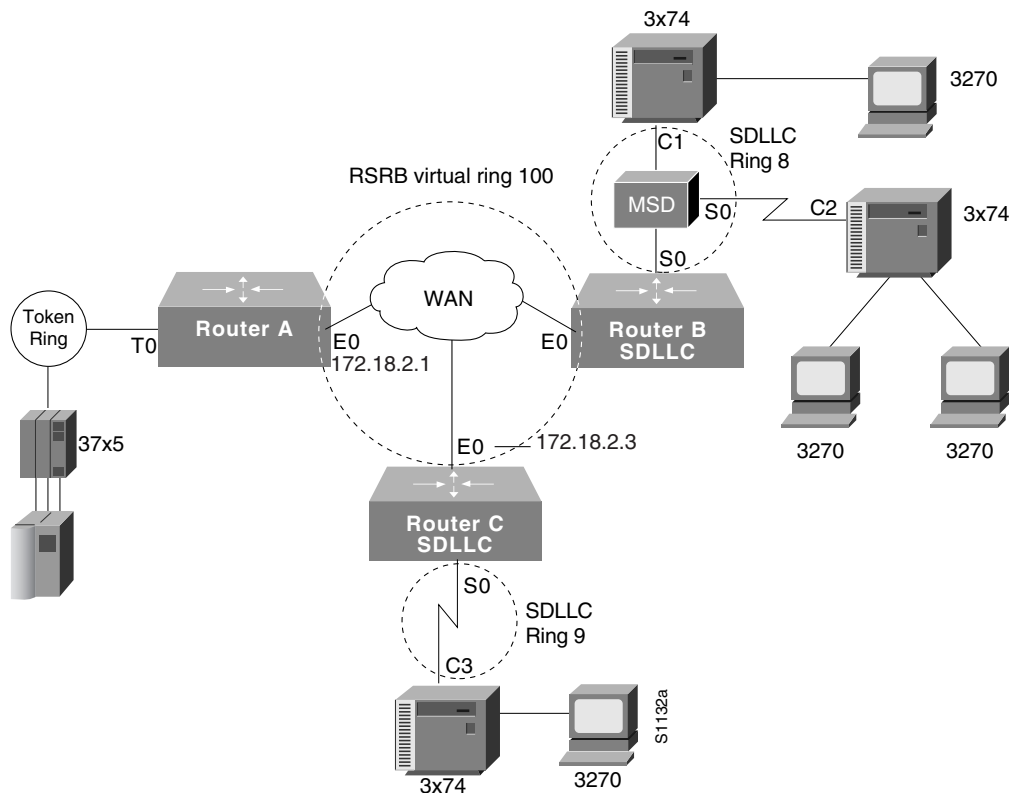
Router B

```
source-bridge ring-group 100
source-bridge remote-peer 100 tcp 172.18.1.1
source-bridge remote-peer 100 tcp 172.18.2.2
!
interface tokenring 0
ip address 172.18.2.2 255.255.255.0
source-bridge 1 1 100
!
interface serial 0
encapsulation sdhc-primary
sdhc address c1
sdllc traddr 0110.2222.3300 8 1 100
sdllc partner 1000.5a7d.8123 c1
sdllc xid c1 17200c1
```

SDLLC with RSRB (Multiple 3x74s) Example

In the setup shown in Figure 180, Router A needs no SDLLC configuration, Router B has the SDLLC configuration and supports multipoint on the SDLC link with a modem-sharing device, and Router C is also configured with SDLLC. For information about the NCP and VTAM system generation (sysgen) parameters that are used in this configuration, see the “NCP and VTAM Sysgen Parameters” section on page 447. (The notes in the sample configuration files refer to the “Notes” section within the “NCP and VTAM Sysgen Parameters” section on page 447.)

Figure 180 SDLLC with RSRB with Multiple 3x74s



The following configuration files describe the network shown in Figure 180.

Router A

```
source-bridge ring-group 100
source-bridge remote-peer 100 tcp 172.18.2.1
source-bridge remote-peer 100 tcp 172.18.2.2
source-bridge remote-peer 100 tcp 172.18.2.3
!
interface tokenring 0
ip address 172.18.1.1 255.255.255.0
source-bridge 10 1 100
!
interface ethernet 0
ip address 172.18.2.1 255.255.255.0
```

Router B

```
source-bridge ring-group 100
source-bridge remote-peer 100 tcp 11.108.2.1
source-bridge remote-peer 100 tcp 11.108.2.2
source-bridge remote-peer 100 tcp 11.108.2.3
!
interface ethernet 0
 ip address 172.18.2.2 255.255.255.0
!
interface serial 0
 encapsulation sdlc-primary
 sdlc address c1
 sdlc address c2
 sdllc traddr 0110.2222.3300 7 1 100
 sdllc partner 1000.5a7d.8123 c1
 sdllc partner 1000.5a7d.8123 c2
 sdllc xid c1 17200c1
 sdllc xid c2 17200c2
```

Router C

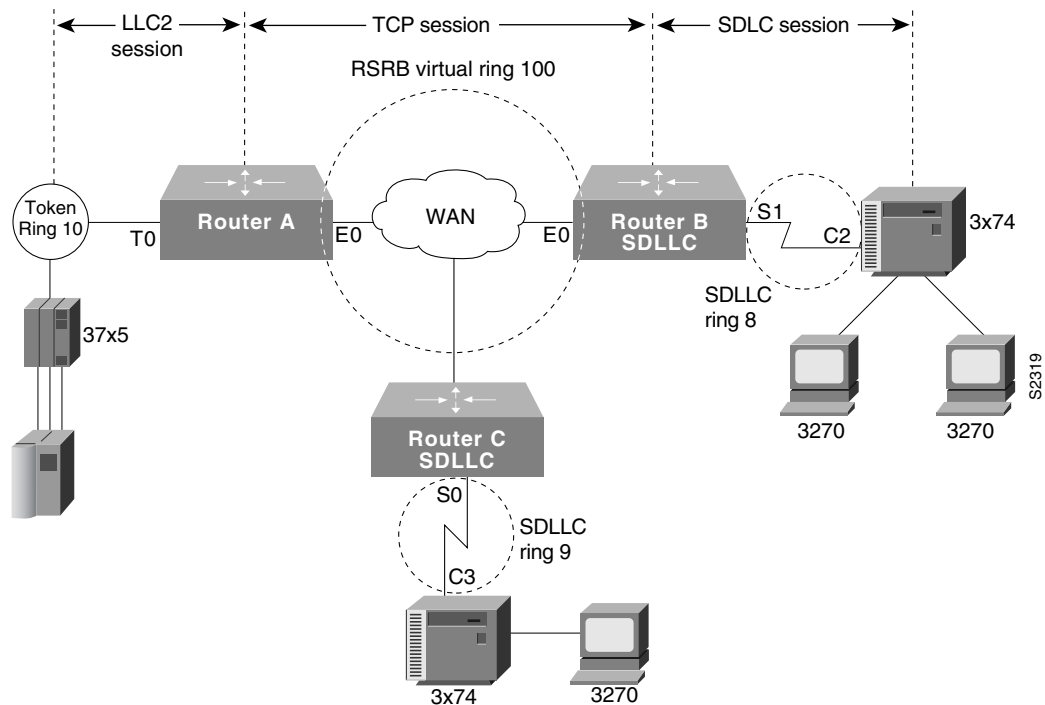
```
source-bridge ring-group 100
source-bridge remote-peer 100 tcp 172.18.2.1
source-bridge remote-peer 100 tcp 172.18.2.2
source-bridge remote-peer 100 tcp 172.18.2.3
!
interface ethernet 0
 ip address 172.18.2.3 255.255.255.0
!
interface serial 0
 encapsulation sdlc-primary
 sdlc address c3
 sdllc traddr 0110.2222.3300 9 1 100
 sdllc partner 1000.5a7d.8123 c3 MUST MATCH TIC LOCADD, NOTE 2
 sdllc xid c3 17200c3 MUST MATCH VTAM IDBLK/IDNUM, NOTE 4
```

SDLLC with RSRB and Local Acknowledgment Example

The configuration shown in Figure 181 enables local acknowledgment for Router B, which means that the LLC session terminates at Router A. However, the LLC2 session between Router A and Router C is not locally acknowledged and terminates at Router C.

For information about the NCP and VTAM system generation (sysgen) parameters that are used in this configuration, see the “NCP and VTAM Sysgen Parameters” section on page 447.

Figure 181 SDLLC with RSRB and Local Acknowledgment



The following sample configuration files describe the network shown in Figure 181. (The notes in the sample configuration files refer to the “Notes” section within the “NCP and VTAM Sysgen Parameters” section on page 447.)

Router A

```
source-bridge ring-group 100
source-bridge remote-peer 100 tcp 172.18.2.1
source-bridge remote-peer 100 tcp 172.18.2.2 local-ack
source-bridge remote-peer 100 tcp 172.18.2.3
!
interface tokenring 0
ip address 172.18.1.1 255.255.255.0
source-bridge 1 1 100
!
interface ethernet 0
ip address 172.18.2.1 255.255.255.0
```

Router B

```
source-bridge ring-group 100
source-bridge remote-peer 100 tcp 172.18.2.1 local-ack
source-bridge remote-peer 100 tcp 172.18.2.2
source-bridge remote-peer 100 tcp 172.18.2.3

source-bridge sdllc local-ack
!
interface ethernet 0
ip address 172.18.2.2 255.255.255.0
!
interface serial 0
encapsulation sdllc-primary
sdllc address c1
sdllc traddr 4000.3174.0b0d 7 1 100
```

```

sdllc partner 1000.5a7d.8123 c1
!Must match TIC LOCADD [See NOTE 2.]
sdllc xid c1 017200c1
!Must match VTAM IDBLK/IDNUM [See NOTE 4.]
!
interface serial 1
encapsulation sdhc-primary
sdhc address c2
sdllc traddr 0110.2222.3200 8 1 100
sdllc partner 1000.5a7d.8123 c2
!Must match TIC LOCADD [See NOTE 2.]
sdllc xid c2 017200c2
!Must match VTAM IDBLK/IDNUM [See NOTE 4.]

```

Router C

```

source-bridge ring-group 100
source-bridge remote-peer 100 tcp 172.18.2.1
source-bridge remote-peer 100 tcp 172.18.2.2
source-bridge remote-peer 100 tcp 172.18.2.3
!
interface ethernet 0
ip address 172.18.2.3 255.255.255.0
!
interface serial 0
encapsulation sdhc-primary
sdhc address c3
sdllc traddr 4000.3174.0c00 9 1 100
sdllc partner 1000.5a7d.8123 c3
Must match TIC LOCADD [See NOTE 2.]
sdllc xid c3 017200c3
!Must match VTAM IDBLK/IDNUM [See NOTE 4.]

```

QLLC Conversion Configuration Examples

The following sections provide QLLC conversion configuration examples and information:

- QLLC Conversion Between a Single 37x5 and a Single 3x74 Example, page 443
- QLLC Conversion Between a Single 37x5 and Multiple 3x74s Example, page 444
- QLLC Conversion Between Multiple 37x5s and Multiple 3x74s Example, page 445
- QLLC Conversion Between a Single 37x5 and Multiple 3x74s Across an Arbitrary WAN Example, page 446

The examples describe four increasingly complex QLLC conversion topologies and possible software configurations for each. Following the examples are sample NCP definitions that the 37x5 FEP in these topologies could use and VTAM definitions that the IBM host in these topologies could use to reflect the routers in the communication path.

QLLC Conversion Between a Single 37x5 and a Single 3x74 Example

Figure 181, shown previously, illustrates the simplest QLLC conversion topology—a single 37x5 FEP on a Token Ring communicating with a single 3x74 cluster controller across an X.25 network. A router connects the Token Ring to the X.25 network. In Figure 181, notice that the router's X.25 interface is treated as a virtual ring for configuration purposes.

The following configuration file configures the Cisco IOS software to support the network topology shown in Figure 181:

```

source-bridge ring-group 100
!
interface serial 0
encapsulation x25
x25 address 31102120100
x25 map qllc 0100.0000.0001 31104150101
qllc srb 0100.0000.0001 201 100
!
! Allow the 3x74 to initiate the connection.
!
qllc partner 0100.0000.0001 4000.0101.0132

interface tokenring 0
source-bridge 1 1 100

```

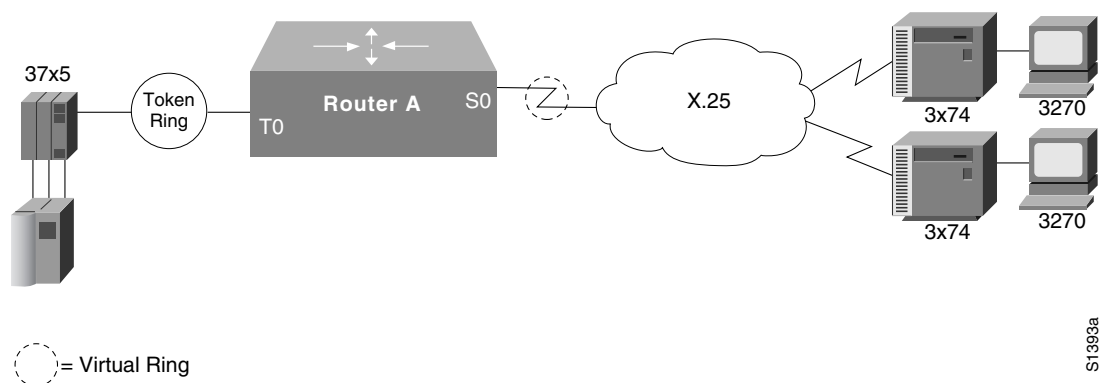
In this configuration file, the **source-bridge ring-group** command defines a virtual ring number 100. The serial 0 interface that connects to the X.25 network is then configured for X.25 DTE operation using the **encapsulation x25** command and assigned the X.121 address of 31102120100 using the **x25 address** command. The **x25 map qllc** command associates the X.121 address of the remote X.25 device (31104150101) with a virtual Token Ring MAC address (0100.0000.0001) the Token Ring device will use to communicate with this remote X.25 device. The **qllc srb** command indicates that the virtual MAC address of the X.25 device will be used to communicate with the real MAC address of the Token Ring device.

The **qllc partner** command enables the software to open a connection to the local Token Ring device at MAC address 4000.0101.0132 on behalf of the remote X.25 device at virtual Token Ring MAC address 0100.0000.0001. The **source-bridge** command configures the router's Token Ring 0 interface for local source-route bridging by associating the router's virtual ring number 100 with the ring number (1) of the local Token Ring and the bridge number (1) that uniquely identifies this bridge interface.

QLLC Conversion Between a Single 37x5 and Multiple 3x74s Example

Figure 182 shows a slightly more complex QLLC conversion topology. The same 37x5 FEP on a Token Ring connects through a router to an X.25 network, but communicates with multiple 3x74 cluster controllers through X.25.

Figure 182 QLLC Conversion Between a Single 37x5 and Multiple 3x74s



The following configuration file configures the Cisco IOS software to support the network topology shown in Figure 182:

```

source-bridge ring-group 100
!
interface serial 0
 encapsulation x25
 x25 address 3137005469
!
! configure the first 3174
!
x25 map qllc 0000.0cff.0001 31370054111
!
! 1001 - virtual ring used by all qllc devices
! 100 - the virtual ring group
!
qllc srb 0000.0cff.0001 1001 100
qllc partner 0000.0cff.0001 4000.1160.0000
qllc xid 0000.0cff.0001 01710017
!
! configure the second 3174
!
x25 map qllc 0000.0cff.0002 313700543247
!
! 1001 - virtual ring used by all qllc devices
! 100 - the virtual ring group
!
qllc srb 0000.0cff.0002 1001 100
qllc partner 0000.0cff.0002 4000.1160.0000
qllc xid 0000.0cff.0002 01710017
!
interface tokenring 0
!
! Since this is a real bridge, we have to define the way it
! bridges to the Qllc virtual ring.
!
source-bridge 1 1 100
source-bridge spanning

```

QLLC Conversion Between Multiple 37x5s and Multiple 3x74s Example

In the following example, two 3x74s on a Token Ring each attach to a different 37x5 on the other side of an X.25 network. Only one Token Ring interface is used. Do not create a bridge from the QLLC virtual ring (1001) to the physical Token Ring (1). Instead, define a virtual ring group (for example, 100).

```

interface serial 0
 encapsulation x25
 x25 address 3137005469
!
! configure the router for the first 3x74
!
x25 map qllc 0000.0cff.0001 31370054111
!
! 1001 - virtual ring used by all qllc devices
! 1 - the local Token Ring number
!
qllc srb 0000.0cff.0001 1001 1
qllc partner 0000.0cff.0001 4000.1160.0000
!
! configure the router for the second 3x74
!
x25 map qllc 0000.0cff.0002 31370053247

```

```

!
! 1001 - virtual ring used by all qllc devices
! 1 - the local Token Ring number
!
! Note that the partner's MAC address and XID are different from
! those in the first 3x74.
!
qllc srb 0000.0cff.0001 1001 1
qllc partner 0000.0cff.0002 4000.1161.1234
!
interface tokenring 0
!
! Since this is a real bridge, we have to define the way it bridges
! to the QLLC virtual ring.
!
source-bridge 1 1 1001
source-bridge spanning

```

QLLC Conversion Between a Single 37x5 and Multiple 3x74s Across an Arbitrary WAN Example

Figure 182 includes an added arbitrary WAN in the communication path between the 37x5 FEP and the multiple 3x74 cluster controllers. The arbitrary WAN can be a multihop network, whereas QLLC conversion treats the X.25 network as a single-hop network.

In Figure 182, notice that the arbitrary WAN and the routers on either side of it form a single virtual ring, as configured using the **source-bridge ring-group** global configuration command.

In this configuration file, Router A uses an IP address of 172.18.2.2 and its Token Ring interface is attached to Token Ring 1. Because Router A connects to the Token Ring, it does not need to be configured for QLLC conversion. Router B, configured for QLLC conversion because it connects directly to the X.25 network through its serial interface, uses an X.121 address of 31102120100 and an IP address of 172.18.1.1. The 37x5 device uses a MAC address of 4000.0101.0132. The virtual MAC address of 0100.0000.0001 has been assigned to the 3x74 device.

Router A

The following configuration file configures the Router A in Figure 182:

```

source-bridge ring-group 100
source-bridge remote-peer 100 tcp 172.18.1.1 local-ack
source-bridge remote-peer 100 tcp 172.18.2.2 local-ack
!
interface ethernet 0
ip address 131.108.3.3 255.255.255.0
!
interface tokenring 0
ip address 172.18.2.2 255.255.255.0
source-bridge 1 1 100
source-bridge spanning

```

Router B

The following configuration file configures the Router B in Figure 182:

```

source-bridge ring-group 100
source-bridge remote-peer 100 tcp 172.18.1.1 local-ack
source-bridge remote-peer 100 tcp 172.18.2.2 local-ack
source-bridge qllc-local-ack
!
interface serial 0

```

```

encapsulation x25
x25 address 31102120100
x25 map qllc 0100.0000.0001 31104150101
x25 map qllc 0100.0000.0002 31104150102
qllc srb 0100.0000.0001 201 100
qllc srb 0100.0000.0002 201 100
!
! Allow the 3174 to initiate the connection.
!
qllc partner 0100.0000.0001 4000.0101.0132
qllc partner 0100.0000.0002 4000.0101.0132
!
interface ethernet 0
ip address 172.18.1.1 255.255.255.0

```

NCP and VTAM Sysgen Parameters

The sample system generation (sysgen) parameters in this section show typical NCP and VTAM values that correspond with configurations for Router A, Router B, and Router C in Figure 180, Figure 181, and Figure 182. Figure 180 and Figure 181 show SDLLC media translation and Figure 182 shows QLLC conversion.

IBM's ACF/NCP uses a function called NTRI (NCP/Token Ring Interconnection) to support Token Ring-attached SNA devices. NTRI also provides translation from Token Ring-attached SNA devices (Physical Units) to switched (dial-up) devices. VTAM provides the resolution for these devices in a Switched Major Node. VTAM treats these devices on NTRI logical lines as switched devices. (For more information consult IBM documentation *NCP/SSP/EP Resource Definition Reference*, SC30-3448-04.)

Using SDLLC, the Cisco IOS software translates SDLC leased line protocol into Token Ring LLC2 protocol, then the NTRI function in ACF/NCP translates Token Ring LLC2 protocol into an SNA switched protocol.

NCP Generation Definitions

```

*****
***          SAMPLES BASED ON ACF/NCP V5 R4.
***          NOT ALL NCP PARAMETERS ARE SHOWN
*****
*
*****
*          OPTIONS DEFINITION STATEMENT
*****
NCPOPTOPTIONSNEWDEFN=YESNTRI GENERATION, MUST BE FIRST STMT
*
*****
*BUILD MACRO
*****
NCPBUBUILDLOCALTO=1.5,NTRI ACK TIMER FOR LOCAL TOKEN RINGS
REMOTTO=2.5,NTRI ACK TIMER FOR REMOTE TOKEN RINGS
USED IN SDLLC CONFIGURATIONS, NOTE 1.
*
*****
*DYNAMIC RECONFIGURATION POOL SPACE
*****
DRPOOLUDRPOOLNUMTYP2=50RESERVE 50 LUS ON PU. T2 PUS
*
*****
*PHYSICAL GROUP FOR NTRI TIC #1, DEFINITIONS FOR THE TOKEN RING
*ADAPTER TO ESTABLISH PHYSICAL CONNECTIVITY
*****

```

```

EPHYGGROUPECLTYPE=PHYSICAL
*
EPHYLLINEADAPTER=TIC2,TYPE OF ADAPTER
ADDRESS=(16,FULL),INTERNAL FEP TIC ADDRESS
PORTADD=0,
LOCADD=10005a7d8123,TIC ADDRESS, NOTE 2.
RCVBUFC=1440,
MAXTSL=2012,
TRSPED=16TOKEN RING SPEED
*
EPHYPPUPU
*
EPHYLULUISTATUS=INACTIVE
*
*****
*NTRI PERIPHERAL LOGICAL LINE GROUP, LINE AND PU PAIRS ARE
*GENERATED BY THE AUTOGEN PARAMETER.
*****
ELOGGGROUPECLTYPE=LOGICAL,
PHYPORT=0,
CALL=INOUT,
AUTOGEN=3ONE PER SDLLC CONTROLLER,
NOTE 3.
*****

```

VTAM Definitions

```

*****
*VTAM SWITCHED MAJOR NODE, BASED ON ACF/VTAM V3 R4.
*THE CODING BELOW SUPPORTS DIAL IN OPERATION ONLY. TYPICALLY,
*NTRI IMPLEMENTATIONS USE ONLY DIAL IN. IF DIAL OUT FROM AN
*APPLICATION IS REQUIRED, PATH MACROS MUST BE USED. CONSULT
*THE APPROPRIATE VTAM INSTALLATION REFERENCE MANUAL.
*****
VSWITCHVBUILDTYPE=SWNET
*
VPU1PUADDR=13,COULD BE ANYTHING (NOT USED)
IDBLK=017,XID PARM, NOTE 4.
IDNUM=200c1,XID PARM, NOTE 4.
MAXOUT=7,
MAXDATA=265,
MODETAB=AMODETAB,
DLOGMOD=US327X,
PUTYPE=2,
USSTAB=USS327X
*
VLU1ALULOCADDR=2,
VLU1BLULOCADDR=3
*
VPU2PUADDR=13,COULD BE ANYTHING (NOT USED)
IDBLK=017,XID PARM, NOTE 4.
IDNUM=200c2,XID PARM, NOTE 4.
MAXOUT=7,
MAXDATA=265,
MODETAB=AMODETAB,
DLOGMOD=US327X,
PUTYPE=2,
USSTAB=USS327X
*
VLU2ALULOCADDR=2,
VLU2BLULOCADDR=3
*
VPU3PUADDR=13,COULD BE ANYTHING (NOT USED)
IDBLK=017,XID PARM, NOTE 4.

```

```
IDNUM=200c3,XID PARM, NOTE 4.
MAXOUT=7,
MAXDATA=265,
MODETAB=AMODETAB,
DLOGMOD=US327X,
PUTYPE=2,
USSTAB=USS327X
*
VLU3ALULOCADDR=2,
VLU3BLULOCADDR=3
*
```

Notes

In these sample definitions:

1. REMOTTO is the NCP's T1 timer for remote Token Rings. All connections use RIF information and therefore look like remote Token Ring devices. The default is 2.5 seconds, which is adequate for most situations; however, when slow-speed links are used, this parameter should be reviewed to ensure enough time for link-level acknowledgments.
2. The LOCADD parameter defines the locally administered address of the TIC in the NCP. The Cisco IOS software, configured for SDLLC, will insert this address as the 802.5 destination address field in TEST and XID frames to establish connectivity and then in data frames during the session. The **sdllc partner** and **qlc partner** commands define this connection in the Cisco IOS software. Each SDLC control unit is defined with an **sdllc partner** or **qlc partner** command.
3. The AUTOGEN parameter specifies the number of LINE and PU pairs that are automatically generated by NDF (Network Definition Facility). Each controller requires a LINE and PU definition in the ELCTYPE LOGICAL group. These represent control block space in the NCP simulating switched line as described earlier.
4. The IDBLK and IDNUM parameters in VTAM are used to identify incoming connection requests. IDBLK is typically unique for each type of IBM device. IDNUM is any five hexadecimal digit combination. The Cisco routers configured for SDLLC or QLLC conversion must associate an IDBLK/IDNUM combination with a controller by using the **sdllc xid** or **qlc xid** command. If not using the **qlc xid** command, then IDBLK/IDNUM must agree with the values of the X.25 attached devices. During activation, an XID will be sent to the NCP containing the specific IDBLK/IDNUM. NCP will send these values to VTAM in an SNA command called REQCONT. VTAM will search its switched major nodes to find a match. If found, VTAM will establish sessions with the device by sending activation commands (ACTPU, ACTLU).

