Configuring DSPU and SNA Service Point Support

This chapter describes Cisco IOS support for Systems Network Architecture (SNA) downstream physical unit (DSPU) devices and SNA Service Point. For a complete description of the DSPU and SNA Service Point commands mentioned in this chapter, refer to the “DSPU and SNA Service Point Configuration Commands” chapter of the Cisco IOS Bridging and IBM Networking Command Reference (Volume 2 of 2). 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:

- Technology Overview, page 1
- DSPU Configuration Task List, page 3
- Configuring SNA Service Point Support, page 16
- Monitoring and Maintaining DSPU and SNA Service Point Feature Status, page 21
- DSPU and SNA Service Point Configuration Examples, page 22

To identify the hardware platform or software image information associated with a feature, use the Feature Navigator on Cisco.com to search for information about the feature or refer to the software release notes for a specific release. For more information, see the “Identifying Platform Support for Cisco IOS Software Features” section on page li in the “Using Cisco IOS Software” chapter.

Technology Overview

DSPU is a software feature that enables the router to function as a physical unit (PU) concentrator for SNA PU type 2 nodes. PU concentration at the device simplifies the task of PU definition at the upstream host while providing additional flexibility and mobility for downstream PU devices.

The DSPU feature allows you to define downstream PU type 2 devices in the Cisco IOS software. DSPU reduces the complexity of host configuration by letting you replace multiple PU definitions that represent each downstream device with one PU definition that represents the router.

Because you define the downstream PUs at the router rather than the host, you isolate the host from changes in the downstream network topology. Therefore you can insert and remove downstream PUs from the network without making any changes on the host.
The concentration of downstream PUs at the router also reduces network traffic on the WAN by limiting the number of sessions that must be established and maintained with the host. The termination of downstream sessions at the router ensures that idle session traffic does not appear on the WAN.

SNA service point support in the Cisco IOS software assumes that NetView or an equivalent product is available at the SNA host. The user interacts with the network management feature in the router and at the SNA host. In the Cisco IOS software, you can configure the host connection and show the status of this connection. At the SNA host, you can use the NetView operator’s console to view alerts and to send and receive Cisco syntax commands to the Cisco device.

Figure 1 shows a router functioning as a DSPU concentrator.

The SSCP sessions established between the router and its upstream host are completely independent of the SSCP sessions established between the router and its downstream PUs. SNA traffic is routed at a logical unit (LU) level using a routing algorithm that maps downstream LUs onto upstream LUs.

Figure 2 illustrates the SNA perspective of DSPU.
DSPU Configuration Task List

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

- Defining DSPU Upstream Hosts, page 3 (Required)
- Defining Downstream PUs, page 4 (Required)
- Defining DSPU LUs, page 6 (Required)
- Configuring DSPU to Use a Data-Link Control, page 7 (Optional)
- Defining the Number of Outstanding, Unacknowledged Activation RUs, page 15 (Optional)

See the “DSPU and SNA Service Point Configuration Examples” section on page 22 for examples.

Defining DSPU Upstream Hosts

The upstream host provides logical units (LUs) that the Cisco IOS software assigns for use by its downstream PUs. Because one upstream host can only provide a maximum of 255 LUs, the DSPU feature supports multiple hosts. Multiple upstream host support allows the DSPU router to provide more than 255 LUs for use by its downstream PUs.

To define a DSPU host over Token Ring, Ethernet, Fiber Distributed Data Interface (FDDI), remote source-route bridging (RSRB), or virtual data-link control connections, use the following command in global configuration mode:

```
Router(config)# dspu host host-name xid-snd xid rmac remote-mac [rsap remote-sap] [lsap local-sap] [interface slot/port] [window window-size] [maxiframe max-iframe] [retries retry-count] [retry-timeout retry-timeout] [focalpoint]
```

Defines a DSPU host over Token Ring, Ethernet, FDDI, RSRB, or virtual data-link control connections.

To define a DSPU host over a Synchronous Data-Link Control (SDLC) connection, use the following command in global configuration mode:
To define a DSPU host over an X.25/Qualified Logical Link Control (QLLC) connection, use the following command in global configuration mode:

```
Router(config)# dspu host host-name xid-snd xid sdlc sdlc-addr [interface slot/port] [window window-size] [maxiframe max-iframe] [retries retry-count] [retry-timeout retry-timeout] [focalpoint]
```

Defines a DSPU host over an SDLC connection.

To define a DSPU host over a Frame Relay connection, use the following command in global configuration mode:

```
Router(config)# dspu host host-name xid-snd xid x25 remote-x121-addr [qllc local-x121-subaddr] [interface slot/port] [window window-size] [maxiframe max-iframe] [retries retry-count] [retry-timeout retry-timeout] [focalpoint]
```

Defines a DSPU host over an X.25/QLLC connection.

To define a DSPU host over a Frame Relay connection, use the following command in global configuration mode:

```
Router(config)# dspu host host-name xid-snd xid dlc1 dlci-number [rsap remote-sap] [lsap local-sap] [interface slot/port] [window window-size] [maxiframe max-iframe] [retries retry-count] [retry-timeout retry-timeout] [focalpoint]
```

Defines a DSPU host over a Frame Relay connection.

## Defining Downstream PUs

To define the downstream PUs, perform either of the tasks in the following sections, depending on your circumstances:

- Explicitly Defining a Downstream PU, page 4
- Enabling the Default PU Option, page 6

### Explicitly Defining a Downstream PU

Explicitly define a downstream PU if you require the Cisco IOS software to perform verification checking on incoming downstream connections or to initiate an outgoing downstream connection.

For Cisco IOS Release 11.3 and later releases, the number of DSPU PUs you can configure is 1024.

To explicitly define a downstream PU over Token Ring, Ethernet, FDDI, RSRB, virtual data-link control, or native client interface architecture (NCIA) connections, use the following command in global configuration mode:
To explicitly define a downstream PU over an SDLC connection, use the following command in global configuration mode:

```
Router(config)# dspu pu pu-name [sdlc sdlc-addr] [xid-rcv xid] [interface slot/port] [window size] [maxiframe max-iframe] [retries retry-count] [retry-timeout retry-timeout]
```

Explicitly defines a downstream PU over an SDLC connection.

To explicitly define a downstream PU over an X.25/QLLC connection, use the following command in global configuration mode:

```
Router(config)# dspu pu pu-name [x25 remote-x121-addr] [qllc local-x121-subaddr] [xid-rcv xid] [interface slot/port] [window size] [maxiframe max-iframe] [retries retry-count] [retry-timeout retry-timeout]
```

Explicitly defines a downstream PU over an X.25/QLLC connection.

To explicitly define a downstream PU over a Frame Relay connection, use the following command in global configuration mode:

```
Router(config)# dspu pu pu-name [dlci dlci-number] [rsap remote-sap] [lsap local-sap] [xid-rcv xid] [interface type slot/port] [window size] [maxiframe max-iframe] [retries retry-count] [retry-timeout retry-timeout]
```

Explicitly defines a downstream PU over a Frame Relay connection.

A PU definition must have either an xid-rcv parameter or an address (rmac, sdlc, x25 or dlci) parameter.

If the Cisco IOS software will perform verification checking on incoming downstream connections, there are several combinations of parameters that you can configure for verification matching. Note that the address parameter, when specified, is considered to be the primary key on the PU definition. Therefore, if both an address and xid-rcv are configured, the matching algorithm will match on the address and ignore the xid-rcv parameter.

- Match on xid-rcv value only
  
  User may define a downstream PU using only the xid-rcv value so that any connecting PU that specifies the value of the configured XID will match that PU definition.

- Match on xid-rcv and interface values
User may define a downstream PU using the xid-rcv and interface values so that any PU connecting into the configured interface that specifies the value of the configured XID will match the PU definition.

- **Match on addressing values only**

  User may define a downstream PU using only the addressing values (RMAC/RSAP/LSAP, SDLC, DLCI/RSAP/LSAP, or X25/QLLC) so that any connecting PU with addressing that matches the configured addressing will match that PU definition. If no PU definition is found to match the incoming RSAP, then a match is accepted on a PU that has the correct RMAC/LSAP or DLCI/LSAP.

- **Match on addressing and interface values**

  User may define a downstream PU using the interface and addressing values (RMAC/RSAP/LSAP, SDLC, DLCI/RSAP/LSAP, or X25/QLLC) so that any PU connecting into the configured interface with addressing that matches the configured addressing will match the PU definition. If no PU definition is found to match the incoming RSAP, then a match is accepted on a PU that has the correct RMAC/LSAP or DLCI/LSAP and interface.

The Cisco IOS software rejects any incoming downstream connections that do not match the parameters of a defined downstream PU unless the default PU option is also enabled.

### Enabling the Default PU Option

Configure the DSPU default PU option if you do not require the Cisco IOS software to verify incoming downstream connections. The default PU option allows the software to accept incoming downstream connections without an explicit definition for the downstream PU.

To enable the default PU option, use the following command in global configuration mode:

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>Router(config)# dspu default-pu [window window-size] [maxiframe max-iframe]</td>
<td>Enables the default PU option.</td>
</tr>
</tbody>
</table>

### Defining DSPU LUs

Specify the LU routing algorithm used to map the upstream LUs to the downstream LUs and to define all LUs for each upstream and downstream PU.

The DSPU feature assigns upstream LUs to downstream LUs based on the selected LU routing algorithm and performs the mapping necessary for SNA data transfer.

The DSPU feature supports two alternative mapping algorithms that are described in the following sections:

- Defining Dedicated LU Routing, page 6
- Defining Pooled LU Routing, page 7

An upstream host PU or downstream PU can support up to 255 LU sessions. The DSPU feature allows each LU to be individually configured for either dedicated LU routing or pooled LU routing.

### Defining Dedicated LU Routing

You can configure an upstream LU so that it is reserved, or dedicated, for use by a specific downstream LU.
To define a dedicated LU or a range of dedicated LUs for an upstream host and downstream PU, use the following command in global configuration mode:

```
Router(config)# dspu lu lu-start [lu-end]
     [host host-name host-lu-start | pool pool-name] [pu pu-name]
```

Defines a dedicated LU or a range of dedicated LUs for a downstream PU.

See the “Dedicated LU Routing Example” section on page 22 for an example of dedicated LU routing.

**Defining Pooled LU Routing**

You can configure an upstream host LU so that it is a member of a pool of LUs. When a downstream connection is established and the downstream LU is configured as a pooled LU, the Cisco IOS software selects an upstream LU from the pool for assignment to the downstream LU.

Pooled LU routing allows a limited number of upstream host LUs to be shared (at different times) among many downstream LUs.

To define a host LU or a range of host LUs in an LU pool, use the following command in global configuration mode:

```
Router(config)# dspu pool pool-name host
     host-name lu lu-start [lu-end] [inactivity-timeout inactivity-minutes]
```

Defines a host LU or a range of host LUs in an LU pool.

You can configure a downstream LU as a pooled LU. When a downstream connection is established and the downstream LU is configured as a pooled LU, the software selects an upstream LU from the specified pool for assignment to the downstream LU.

To define a pooled LU or a range of pooled LUs for a downstream PU, use the following command in global configuration mode:

```
Router(config)# dspu lu lu-start [lu-end]
     pool pool-name pu pu-name
```

Defines a pooled LU or a range of pooled LUs for a downstream PU.

See the “Pooled LU Routing Example” section on page 23 for an example of pooled LU routing.

**Configuring DSPU to Use a Data-Link Control**

The final step in configuring DSPU is to define the data-link controls that will be used for upstream host and downstream PU connections.

The DSPU feature supports the data-link controls described in the following sections:

- Configuring DSPU to Use Token Ring, Ethernet, or FDDI, page 8
- Configuring DSPU to Use RSRB, page 8
- Configuring DSPU to Use RSRB with Local Acknowledgment, page 10
Configuring DSPU to Use Token Ring, Ethernet, or FDDI

You can configure DSPU to use the Token Ring, Ethernet, or FDDI data-link controls by enabling a service access point (SAP) address on the interface. Each interface can support up to 254 local SAPs enabled for either upstream or downstream connections; a local SAP cannot be enabled for both upstream and downstream connections on the same interface.

To enable a local SAP on the Token Ring, Ethernet, or FDDI interfaces for use by upstream hosts, use the following command in interface configuration mode:

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>Router(config-if)# dspu enable-host [lsap local-sap]</td>
<td>Enables local SAP for upstream hosts.</td>
</tr>
</tbody>
</table>

To enable a local SAP on the Token Ring, Ethernet, or FDDI interfaces for use by downstream PUs, use the following command in interface configuration mode:

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>Router(config-if)# dspu enable-pu [lsap local-sap]</td>
<td>Enables local SAP for downstream PUs.</td>
</tr>
</tbody>
</table>

Once a local SAP is enabled, it is ready to accept incoming connection attempts from the remote device (upstream host or downstream PU). Alternately, initiate an outgoing connection to the remote device by using the following command in interface configuration mode:

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>Router(config-if)# dspu start {host-name</td>
<td>pu-name}</td>
</tr>
</tbody>
</table>

Configuring DSPU to Use RSRB

To configure DSPU to use RSRB, you must create a DSPU/RSRB data-link control.

Cisco’s implementation of DSPU/RSRB data-link control uses the concept of a virtual Token Ring device residing on a virtual Token Ring to represent the Cisco IOS software to upstream hosts and downstream PUs across an RSRB network. This is similar to Cisco’s implementation of SDLLC.

Because the upstream host and downstream PU expects its peer to also be on a Token Ring, you must assign a virtual Token Ring address (the DSPU virtual MAC address) to the DSPU/RSRB data-link control. Like real Token Ring addresses, the DSPU virtual MAC address must be unique across the network.
In addition to assigning the DSPU virtual MAC address, you must also assign a DSPU virtual ring number to the DSPU/RSRB data-link control. The DSPU virtual ring number must be unique across the network.

**Note**
The DSPU virtual ring number is a different number from the virtual ring group numbers that you use to configure RSRB and multiport bridging.

The combination of the DSPU virtual MAC address and the DSPU virtual ring number identifies the DSPU/RSRB data-link control interface to the rest of an RSRB network.

When an end station (either an upstream host or a downstream PU) attempts to connect with the DSPU software, the following events occur:

1. The end station sends explorer packets with the locally administered MAC address on the router interface to which the end station is connected.
2. The router configured with that locally administered MAC address or with the hardware MAC address intercepts the frame, fills in the DSPU virtual ring number and the DSPU bridge number in the routing information field (RIF), and sends a response to the end station.
3. The end station establishes a session with the DSPU router.

To define the DSPU/RSRB data-link control interface, use the following commands in global configuration mode:

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>Step 1</td>
<td>Router(config)# source-bridge ring-group ring-group [virtual-mac-address]</td>
</tr>
<tr>
<td>Step 2</td>
<td>Router(config)# source-bridge remote-peer ring-group tcp ip-address local-ack</td>
</tr>
<tr>
<td>Step 3</td>
<td>Router(config)# dspu rsrb local-virtual-ring bridge-number target-virtual-ring virtual-macaddr</td>
</tr>
</tbody>
</table>

After you define the DSPU RSRB data-link control, configure DSPU to use the RSRB data-link control by enabling a local SAP for either upstream or downstream connections.

To enable a local SAP on RSRB for use by upstream hosts, use the following command in global configuration mode:

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>Router(config)# dspu rsrb enable-host [lsap local-sap]</td>
<td>Enables local SAP for upstream hosts.</td>
</tr>
</tbody>
</table>

To enable a local SAP on RSRB for use by downstream PUs, use the following command in global configuration mode:

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>Router(config)# dspu rsrb enable-pu [lsap local-sap]</td>
<td>Enables local SAP for downstream PUs.</td>
</tr>
</tbody>
</table>
Once a local SAP is enabled, it is ready to accept incoming connection attempts from the remote device (upstream host or downstream PU) over RSRB. Alternatively, initiate an outgoing connection to the remote device by using the following command in global configuration mode:

```
Router(config)# dspu rsrb start (host-name | pu-name)
```

### Configuring DSPU to Use RSRB with Local Acknowledgment

Configuring DSPU to use RSRB with local acknowledgment is identical to configuring RSRB with local acknowledgment. If you add the `local-ack` keyword to the `source-bridge remote-peer` configuration command, DSPU will use local acknowledgment for any end stations that connect to DSPU from that peer.

To configure DSPU to use RSRB with local acknowledgment, use the following commands in global configuration mode:

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
</table>
| Step 1  | Router(config)# source-bridge ring-group ring-group \( [virtual-mac-address] \)  
          | Defines an RSRB ring group. |
| Step 2  | Router(config)# source-bridge remote-peer  
          | ring-group tcp ip-address local-ack  
          | Defines a remote peer with the local acknowledgment feature. |
| Step 3  | Router(config)# dspu rsrb local-virtual-ring  
          | bridge-number target-virtual-ring virtual-macaddr  
          | Defines the DSPU/RSRB interface. |

### Configuring DSPU to Use Virtual Data-Link Control

To configure DSPU to use virtual data-link control, you must create a DSPU virtual data-link control interface.

Similar to our implementation of SDLLC, the DSPU virtual data-link control interface uses the concept of a virtual Token Ring device residing on a virtual Token Ring to represent the Cisco IOS software to upstream hosts and downstream PUs across a network.

Because the upstream host and downstream PU expects its peer to also be on a Token Ring, you must assign a virtual Token Ring address (the DSPU virtual MAC address) to the DSPU virtual data-link control interface. Like real Token Ring addresses, the DSPU virtual MAC address must be unique across the network.

In addition to assigning the DSPU virtual MAC address, you must also identify the source-route bridging virtual ring number with which the DSPU virtual MAC address will be associated. The source-route bridging virtual ring number is set using the `source-bridge ring-group` command. This is documented in the “Source-Route Bridging Commands” chapter of the *Cisco IOS Bridging and IBM Networking Command Reference* (Volume 1 of 2).

The combination of the DSPU virtual MAC address and the source-route bridging virtual ring number identifies the DSPU virtual data-link control interface to the rest of the DLSw+ network.
When an end station (either an upstream host or a downstream PU) attempts to connect with the DSPU software, the following events occur:

1. The end station sends explorer packets with the locally administered MAC address on the router interface to which the end station is connected.

2. The router configured with that locally administered MAC address intercepts the frame, DLSw+ adjusts the routing information field (RIF), and sends a response to the end station.

3. The end station establishes a session with the DSPU router.

Prior to creating the DSPU virtual data-link control interface, you must also configure DLSw+ peers so that DLSw+ can provide the communication path. The commands for defining DLSw+ local and remote peers are documented in the “DLSw+ Configuration Commands” chapter of the *Cisco IOS Bridging and IBM Networking Command Reference* (Volume 1 of 2).

To define the DSPU virtual data-link control interface, use the following command in global configuration mode:

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>Router(config)# dspu vdlc ring-group virtual-mac-address</td>
<td>Defines the DSPU virtual data-link control interface.</td>
</tr>
</tbody>
</table>

After you define the DSPU virtual data-link control interface, configure DSPU to use virtual data-link control by enabling a local SAP for either upstream or downstream connections.

To enable a local SAP on the virtual data-link control for use by upstream hosts, use the following command in global configuration mode:

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>Router(config)# dspu vdlc enable-host [lsap local-sap]</td>
<td>Enables local SAP for upstream hosts.</td>
</tr>
</tbody>
</table>

To enable a local SAP on the virtual data-link control for use by downstream PUs, use the following command in global configuration mode:

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>Router(config)# dspu vdlc enable-pu [lsap local-sap]</td>
<td>Enables local SAP for downstream PUs.</td>
</tr>
</tbody>
</table>

Once a local SAP is enabled, it is ready to accept incoming connection attempts from the remote device (upstream host or downstream PU) using virtual data-link control. Alternately, initiate an outgoing connection to the remote device by using the following command in global configuration mode:

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>Router(config)# dspu vdlc start (host-name</td>
<td>pu-name)</td>
</tr>
</tbody>
</table>

**Configuring DSPU to Use SDLC**

Before DSPU may be configured to use the SDLC data-link control, the serial interface must be defined for SDLC encapsulation and assigned an SDLC role.
To define the serial interface to use SDLC and specify the SDLC role, use the following commands in interface configuration mode:

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>Router(config-if)# encapsulation sdlc</code></td>
<td>Enables SDLC encapsulation on the serial interface.</td>
</tr>
<tr>
<td>`Router(config-if)# sdlc role {none</td>
<td>primary</td>
</tr>
</tbody>
</table>

For the connection to be established without XID exchange, the SDLC role must be **primary** if DSPU will be initiating connections to the SDLC partner. The SDLC role must be **secondary** or **none** if the SDLC partner will be initiating connections with DSPU.

When an XID exchange is required, the SDLC role must be **prim-xid-poll** or **none** if DSPU will be initiating connections to the SDLC partner. The role must be **none** if the SDLC partner will be initiating connections with DSPU.

The SDLC addresses used on the SDLC link must also be defined. If DSPU is configured to initiate the connection, then the SDLC address identifies the SDLC partner. If the remote SDLC device initiates the connection, then the SDLC address identifies the address for which a connection will be accepted.

To configure the SDLC address, use the following command in interface configuration mode:

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>Router(config-if)# sdlc address hexbyte</code></td>
<td>Defines the SDLC address.</td>
</tr>
</tbody>
</table>

Finally, the SDLC address must be enabled for use by DSPU. Each interface can support up to 255 SDLC addresses enabled for either upstream or downstream connections; an SDLC address cannot be enabled for both upstream and downstream connections on the same interface. If the SDLC role is **none**, there can be only one SDLC address on that interface.

To enable an SDLC address for use by upstream host connections, use the following command in interface configuration mode:

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>Router(config-if)# dspu enable-host sdlc sdlc-address</code></td>
<td>Enables the SDLC address for an upstream host.</td>
</tr>
</tbody>
</table>

To enable an SDLC address for use by downstream PU connections, use the following command in interface configuration mode:

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>Router(config-if)# dspu enable-pu sdlc sdlc-address</code></td>
<td>Enables the SDLC address for the downstream PU.</td>
</tr>
</tbody>
</table>

When the SDLC role is configured as **primary**, DSPU initiates a connection with the remote device by sending set normal response mode (SNRM) when the SDLC address is enabled for DSPU.

When the SDLC role is configured as **prim-xid-poll**, DSPU initiates a connection with the remote device by sending a NULL XID when the SDLC address is enabled for DSPU.

When the SDLC role is configured as **secondary**, DSPU will not be ready to respond to SNRM until a `dspu start pu-name` command is issued.
When the SDLC role is configured as `none`, DSPU is ready to respond to a received XID or SNRM when the SDLC address is enabled for DSPU; otherwise, the connection may be initiated by issuing the `dspu start pu-name` command.

To configure DSPU to respond to SNRM when the SDLC role is configured as `secondary`, or to initiate a connection when the SDLC role is configured as `none`, use the following command in interface configuration mode:

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>`Router(config-if)# dspu start (host-name</td>
<td>pu-name)`</td>
</tr>
</tbody>
</table>

### Configuring DSPU to Use QLLC

Before DSPU may be configured to use the QLLC data-link control, the serial interface must be defined for X.25 encapsulation and assigned an X.121 address.

To define the serial interface to use X.25, use the following commands in interface configuration mode:

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>Router(config-if)# encapsulation x25 [dce]</code></td>
<td>Enables X.25 encapsulation on the serial interface.</td>
</tr>
<tr>
<td><code>Router(config-if)# x25 address x121-addr</code></td>
<td>Defines an X.121 address.</td>
</tr>
</tbody>
</table>

X.25 routing must also be configured so that incoming calls to the local X.121 address can be appropriately routed to the serial interface and mapped into the QLLC data-link control.

To define X.25 routing, use the following commands in global configuration mode:

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>Router(config)# x25 routing</code></td>
<td>Enables X.25 routing.</td>
</tr>
<tr>
<td><code>Router(config)# x25 route *local-x121-addr.* alias serial slot/port</code></td>
<td>Enables routing of X.25 packets to the serial interface.</td>
</tr>
</tbody>
</table>

To define which calls get mapped into QLLC, use the following command in interface configuration mode:

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>Router(config-if)# x25 map qllc x121-addr</code></td>
<td>Defines the remote X.121 address for mapping into QLLC.</td>
</tr>
</tbody>
</table>

Finally, the local X.121 subaddress must be enabled for use by DSPU. An X.121 subaddress can be enabled for either upstream or downstream connections; an X.121 subaddress cannot be enabled for both upstream and downstream connections on the same interface.

To enable an X.121 subaddress for use by upstream host connections via QLLC, use the following command in interface configuration mode:

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>Router(config-if)# dspu enable-host qllc x121-subaddress</code></td>
<td>Enables an X.121 subaddress for an upstream host.</td>
</tr>
</tbody>
</table>
To enable an X.121 subaddress for use by downstream PU connections via QLLC, use the following command in interface configuration mode:

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>Router(config-if)# dspu enable-pu qllc x121-subaddress</td>
<td>Enables an X.121 subaddress for a downstream PU.</td>
</tr>
</tbody>
</table>

Once an X.121 subaddress is enabled, it is ready to accept incoming connection attempts from the remote device (upstream host or downstream PU) over QLLC. Alternatively, initiate an outgoing connection to the remote device by using the following command in interface configuration mode:

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>Router(config-if)# dspu start {host-name</td>
<td>pu-name}</td>
</tr>
</tbody>
</table>

### Configuring DSPU to Use Frame Relay

Before DSPU may be configured to use the LLC2/Frame Relay data-link control, the serial interface must be defined for Frame Relay encapsulation.

To define the serial interface for Frame Relay encapsulation, use the following command in interface configuration mode:

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>Router(config-if)# encapsulation frame-relay ietf</td>
<td>Enables Frame Relay encapsulation on a serial interface.</td>
</tr>
</tbody>
</table>

The DLCI used on the Frame Relay link must be mapped into LLC2.

To configure the mapping of a DLCI into LLC2, use the following command in interface configuration mode:

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>Router(config-if)# frame-relay map llc2 dlci-number</td>
<td>Configures DLCI mapping into LLC2.</td>
</tr>
</tbody>
</table>

Finally, the local SAP address must be enabled for use by DSPU. A SAP address can be enabled for either upstream or downstream connections; a SAP address cannot be enabled for both upstream and downstream connections on the same interface.

To enable a local SAP on the LLC2/Frame Relay interface for use by upstream hosts, use the following command in interface configuration mode:

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>Router(config-if)# dspu enable-host [lsap local-sap]</td>
<td>Enables local SAP for upstream hosts.</td>
</tr>
</tbody>
</table>

To enable a local SAP for the LLC2/Frame Relay interface for use by downstream PUs, use the following command in interface configuration mode:
Once a local SAP is enabled, it is ready to accept incoming connection attempts from the remote device (upstream host or downstream PU) over Frame Relay. Alternatively, initiate an outgoing connection to the remote device by using the following command in interface configuration mode:

```
Router(config-if)# dspu enable-pu [lsap local-sap]
```

Enables local SAP for downstream PUs.

### Configuring DSPU to Use NCIA

To configure DSPU to use NCIA, you must perform the following tasks:

- Configure the NCIA server as the underlying transport mechanism.
- Enable a local SAP on the NCIA server for use by downstream PUs.

To configure the NCIA server as the underlying transport mechanism, use the following command in global configuration mode:

```
Router(config)# dspu ncia server-number
```

Configures the NCIA server as the underlying transport mechanism.

To enable a local SAP on the NCIA server for use by downstream PUs, use the following command in global configuration mode:

```
Router(config)# dspu ncia enable-pu [lsap local-sap]
```

Enables local SAP for downstream PUs.

### Defining the Number of Outstanding, Unacknowledged Activation RUs

The DSPU feature allows you to define the number of activation request/response units (RUs) such as ACTLUs or DDDLUs that can be sent by the Cisco IOS software before waiting for responses from the remote PU.

The DSPU activation window provides pacing to avoid depleting the router buffer pool during PU activation. Increasing the window size allows more LUs to become active in a shorter amount of time (assuming the required buffers for activation RUs are available). Decreasing the window size limits the amount of buffers the DSPU may use during PU activation. Typically, you do not need to change the default window size.

To define the number of unacknowledged activation RUs that can be outstanding, use the following command in global configuration mode:
Configuring SNA Service Point Support

Cisco’s implementation of SNA Service Point support includes support for three commands: Alerts, RUNCMD, and Vital Product Data support.

Alert support is provided as the Cisco IOS software sends unsolicited Alerts to NetView (or an equivalent network management application) at the host. This function occurs at the various router interfaces and protocol layers within the device.

RUNCMD support enables you to send commands to the router from the NetView console using the NetView RUNCMD facility, and the router sends the relevant replies back to the RUNCMD screen. Some commands, such as telnet, rsh, rlogin, and tn3270, are not supported.

Vital Product Data support allows you to request Vital Product Data from the NetView console. The router replies to NetView with the relevant information.

To configure SNA Service Point support, perform the tasks in the following sections:

- Defining a Link to an SNA Host, page 16
- Configuring Service Point Support to Use a Data-Link Control, page 17
- Specifying Names for All Attached LANs, page 21
- Specifying the Physical Location of the Router, page 21

Note: You must define the Service Point PU at the SNA host by using either ANS=STOP, or you can omit the ANS keyword. Do not use ANS=CONTINUE to define the Service Point PU at the SNA host. Coordinate this with your SNA host systems programmer.

Note: You do not need to perform the tasks in the next section if you have configured a DSPU host with the focalpoint parameter.

Defining a Link to an SNA Host

To define a link to an SNA host over Token Ring, Ethernet, FDDI, RSRB, or virtual data-link control connections, use the following command in global configuration mode:

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>Router(config)# sna host host-name xid-xnd xid rmac remote-mac [rsap remote-sap] [lsap local-sap] [interface slot/port] [window window-size] [maxiframe max-iframe] [retries retry-count] [retry-timeout retry-timeout] [focalpoint]</td>
<td>Defines a link to an SNA host over Token Ring, Ethernet, FDDI, RSRB, or virtual data-link control connections.</td>
</tr>
</tbody>
</table>
Configuring DSPU and SNA Service Point Support

To define a link to an SNA host over an SDLC connection, use the following command in global configuration mode:

Command
Router(config)# sna host host-name xid-end
    xid sdlc sdlc-addr [interface slot/port]
    [window window-size] [maxframe max-iframe]
    [retries retry-count] [retry-timeout retry-timeout] [focalpoint]

Purpose
Defines a link to an SNA host over an SDLC connection.

To define a link to an SNA host over an X.25/QLLC connection, use the following command in global configuration mode:

Command
Router(config)# sna host host-name xid-end
    xid x25 remote-x121-addr [qllc]
    local-x121-subaddr [interface slot/port]
    [window window-size] [maxframe max-iframe]
    [retries retry-count] [retry-timeout retry-timeout] [focalpoint]

Purpose
Defines a link to an SNA host over an X.25/QLLC connection.

To define a link to an SNA host over a Frame Relay connection, use the following command in global configuration mode:

Command
Router(config)# sna host host-name xid-end
    xid dcli dcli-number [rsap remote-sap] [lsap]
    local-sap [interface slot/port] [window window-size] [maxframe max-iframe]
    [retries retry-count] [retry-timeout retry-timeout] [focalpoint]

Purpose
Defines a link to an SNA host over a Frame Relay connection.

Configuring Service Point Support to Use a Data-Link Control

To configure Service Point to use a data-link control, perform the tasks in one of the following sections:

- Configuring Service Point to Use Token Ring, Ethernet, or FDDI, page 18
- Configuring Service Point to Use RSRB, page 18
- Configuring Service Point to Use RSRB with Local Acknowledgment, page 18
- Configuring Service Point to Use Virtual Data-Link Control, page 19
- Configuring Service Point Support for Frame Relay, page 20
- Configuring Service Point Support for SDLC, page 20
- Configuring Service Point Support for X.25, page 20

Note
You do not need to perform this task if you have configured a DSPU host with the focalpoint parameter and have configured the DSPU host to use a data-link control.
Configuring Service Point to Use Token Ring, Ethernet, or FDDI

To enable a local SAP on the Token Ring, Ethernet, or FDDI interfaces for use by SNA Service Point, use the following command in interface configuration mode:

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>Router(config-if)# sna enable-host [lsap lsap-address]</td>
<td>Enables local SAP for Service Point.</td>
</tr>
</tbody>
</table>

Once a local SAP is enabled, it is ready to accept incoming connection attempts from the remote host. Alternately, initiate an outgoing connection to the remote host by using the following command in interface configuration mode:

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>Router(config-if)# sna start host-name</td>
<td>Initiates a connection with a host via Token Ring, Ethernet, or FDDI.</td>
</tr>
</tbody>
</table>

Configuring Service Point to Use RSRB

To define the Service Point/RSRB data-link control interface, use the following commands in global configuration mode:

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>Step 1: Router(config)# source-bridge ring-group ring-group [virtual-mac-address]</td>
<td>Defines an RSB ring group.</td>
</tr>
<tr>
<td>Step 2: Router(config)# sna rsrb local-virtual-ring bridge-number target-virtual-ring virtual-macaddr</td>
<td>Defines the Service Point/RSRB interface.</td>
</tr>
</tbody>
</table>

To enable a local SAP on RSB for use by hosts, use the following command in global configuration mode:

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>Router(config)# sna rsrb enable-host [lsap local-sap]</td>
<td>Enables local SAP for hosts.</td>
</tr>
</tbody>
</table>

Once a local SAP is enabled, it is ready to accept incoming connection attempts from the remote host over RSB. Alternatively, initiate an outgoing connection to the remote host by using the following command in global configuration mode:

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>Router(config)# sna rsrb start host-name</td>
<td>Initiates a connection with a host via RSB.</td>
</tr>
</tbody>
</table>

Configuring Service Point to Use RSB with Local Acknowledgment

To configure Service Point to use RSB with local acknowledgment, use the following commands in global configuration mode:

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>Router(config)# sna rsrb enable-host [lsap local-sap]</td>
<td>Enables local SAP for hosts.</td>
</tr>
</tbody>
</table>

Once a local SAP is enabled, it is ready to accept incoming connection attempts from the remote host over RSB. Alternatively, initiate an outgoing connection to the remote host by using the following command in global configuration mode:
Configuring Service Point to Use Virtual Data-Link Control

To configure SNA Service Point to use virtual data-link control, you must create an SNA virtual data-link control interface.

Similar to our implementation of SD LLC, the SNA virtual data-link control interface uses the concept of a virtual Token Ring device residing on a virtual Token Ring to represent the Cisco IOS software to upstream hosts and downstream PUs across a network.

Because the upstream host and downstream PU expect their peer to also be on a Token Ring, you must assign a virtual Token Ring address (the SNA virtual data-link control virtual MAC address) to the SNA virtual data-link control interface. Like real Token Ring addresses, the SNA virtual MAC address must be unique across the network.

You must also identify the source-route bridging virtual ring number with which the SNA virtual MAC address will be associated. The source-route bridging virtual ring number is set using the `source-bridge ring-group` command, which is documented in the “Source-Route Bridging Commands” chapter of the Cisco IOS Bridging and IBM Networking Command Reference (Volume 1 of 2).

The combination of the SNA virtual MAC address and the source-route bridging virtual ring number identifies the SNA virtual data-link control interface to the rest of the DLSw+ network.

When an end station (either an upstream host or a downstream PU) attempts to connect with the SNA Service Point software, the following events occur:

1. The end station sends explorer packets with the locally administered MAC address on the router interface to which the end station is connected.
2. The router configured with that locally administered MAC address intercepts the frame, DLSw+ adjusts the RIF and sends a response to the end station.
3. The end station establishes a session with the SNA Service Point router.

Prior to creating the SNA virtual data-link control interface, you must also configure DLSw+ peers so that DLSw+ can provide the communication path. The commands for defining DLSw+ local and remote peers are documented in the “DLSw+ Configuration Commands” chapter of the Cisco IOS Bridging and IBM Networking Command Reference (Volume 1 of 2).

To define the Service Point virtual data-link control interface, use the following command in global configuration mode:

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>sna vdlc ring-group virtual-mac-address</code></td>
<td>Defines the Service Point virtual data-link control interface.</td>
</tr>
</tbody>
</table>
After you create the SNA virtual data-link control interface, configure SNA Service Point to use virtual data-link control by enabling a local SAP for upstream connections. To enable a local SAP on virtual data-link control for use by hosts, use the following command in global configuration mode:

```
Router(config)# sna vdlc enable-host [lsap local-sap]
```

Once a local SAP is enabled, it is ready to accept incoming connection attempts from the remote host using virtual data-link control. Alternatively, initiate an outgoing connection to the remote host by using the following command in global configuration mode:

```
Router(config)# sna vdlc start host-name
```

### Configuring Service Point Support for Frame Relay

To configure Service Point support for Frame Relay, use the following commands in interface configuration mode:

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>Step 1</td>
<td></td>
</tr>
<tr>
<td>Router(config-if)# frame-relay map llc2 dlci-number</td>
<td>Defines DLCI mapping into LLC2.</td>
</tr>
<tr>
<td>Step 2</td>
<td></td>
</tr>
<tr>
<td>Router(config-if)# sna enable-host lsap lsap-address</td>
<td>Enables a local SAP for hosts.</td>
</tr>
</tbody>
</table>

### Configuring Service Point Support for SDLC

To configure Service Point support for SDLC, use the following commands in interface configuration mode:

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>Step 1</td>
<td></td>
</tr>
<tr>
<td>Router(config-if)# sdlc role {none</td>
<td>primary</td>
</tr>
<tr>
<td>Step 2</td>
<td></td>
</tr>
<tr>
<td>Router(config-if)# sdlc address hexbyte</td>
<td>Defines the SDLC address.</td>
</tr>
<tr>
<td>Step 3</td>
<td></td>
</tr>
<tr>
<td>Router(config-if)# sna enable-host sdlc sdlc-address</td>
<td>Enables the SDLC address for the host.</td>
</tr>
</tbody>
</table>

### Configuring Service Point Support for X.25

To configure Service Point support for X.25, use the following commands in interface configuration mode:
Specifying Names for All Attached LANs

You can specify names for all Token Ring or Ethernet LANs attached to the router. These names are used to identify the LAN when the Cisco IOS software sends an Alert to the host. To specify names for all attached LANs, use the following command in interface configuration mode:

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>Router(config-if)# x25 address x121-address</td>
<td>Defines the name of an attached LAN.</td>
</tr>
</tbody>
</table>

Specifying the Physical Location of the Router

You can specify the physical location of the router if you intend requesting vital product information from the router. To specify the physical location, use the following command in interface configuration mode:

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>Router(config-if)# location location-description</td>
<td>Defines the physical location of the router.</td>
</tr>
</tbody>
</table>

Monitoring and Maintaining DSPU and SNA Service Point Feature Status

You can monitor the status of the DSPU and SNA Service Point features. To display information about the state of the DSPU and SNA Service Point features, use the following commands in privileged EXEC mode:

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>Router# show dspu</td>
<td>Shows the status of all DSPU resources.</td>
</tr>
<tr>
<td>Router# show dspu pu (host-name</td>
<td>pu-name) [all]</td>
</tr>
<tr>
<td>Router# show dspu pool pool-name [all]</td>
<td>Shows the status of a DSPU pool.</td>
</tr>
</tbody>
</table>
To control the reporting of DSPU notification events (DSPU-specific SNMP Traps and Unsolicited SNA Messages to Operator), use the following command in global configuration mode:

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>Router# show sna</td>
<td>Shows the status of all SNA hosts.</td>
</tr>
<tr>
<td>Router# show sna pu host-name [all]</td>
<td>Shows the status of an SNA host.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>Router(config)# dspu notification-level {off</td>
<td>low</td>
</tr>
</tbody>
</table>

### DSPU and SNA Service Point Configuration Examples

The following sections provide DSPU and SNA Service Point configuration examples:

- Dedicated LU Routing Example, page 22
- Pooled LU Routing Example, page 23
- Upstream Host via RSRB DSPU Configuration Example, page 24
- DSPU over DLSw+ using Virtual Data-Link Control Configuration Example, page 24
- Downstream PU via SDLC DSPU Configuration Example, page 25
- Upstream Host via SDLC DSPU Configuration Example, page 25
- Downstream PU via QLLC/X.25 DSPU Configuration Example, page 26
- Upstream Host via Frame Relay DSPU Configuration Example, page 26
- DSPU NCIA Configuration Example, page 27
- SNA Service Point Support Configuration Example, page 27
- SNA Service Point over DLSw+ Using Virtual Data-Link Control Configuration Example, page 28

### Dedicated LU Routing Example

Figure 3 illustrates the use of dedicated LU routing. Each upstream host LU is dedicated for use by a specific downstream LU.
Figure 3  Dedicated LU Routing

The following is a configuration file for the dedicated LU routing shown in Figure 3:

dspu host ciscohost xid-snd 06500001 rmac 4000.3745.0001
dspu pu ciscopu-a xid-rcv 05D00001 rmac 1000.5AED.0001
dspu lu 1 2 host ciscohost 2
dspu lu 3 3 host ciscohost 20
dspu pu ciscopu-b xid-rcv 05D00002 rmac 1000.5AED.0002
dspu lu 1 2 host ciscohost 4
dspu lu 3 3 host ciscohost 21

Pooled LU Routing Example

Figure 4  Pooled LU Routing

The following is a configuration file for the pooled LU routing shown in Figure 4:

ciscoHost

2 3 4..........................20 21

ciscoPU-A

1 2 3

ciscoPU-B

1 2 3

LU Pool

1 2 3
ciscoPU-C

The following is a configuration file for the pooled LU routing shown in Figure 4:
Upstream Host via RSRB DSPU Configuration Example

The following configuration example represents one possible definition for the network topology shown in Figure 3. This example demonstrates the configuration of an upstream host via RSRB (with local acknowledgment) and downstream PUs via Token Ring.

```
source-bridge ring-group 99
source-bridge remote-peer 99 tcp 150.10.13.1
source-bridge remote-peer 99 tcp 150.10.13.2 local-ack

dspu rsrb 88 1 99 4000.fffd.0001

dspu host ciscohost xid-snd 06500001 rmac 4000.3172.0001 rsap 4 lsap 4

dspu rsrb enable-host lsap 4

dspu host ciscohost xid-snd 06500001 rmac 4000.3172.0001 rsap 4 lsap 4

dspu pool ciscopool host ciscohost lu 2 8

dspu rsrb start ciscohost

dspu pu ciscopu1 xid-rcv 05d00001

dspu pu ciscopu2 xid-rcv 05d00002

dspu pu ciscopu3 xid-rcv 05d00003

dspu pu ciscopu4 xid-rcv 05d00004

dspu lu 2 3 host ciscohost 9

interface tokenring 0
description tokenring connection for downstream PUs
ing-speed 16
dspu enable-pu lsap 8
```

DSPU over DLSw+ using Virtual Data-Link Control Configuration Example

The following example illustrates pooled LU routing over DLSw+ using virtual data-link control:

```
source-bridge ring-group 99

dlaw local-peer peer-id 150.10.16.2
dlaw remote-peer 0 tcp 150.10.16.1

dspu vdlc 99 4000.4500.01f0

dspu vdlc enable-pu lsap 8

dspu vdlc enable-host lsap 12

dspu host HOST-B xid-snd 065bb0001 rmac 4000.7000.01f1 lsap 4 lsap 12 focalpoint

dspu pool pool-B host HOST-B lu 2 254
```
Downstream PU via SDLC DSPU Configuration Example

The following example demonstrates the configuration of downstream PUs via SDLC and an upstream host via Token Ring:

dspu host ciscohost xid-snd 06500001 rmac 4000.3172.0001 rsap 4 lsap 12
dspu pool ciscopool host ciscohost lu 2 11
!
dspu pu pu-sdlc0 sdlc C1 interface serial 0
dspu lu 2 6 pool ciscopool
!
dspu pu pu-sdlc1 sdlc C1 interface serial 1
dspu lu 2 6 pool ciscopool
!
interface serial 0
description SDLC connection for pu-sdlc0
encapsulation sdlc
sdlc role primary
sdlc address C1
dspu enable-pu sdlc C1
clockrate 56000
!
interface serial 1
description SDLC connection for pu-sdlc1
encapsulation sdlc
sdlc role primary
sdlc address C1
dspu enable-pu sdlc C1
clockrate 56000
!
interface tokenring 0
description Token Ring connection for ciscohost
ring-speed 16
dspu enable-host lsap 12
dspu start ciscohost

Upstream Host via SDLC DSPU Configuration Example

The following example demonstrates the configuration of an upstream host via SDLC and downstream PUs via Token Ring and Ethernet:

dspu host ciscohost xid-snd 06500001 sdlc C1 interface serial 0
dspu pool ciscopool host ciscohost lu 2 11
!
Configuring DSPU and SNA Service Point Support

DSPU and SNA Service Point Configuration Examples

```
dspu pu pu-token rmac 4000.4444.0001 rsap 4 lsap 8

dspu pu 2 6 pool ciscopool
!

dspu pu pu-ether rmac 0200.2222.0001 rsap 4 lsap 8

dspu pu 2 6 pool ciscopool
!
```

```
interface serial 0
   description SDLC connection for ciscohost
   encapsulation sdlc
   sdlc role secondary
   sdlc address C1
   dspu enable-host sdlc C1
   clockrate 56000
   dspu start ciscohost
!
```

```
interface tokenring 0
   description tokenring connection for pu-token
   ring-speed 16
   dspu enable-pu lsap 8
!
```

```
interface ethernet 0
   description Ethernet connection for pu-ether
   dspu enable-pu lsap 8
```

Downstream PU via QLLC/X.25 DSPU Configuration Example

The following example demonstrates the configuration of a downstream PU via QLLC/X.25 and upstream host via Ethernet:

```
x25 routing
!

dspu host ciscohost xid-snd 06500001 rmac 0200.2222.0001 rsap 4 lsap 12

dspu pool ciscopool host ciscohost lu 2 11
!

dspu pu pu-qllc x25 320108 qllc 08

dspu lu 2 11 pool ciscopool
!
```

```
interface serial 0
   description QLLC connection for pu-qllc
   encapsulation x25
   x25 address 3202
   x25 map qllc 320108
   dspu enable-pu qllc 8
!
```

```
interface ethernet 0
   description Ethernet connection for pu-ether
   dspu enable-host lsap 12
   dspu start ciscohost
!
```

```
x25 route ^3202.* alias serial 0
```

Upstream Host via Frame Relay DSPU Configuration Example

The following example demonstrates the configuration of an upstream host via Frame Relay and downstream PUs via Token Ring and Ethernet:

```
dspu host ciscohost xid-snd 06500001 dlci 200 rsap 4 lsap 12

dspu pool ciscopool host ciscohost lu 2 11
!

dspu pu pu-token rmac 4000.4444.0001 rsap 4 lsap 8
```
Configuring DSPU and SNA Service Point Support

DSPU and SNA Service Point Configuration Examples

```
dsru lu 2 6 pool ciscopool
! 
dsru pu pu-ether rmac 0200.2222.0001 rsap 4 lsap 8

dsru lu 2 6 pool ciscopool
!
interface serial 0
description Frame Relay connection for ciscohost
enapsulation frame-relay ietf
frame-relay map llc2 200

dsru enable-host lsap 12

dsru start ciscohost
!
interface tokenring 0
description tokenring connection for pu-token
ring-speed 16

dsru enable-pu lsap 8
!
interface ethernet 0
description Ethernet connection for pu-ether

dsru enable-pu lsap 8
```

**DSPU NCIA Configuration Example**

The following example illustrates an NCIA client/server session using DSPU:

```
cnia server 1 10.2.20.4 4000.3745.0001 1000.0000.0001 128
!

dsru ncia 1

dsru ncia enable-pu lsap 8
!

dsru host HOST-9370 xid-snd 11100001 rmac 4000.1060.1000 rsap 4 lsap 4
!

dsru pu CISCOPU-A xid-rcv 01700001

dsru lu 2 6 host HOST-9370 2
!

dpni tokenring 0
ring-speed 16
llc2 xid-retry-time 0

dsru enable-host lsap 4

dsru start HOST-9370
```

**SNA Service Point Support Configuration Example**

The following is an example of an RSRB configuration that implements SNA Service Point:

```
source-bridge ring-group 99
source-bridge remote-peer 99 tcp 150.10.13.2 local-ack
!
sna rsrb 88 1 99 4000.ffff.0001
!
sna host CNM02 xid-snd 05db000 rmac 4001.3745.1088 rsap 4 lsap 4 focalpoint
sna rsrb enable-host lsap 4
sna rsrb start CNM02
```
SNA Service Point over DLSw+ Using Virtual Data-Link Control Configuration Example

The following is an example of an SNA Service Point configuration that uses virtual data-link control over DLSw+:

source-bridge ring-group 99
dlw local-peer peer-id 150.10.16.2
dlw remote-peer 0 tcp 150.10.16.1
dlw vdlc 99 4000.4500.01f0
dlw vdlc enable-host lsap 12
!
sna host HOST-B xid-snd 065bmmm0 rmac 4000.7000.01f1 rsap 4 lsap 12 focalpoint
!
sna vdlc start HOST-B
!
interface serial 3
description IP connection to dspu7k
ip address 150.10.16.2 255.255.255.0
clockrate 4000000