

# Troubleshooting DLSw: SDLC

Document ID: 17566

## Contents

### Introduction

#### Prerequisites

- Requirements
- Components Used
- Conventions

#### Troubleshoot SDLC

- PU Type
- Common SDLC Issues
- Sample Session Establishment Flows for a PU 2.0 Device
- Sample Session Establishment Flows for a PU 2.1 Device
- Debug SDLC Events or Packets
- SDLC Packets During DLSw with SDLC for PU 2.1

#### Related Information

## Introduction

This document helps you troubleshoot problems that may occur in a network when a Synchronous Data Link Control (SDLC)-attached end device connects to a data center, for example, over Data-Link Switching (DLSw).

## Prerequisites

### Requirements

There are no specific requirements for this document.

### Components Used

This document is not restricted to specific software or hardware versions.

The information in this document was created from the devices in a specific lab environment. All of the devices used in this document started with a cleared (default) configuration. If your network is live, make sure that you understand the potential impact of any command.

### Conventions

For more information on document conventions, refer to the Cisco Technical Tips Conventions.

## Troubleshoot SDLC

Issue the **show interface serial x** command at the router to begin to troubleshoot SDLC. The output of this command contains information that might help you to locate the problem.

```
Serial1/0 is up, line protocol is up  
  
!--- If line is down/down, then check CLOCKING.
```

*!--- If line is up/down, then check NRZI\_ENCODING.  
!--- If line is cycling between up/up and up/down, then check DUPLEX.  
!--- A modem sharing device (MSD) uses full duplex.*

```
Hardware is CD2430 in sync mode
Description SDLC PU2.1 PRIMARY
MTU 1500 bytes, BW 128 Kbit, DLY 20000 usec, rely 255/255, load 1/255
Encapsulation SDLC, loopback not set
Router link station role: PRIMARY (DCE)
```

*!--- DCE has to provide the clock. It is responsible for raising DCD, CTS,  
!--- and DSR. Issue the **show controllers** command to check DTE, DCE, and  
!--- cable type.*

```
Router link station metrics:
  slow-poll 10 seconds
  T1 (reply time out) 3000 milliseconds
```

*!--- The **sdlc t1 <milliseconds>** command sets the amount of time waited  
!--- for an acknowledgement to an SDLC frame, where <milliseconds> is a  
!--- numeric value in milliseconds between 1 and 64000 (default is 3000).*

```
N1 (max frame size) 12016 bits
```

*!--- The **sdlc n1 <bit-count>** commands sets the maximum size of an  
!--- incoming frame, where <bit-count> is a numeric value from 1 to 12000  
!--- (default is 12000).*

```
N2 (retry count) 20
```

*!--- The **sdlc n2 <retry-count>** command sets the number of times that an  
!--- SDLC frame is sent before the session is terminated, where <retry-count>  
!--- is a numeric value between 1 and 255 (default is 20).*

```
poll-pause-timer 200 milliseconds
```

*!--- Set this with the **sdlc poll-pause-timer <milliseconds>** command,  
!--- where <milliseconds> is a numeric value in milliseconds from 1 to 10000.  
!--- Set this value to a minimum of 2000 before you run SDLC debugs; otherwise,  
!--- you will flood the console with SDLC polling messages.*

```
poll-limit-value 1
```

*!--- Set this with the **sdlc poll-limit-value <count>** command, where <count>  
!--- is a numeric value from 1 to 10.  
!--- Use this command on multidrops to determine the number of polls that are  
!--- dedicated to each secondary device. Higher value allows a single secondary  
!--- to send more data but can decrease overall secondary servicing efficiency.*

```
k (window-size) 1
modulo 8
```

*!--- Set K with the **sdlc k <window-size>** command, where <window-size> is a  
!--- numeric value of 1 through 7 (if modulo 7) or 1 through 127 (if modulo 128).  
!--- rrrz sss0*

*!--- rrr = Frame number of the block that is expected to be received next  
!--- (rrrrrrr if modulo 128)*

*!--- z = Poll/Final bit, which may be 0 or 1.*

*!--- sss = Frame number of the block that is expected to be sent next  
!--- (sssssss if modulo 128)*

*!--- The K value determines how many frames after which the poll bit is set to 1,  
!--- which indicates that it is the other side???'s turn to send.*

```
sdhc vmac: 4000.1555.21--
sdhc addr 01 state is CONNECT
```

*!--- Refer to SDLC States*

cls\_state is CLS\_IN\_SESSION

*!--- See Table 1 ??? CLS States.*

VS 6, VR 6, Remote VR 6, Current retransmit count 0  
Hold queue: 0/200 IFRAMEs 2649/683  
TESTs 0/0 XIDs 0/0, DMs 0/0 FRMRs 0/0

*!--- FRMRs could indicate a bug in the end station SDLC emulation package.  
!--- Check the values in the FRMR frame against the FRMR frame description.*

RNRs 1797153/2291 SNRMs 222/0 DISC/RDs 12/0 REJs 0/0

*!--- If you see a steady increase in RNRs, then check for congestion on the DLSw  
!--- peer (the value under the TCP column in **show dlsw peer** command output).  
!--- If RNRs are greater than 50 percent of the default TCP queue depth 200, then  
!--- there is congestion.*

Poll: clear, Poll count: 0, ready for poll, chain: 01/01  
Last input 00:00:00, output 00:00:00, output hang never  
Last clearing of "show interface" counters never  
Queueing strategy: fifo  
Output queue 0/40, 0 drops; input queue 0/75, 0 drops

*!--- Check that the input and output queues are not wedged (41/40 or 76/75).  
!--- If the queue is wedged, then the router usually must be reloaded to recover.*

5 minute input rate 0 bits/sec, 4 packets/sec  
5 minute output rate 0 bits/sec, 4 packets/sec  
2857443 packets input, 5738306 bytes, 0 no buffer  
Received 409483 broadcasts, 0 runts, 0 giants, 0 throttles  
1 input errors, 0 CRC, 0 frame, 0 overrun, 0 ignored, 1 abort

*!--- Giants and input errors might indicate a wrong NRZI value (NRZI-ENCODING).*

2857874 packets output, 6029620 bytes, 0 underruns  
0 output errors, 0 collisions, 60523 interface resets  
0 output buffer failures, 0 output buffers swapped out  
53 carrier transitions  
DCD=up DSR=up DTR=up RTS=down CTS=up

*!--- RTS and CTS are always up, with full duplex.  
!--- RTS and CTS will cycle between up and down, with half duplex.*

**State Meaning** CLS\_STN\_CLOSED No line activation process has started yet. CLS\_ROSCNF\_PEND ReqOpenStn has been sent to PU; awaiting ReqOpenStnCfm. CLS\_STN\_OPENED ReqOpenStnCfm received from PU. CLS\_CONNECT\_RSP\_PEND Sent SNRM; awaiting UA from PU.

CLS\_DISCCNF\_PEND PU sends DISC (if primary) or RDISC (if secondary).

CLS\_CONNECT\_REQ\_PEND Awaiting a connect response. CLS\_FULL\_XID\_PEND Awaiting a response to the Null XID that was sent. CLS\_CONNECTED\_IND\_PEND Connect.Rsp received from DLU.

CLS\_DISC\_IND\_SENT Disconnect.Ind has been sent. CLS\_IN\_SESSION Circuit establishment has completed. CLS\_CLOSING Cisco Link Services (CLS) is in a closing state.

## PU Type

For SDLC-attached controllers, it is important to know the Physical Unit (PU) type that is being used (for example, PU 2.0 or PU 2.1) and the SDLC role.

Table 2 shows some of the most common devices and the PU type that they represent. The PU type determines the configuration that should be adopted, as illustrated in the PU 2 with SDLC Station Role Set to Secondary section.

**Device PU Type** 5294 1 5394 1 5394 +RPQ 8Q0775 2.1 5494 2.1 3276 2.0 3274 2.0 3174 2.0 / 2.1 3745 4  
3172 No PU XCA node S/38 2.0 36XX 2.0 Netware/SAA 2.0 / 2.1 SNA Server NT 2.0 / 2.1

### PU 2 with SDLC Station Role Set to Secondary

```
interface serial x

encapsulation sdhc
sdhc role primary

!--- Assumes SDLC station role secondary for the attached SDLC controller.

sdhc vmac 1234.3174.0000

!--- Virtual MAC address given to the SDLC controller, which has the
!--- SDLC address (D2) appended to it.

!--- For more information about the sdhc vmac command, refer to
!--- LLC2 and SDLC Commands.

sdhc address D2

!--- SDLC address obtained from SDLC controller configuration.

sdhc xid D2 01730020

!--- D2 is the SDLC address, and 01730020 is the IDBLK and IDNUM, which is
!--- obtained from the Switched Major Node on the host.

sdhc partner 1000.5aed.1f53 D2

!--- 1000.5aed.1f53 is the MAC address of the host, and D2 is the SDLC address.

sdhc dlsw D2
```

### PU 2 with SDLC Station Role Set to Primary

```
interface serial x

sdhc role secondary
sdhc vmac 1234.3174.0000
sdhc address D2
sdhc xid D2 01730020
sdhc partner 1000.5aed.1f53 D2
sdhc dlsw D2
```

### Node Type 2.1 with SDLC Station Role Set to Negotiable or Primary

```
interface serial x

encapsulation sdhc
sdhc role none
sdhc vmac 1234.3174.0000
sdhc address D2
sdhc partner 1000.5aed.1f53 D2
sdhc dlsw D2
```

## Node Type 2.1 with SDLC Station Role Set to Secondary

```
interface serial x

encapsulation sdlc
sdlc role prim-xid-poll
sdlc vmac 1234.3174.0000
sdlc address D2
sdlc partner 1000.5aed.1f53 D2
sdlc dlsw D2
```

**Note:** For multidrop SDLC for PU 2.0 or PU 2.1, and a combination of PU 2.0 and PU 2.1, refer to the DLSw+ with SDLC Multidrop Support Configuration Example section of Configuring Data-Link Switching Plus.

## PU 4.0 with SDLC

```
interface serial x

no ip address
encapsulation sdlc
no keepalive
clock rate 19200
sdlc vmac 4000.3745.0100
sdlc address 01 seconly
sdlc partner 4000.3745.2176 01
sdlc dlsw 1
```

For more information on SDLC to Logical Link Control, type 2 (LLC2) conversion for Format Indicator 4 (FID4) frames, refer to DLSw+ FID4 LLC2-to-SDLC Conversion for PU4/5 Devices.

There is a direct relationship between Cisco Link Services and SDLC. For Cisco Link Services, no changes occur until the Set Normal Response Mode (SNRM) is acknowledged by an Unnumbered Acknowledgment (UA). Once a UA is obtained, the router sends a Receiver Not Ready (RNR, USBUSY) to the SDLC station, to keep it quiescent while DLSw brings up the DLSw circuit with the host (SDLC role primary). The SDLC code sends a null Exchange Identification (XID) internally to the Cisco Link Services code, to initiate this. These Cisco Link Services states can be seen:

- **CLS\_STN\_CLOSED**??The CANUREACH explorer (CUR-ex) is sent to the DLSw peer, but an ICANREACH explorer (ICR-ex) response is not received yet. The problem is probably an incorrect MAC address, or the host adapter is not open or active.
- **CLS\_STN\_OPENED**??A null XID is sent but receives no response from the host. The problem is probably an incorrect destination Service Access Point (SAP), or no logical lines are available.
- **CLS\_CONNECT\_REQ\_PEND**??A Systems Network Architecture (SNA) XID is sent, and there is no response from the host. The problem is probably a Switched Major Node that is incorrect, not active, or activated by another device.

## Common SDLC Issues

This section lists some of the most common SDLC issues.

- Wrong SDLC address.

For more information about **sdlc address**, refer to LLC2 and SDLC Commands.

- Incorrect encoding: Non-Return to Zero (NRZ) or Non-Return to Zero Inverted (NRZI).

For more information about **nrzi-encoding**, refer to Synchronous Serial Port Setup Commands.

- Turned-off or broken SDLC station.
- The DCE sends a DSR instead of a Data Carrier Detect (DCD) signal (the router serial interface is operating in DTE mode).
- Missing **clock rate** interface command.

For more information about the **clock rate** command, refer to the Interface Commands.

- The DTE is not raising a Data Terminal Ready (DTR) signal (the router serial interface is operating in DCE mode).
- Full-duplex or half-duplex operation.

Refer to the Configure an SDLC Interface for Half-Duplex Mode section in Configuring LLC2 and SDLC Parameters.

- Incorrect cable pinouts.

For more information about cable pinouts, refer to Hardware Specifications and Cable Pinouts .

- The cable length limit is exceeded.

Refer to the Distance Limitations for Interface Cables section in Planning Your Installation.

- Incorrect SDLC station role.

See the PU Type section in this document.

## Wrong SDLC Address

The SDLC address that is configured on the router needs to match the SDLC address of the attached SDLC controller. For example, with a 3174 cluster controller, this is configuration line number 104. If the router is configured for SDLC role primary, and the SDLC state is stuck in `SNRMSSENT`, then it is possible that the two addresses do not match. A useful command to issue to test the SDLC line and the controller is **sdlc test serial**; refer to **sdlc test serial** in LLC2 and SDLC Commands. Similar to **IP ping**, it sends out ten test frames; if all ten are received, then the test is considered a `??pass.??` This test also verifies that you have the correct encoding (NRZ or NRZI); refer to **nrzi-encoding** in Synchronous Serial Port Setup Commands. Similar to the SDLC address parameter, the encoding needs to match on the router serial interface and on the SDLC controller. In the example of a 3174, this is configuration line number 313: 0 means NRZ, and 1 means NRZI. The default on the router is 0 (NRZ).

## DCE Sends a DSR Instead of a DCD Signal

Another common SDLC issue is the use of DCE or DTE, and clocking issues. Typically, the Cisco router provides the clocking and has a DCE cable connected. This makes the router serial interface act as a DCE and makes the attached controller act as a DTE. This setup can also be reversed: the router serial interface has a DTE cable connected and the attached controller provides the clock. By default, when the serial interface operates in DTE mode, it monitors the DCD signal as the line-up-or-down indicator. Typically, the attached DCE device sends the DCD signal. When the DTE interface detects the DCD signal, it changes the state of the interface to up. In some configurations, such as an SDLC multidrop environment, the DCE device sends the DSR signal instead of the DCD signal, which does not allow the interface to come up. To get the interface to monitor the DSR signal instead of the DCD signal as the line-up-or-down indicator, issue the **ignore-dcd** command in interface configuration mode. Refer to **ignore-dcd** in Synchronous Serial Port Setup Commands.

## DTE Is Not Raising a DTR Signal

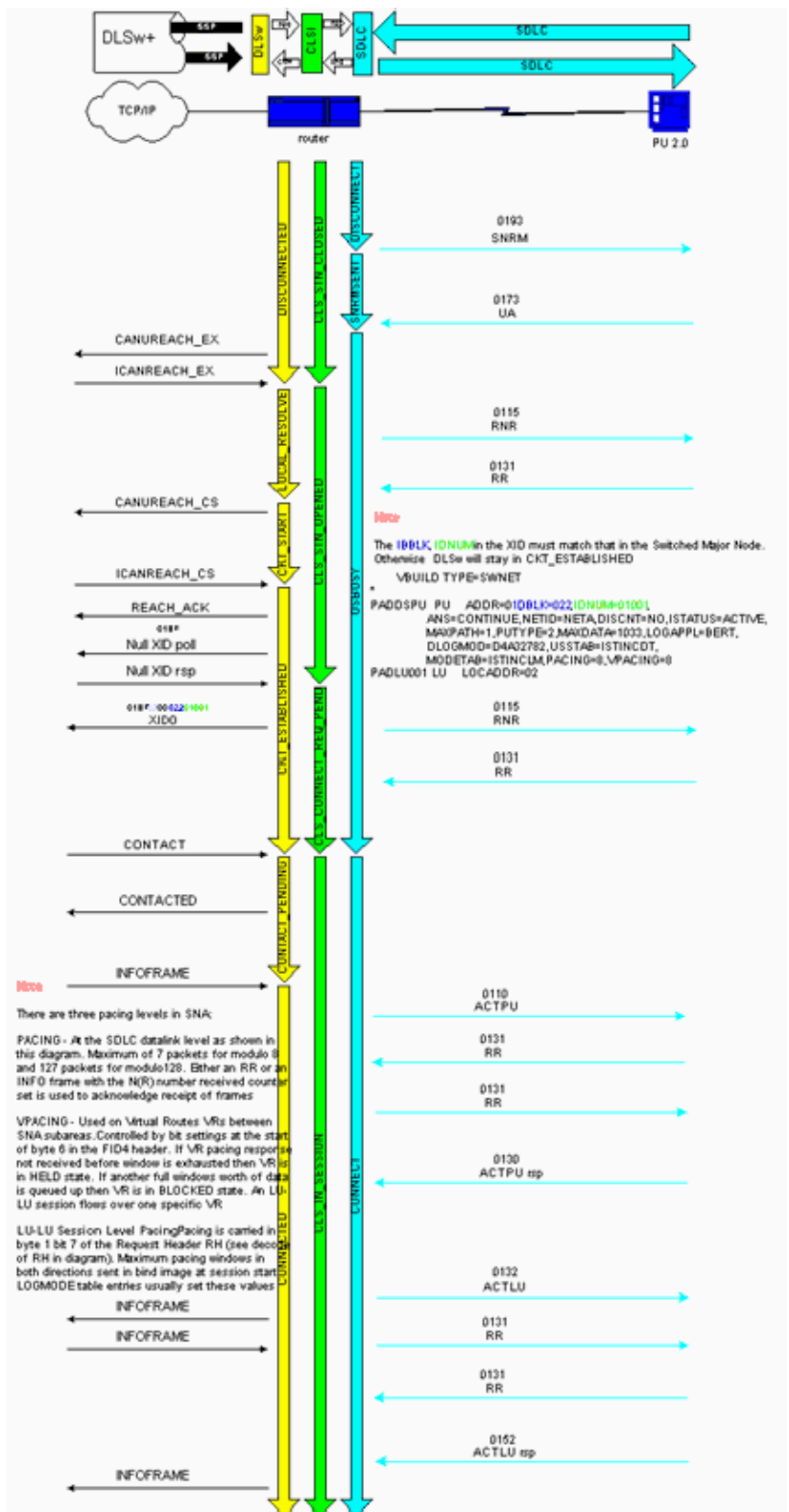
When the router serial interface acts as a DCE, one possible issue might be a failure of the DTE to raise the DTR signal. This can be verified by the last line of display output from the **show interface** command. The problem might be due to bad cabling, due to an incorrect pinout (refer to Hardware Specifications and Cable Pinouts), or due to the SDLC controller failing to power up properly. Use a break-out box to verify all signals

from both the DCE and the DTE side. To determine the type of cable that is attached to the router serial interface, issue the **show controllers serial** command. Refer to **show controllers serial** in the Interface Commands.

### **Full-Duplex or Half-Duplex Operation**

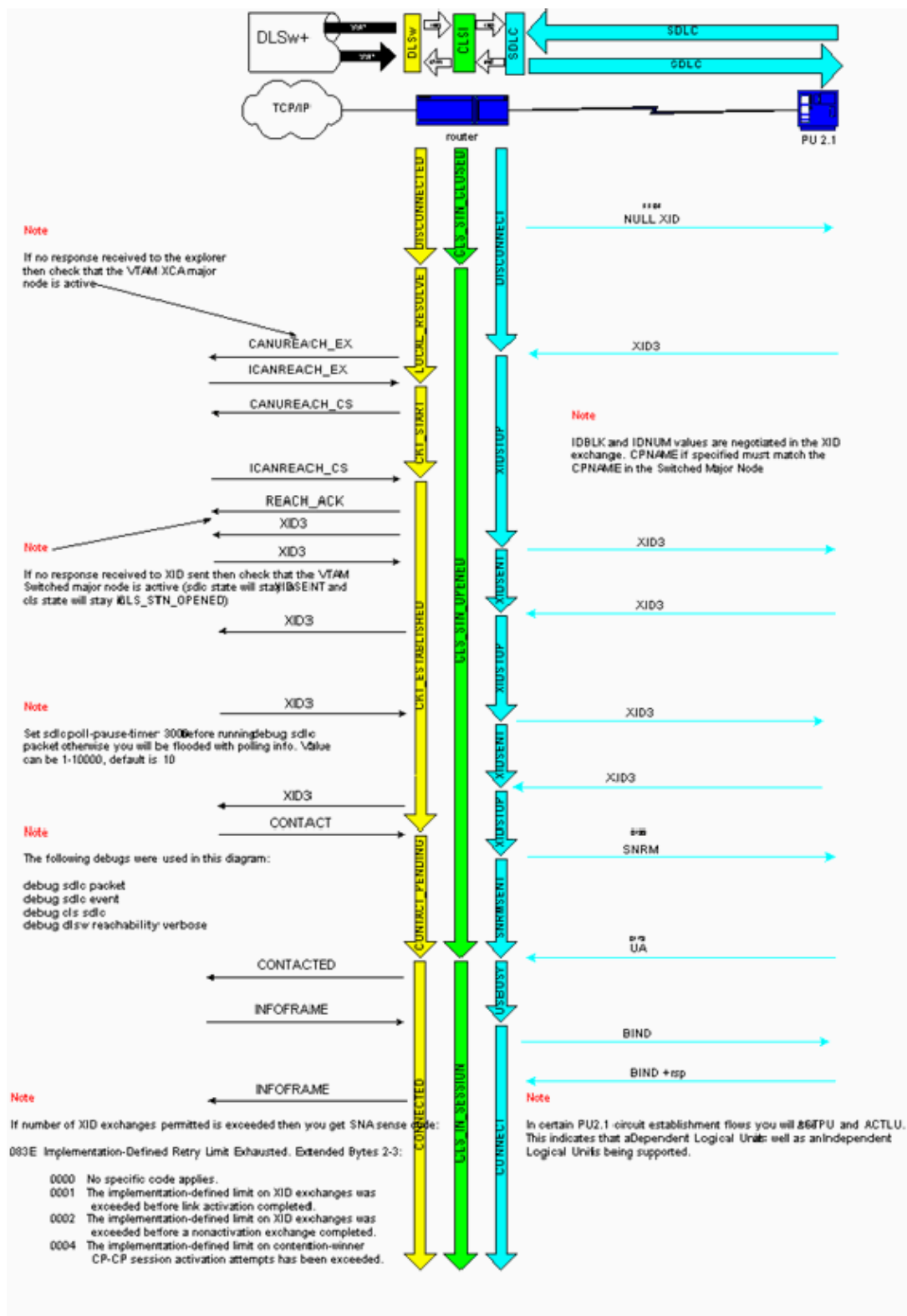
The duplex speed is another common culprit in SDLC connections. The router interface and the SDLC controller need to have identical duplex speed settings; either half or full. For example, with a 3174 cluster controller, this is configuration line number 318: 0 means full-duplex speed, and 1 means half-duplex speed. The router serial interface defaults to full duplex. If the router is connected to a Modem Sharing Device (MSD), the router serial interface and the MSD should run full duplex. Refer to the Configure an SDLC Interface for Half-Duplex Mode section in Configuring LLC2 and SDLC Parameters.

### **Sample Session Establishment Flows for a PU 2.0 Device**



Sample Session Establishment Flows for a PU 2.1 Device





## Debug SDLC Events or Packets

The most common **debug** commands for SDLC are **debug sdcic event** and **debug sdcic packet**. They can be used when an SDLC analyzer is not available and a quick diagnosis is required. If you have multiple SDLC addresses configured, you may get **debug** output for all addresses. Use **debug sdcic event**, which shows every packet, rather than **debug sdcic packet**, which shows only events.

**Note:** If you have multiple SDLC serial interfaces, the command generates debugs from all SDLC-configured

interfaces.

To limit output to just one interface, issue these commands:

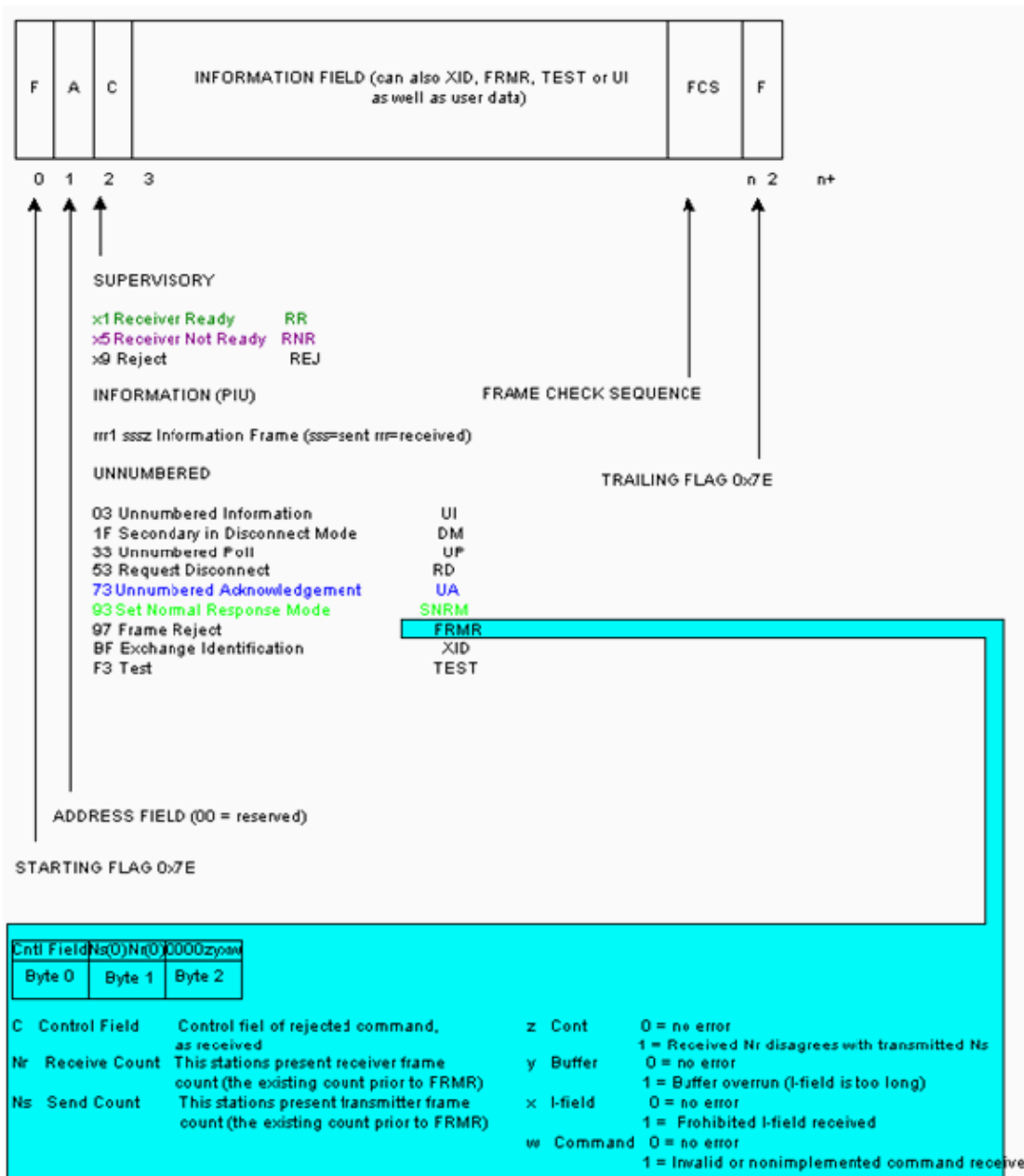
- **debug list serial** *x* , where *x* is the interface number
- **debug sdlc event**

Do not issue the **debug sdlc packet** command, because it bypasses the filter.

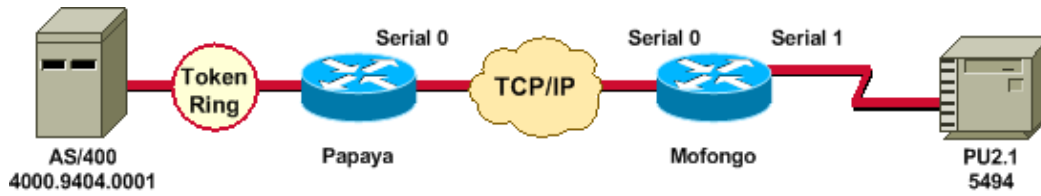


**Caution:** The **debug sdlc** command can cause severe performance degradation, especially when issued on a router that has multiple SDLC addresses configured. Before you attempt this **debug** command, refer to Important Information on Debug Commands.

### SDLC Frame Formats



# SDLC Packets During DLSw with SDLC for PU 2.1



```
%LINK-3-UPDOWN: Interface Serial1, changed state to up
```

After you issue the **debug sdhc packet** command, these events occur:

1. An **XID**, or **BF**, is sent to **SDLC** broadcast address **FF**.

```
Serial1 SDLC output???????? FFBF
```

2. An **XID** is issued from the 5494. This is an **XID** format 3 type 2, which you can issue the **debug sdhc packet** command to view.

```
Serial1 SDLC input
0046C930: DDBF3244 073000DD 0000B084 00000000?? .....d....
0046C940: 00000001 0B000004 09000000 00070010?? .....
0046C950: 17001611 01130012 F5F4F9F4 F0F0F2F0?? .....54940020
0046C960: F0F0F0F0 F0F0F0F0 0E0CF4D5 C5E3C14B?? 00000000..4NETA.
0046C970: C3D7F5F4 F9F4???????????????????????????????????????? CP5494
```

Though this document does not provide the full details necessary to parse this **XID**, this is a description of some of the fields:

- ◆ 073000DD???The block ID and the ID number that is configured in the 5494. Together, they are referred to as the **XID**, and they are sent by the 5494 to the peer, during session negotiation.
- ◆ NETA???The Advanced Peer-to-Peer Networking (APPN) Network Identifier (NETID) that is being used. It should normally match the NETID that is configured in the peer. In this case, the peer is an AS/400.
- ◆ CP5494???The control point name of the 5494.

3. The **XID** is issued from the AS/400.

```
Serial1 SDLC output
004BC070:???????? FFBF 324C0564 52530000 000A0800???????? ...<.....
004BC080: 00000000 00010B30 0005BA00 00000007?? .....
004BC090: 000E0DF4 D5C5E3C1 4BD9E3D7 F4F0F0C1?? ..4NETA.RTP400A
004BC0A0: 1017F116 11011300 11F9F4F0 F4C6F2F5?? ..1.....9404F25
004BC0B0: F1F0F0F0 F4F5F2F5 F3460505 80000000?? 100045253.....
004BC0C0:
SERIAL1 SDLC INPUT
0046C270:???????????????????????????????????????? DDBF3244 073000DD????????????????????????? ..
0046C280: 0000B084 00000000 00000001 0B000004?? ...D.....
0046C290: 09000000 00070010 17001611 01130012?? .....
0046C2A0: F5F4F9F4 F0F0F2F0 F0F0F0F0 F0F0F0F0?? 5494002000000000
0046C2B0: 0E0CF4D5 C5E3C14B C3D7F5F4 F9F4????????????? ..4NETA.CP5494
SERIAL1 SDLC OUTPUT
004C0B10:???????? FFBF 324C0564 52530000 00F6C800???????? ...<.....6H.
004C0B20: 00000080 15010B10 0005BA00 00000007?? .....
004C0B30: 000E0DF4 D5C5E3C1 4BD9E3D7 F4F0F0C1?? ..4NETA.RTP400A
004C0B40: 1017F116 11011300 11F9F4F0 F4C6F2F5?? ..1.....9404F25
004C0B50: F1F0F0F0 F4F5F2F5 F3460505 80150000?? 100045253.....
004C0B60:
SERIAL1 SDLC INPUT
0046BBC0: DDBF3244 073000DD 0000B084 00000000?? .....D....
```

```
0046BBD0: 00000001 0B000004 09000000 00070010?? .....
0046BBE0: 17001611 01130012 F5F4F9F4 F0F0F2F0?? .....54940020
0046BBF0: F0F0F0F0 F0F0F0F0 0E0CF4D5 C5E3C14B?? 00000000..4NETA.
0046BC00: C3D7F5F4 F9F4???????????????????????????????????????? CP5494
```

- ◆ 05645253??The block ID and the ID number of the AS/400.
- ◆ RTP400A??The control point name of the AS/400.

This can be found in the Display Network Attributes (DSPNETA) file on the AS/400.

4. The SNRM (93) and UA (73) can be seen on the line. Prior to the SNRM, the router always used the broadcast address. Henceforth, the router always uses the actual polling address of DD.

```
Serial1 SDLC output????????? DD93
Serial1 SDLC input???????????? DD73
Serial1 SDLC output????????? DD11
Serial1 SDLC input???????????? DD11
```

If you vary the controller off on the AS/400, you can see the DISC (53) and UA (73) that results on the SDLC side of the session.

```
Serial1 SDLC output DD53
Serial1 SDLC input
```

The rest of the debugs have been omitted.

## Related Information

- [DLSw and DLSw+ Support](#)
- [Technology Support](#)
- [Product Support](#)
- [Technical Support – Cisco Systems](#)