

SDLC-LLC 네트워크 미디어 변환 이해 및 문제 해결

목차

[소개](#)

[사전 요구 사항](#)

[요구 사항](#)

[사용되는 구성 요소](#)

[표기 규칙](#)

[SDLLC](#)

[SDLC 구성](#)

[SDLLC 구성](#)

[SDLLC 디버깅](#)

[DLSw 미디어 변환](#)

[show 명령](#)

[PU2.1용 DLSw/SDLC 중 SDLC 패킷 디버깅](#)

[DLSw 미디어 변환 예](#)

[DLSw 역미디어 변환 수행](#)

[로컬 DLSw 미디어 변환](#)

[관련 정보](#)

[소개](#)

이 문서에서는 SDLC(Synchronous Data Link Control)에서 LLC(Logical Link Control) 네트워크 미디어 변환을 이해하고 문제를 해결하는 데 필요한 정보를 제공합니다.

[사전 요구 사항](#)

[요구 사항](#)

이 문서에 대한 특정 요건이 없습니다.

[사용되는 구성 요소](#)

이 문서는 특정 소프트웨어 및 하드웨어 버전으로 한정되지 않습니다.

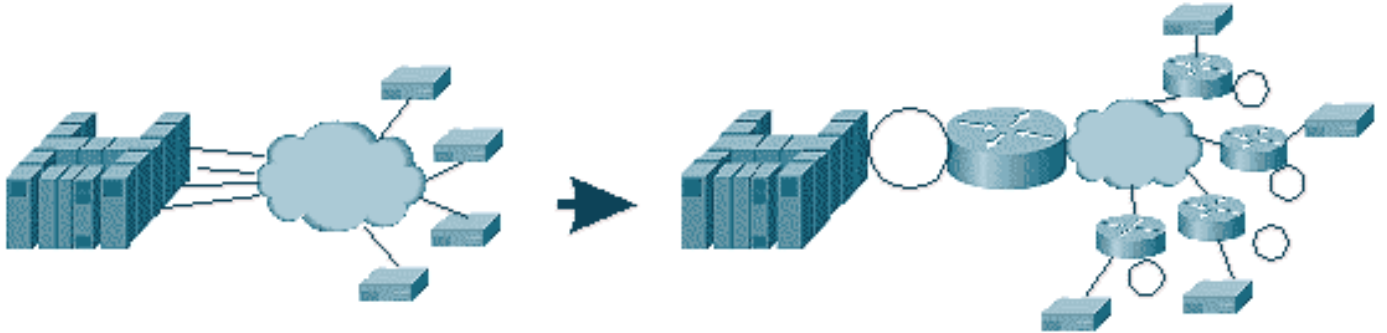
[표기 규칙](#)

문서 규칙에 대한 자세한 내용은 [Cisco 기술 팁 표기 규칙을 참고하십시오.](#)

SDLLC

SDLC-LAN 변환(SDLLC)은 PU2.0(Physical Unit 2) 디바이스의 SDLC 세션을 LLC2(Logical Link Control) 세션으로 변환하는 데 사용됩니다.이 기능은 많은 양의 원격 컨트롤러가 FIP(Front-End Processor)의 단일 토큰 링 포트에 공급되는 경우에 매우 유용합니다.

이 다이어그램의 왼쪽에는 원격 위치로 떠나는 많은 SDLC 회선이 있는 FAP가 표시됩니다.이 다이어그램의 오른쪽에는 Cisco 라우터와 동일한 시나리오가 표시됩니다.



라우터를 통해 FAP는 토큰 링 인터페이스만 가질 수 있습니다.이 시점부터 호스트에 대해 SDLLC를 수행하는 여러 원격 위치와 일반 SRB(source-route bridge) 트래픽이 있습니다.

참고: LLC에서 SDLC로의 변환에 SDLLC를 사용하는 것은 PU2.0 디바이스에만 적용되며 물리적 유닛 유형 2.1(PU2.1)에는 적용되지 않습니다.PU2.1은 DLSw(data-link switching)에서 지원됩니다.

SDLLC를 구성하려면 라우터에 SRB가 필요합니다.SRB를 [구성하는](#) 방법에 대한 자세한 내용은 [로컬 소스](#) 경로 브리징 이해 및 문제 해결을 참조하십시오.

SDLC 구성

SDLLC는 SDLC 인터페이스에서 변환되므로 먼저 SDLC를 올바르게 구성해야 합니다.SDLC를 구성하려면 다음 단계를 완료합니다.

1. SDLC로 직렬 캡슐화를 변경하려면 `encapsulation sdlc` 명령을 실행합니다.
2. `sdlc role primary` 명령을 실행하여 라우터의 역할을 SDLC 행에서 primary로 변경합니다.**참고:** STUN(Serial Tunneling) 환경에서는 기본 및 보조 역할이 있습니다.자세한 [내용은 STUN\(Serial Tunneling\) 구성 및 문제 해결](#)을 참조하십시오.
3. `sdlc address xx` 명령을 실행하여 SDLC 폴링 주소를 구성합니다.

SDLLC 구성

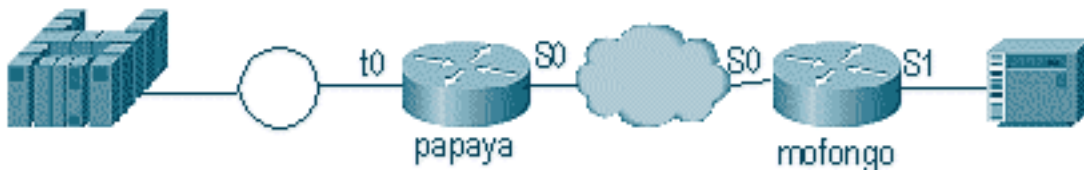
SDLLC를 구성하려면 첫 번째 명령이 `traddr`입니다.이 명령은 LLC2 환경에서 SDLC가 변환하는 내용을 정의합니다.SDLLC를 구성하려면 다음 단계를 완료합니다.

1. SDLLC `traddr xxxx.xxxx.xx00 lr bn tr` 명령을 실행하여 직렬 인터페이스에서 SDLLC 미디어 변환을 활성화합니다.이 명령은 라우터에 SDLC 스테이션의 가상 MAC 주소를 알려줍니다.그런 다음 명령은 로컬 벨소리 번호(lr), 브리지 번호(bn) 및 대상 벨소리 번호(tr)를 지정합니다. lr은 네트워크에서 고유해야 합니다.bn은 1에서 15 사이의 값일 수 있습니다. tr은 라우터의 가상 링이어야 합니다.로컬 SDLLC를 구성하는 경우 이 점을 라우터의 가상 링 또는 인터페이스(토큰 링 인터페이스에 연결된 물리적 링)로 설정할 수 있습니다.**참고:** 이 명령에서 MAC 주

소의 마지막 두 자릿수는 **00입니다**. 라우터가 이 숫자를 사용하여 이 줄의 SDLC 주소를 삽입하기 때문에 마지막 두 자리 숫자를 설정할 수 없습니다.마지막 두 숫자를 지정하면 라우터가 SDLC 주소로 바꿉니다.그러면 호스트가 해당 MAC 주소에 응답하지 않습니다.예를 들어 traddr MAC이 4000.1234.5678으로 구성되고 SDLC 주소가 0x01이면 라우터는 LLC 도메인에서 SDLC 디바이스를 나타내기 위해 4000.1234.5601의 MAC을 사용합니다.또한 traddr MAC은 토큰 링 프레임과 동일한 형식인 비정규 형식입니다.

2. `sdllc xid address xxxxxxxxxx` 명령을 실행하여 SDLC 스테이션에서 VTAM(Virtual Telecommunications Access Method) 값과 일치하는 XID(Exchange Identification) 값을 지정합니다.이는 VTAM의 스위치 주 노드의 IDBLK 및 IDNUM에서 결정됩니다.일치하지 않으면 XID 교환이 실패합니다.
3. `sdllc partner mac-address sdllc-address` 명령을 실행하여 SDLLC에 대한 연결을 활성화합니다.파트너의 MAC 주소(일반적으로 호스트)를 지정합니다.

간단한 SDLLC 샘플 컨피그레이션이 표시됩니다.SDLC 연결 컨트롤러는 FAP에 연결된 로컬 토큰 링 장치로 나타납니다.

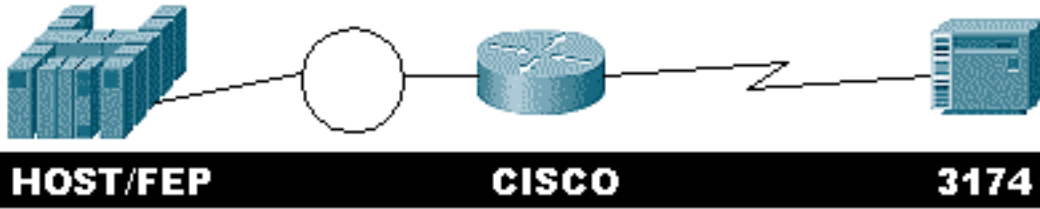


파파야	모풍고
<pre>source-bridge ring-group 100 source-bridge remote- peer 100 tcp 1.1.1.1 source-bridge remote- peer 100 tcp 1.1.2.1 local-ack interface tokenring 0 ip address 1.1.3.1 255.255.255.0 source-bridge 33 2 100 source-bridge spanning interface loopback 0 ip address 1.1.1.1 255.255.255.0</pre>	<pre>source-bridge ring group 100 source-bridge remote-peer 100 tcp 1.1.2.1 source-bridge remote-peer 100 tcp 1.1.1.1 local-ack source-bridge sdllc local-ack interface serial 0 encapsulation sdllc-primary sdllc address c6 sdllc traddr 4000.3174.1100 333 3 100 sdllc partner 4000.1111.1111 c1 sdllc xid c1 17200c6 interface loopback 0 ip address 1.1.2.1 255.255.255.0</pre>

SDLLC 디버깅

SDLLC 문제를 해결하려면 두 가지 다른 환경을 해결해야 합니다.SDLC World와 LLC2(Logical Link Control)를 사용하여 프레임을 변환할 수 있습니다.컨트롤러 유형은 하나만 사용할 수 있으므로 SDLLC를 디버깅하는 것이 DLSw(data-link switching)/SDLC보다 쉽습니다.

먼저 이 특정 세션 시작에 대한 플로우를 확인합니다.



컨트롤러에서 SNRM(Set Normal Response Mode) 응답을 확인합니다.SDLC 부분이 작동 및 실행 될 때까지 라우터는 LLC 부분을 시작하지 않습니다.

다음 명령을 실행하여 SNRM 응답을 확인합니다.

- **sdhc_상태**
- **sdllc_state**

이 예에서는 SNRM이 컨트롤러에 전송되며, 이 경우 회선 상태가 SNRMSSENT로 변경됩니다.라우터가 이 상태로 유지되면 컨트롤러에서 번호가 지정되지 않은 승인(UA)을 받지 못했습니다.이는 SDLC 라인에 문제가 있음을 의미할 수 있습니다.이 경우 디버그는 다음과 같이 표시됩니다.

```
%LINEPROTO-5-UPDOWN: Line protocol on Interface Serial1, changed state to up
s4f#
SDLLC_STATE: Serial1 C6 DISCONNECT
-> SDLC PRI WAIT
SDLC_STATE: (5234984) Serial1 C6 DISCONNECT
-> SNRMSSENT
%SYS-5-CONFIG_I: Configured from console by console
%LINK-3-UPDOWN: Interface Serial1, changed state to up
Serial1 SDLC output C693
Serial1 SDLC input C673
SDLC_STATE: (5235700) Serial1 C6 SNRMSSENT
-> CONNECT
SDLLC_STATE: Serial1 C6 SDLC PRI WAIT
-> NET UP WAIT
SDLC_STATE: (5235700) Serial1 C6 CONNECT
-> USBUSY
```

라우터가 UA를 수신하면 **sdhc_state**가 **SNRM_SENT**에서 **CONNECT**로 이동합니다.다음으로 **SDLLC** 상태가 **SDLC_PRI_WAIT**에서 **NET_UP_WAIT**로 이동합니다.이 경우 라우터는 연결의 LLC 측면을 시작할 수 있습니다.마지막 작업은 SDLC 행에 **RNR**(Receive Not Ready)을 보내기 시작하는 것입니다.이렇게 하면 LLC 측에서 작동할 때까지 컨트롤러가 정보를 보내지 않습니다.

그런 다음 라우터가 탐색기를 보내 파트너의 위치를 찾습니다.

```
SDLLC: O TEST, dst 4000.1111.1111 src 4000.3174.11c6 dsap 0 ssap 0
To0: out: MAC: acfc: 0x8040 Dst: 4000.1111.1111 Src: c000.3174.11c6 bf: 0x82 0x304A210
To0: out: RIF: 8800.14D3.0642.0210
To0: out: LLC: 0000F300 00800000 000C3BF0 7D000000 00800000 000C3BF0 ln: 25
SDLLC: NET UP WAIT      recv FORWARD TEST P/F(F3) 4000.3174.11c6 c000.1111.1111 00 01 -> Serial1
C6
caching rif
```

앞의 출력에는 전송 및 수신되는 테스트 폴링이 표시됩니다. 이 예에는 로컬에 연결된 컨트롤러와 토큰 링이 있으므로 테스트 폴링은 라우터에서 파트너 주소를 검색하도록 합니다. 라우터가 테스트 프레임을 수신한 후 XID 교환을 시작합니다. 라우터는 이 세션에 대한 RIF(Routing Information Field)를 캐시하며, **show rif** 명령으로 이를 확인할 수 있습니다. PU2.0이므로 라우터는 XID null에 대한 응답 후 호스트에 Format 0 Type 2 XID를 보냅니다.

```
SDLLC: O xid(null), 4000.1111.1111 4000.3174.11c6 4 4 [1000.14D3.0641.0051.12C2.0194.01F1.02C0]
SDLLC: NET UP WAIT      recv FORWARD XID P/F(BF) 4000.3174.11c6 c000.1111.1111 04 05
-> Serial1 C6
SDLLC: O xid(0T2), 4000.1111.1111 4000.3174.11c6 4 4 [1000.14D3.0641.0051.12C2.0194.01F1.02C0]
SDLLC: NET UP WAIT      recv FORWARD SABME P/F(7F) 4000.3174.11c6 c000.1111.1111 04 04
-> Serial1 C6
SDLLC: SABME for Serial1 C6 in NET UP WAIT
%SDLLC-5-ACT_LINK: SDLLC: Serial1 LINK address C6 ACTIVATED: Net connect
SDLLC_STATE: Serial1 C6 NET UP WAIT      -> CONNECT
```

XID 교환 후 라우터는 호스트에서 SABME(Set Asynchronous Balanced Mode Extended)를 수신합니다. 그러면 시작 절차가 완료되고 라우터가 호스트에 UA로 응답합니다. 이제 SDLC 라인의 상태가 USBUSY에서 CONNECT로 변경되고 1-프레임이 라우터를 통과할 수 있습니다.

```
SDLC_STATE: (5235944) Serial1 C6 USBUSY
-> CONNECT
Serial1 SDLC output      C611
Serial1 SDLC input       C611
s4f#
```

DLSw 미디어 변환

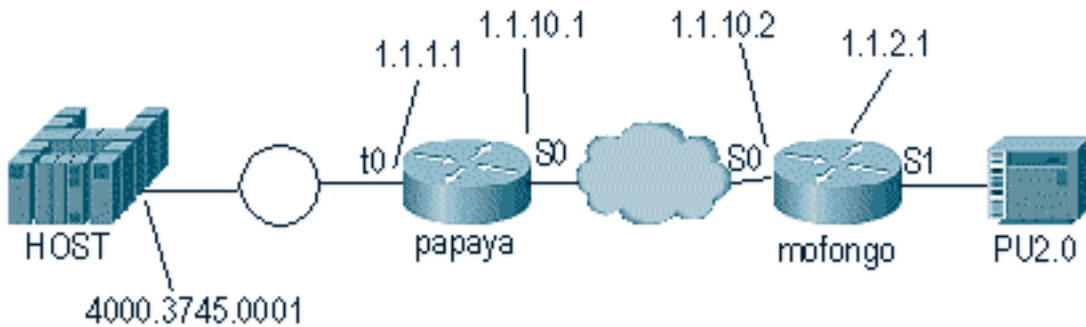
DLSw는 PU2.1을 지원하므로 미디어 변환을 대폭 개선할 수 있습니다. 이를 통해 5494 및 5394(PU2.1 - IBM RPQ 8Q0775)에서 AS/400s로 SDLLC에서 LLC2 변환을 수행할 수 있습니다. 이렇게 하면 STUN 및 잘못된 AS/400 멀티포인트 회선이 필요하지 않습니다.

DLSw 미디어 변환의 컨피그레이션 매개변수는 SDLLC 매개변수와 약간 다릅니다. 추가된 DLSw 명령이 하나 있으며 나머지는 SDLC 명령입니다. DLSw 미디어 변환을 구성하려면 다음 단계를 완료합니다.

1. SDLC로 직렬 캡슐화를 변경하려면 **encapsulation sdlc** 명령을 실행합니다. 라우터에서 SDLC 회선을 종료하려면 라우터가 폴링을 위해 기본 역할을 해야 합니다. 이는 주 유닛이 HOST 또는 AS/400이 되기 때문에 STUN과 다릅니다.
2. **sdlc role primary** 명령을 실행하여 라우터의 역할을 SDLC 행에서 primary로 변경합니다.
3. **sdlc address xx** 명령을 실행하여 SDLC 폴링 주소를 구성합니다. 여기서 DLSw는 SDLLC와 다릅니다. SDLLC에서 **sdllc** 키워드로 명령을 지정합니다. DLSw에서 **sdlc** 키워드로 명령을 지정합니다.

4. `sdhc vmac xxxx.xxxx.xx00` 명령을 실행하여 SDLC 컨트롤러의 가상 MAC 주소를 구성합니다. 이 매개변수는 라우터에 LLC2 환경에서 이 SDLC 컨트롤러의 가상 MAC 주소를 알려줍니다. 폴링 주소가 `00`으로 추가되었으므로 마지막 바이트를 계속 **설정해야** 합니다.
5. 이 PU 2.0에 대한 XID를 구성하려면 `sdhc xid nxxxxxx` 명령을 실행합니다. 이 명령에서 **nn은 컨트롤러의 폴링 주소**이며 xxxxxxx는 이 PU2.0의 XID(VTAM의 스위치 주 노드에 코딩된 IDBLOCK 및 IDNUM)입니다. **참고:** PU2.1이 있는 경우 XID 협상이 이루어집니다. 따라서 명령이 변경됩니다.
6. 이 PU 2.1에 대한 XID를 구성하려면 `sdhc xid nxid-poll` 명령을 실행합니다. 이 명령에서 *nn*은 스테이션의 폴링 주소입니다.
7. `sdhc partner xxxx.xxxx.xxxx nn` 명령을 실행하여 라우터 파트너 MAC 주소를 구성합니다. 이 명령에서 **nn은 해당 컨트롤러의 폴링 주소**입니다. 다중 지점 회선에는 한 호스트로 향하는 컨트롤러와 다른 호스트로 향하는 컨트롤러가 있을 수 있으므로 컨트롤러 주소를 지정하는 것이 중요합니다.
8. 특정 컨트롤러에 대한 DLSw를 구성하려면 `sdhc dlsw nn` 명령을 실행합니다. 이 명령에서 **nn은 멀티드롭에 있는 컨트롤러 또는 컨트롤러의 폴링 주소**입니다. 이 명령을 사용하면 하나의 명령에서 여러 폴링 주소를 지정할 수 있습니다. **참고:** 버그 #CSCdi75481을 주의하십시오. 자세한 내용은 [버그 툴킷\(등록된 고객만 해당\)](#)을 참조하십시오. 라우터의 SDLC 주소를 변경하기 전에 `sdhc dlsw` 명령을 제거하지 않으면 CLS 코드가 SDLC 인터페이스와 DLSw를 올바르게 통신할 수 없습니다. 그러면 인터페이스가 구성된 것이 없는 것처럼 동작합니다. 이 버그는 Cisco IOS® Software 릴리스 11.1(8.1) 11.1(8.1)AA01(01.03) 11.1(8.1)AA01(01.02) 이상에서 수정되었습니다.

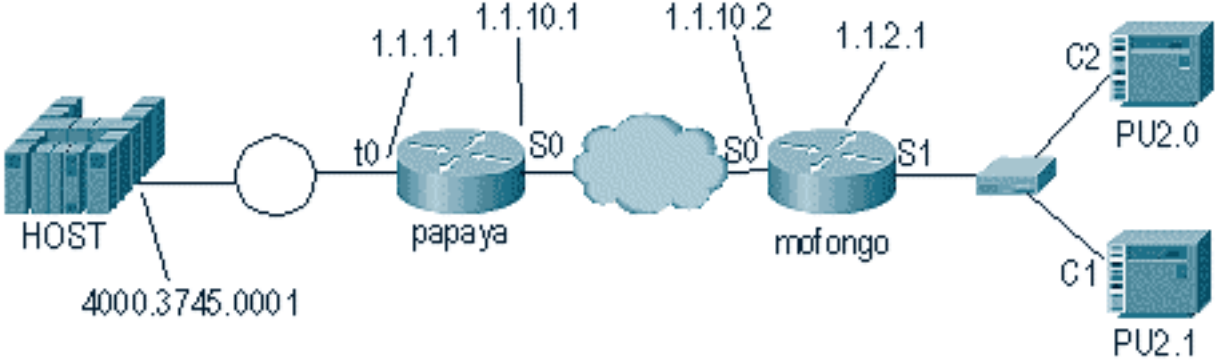
DLSw SDLC PU2.0 컨트롤러의 샘플 컨피그레이션이 표시됩니다.



파파야	모풍고
<pre>source-bridge ring-group 100 dlsw local-peer peer-id 1.1.1.1 dlsw remote-peer 0 tcp 1.1.2.1 ! interface serial 0 ip address 1.1.10.1 255.255.255.0 ! interface tokenring 0 ip address 1.1.1.1 255.255.255.0 ring-speed 16 source-bridge 1 1 100 source-bridge spanning</pre>	<pre>dlsw local-peer peer-id 1.1.2.1 dlsw remote-peer 0 tcp 1.1.1.1 ! interface loopback 0 ip address 1.1.2.1 ! interface serial 0 ip address 1.1.10.2 255.255.255.0 ! interface serial 1 no ip address encapsulation sdhc sdhc role primary sdhc vmac 4000.3174.0000 sdhc address c1 sdhc xid c1 01767890 sdhc partner 4000.3745.0001 c1</pre>

```
sdlc dlsw c1
```

멀티드롭을 코딩할 때는 PU2.1s가 더 지능적이며 일반 PU2.0 디바이스보다 더 많은 정보를 교환한다는 점을 기억하십시오. PU2.0 디바이스의 기본 코드로 회선을 코딩해야 하기 때문에 멀티드롭 환경을 구성할 때 중요합니다. 또한 코드가 각 컨트롤러를 사용하여 수행할 작업을 이해할 수 있도록 PU2.1 디바이스의 SDLC 주소에 대해 xid-poll을 추가해야 합니다. 다음은 컨피그레이션의 예입니다



파파야	모풍고
<pre>source-bridge ring-group 100 dlsw local-peer peer-id 1.1.1.1 dlsw remote-peer 0 tcp 1.1.2.1 ! interface serial 0 ip address 1.1.10.1 255.255.255.0 ! interface tokenring 0 ip address 1.1.1.1 255.255.255.0 ring-speed 16 source-bridge 1 1 100 source-bridge spanning</pre>	<pre>dlsw local-peer peer-id 1.1.2.1 dlsw remote-peer 0 tcp 1.1.1.1 ! interface loopback 0 ip address 1.1.2.1 ! interface serial 0 ip address 1.1.10.2 255.255.255.0 ! interface serial 1 no ip address encapsulation sdhc sdhc role primary sdhc vmac 4000.3174.0000 sdhc address c1 xid-poll sdhc partner 4000.9404.0001 c1 sdhc address c2 01767890 sdhc partner 4000.9404.0001 c2 sdhc dlsw c1 c2</pre>

show 명령

DLSw 미디어 변환에 사용되는 show 명령에 대한 자세한 내용은 Data-Link Switching Plus를 참조하십시오.

PU2.1용 DLSw/SDLC 중 SDLC 패킷 디버깅

%LINK-3-UPDOWN: Interface Serial2, changed state to up
가장 먼저 발생하는 일은 FF의 SDLC 브로드캐스트 주소에 대한 XID 또는 BF입니다.

다음으로, 5494에서 XID를 수신합니다. 이는 XID 형식 2 유형 3이며, 이 XID 형식은 **debug sdhc packet** 명령 출력에 표시됩니다.

```
Serial2 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
```

다음은 이 명령의 여러 필드에 대한 설명입니다.

- **073000DD** — 이 필드는 5494에 구성된 블록 ID 및 ID 번호입니다. 블록 ID 및 ID 번호는 XID라고 하며 세션 협상 중에 5494가 피어로 전송합니다.
- **NETA** — 이 필드는 사용 중인 NETID(Advanced Peer-to-Peer Networking)입니다. 일반적으로 이 필드는 피어에 구성된 NETID와 일치합니다. 이 경우 피어는 AS/400입니다.
- **CP5494** — 이 필드는 5494의 CP(Control Point) 이름입니다.
- **DD** — 이 필드는 SDLC 주소입니다.

다음으로 AS/400에서 XID를 수신합니다.

```
Serial2 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:
```

```
Serial2 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
```

```
Serial2 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:
```

```
Serial2 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** — 이 필드는 AS/400의 블록 ID 및 ID 번호입니다.
- **RTP400A** — 이 필드는 AS/400의 CP 이름입니다. CP 이름은 AS/400의 DSPNETA(Display Network Attributes) 파일에 있습니다.

그러면 SNRM(93) 및 UA(73)가 회선에 표시됩니다. SNRM 이전에는 라우터가 항상 브로드캐스트 주소를 사용합니다. 이 시점부터 라우터는 항상 DD의 실제 폴링 주소를 사용합니다.


```
Serial2 SDLC input      DD73
Serial2 SDLC output    DD11
Serial2 SDLC input      DD11
```

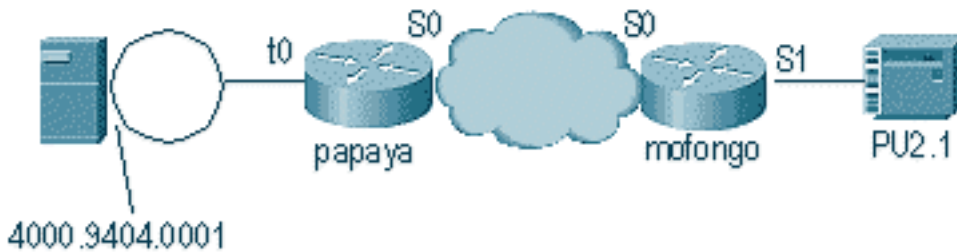
이 시점에서 라우터와 5494 간의 RR(Receiver Ready) 상태 때문에 연결이 일시 중단됩니다.

참고: 디버그를 실행해야 하는 라우터에 다른 SDLC 인터페이스가 있고 버퍼링된 로깅이 없으면 라우터가 일시 중단할 수 있습니다. 터미널에 대한 디버그를 실행할 수 있는 경우와 로깅을 실행할 수 있는 경우를 이해하는 것은 환경을 제공합니다. 확실하지 않은 경우, 항상 logging buffered 및 show log 명령을 사용하여 SDLC 디버그를 표시합니다.

AS/400에서 컨트롤러를 꺼냅니다. 이렇게 하면 세션의 SDLC측에서 DISK(53) 및 UA(73)를 볼 수 있습니다.

```
Serial2 SDLC output    DD53
Serial2 SDLC input      DD73
```

DLSw 미디어 변환 예



인터페이스가 작동되고 나면 라우터가 원격 컨트롤러의 위치를 확인하여 프로세스를 시작합니다.

```
%LINEPROTO-5-UPDOWN: Line protocol on Interface Serial4, changed state to up
DLSW Received-ctlQ : CLSI Msg : ID_STN.Ind  dlen: 46
CSM: Received CLSI Msg : ID_STN.Ind  dlen: 46 from Serial4
CSM: smac 4000.5494.00dd, dmac 4000.9404.0001, ssap 4 , dsap 4
%DLSWC-3-RECVSSP: SSP OP = 4( ICR ) -explorer from peer 10.17.2.198(2065)
DLSw: new_ckt_from_clsi(): Serial4 4000.5494.00dd:4->4000.9404.0001:4
```

ICR 프레임을 수신한 후 DLSw는 이 세션에 대한 FSM(Finite State Machine)을 시작합니다. 이 작업은 DLSw와 CLSI(Cisco Link Services Interface) 간에 있는 REQ_OPNSTN Req 및 REQ_OPNSTN.Cfm 메시지에 의해 수행됩니다.

```
DLSw: START-FSM (488636): event:DLC-Id state:DISCONNECTED
DLSw: core: dlsw_action_a()
DISP Sent : CLSI Msg : REQ_OPNSTN.Req  dlen: 106
DLSw: END-FSM (488636): state:DISCONNECTED->LOCAL_RESOLVE
```

```
DLSW Received-ctlQ : CLSI Msg : REQ_OPNSTN.Cfm CLS_OK dlen: 106
DLSw: START-FSM (488636): event:DLC-ReqOpnStn.Cnf state:LOCAL_RESOLVE
DLSw: core: dlsw_action_b()
CORE: Setting lf size to FF
```

CLSI와의 대화 후 DLSw는 세션 시작 CUR 프레임을 원격 라우터로 전송합니다. 이는 두 라우터 사이에서만 발생합니다.

```
%DLSWC-3-SENDSSP: SSP OP = 3( CUR ) to peer 10.17.2.198(2065) success
```

DLSw: END-FSM (488636): state:LOCAL_RESOLVE->CKT_START

%DLSWC-3-**RECVSSP**: SSP OP = 4(**ICR**) from peer 10.17.2.198(2065)

DLSw: 488636 recv FCI 0 - s:0 so:0 r:0 ro:0

DLSw: recv RWO

DLSw: START-FSM (488636): event:WAN-ICR state:CKT_START

DLSw: core: dlsw_action_e()

DLSw: sent RWO

DLSw: 488636 sent FCI 80 on ACK - s:20 so:1 r:20 ro:1

%DLSWC-3-**SENDSSP**: SSP OP = 5(**ACK**) to peer 10.17.2.198(2065) success

DLSw: END-FSM (488636): state:CKT_START->CKT_ESTABLISHED

회로가 설정되면 라우터는 저장된 XID를 전송하고 XID 교환을 시작합니다.XID의 위치를 파악하는 것이 중요합니다.이 예에서 DLC(data-link control)-Id는 XID가 로컬 DLC 스테이션에서 오고 WAN-XID가 원격 라우터 또는 원격 스테이션에서 제공되었음을 의미합니다.

DLSw: START-FSM (488636): event:**DLC-Id** state:CKT_ESTABLISHED

DLSw: core: dlsw_action_f()

DLSw: 488636 sent FCA on XID

%DLSWC-3-**SENDSSP**: SSP OP = 7(**XID**) to peer 10.17.2.198(2065) success

DLSw: END-FSM (488636): state:CKT_ESTABLISHED->CKT_ESTABLISHED

%DLSWC-3-**RECVSSP**: SSP OP = 7(**XID**) from peer 10.17.2.198(2065)

DLSw: 488636 recv FCA on XID - s:20 so:0 r:20 ro:0

DLSw: START-FSM (488636): event:**WAN-XID** state:CKT_ESTABLISHED

DLSw: core: dlsw_action_g()

DISP Sent : CLSI Msg : ID.Rsp dlen: 12

DLSw: END-FSM (488636): state:CKT_ESTABLISHED->CKT_ESTABLISHED

%DLSWC-3-**RECVSSP**: SSP OP = 7(**XID**) from peer 10.17.2.198(2065)

DLSw: START-FSM (488636): event:WAN-XID state:CKT_ESTABLISHED

DLSw: core: dlsw_action_g()

DISP Sent : CLSI Msg : ID.Req dlen: 88

DLSw: END-FSM (488636): state:CKT_ESTABLISHED->CKT_ESTABLISHED

DLSW Received-ctlQ : CLSI Msg : ID.Ind dlen: 82

DLSw: START-FSM (488636): event:**DLC-Id** state:CKT_ESTABLISHED

DLSw: core: dlsw_action_f()

%DLSWC-3-**SENDSSP**: SSP OP = 7(**XID**) to peer 10.17.2.198(2065) success

DLSw: END-FSM (488636): state:CKT_ESTABLISHED->CKT_ESTABLISHED

%DLSWC-3-**RECVSSP**: SSP OP = 7(**XID**) from peer 10.17.2.198(2065)

DLSw: START-FSM (488636): event:**WAN-XID** state:CKT_ESTABLISHED

DLSw: core: dlsw_action_g()

DISP Sent : CLSI Msg : ID.Rsp dlen: 88

DLSw: END-FSM (488636): state:CKT_ESTABLISHED->CKT_ESTABLISHED

DLSW Received-ctlQ : CLSI Msg : ID.Ind dlen: 82

DLSw: START-FSM (488636): event:**DLC-Id** state:CKT_ESTABLISHED

DLSw: core: dlsw_action_f()

%DLSWC-3-SENDSSP: SSP OP = 7(**XID**) to peer 10.17.2.198(2065) success

DLSw: END-FSM (488636): state:CKT_ESTABLISHED->CKT_ESTABLISHED

%DLSWC-3-RECVSSP: SSP OP = 7(**XID**) from peer 10.17.2.198(2065)

DLSw: START-FSM (488636): event:**WAN-XID** state:CKT_ESTABLISHED

DLSw: core: dlsw_action_g()

DISP Sent : CLSI Msg : ID.Rsp dlen: 88

DLSw: END-FSM (488636): state:CKT_ESTABLISHED->CKT_ESTABLISHED

DLSW Received-ctlQ : CLSI Msg : ID.Ind dlen: 82

DLSw: START-FSM (488636): event:**DLC-Id** state:CKT_ESTABLISHED

DLSw: core: dlsw_action_f()

%DLSWC-3-SENDSSP: SSP OP = 7(**XID**) to peer 10.17.2.198(2065) success

DLSw: END-FSM (488636): state:CKT_ESTABLISHED->CKT_ESTABLISHED

라우터는 AS/400(SABME)에서 CONQ를 수신합니다. 이는 직렬 회선으로 SNRM으로 변환됩니다. 그런 다음 라우터는 직렬 회선(CONNECT.Cfm)에서 UA를 기다리고 CONR을 다른 쪽으로 전송합니다. 이렇게 하면 세션 상태가 CONNECTED로 변경됩니다.

%DLSWC-3-RECVSSP: SSP OP = 8(CONQ) from peer 10.17.2.198(2065)

DLSw: START-FSM (488636): event:WAN-CONQ state:CKT_ESTABLISHED

DLSw: core: dlsw_action_i()

DISP Sent : CLSI Msg : CONNECT.Req dlen: 16

DLSw: END-FSM (488636): state:CKT_ESTABLISHED->CONTACT_PENDING

DLSW Received-ctlQ : CLSI Msg : CONNECT.Cfm CLS_OK dlen: 8

DLSw: START-FSM (488636): event:DLC-Connect.Cnf state:CONTACT_PENDING

DLSw: core: dlsw_action_j()

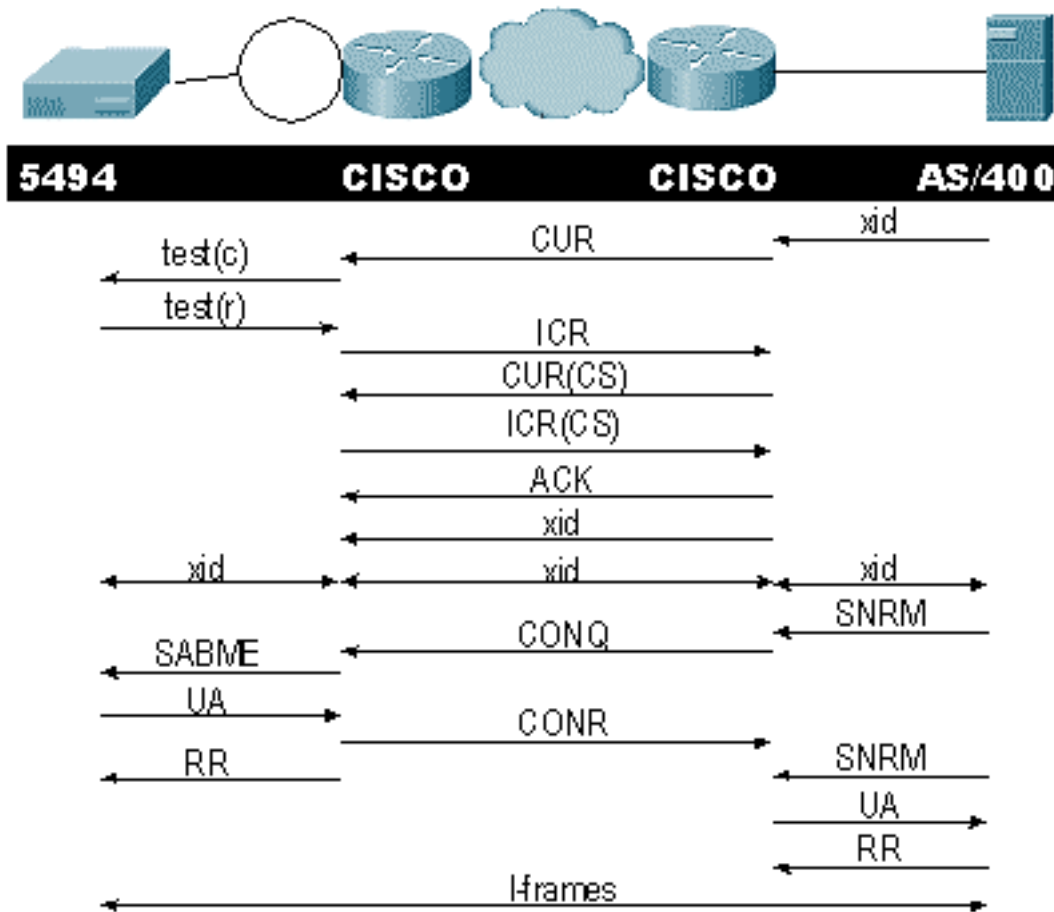
%DLSWC-3-SENDSSP: SSP OP = 9(CONR) to peer 10.17.2.198(2065) success

DISP Sent : CLSI Msg : FLOW.Req dlen: 0

DLSw: END-FSM (488636): state:CONTACT_PENDING->CONNECTED

DLSw 역미디어 변환 수행

또 다른 일반적인 설정은 reverse-sdlc입니다. 역방향 SDLLC에서 기본 스테이션은 SDLC 회선을 통해 라우터에 연결됩니다. 이는 일반적으로 사용자가 호스트를 Token Ring 첨부 파일로 마이그레이션하려는 경우 호스트 환경에서 나타납니다. Reverse SDLLC는 원격 CPU가 활성 상태인지 여부를 알 수 없기 때문에 DLSw에서 SDLC 라인을 처리하는 방식을 변경합니다.



첫째, AS/400은 이 경우 기본 항목이거나 역할에서 협상 가능으로 설정되어 있으므로 세션을 시작해야 합니다. AS/400이 직렬 회선이 작동된 후 첫 번째 XID를 전송하면 라우터가 원격 컨트롤러에 대한 검색 프로세스를 시작합니다. 회로가 설정되면 XID 협상은 회선에서 시작할 수 있습니다.

XID 협상이 완료되면 AS/400은 라우터에 SNRM을 전송합니다.이렇게 하면 라우터가 CONQ를 전송하고 원격 라우터에서 CONR을 기대합니다.라우터는 SNRM이 표시되고 CONR이 수신될 때까지 UA로 응답할 수 없습니다.거의 모든 버전의 코드에서 라우터는 세션 시간 초과가 될 때까지 30초 동안 기다립니다.이는 운영 디바이스가 원격 호스트에서 CONR을 수신하면 운영 디바이스에서 SNRM을 수신하는 것과 관련된 것입니다.

최신 Cisco IOS 11.1 코드에서 기본값은 30초가 아닌 1분으로 변경되었습니다.AS/400에서 이 시간 초과는 비생산적인 응답 타이머라고 하며 기본값은 32초입니다.

로컬 DLSw 미디어 변환



```

%LINEPROTO-5-UPDOWN: Line protocol on Interface Serial2, changed state to up
%SYS-5-CONFIG_I: Configured from console by console
DLSW Received-ctlQ : CLSI Msg : ID_STN.Ind dlen: 46
CSM: Received CLSI Msg : ID_STN.Ind dlen: 46 from Serial2
  
```

DLSw 로컬에서 처음 확인할 수 있는 것은 시리얼 쪽의 XID입니다.라우터가 LLC 테스트 프레임/응답을 보낼 때까지 이 XID를 저장해야 합니다.

```

CSM: smac 4000.5494.00dd, dmac 4000.9404.0001, ssap 4 , dsap 4
DISP Sent : CLSI Msg : TEST_STN.Req dlen: 46
DISP Sent : CLSI Msg : TEST_STN.Req dlen: 46
DISP Sent : CLSI Msg : TEST_STN.Req dlen: 46
CSM: Write to all peers not ok - PEER_NO_CONNECTIONS
DLSW Received-ctlQ : CLSI Msg : TEST_STN.Ind dlen: 43
CSM: Received CLSI Msg : TEST_STN.Ind dlen: 43 from TokenRing0
CSM: smac c000.9404.0001, dmac 4000.5494.00dd, ssap 0 , dsap 4
  
```

다음으로, 테스트 스테이션은 라우터를 남겨두고 AS/400에서 응답을 반환합니다. 이제 라우터가 로컬 FSM을 생성할 수 있습니다.

참고: 이 세션은 로컬 세션입니다.

```

DLSw: csm_to_local(): Serial2-->TokenRing0 4000.5494.00dd:4->4000.9404.0001:4
DLSw: START-LFSM TokenRing0 (4000.9404.0001->4000.5494.00dd) event:ADMIN-START
DLSw: LFSM-A: Opening DLC station
DISP Sent : CLSI Msg : REQ_OPNSTN.Req dlen: 106
DLSw: END-LFSM (4000.9404.0001->4000.5494.00dd): state:DISCONNECTED ->OPN_STN_PEND

DLSw: START-LFSM Serial2 (4000.5494.00dd->4000.9404.0001) event:ADMIN-START
DLSw: LFSM-A: Opening DLC station
DISP Sent : CLSI Msg : REQ_OPNSTN.Req dlen: 106
DLSw: END-LFSM (4000.5494.00dd->4000.9404.0001): state:DISCONNECTED ->OPN_STN_PEND

DLSW Received-ctlQ : CLSI Msg : REQ_OPNSTN.Cfm CLS_OK dlen: 106
DLSw: START-LFSM TokenRing0 (4000.9404.0001->4000.5494.00dd) event:DLC-ReqOpnStn.Cnf
  
```

DLSw: LFSM-B: DLC station opened
DLSw: END-LFSM (4000.9404.0001->4000.5494.00dd): state:OPN_STN_PEND ->ESTABLISHED

DLSW Received-ctlQ : CLSI Msg : **REQ_OPNSTN.Cfm** CLS_OK dlen: 106
DLSw: START-LFSM Serial2 (4000.5494.00dd->4000.9404.0001) event:DLC-ReqOpnStn.Cnf
DLSw: LFSM-B: DLC station opened
DLSw: processing saved clsi message

라우터가 FSM이 준비되었음을 로컬로 확인한 후 파트너에게 XID를 보낼 수 있습니다.이 예에서 파트너는 AS/400(ID Req)입니다.

DLSw: START-LFSM Serial2 (4000.5494.00dd->4000.9404.0001) event:DLC-Id
DLSw: LFSM-X: forward XID to partner
DISP Sent : CLSI Msg : **ID Req** dlen: 12
DLSw: END-LFSM (**4000.5494.00dd->4000.9404.0001**): state:ESTABLISHED ->ESTABLISHED

DLSw: END-LFSM (4000.5494.00dd->4000.9404.0001): state:OPN_STN_PEND ->ESTABLISHED

DLSW Received-ctlQ : CLSI Msg : ID.Cfm CLS_OK dlen: 32
DLSw: START-LFSM TokenRing0 (**4000.9404.0001->4000.5494.00dd**) event:DLC-Id
DLSw: LFSM-X: forward XID to partner
DISP Sent : CLSI Msg : **ID.Rsp** dlen: 12
DLSw: END-LFSM (4000.9404.0001->4000.5494.00dd): state:ESTABLISHED ->ESTABLISHED

그러면 토큰 링에서 XID가 수신됩니다.ID.Ind의 길이는 108입니다. 라우터가 이 XID를 SDLC 라인인 이 시나리오에서 파트너에게 전달합니다.이는 전송된 ID Req로 표시됩니다.라우터가 패킷을 수신할 때마다 LFSM(linear finite state machine)을 시작해야 합니다. 이 디버그가 시작되는 위치와 진행 중인 점을 알려주기 때문에 이 디버그를 이해하는 데 중요한 요소입니다.

DLSW Received-ctlQ : CLSI Msg : **ID.Ind** dlen: 108
DLSw: START-LFSM **TokenRing0** (**4000.9404.0001->4000.5494.00dd**) event:DLC-Id
DLSw: LFSM-X: forward XID to partner
DISP Sent : CLSI Msg : **ID Req** dlen: 88
DLSw: END-LFSM (4000.9404.0001->4000.5494.00dd): state:ESTABLISHED ->ESTABLISHED

다음으로, XID 응답은 직렬 회선에서 수신되고 파트너(이 예의 토큰 링 스테이션)에게 전달됩니다. 이 PU2.1 디바이스에 대해 XID 교환이 완료될 때까지 계속됩니다.

DLSW Received-ctlQ : CLSI Msg : **ID.Ind** dlen: 82
DLSw: START-LFSM Serial2 (**4000.5494.00dd->4000.9404.0001**) event:DLC-Id
DLSw: LFSM-X: forward XID to partner
DISP Sent : CLSI Msg : ID.Rsp dlen: 80
DLSw: END-LFSM (4000.5494.00dd->4000.9404.0001): state:ESTABLISHED ->ESTABLISHED

DLSW Received-ctlQ : CLSI Msg : **ID.Ind** dlen: 108
DLSw: START-LFSM TokenRing0 (**4000.9404.0001->4000.5494.00dd**) event:DLC-Id
DLSw: LFSM-X: forward XID to partner
DISP Sent : CLSI Msg : **ID.Rsp** dlen: 88
DLSw: END-LFSM (4000.9404.0001->4000.5494.00dd): state:ESTABLISHED ->ESTABLISHED

DLSW Received-ctlQ : CLSI Msg : ID.Ind dlen: 82
DLSw: START-LFSM Serial2 (4000.5494.00dd->4000.9404.0001) event:DLC-Id
DLSw: LFSM-X: forward XID to partner
DISP Sent : CLSI Msg : **ID.Rsp** dlen: 80
DLSw: END-LFSM (4000.5494.00dd->4000.9404.0001): state:ESTABLISHED ->ESTABLISHED

DLSW Received-ctlQ : CLSI Msg : ID.Ind dlen: 108
DLSw: START-LFSM TokenRing0 (4000.9404.0001->4000.5494.00dd) event:DLC-Id
DLSw: LFSM-X: forward XID to partner
DISP Sent : CLSI Msg : **ID.Rsp** dlen: 88

DLSw: END-LFSM (4000.9404.0001->4000.5494.00dd): state:ESTABLISHED ->ESTABLISHED

%LINK-3-UPDOWN: Interface Serial2, changed state to up

DLSW Received-ctlQ : CLSI Msg : ID.Ind dlen: 82

DLSw: START-LFSM Serial2 (4000.5494.00dd->4000.9404.0001) event:DLC-Id

DLSw: LFSM-X: forward XID to partner

DISP Sent : CLSI Msg : **ID.Rsp** dlen: 80

DLSw: END-LFSM (4000.5494.00dd->4000.9404.0001): state:ESTABLISHED ->ESTABLISHED

XID 교환 후 라우터는 CONNECT.Ind를 통해 AS/400에서 SABME를 수신합니다.그러면 라우터가 SNRM인 SDLC 라인에 CONNECT.Req를 전송하도록 지시합니다.그런 다음 CONNECT.CFM(UA) 메시지가 직렬 회선에서 수신되므로 DLSw 코드가 CONNECT.Rsp(UA)를 AS/400으로 보냅니다.

DLSW Received-ctlQ : CLSI Msg : **CONNECT.Ind** dlen: 8

DLSw: START-LFSM TokenRing0 (4000.9404.0001->4000.5494.00dd) event:DLC-Connect.Ind

DLSw: LFSM-C: starting local partner

DLSw: START-LFSM Serial2 (4000.5494.00dd->4000.9404.0001) event:ADMIN-CONN

DLSw: LFSM-D: sending connect request to station

DISP Sent : CLSI Msg : **CONNECT.Req** dlen: 16

DLSw: END-LFSM (4000.5494.00dd->4000.9404.0001): state:ESTABLISHED ->CONN_OUT_PEND

DLSw: END-LFSM (4000.9404.0001->4000.5494.00dd): state:ESTABLISHED ->CONN_IN_PEND

DLSW Received-ctlQ : CLSI Msg : **CONNECT.Cfm** CLS_OK dlen: 8

DLSw: START-LFSM Serial2 (4000.5494.00dd->4000.9404.0001) event:DLC-Connect.Cnf

DLSw: LFSM-E: station accepted the connection

DLSw: START-LFSM TokenRing0 (4000.9404.0001->4000.5494.00dd) event:ADMIN-CONN

DLSw: LFSM-F: accept incoming connection

DISP Sent : CLSI Msg : **CONNECT.Rsp** dlen: 20

DLSw: END-LFSM (4000.9404.0001->4000.5494.00dd): state:CONN_IN_PEND ->CONNECTED

DISP Sent : CLSI Msg : FLOW.Req dlen: 0

DLSw: END-LFSM (4000.5494.00dd->4000.9404.0001): state:CONN_OUT_PEND->CONNECTED

컨트롤러(SDLC)가 종료될 때의 세션이 표시됩니다.

%LINEPROTO-5-UPDOWN: Line protocol on Interface Serial2, changed state to down

%LINK-5-CHANGED: Interface Serial2, changed state to administratively down

DLSW Received-ctlQ : CLSI Msg : **DISCONNECT.Ind** dlen: 8

DLSw: START-LFSM Serial2 (4000.5494.00dd->4000.9404.0001) event:DLC-Disc.Ind

DLSw: LFSM-Q: acknowledge disconnect

DISP Sent : CLSI Msg : **DISCONNECT.Rsp** dlen: 4

다음으로 라우터는 AS/400(DISCONNECT.Rsp)에 디스크를 전송합니다. 그리고 나서, 지역 회로를 해체하기 시작합니다.

DLSw: START-LFSM TokenRing0 (4000.9404.0001->4000.5494.00dd) event:ADMIN-STOP

DLSw: LFSM-Z: close dlc station request

DISP Sent : CLSI Msg : CLOSE_STN.Req dlen: 4

DLSw: END-LFSM (4000.9404.0001->4000.5494.00dd): state:ESTABLISHED ->CLOSE_PEND

DISP Sent : CLSI Msg : CLOSE_STN.Req dlen: 4

DLSw: END-LFSM (4000.5494.00dd->4000.9404.0001): state:ESTABLISHED ->CLOSE_PEND

DLSW Received-ctlQ : CLSI Msg : CLOSE_STN.Cfm CLS_OK dlen: 8

DLSw: START-LFSM TokenRing0 (4000.9404.0001->4000.5494.00dd) event:DLC-CloseStn.Cnf

DLSw: LFSM-Y: driving partner to close circuit

DLSw: START-LFSM Serial2 (4000.5494.00dd->4000.9404.0001) event:ADMIN-STOP

DLSw: END-LFSM (4000.5494.00dd->4000.9404.0001): state:CLOSE_PEND ->CLOSE_PEND

DLSw: END-LFSM (4000.9404.0001->4000.5494.00dd): state:CLOSE_PEND ->DISCONNECTED

```
DLSW Received-ctlQ : CLSI Msg : DISCONNECT.Ind   dlen: 8  
DLSw: START-LFSM Serial2 (4000.5494.00dd->4000.9404.0001) event:DLC-Disc.Ind  
DLSw: END-LFSM (4000.5494.00dd->4000.9404.0001): state:CLOSE_PEND   ->CLOSE_PEND
```

```
DLSW Received-ctlQ : CLSI Msg : CLOSE_STN.Cfm CLS_OK dlen: 8  
DLSw: START-LFSM Serial2 (4000.5494.00dd->4000.9404.0001) event:DLC-CloseStn.Cnf  
DLSw: LFSM-Y: removing local switch entity  
DLSw: END-LFSM (4000.5494.00dd->4000.9404.0001): state:CLOSE_PEND   ->DISCONNECTED
```

라우터가 AS/400에서 **DISCONNECT.Ind(UA)**를 수신하면 세션 지우기가 완료되고 연결 해제 상태로 전환됩니다.

[관련 정보](#)

- [IBM 기술](#)
- [기술 지원 및 문서 - Cisco Systems](#)