

Entendendo e Troubleshooting de SDLC para Tradução de Mídia de Rede LLC

Contents

[Introduction](#)

[Prerequisites](#)

[Requirements](#)

[Componentes Utilizados](#)

[Conventions](#)

[SDLLC](#)

[Configuração de SDLC](#)

[Configuração do SDLLC](#)

[Depuração de SDLLC](#)

[Tradução de mídia DLSw](#)

[comandos show](#)

[Depurando pacotes SDLC durante DLSw/SDLC para PU2.1](#)

[Exemplo de tradução de mídia DLSw](#)

[DLSw executando tradução de mídia reversa](#)

[Tradução de mídia DLSw local](#)

[Informações Relacionadas](#)

[Introduction](#)

Este documento fornece informações para entender e solucionar problemas de uma tradução de mídia de rede SDLC (Synchronous Data Link Control) para LLC (Logical Link Control).

[Prerequisites](#)

[Requirements](#)

Não existem requisitos específicos para este documento.

[Componentes Utilizados](#)

Este documento não se restringe a versões de software e hardware específicas.

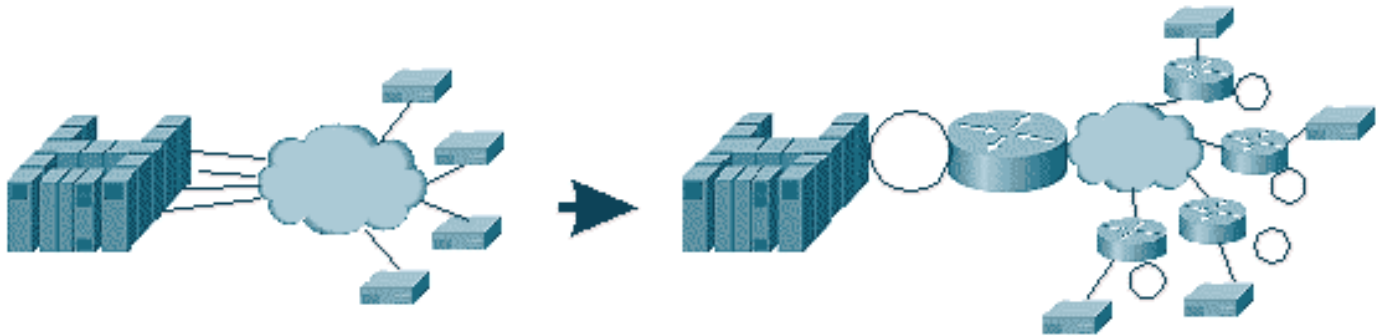
[Conventions](#)

Consulte as [Convenções de Dicas Técnicas da Cisco para obter mais informações sobre convenções de documentos](#).

SDLLC

A conversão SDLC-to-LAN (SDLLC) é usada para converter uma sessão SDLC para um dispositivo de Unidade Física 2 (PU2.0) em uma sessão Logical Link Control, tipo 2 (LLC2). Isso é muito útil se você tiver uma grande quantidade de controladores remotos alimentados em uma única porta Token Ring em um processador de front-end (FEP).

O lado esquerdo deste diagrama exibe um FEP com muitas linhas SDLC saindo para locais remotos. O lado direito deste diagrama exibe o mesmo cenário com os roteadores Cisco.



Os roteadores permitem que o FEP tenha apenas a interface Token Ring. A partir desse ponto, há vários locais remotos executando SDLLC para o host, bem como tráfego regular de bridge de rota de origem (SRB).

Observação: o uso de SDLLC para LLC para conversão de SDLC aplica-se somente a dispositivos PU2.0, não à unidade física tipo 2.1 (PU2.1). PU2.1 é suportado em DLSw (Data-Link Switching).

Para configurar o SDLLC, você precisa de um SRB no roteador. Consulte [Entendendo e Troubleshooting Local Source-Route Bridging](#) para obter informações sobre como configurar um SRB.

Configuração de SDLC

Como o SDLLC converte de uma interface SDLC, primeiro você precisa do SDLC configurado corretamente. Conclua estes passos para configurar o SDLC:

1. Emita o comando **encapsulation sdlc** para alterar o encapsulamento serial para SDLC.
2. Emita o comando **sdlc role primary** para alterar a função do roteador para primary na linha SDLC. **Observação:** em ambientes de STUN (Serial Tunneling, túnel serial), há funções primárias e secundárias. Consulte [Configuração e Troubleshooting de Serial Tunneling \(STUN\)](#) para obter mais informações.
3. Emita o comando **sdlc address xx** para configurar o endereço de pesquisa do SDLC.

Configuração do SDLLC

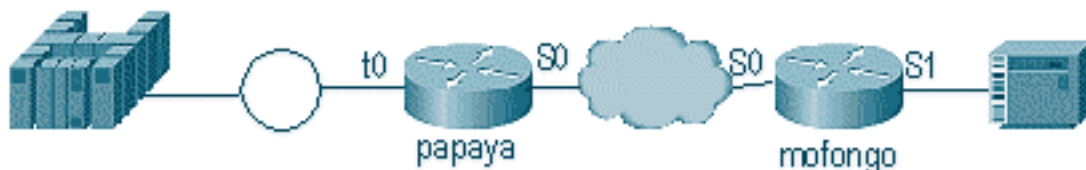
Para configurar SDLLC, o primeiro comando emitido é **traddr**. Esse comando define o que o SDLC converte no ambiente LLC2. Conclua estes passos para configurar SDLLC:

1. Emita o comando **sdllc traddr xxx.xxxx.xx00 lr bn tr** para habilitar a tradução de mídia

SDLLC em uma interface serial. Esse comando informa ao roteador o endereço MAC virtual da estação SDLC. Em seguida, o comando especifica o número do anel local (**lr**), o número da bridge (**bn**) e o número do anel de destino (**tr**). O **lr** deve ser exclusivo na rede. O **bn** pode ser um valor de 1 a 15. A **trn** deve ser o anel virtual no roteador. Se estiver configurando o SDLLC local, você poderá fazer com que ele aponte para um anel virtual ou para uma interface (anel físico conectado à interface Token Ring) no roteador. **Observação:** os dois últimos dígitos do endereço MAC neste comando são **00**. Não é possível definir os dois últimos dígitos de **traddr** porque o roteador usa esses dígitos para inserir o endereço SDLC dessa linha. Se você especificar os dois últimos dígitos, o roteador os substituirá pelo endereço SDLC. Em seguida, o host não responde para esse endereço MAC. Por exemplo, se o endereço MAC de negociação estiver configurado como 4000.1234.5678 e o endereço SDLC for 0x01, o roteador usará o MAC de 4000.1234.5601 para representar o dispositivo SDLC no domínio LLC. Além disso, o MAC de troca está em formato não canônico, que é o mesmo formato do quadro Token Ring.

2. Emita o comando **sdllc xid address xxxxxxxx** para especificar o valor XID (exchange identification) apropriado para a estação SDLC corresponder aos valores do Virtual Telecommunications Access Method (VTAM). Isso é determinado a partir do IDBLK e do IDNUM no nó principal do switch no VTAM. Se isso não coincidir, a troca XID falhará.
3. Emita o comando **sdllc partner mac-address sdllc-address** para ativar conexões para SDLLC. Isso especifica o endereço MAC do parceiro, que geralmente é o host.

Uma configuração de exemplo simples de SDLLC é exibida. A controladora conectada SDLC aparece como um dispositivo local conectado a Token Ring ao FEP.



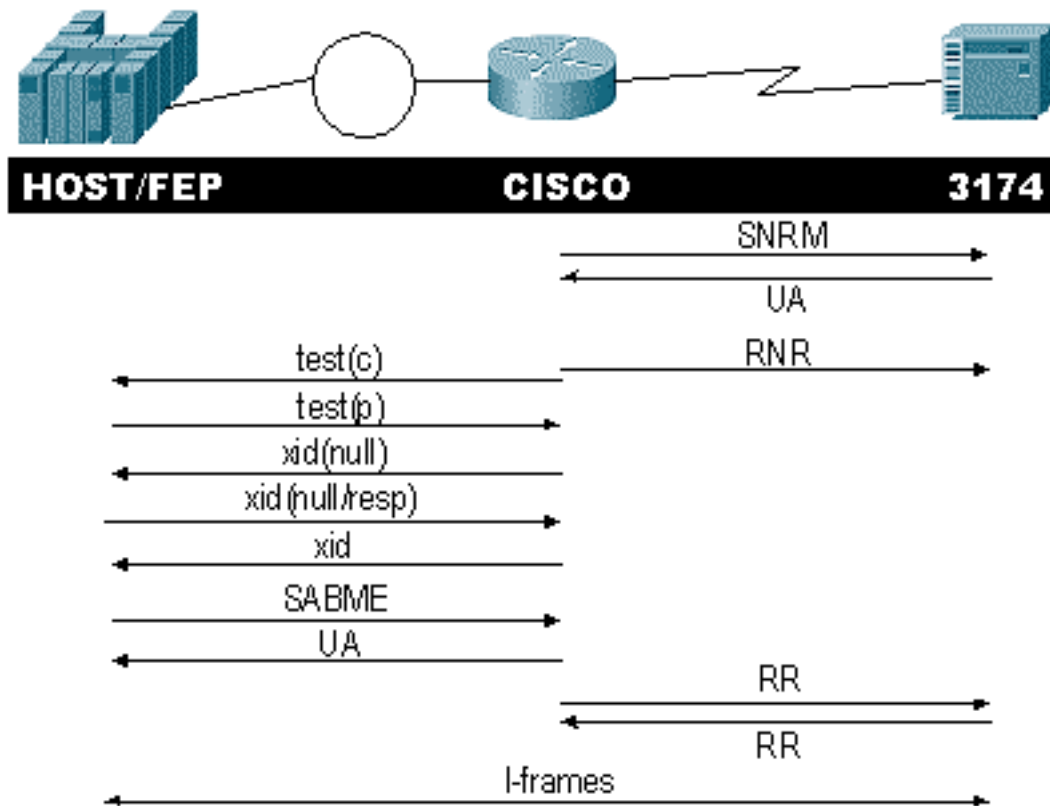
Papaya	Mofongo
<pre>source-bridge ring-group 100 source-bridge remote- peer 100 tcp 1.1.1.1 source-bridge remote- peer 100 tcp 1.1.2.1 local-ack interface tokenring 0 ip address 1.1.3.1 255.255.255.0 source-bridge 33 2 100 source-bridge spanning interface loopback 0 ip address 1.1.1.1 255.255.255.0</pre>	<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>

[Depuração de SDLLC](#)

Um problema de SDLLC requer que você solucione dois ambientes diferentes: o mundo SDLC e

o mundo Logical Link Control, tipo 2 (LLC2) para onde você está traduzindo os quadros. Como você pode ter apenas um tipo de controlador, a depuração de SDLLC é mais fácil de entender do que DLSw/SDLC.

Primeiro, observe os fluxos para a inicialização desta sessão específica:



Verifique a resposta do modo de resposta normal definido (SNRM) da controladora. O roteador não inicia a parte do LLC até que a parte do SDLC esteja ativa e em execução.

Emita estes comandos para verificar a resposta do SNRM:

- **sdlc_state**
- **sdllc_state**

Neste exemplo, o SNRM é enviado ao controlador, o que altera o estado da linha para SNRMSSENT. Se o roteador permanecer nesse estado, ele não recebeu a confirmação não numerada (UA) da controladora. Isso pode significar que algo está errado com a linha SDLC. Se isso ocorrer, a depuração será exibida como:

```
%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
```

Se o roteador receber o UA, o **sdlc_state** se move de SNRM_SENT para CONNECT. Em seguida, o estado SDLLC se move de SDLC_PRI_WAIT para NET_UP_WAIT. Quando isso ocorre, o roteador pode começar a ativar o lado LLC da conexão. A ação final é começar a enviar RNRs (não pronto para recebimento) para a linha SDLC. Isso desabilita o controlador de enviar qualquer informação até que o lado do LLC esteja operacional.

Em seguida, o roteador envia um explorador para encontrar a localização de seu parceiro.

```
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 rcv FORWARD TEST P/F(F3) 4000.3174.11c6 c000.1111.1111 00 01 -> Serial1
C6
caching rif
```

A saída anterior exibe a pesquisa de teste sendo enviada e recebida. Como este exemplo tem um controlador conectado localmente e um Token Ring, a pesquisa de teste deixa o roteador procurando o endereço do parceiro. Depois que o roteador recebe o quadro de teste, ele inicia a troca XID. O roteador coloca em cache o Campo de Informações de Roteamento (RIF) para esta sessão, que você pode verificar com o comando **show rif**. Como este é um PU2.0, o roteador envia um XID do tipo 2 do formato 0 para o host após a resposta ao XID nula.

```
SDLLC: O xid(null), 4000.1111.1111 4000.3174.11c6 4 4 [1000.14D3.0641.0051.12C2.0194.01F1.02C0]
SDLLC: NET UP WAIT rcv 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 rcv 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
```

Após a troca de XID, o roteador recebe o SABME (Set Asynchronous Balanced Mode Extended) do host. Isso finaliza o procedimento de inicialização e o roteador responde com um UA ao host. Agora, o estado da linha SDLC muda de USBUSY para CONNECT, e os I-frames podem passar pelo roteador.

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

[Tradução de mídia DLSw](#)

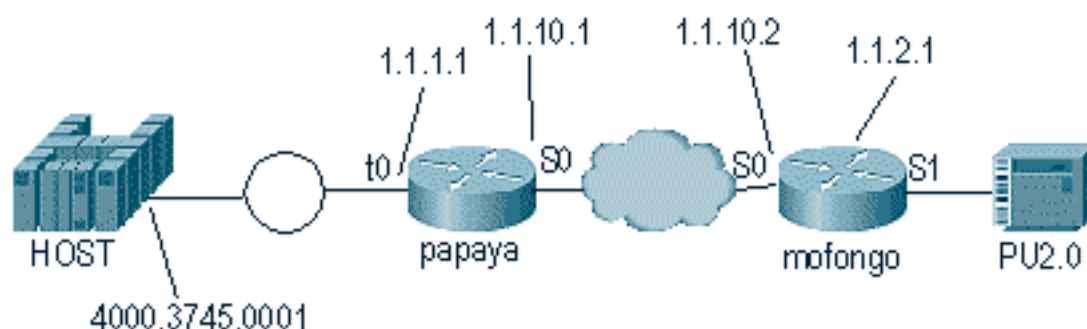
O DLSw oferece um grande avanço para a tradução de mídia porque suporta PU2.1. Isso permite que ele tenha conversão de SDLLC para LLC2 para controladores, como o 5494 e o 5394 (com opção de atualização para PU2.1 - IBM RPQ 8Q0775) para AS/400s. Isso remove a necessidade de STUN e linhas AS/400 multiponto inválidas.

Os parâmetros de configuração para tradução de mídia DLSw são um pouco diferentes dos

parâmetros SDLLC. Há um comando DLSw adicionado, o restante são comandos SDLC. Conclua estes passos para configurar a tradução de mídia DLSw:

1. Emita o comando **encapsulation sdlc** para alterar o encapsulamento serial para SDLC. Como você vai encerrar a linha SDLC no roteador, o roteador deve atuar como primário para fins de pesquisa. Isso é diferente do STUN porque o principal será o HOST ou o AS/400.
2. Emita o comando **sdlc role primary** para alterar a função do roteador para primary na linha SDLC.
3. Emita o comando **sdlc address xx** para configurar o endereço de pesquisa do SDLC. É aqui que o DLSw difere do SDLLC. No SDLLC, você especifica comandos com a palavra-chave **sdllc**. No DLSw, especifique comandos com a palavra-chave **sdlc**.
4. Emita o comando **sdlc vmac xxxx.xxxx.xx00** para configurar o endereço MAC virtual para o controlador SDLC. Esse parâmetro informa ao roteador o endereço MAC virtual desse controlador SDLC no ambiente LLC2. Lembre-se de deixar o último byte definido como **00** porque o endereço de pesquisa é adicionado ali (**endereço sdlc**).
5. Emita o comando **sdlc xid nn xxxxxxx** para configurar o XID para esta PU 2.0. Neste comando, **nn** é o endereço de pesquisa da controladora e **xxxxxxx** é o XID para esta PU2.0 (o IDBLOCK e o IDNUM que é codificado no nó principal do switch em VTAM). **Observação:** se você tem uma PU2.1, há negociação de XID. Assim, o comando muda.
6. Emita o comando **sdlc xid nn xid-poll** para configurar o XID para esta PU 2.1. Neste comando, **nn** é o endereço de pesquisa da estação.
7. Emita o comando **sdlc partner xxxx.xxxx.xxxx nn** para configurar o endereço MAC do parceiro do roteador. Nesse comando, **nn** é o endereço de pesquisa do controlador em questão. É importante especificar o endereço do controlador, pois em linhas multiponto pode haver um controlador direcionado para um host e outro controlador direcionado para um host diferente.
8. Emita o comando **sdlc dlsw nn** para configurar DLSw para o controlador específico. Neste comando, **nn** é o endereço de pesquisa da controladora ou controladoras no multidrop. Esse comando permite especificar vários endereços de pesquisa em um comando. **Nota:** Cuidado com o bug #CSCdi75481. Consulte o [Bug Toolkit](#) (somente clientes [registrados](#)) para obter mais informações. Se o comando **sdlc dlsw nn** não for removido antes de alterar o endereço SDLC do roteador, o código CLS não poderá comunicar corretamente DLSw com a interface SDLC. Isso faz com que a interface se comporte como se nada tivesse sido configurado. Este bug foi corrigido no Cisco IOS® Software Release 11.1(8.1) 11.1(8.1)AA01(01.03) 11.1(8.1)AA01(01.02) e mais recente.

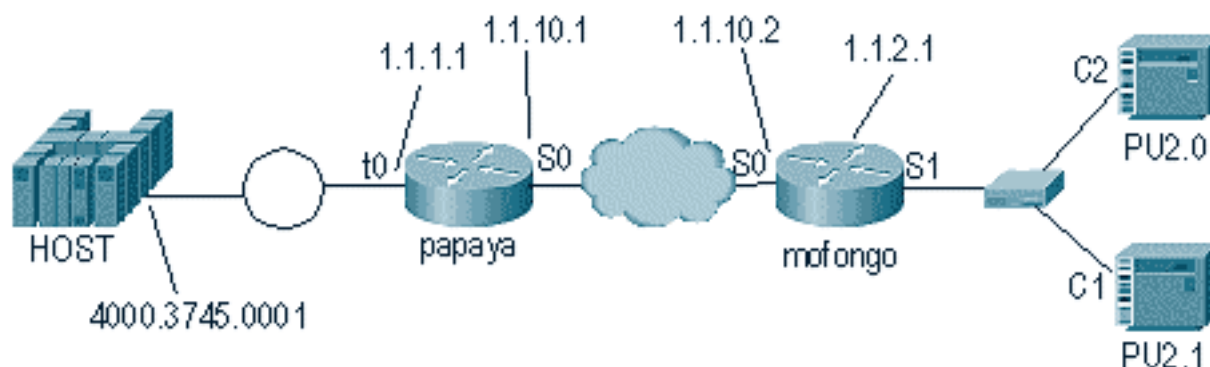
Um exemplo de configuração para um controlador DLSw SDLC PU2.0 é exibido.



Papaya	Mofongo

<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 sdhc dlsw c1 </pre>
---	--

Ao codificar um multidrop, lembre-se de que PU2.1s é mais inteligente e tem mais informações para trocar do que um dispositivo PU2.0 normal. Isso é importante ao configurar um ambiente multidrop, pois você precisa codificar a linha como primária para o dispositivo PU2.0. Você também precisa adicionar o **xid-poll** para o endereço SDLC do dispositivo PU2.1 para que o código entenda o que fazer com cada um dos controladores. Este é um exemplo da configuração.



Papaya	Mofongo
<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>

comandos show

Consulte [Data-Link Switching Plus](#) para obter mais informações sobre os comandos show usados para tradução de mídia DLSw.

Depurando pacotes SDLC durante DLSw/SDLC para PU2.1

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

A primeira coisa a ocorrer é um **XID**, ou **BF** para o endereço de broadcast SDLC de **FF**.

```
Serial2 SDLC output      FFBF
```

Em seguida, um **XID** é recebido do 5494. Este é um formato **XID 2** tipo 3, que é exibido nesta saída do comando **debug sdlc packet**:

```
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
```

Estas são explicações de vários campos desse comando:

- **073000DD** — Este campo é o número de ID e ID do bloco configurado no 5494. O ID do bloco e o número da ID são conhecidos como **XID** e enviados pelo 5494 ao peer durante a negociação da sessão.
- **NETA** — Este campo é o Identificador de Rede Ponto-a-Ponto Avançado (APPN - Advanced Peer-to-Peer Networking) que está sendo usado. Normalmente, esse campo corresponde ao NETID configurado no peer. Nesse caso, o peer é um AS/400.
- **CP5494** — Este campo é o nome do Ponto de Controle (CP) do 5494.
- **DD** — Este campo é o endereço SDLC.

Em seguida, o **XID** é recebido do AS/400:

```
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** — Este campo é o ID do bloco e o número de ID do AS/400.
- **RTP400A** — Este campo é o nome CP do AS/400. O nome do CP é encontrado no arquivo Display Network Attributes (DSPNETA) no AS/400.

Em seguida, o SNRM (93) e o UA (73) são exibidos na linha. Antes do SNRM, o roteador sempre usa o endereço de broadcast. A partir desse ponto, o roteador sempre usa o endereço de pesquisa real de DD.

```

Serial2 SDLC output      DD93
Serial2 SDLC input      DD73
Serial2 SDLC output      DD11
Serial2 SDLC input      DD11

```

Nesse ponto, a conexão é suspensa devido ao estado fixo do RR (Reciever Ready) entre o roteador e o 5494.

Observação: se o roteador no qual você precisa executar o debug tiver outras interfaces SDLC e você não estiver em buffer de registro, o roteador poderá suspender. Entender quando você pode executar uma depuração no terminal versus o registro vem com experiência. Se você não tiver certeza, use sempre o logging buffered e o comando **show log** para exibir as depurações de SDLC

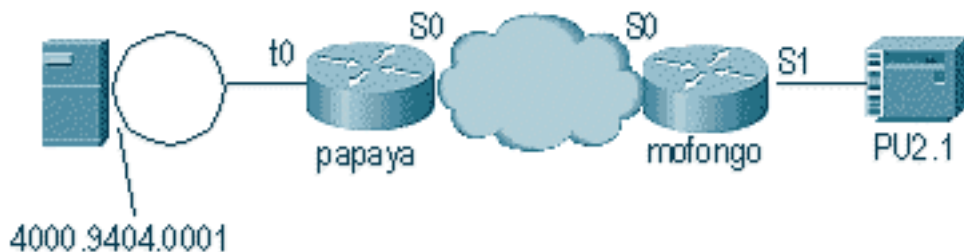
Desligue a controladora no AS/400. Isso permite ver DISK (53) e UA (73) que resultam no lado SDLC da sessão.

```

Serial2 SDLC output      DD53
Serial2 SDLC input      DD73

```

Exemplo de tradução de mídia DLSw



Depois que a interface é ativada e ativada, o roteador inicia o processo determinando a localização do controlador remoto.

```

%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
```

Depois de receber o quadro ICR, o DLsw inicia a máquina de estado finito (FSM) para esta sessão. Isso é realizado pelas mensagens **REQ_OPNSTN.Req** e **REQ_OPNSTN.Cfm** que estão entre DLsw e Cisco Link Services Interface (CLSI).

```
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
```

Após a conversação com CLSI, o DLsw envia quadros **CUR** de inicialização da sessão para o roteador remoto. Isso ocorre somente entre os dois roteadores.

```
%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
```

Quando o circuito é estabelecido, o roteador envia o **XID** que foi armazenado e inicia a troca de **XID**. É importante entender onde estão os **XIDs**. Neste exemplo, o Data-Link Control (DLC)-Id significa que o **XID** veio da estação **DLC** local e o **WAN-XID** veio do roteador remoto, ou estação remota.

```
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
```

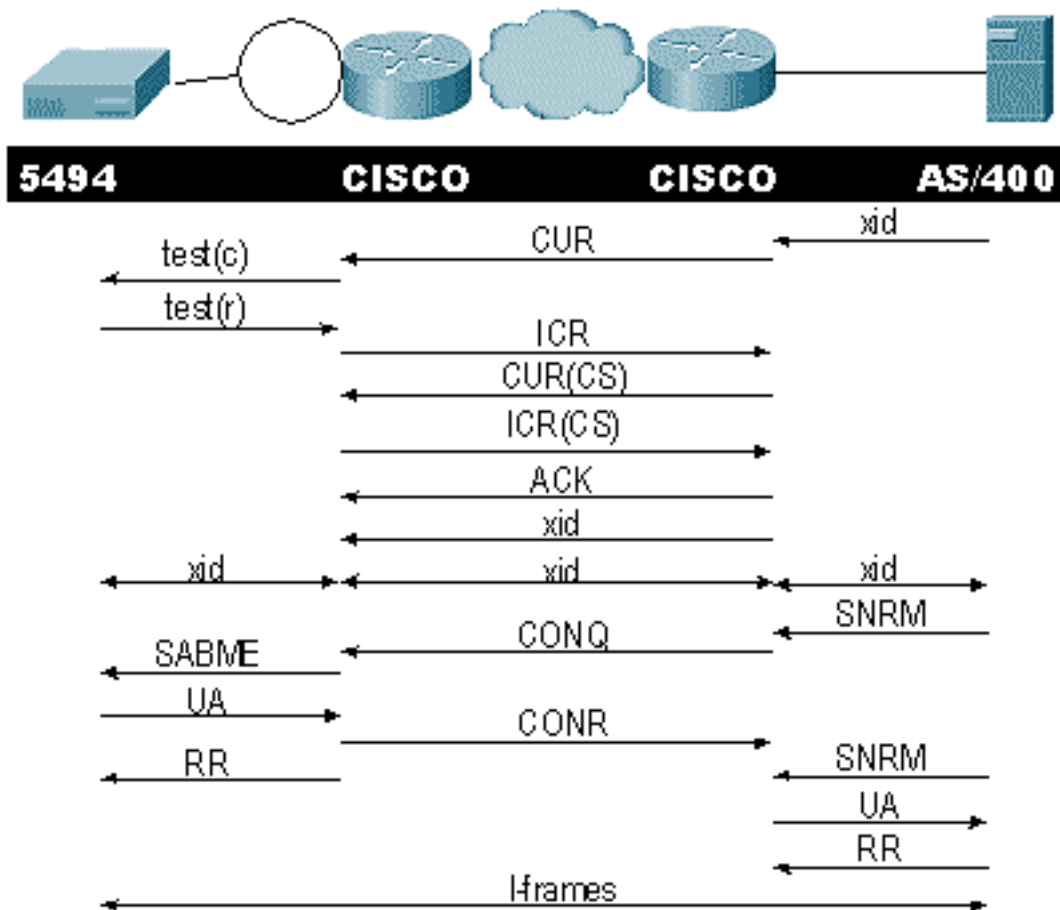
O roteador recebe o **CONQ** do AS/400 (SABME). Isso é traduzido para a linha serial como um **SNRM**. Em seguida, o roteador aguarda o **UA** na linha serial (**CONNECT.Cfm**) e envia o **CONR** para o outro lado. Isso altera o estado da sessão para **CONNECTED**.

```
%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 executando tradução de mídia reversa](#)

Outra configuração comum é a **sdllc inversa**. Em SDLLC reverso, a estação primária é conectada por uma linha SDLC ao roteador. Isso é geralmente visto em ambientes de host quando os usuários desejam migrar o host para um anexo Token Ring. O SDLLC inverso altera a maneira como o DLsw lida com a linha SDLC porque geralmente não está claro se a PU remota está ativa ou não.



Primeiro, como o AS/400 é principal neste caso, ou definido para ser negociável na função, ele precisa iniciar a sessão. Quando o AS/400 envia o primeiro XID depois que a linha serial se torna operacional, o roteador inicia o processo de pesquisa do controlador remoto. Depois que o circuito é configurado, a negociação de XID pode começar na linha.

Quando a negociação XID é concluída, o AS/400 envia o SNRM ao roteador. Isso faz com que o roteador envie o CONQ e espera o CONR do roteador remoto. O roteador não pode responder com o UA até ver um SNRM e depois de receber o CONR. Em quase todas as versões do código, o roteador espera 30 segundos até que expire a sessão. Isso ocorre em relação ao recebimento de SNRMs do dispositivo primário quando o dispositivo primário recebe o CONR do host remoto.

No código mais recente do Cisco IOS 11.1, os padrões foram alterados para um minuto em vez de 30 segundos. No AS/400, esse tempo limite é chamado de **temporizador de resposta não produtiva** e o padrão é 32 segundos.

Tradução de mídia DLSw local



%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
```

A primeira coisa que você percebe no DLSw local é o XID do lado serial. Esse XID precisa ser armazenado até que o roteador envie os quadros/respostas de teste de LLC.

```
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
```

Em seguida, a estação de teste sai do roteador e a resposta retorna do AS/400. Agora, o roteador pode criar o FSM local.

Observação: lembre-se de que esta é uma sessão local.

```
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
```

Depois que o roteador confirmar localmente que o FSM está pronto, ele pode enviar o XID para o parceiro. Neste exemplo, o parceiro é o AS/400 (**ID.Req**).

```
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
```

Em seguida, um XID é recebido do Token Ring. O **ID.Ind** tem um comprimento de 108. O roteador encaminha esse XID para o parceiro neste cenário, que é a linha SDLC. Isso é indicado pelo

ID.Req que foi enviado. Cada vez que o roteador recebe um pacote, ele precisa iniciar a máquina linear de estado finito (LFSM). Essa é a chave para entender essa depuração, pois ela informa onde ela começa e quais pontos ela está indo.

```
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
```

Em seguida, a resposta XID é recebida da linha serial e é encaminhada ao parceiro (a estação Token Ring neste exemplo). Isso continua até que a troca XID seja concluída para este dispositivo PU2.1.

```
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
```

Após a troca XID, o roteador recebe um **SABME** do AS/400 através do **CONNECT.Ind**. Isso instrui o roteador a enviar um **CONNECT.Req** à linha de SDLC, que é o SNRM. Em seguida, uma mensagem **CONNECT.Cfm** (UA) é recebida da linha serial, o que faz com que o código DLSw envie um **CONNECT.Rsp** (UA) ao AS/400.

```
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
```

A sessão quando o controlador (SDLC) é desligado é exibida.

```
%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
```

Em seguida, o roteador envia um DISCO ao AS/400 (DISCONNECT.Rsp). Em seguida, ele começa a derrubar o circuito local.

```
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
```

Depois que o roteador recebe o **DISCONNECT.Ind** (UA) do AS/400, ele termina de limpar a sessão e move-se para um estado de desconexão.

[Informações Relacionadas](#)

- [Tecnologias IBM](#)
- [Suporte Técnico e Documentação - Cisco Systems](#)