

SDLC から LLC へのネットワークメディア変換の説明とトラブルシューティング

内容

[概要](#)

[前提条件](#)

[要件](#)

[使用するコンポーネント](#)

[表記法](#)

[SDLLC](#)

[SDLC の設定](#)

[SDLLC の設定](#)

[SDLLC のデバッグ](#)

[DLSw メディア変換](#)

[show コマンド](#)

[PU2.1 用 DLSw/SDLC 実行時の SDLC パケットのデバッグ](#)

[DLSw メディア変換例](#)

[逆メディア変換を実行する DLSw](#)

[ローカル DLSw メディア変換](#)

[関連情報](#)

概要

このドキュメントでは、同期データ リンク制御 (SDLC) から論理リンク制御 (LLC) へのネットワーク メディア変換を理解し、トラブルシューティングするための情報を提供しています。

前提条件

要件

このドキュメントに特有の要件はありません。

使用するコンポーネント

このドキュメントの内容は、特定のソフトウェアやハードウェアのバージョンに限定されるものではありません。

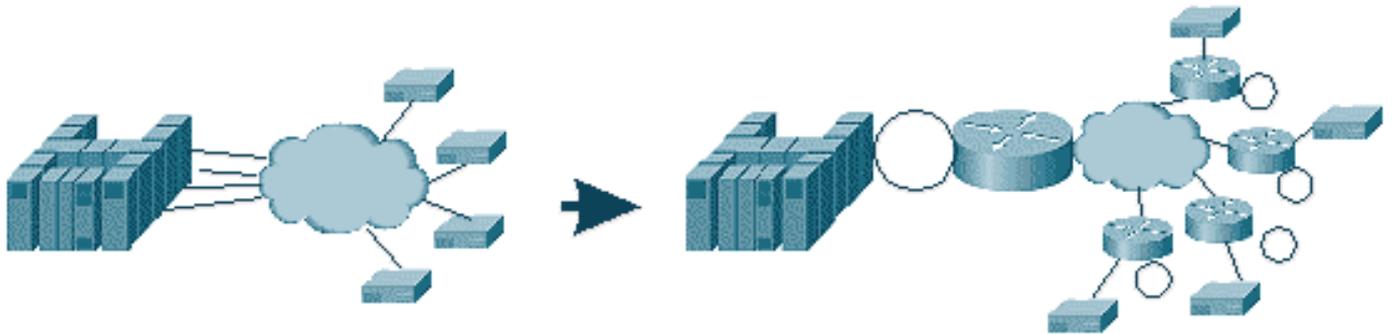
表記法

ドキュメント表記の詳細は、『[シスコ テクニカル ティップスの表記法](#)』を参照してください。

SDLLC

SDLC-to-LAN 変換 (SDLLC) は、物理ユニット 2 (PU2.0) デバイスを論理リンク制御タイプ 2 (LLC2) セッションに変換するために使用されます。これはフロント エンド プロセッサ (FEP) の単一のトークン リンク ポートにデータを送信する大量のリモート コントローラがある場合に非常に便利です。

この図の左側が、遠隔地に繋がる多くの SDLC の回線を持つ FEP を示します。この図の右側は、シスコのルータでの同じ状況を示します。



ルータでは、FEP はトークン リンク インターフェイスだけを持つことができます。その時点から、通常のソースルートブリッジング (SRB) 同様にホストへの SDLLC を実行する複数の遠隔地があります。

注：LLCからSDLCへの変換にSDLLCを使用すると、物理ユニットタイプ2.1(PU2.1)ではなく、PU2.0デバイスにのみ適用されます。PU2.1は、データリンクスイッチング (DLSw) でサポートされています。

SDLLC を設定するには、ルータで SRB が必要です。SRB を設定する方法の詳細については、『[ローカル ソースルートブリッジングのトラブルシューティング](#)』を参照してください。

SDLC の設定

SDLLC では SDLC インターフェイスから変換するため、最初に SDLC が正しく設定されている必要があります。次の手順を実行して、SDLC を設定します。

1. `encapsulation sdlc` コマンドを発行して SDLC へのシリアル カプセル化を変更します。
2. `sdlc role primary` コマンドを発行して SDLC 回線内のルータのロールをプライマリに変更します。注：シリアルトンネリング(STUN)環境では、プライマリとセカンダリの役割があります。詳細については、『[シリアルトンネリング \(STUN \) の設定およびトラブルシューティング](#)』を参照してください。
3. `sdlc address xx` コマンドを発行して SDLC ポーリング アドレスを設定します。

SDLLC の設定

SDLLC を設定するために最初に発行するコマンドは `traddr` です。このコマンドは LLC2 環境で SDLC が変換する対象を定義します。次の手順を実行して、SDLLC を設定します。

1. `sdllc traddr xxxx.xxxx.xx00 lr bn tr` コマンドを発行してシリアル インターフェイスでの SDLLC メディア変換をイネーブルにします。このコマンドによって、ルータに SDLC ステ

ーションの仮想 MAC アドレスが伝えられます。次にこのコマンドによってローカル リング番号 (lr)、ブリッジ番号 (bn.)、およびターゲットリング番号 (tr) が指定されます。lr は、ネットワーク内で一義的である必要があります。bn は 1 ~ 15 の値にすることができます。tr はルータの仮想リングである必要があります。ローカル SDLLC を設定している場合は、これが仮想リングまたはルータ内のインターフェイス (トークンリング インターフェイスに接続されている物理的リング) をポイントするようにすることができます。注: このコマンドの MAC アドレスの最後の 2 桁は 00 です。ルータは、この行の SDLC アドレスを挿入するためにこれらの桁を使用するため、traddr の最後の 2 桁を設定できません。最後の 2 桁を指定した場合、ルータによって SDLC アドレスに置き換えられます。そのため、ホストはその MAC アドレスに対して応答しません。たとえば、traddr MAC が 4000.1234.5678 および SDLC アドレスが 0x01 として設定されている場合、ルータは LLC ドメイン内の SDLC のデバイスを表すために 4000.1234.5601 の MAC を使用します。さらに、traddr MAC はトークンリング フレームと同じ形式の非標準形式で示されます。

2. `sdllc xid address xxxxxxxx` コマンドを発行して仮想記憶通信アクセス方式 (VTAM) 値に一致する SDLC ステーションの交換識別子 (XID) 値を指定します。これは VTAM 内のスイッチ メジャー ノードの IDBLK と IDNUM から決定されます。これが一致しない場合、XID 交換が失敗します。
3. `sdllc partner mac-address sdllc-address` コマンドを発行して SDLLC への接続をイネーブルにします。これによって、通常はホストであるパートナーの MAC アドレスが指定されます

簡単な SDLLC の例の設定を示します。SDLC 接続コントローは、FEP へのローカル トークンリング接続デバイスとして認識されます。



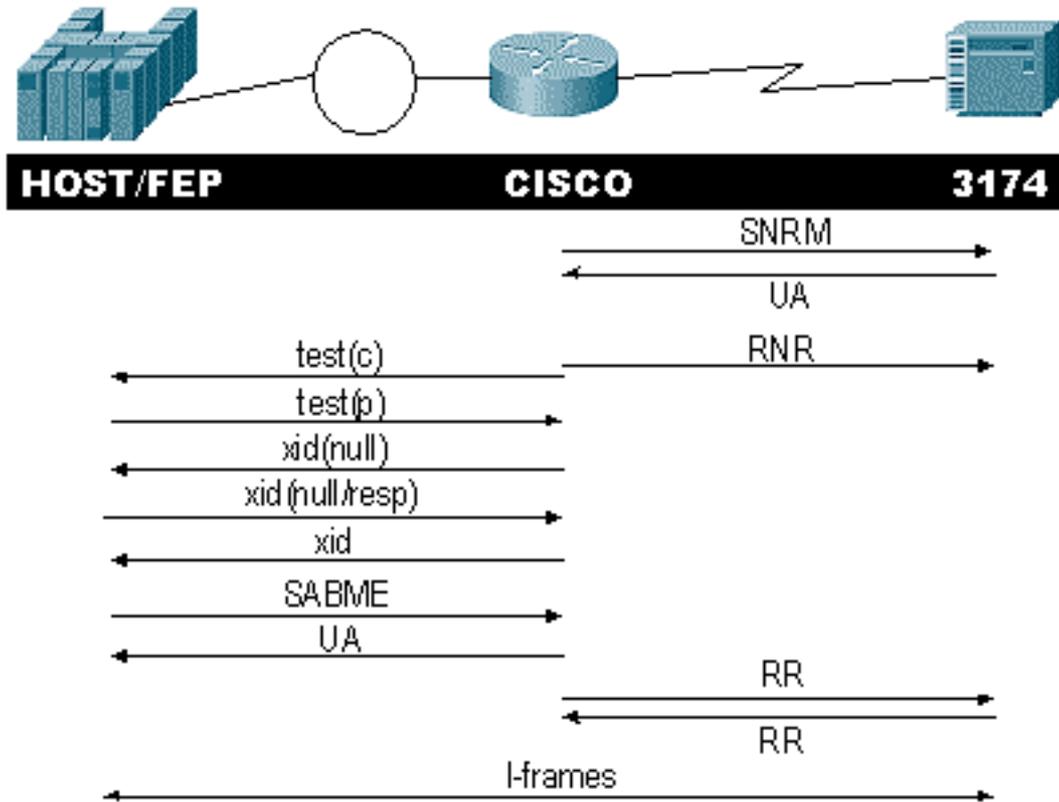
Papaya	Mofongo
<pre>source-bridge ring-group 100 source-bridge remote- peer 100 tcp 1.1.1.1 source-bridge remote- peer 100 tcp 1.1.2.1 local-ack interface tokenring 0 ip address 1.1.3.1 255.255.255.0 source-bridge 33 2 100 source-bridge spanning interface loopback 0 ip address 1.1.1.1 255.255.255.0</pre>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

SDLLC のデバッグ

SDLLC 問題では、次の 2 つの異なる環境をトラブルシュータする必要があります。SDLC の環境

、およびフレームを変換している論理リンク制御タイプ 2 (LLC2) の環境。1 種類のコントローラしか持てないため、SDLLC のデバッグはデータリンク スイッチング (DLSw) /SDLC より理解が容易です。

最初に、次の特定のセッション起動のフローに注目します。



コントローラからの正規応答モードの設定 (SNRM) の応答を確認してください。ルータは、SDLC 部分が起動し実行するまで LLC 部分を開始しません。

次のコマンドを発行して SNRM 応答を確認します。

- `sdlc_state`
- `sdllc_state`

この例では、回線の状態を SNRMSSENT に変更するコントローラに SNRM が送信されます。ルータがこの状態のままの場合は、コントローラから確認応答 (UA) を受信していません。これは SDLC 回線になにかエラーがあることを意味します。これが発生すると、デバッグは、次のように表示されます：

```
%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
```

ルータが UA を受信すると、[sdlc_state] は [SNRM_SENT] から [CONNECT] に変わります。次に、SDLLC の状態が [SDLC_PRI_WAIT] から [NET_UP_WAIT] に変わります。これが発生すると、ルータは接続の LLC 側の起動を開始できます。最終アクションは SDLC 回線への Receive Not Ready (RNR) の送信開始です。これによって LLC 側が動作可能になるまでコントローラの情報の送信をディセーブルにします。

次に、ルータはパートナーのロケーションを見つけるために EXPLORER を送信します。

```
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
```

上記の出力は、送信および受信されるテスト ポーリングを示します。この例にはローカルに接続されたコントローラおよびトークン リングがあるため、テスト ポーリングはパートナー アドレスを検索するルータから発生します。ルータがテスト フレームを受信した後、XID 交換が開始されます。ルータは、show rif コマンドで確認できるセッションのルーティング情報フィールド (RIF) をキャッシュします。これは PU2.0 であるため、ルータは XID のヌルへの応答の後でホストに形式 0 タイプ 2 の XID を送信します。

```
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 交換の後、ルータはホストから非同期平衡モードの設定 (SABME) を受信します。これで起動手続きが終了し、ルータがホストに UA で応答します。ここで、SDLC 回線が USBUSY から CONNECT に変わり、1 フレームがルータを通過できます。

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

[DLSw メディア変換](#)

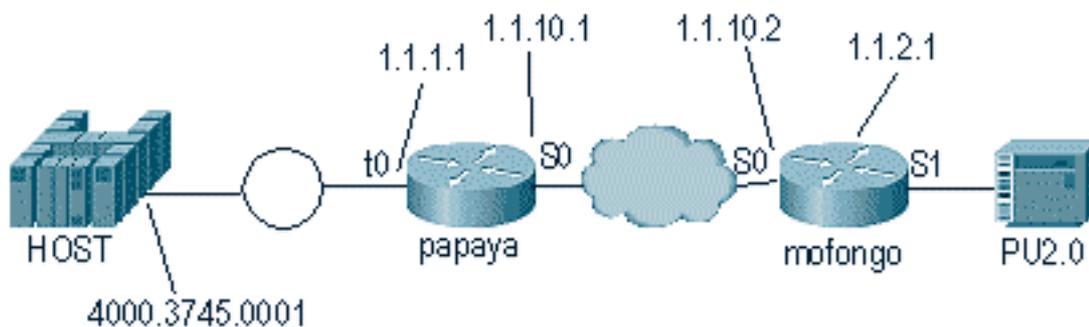
DLSw は PU2.1 をサポートしているため、メディア変換に大きな効果を発揮します。これにより、5494 や 5394 (PU2.1 - IBM RPQ 8Q0775 へのアップグレードオプション付き) などのコントローラに SDLLC から LLCLLC-LLC-LLC 100 秒これによって STUN および AS/400 の不正なマルチポイント回線が不要になります。

DLSw メディア変換の設定パラメータと SDLLC パラメータとの違いはわずかです。DLSw コマンドが 1 つ追加されていて、残りは SDLC コマンドです。次の手順を実行して、DLSw メディア

変換を設定します。

1. `encapsulation sdhc` コマンドを発行して SDLC へのシリアル カプセル化を変更します。ルータの SDLC 回線を終端させようとしているので、ルータはポーリングの目的のプライマリとして動作する必要があります。プライマリはホストまたは AS/400 になるので、これは STUN とは異なります。
2. `sdhc role primary` コマンドを発行して SDLC 回線内のルータのロールをプライマリに変更します。
3. `sdhc address xx` コマンドを発行して SDLC ポーリング アドレスを設定します。これが DLSw が SDLLC と異なる部分です。SDLLC では、`sdhc` キーワードと一緒にコマンドを指定します。DLSw では、`sdhc` キーワードと一緒にコマンドを指定してください。
4. `sdhc vmac xxxx.xxxx.xx00` コマンドを発行して SDLC コントローラの仮想 MAC アドレスを設定します。このパラメータによって、LLC2 環境でのこの SDLC コントローラの仮想 MAC アドレスがルータに伝えられます。ポーリング アドレスが最後の 2 バイト (SDLC アドレス) に追加されるため、最後の 2 バイトを 00 に設定しておくことを忘れないでください。
5. `sdhc xid nn xxxxxxxx` コマンドを発行してこの PU 2.0 の XID を設定します。このコマンドで *nn* はコントローラのポーリングアドレスで、*xxxxxxx* は、この PU2.0 の XID です (VTAM のスイッチメジャーノードでコードされる IDBLOCK および IDNUM)。注：PU2.1 を使用している場合、XID のネゴシエーションがあります。したがって、コマンドが変化します。
6. `sdhc xid nn xid-poll` コマンドを発行して、この PU 2.1 の XID を設定します。このコマンドでは、*nn* はステーションのポーリング アドレスです。
7. `sdhc partner xxxx.xxxx.xxxx nn` コマンドを発行してルータ パートナーの MAC アドレスを設定します。このコマンドでは、*nn* は、対象のコントローラのポーリング アドレスです。マルチポイント回線では 1 台のコントローラが 1 台のホストに送信し、別のコントローラが別のホストに送信することがあるので、コントローラ アドレスを指定することが重要です。
8. `sdhc dlsw nn` コマンドを発行して特定のコントローラの DLSw を設定します。このコマンドでは、*nn* はコントローラまたはマルチドロップ内のコントローラのポーリング アドレスです。このコマンドでは複数のポーリング アドレスを 1 つのコマンドで指定することができます。注：バグ #CSCdi75481 に注意してください。詳細は [Bug Toolkit](#) (登録 ユーザ専用) を参照してください。ルータの SDLC アドレスを設定する前に `sdhc dlsw nn` コマンドが削除されていない場合、CLS コードは SDLC インターフェイスで DLSw と正常に通信できません。これは、インターフェイスが何も設定されていないかのように動作する原因になります。この不具合は、Cisco IOS® のソフトウェア リリース 11.1(8.1) 11.1(8.1)AA01(01.03) 11.1(8.1)AA01(01.02) 以降で修正されています。

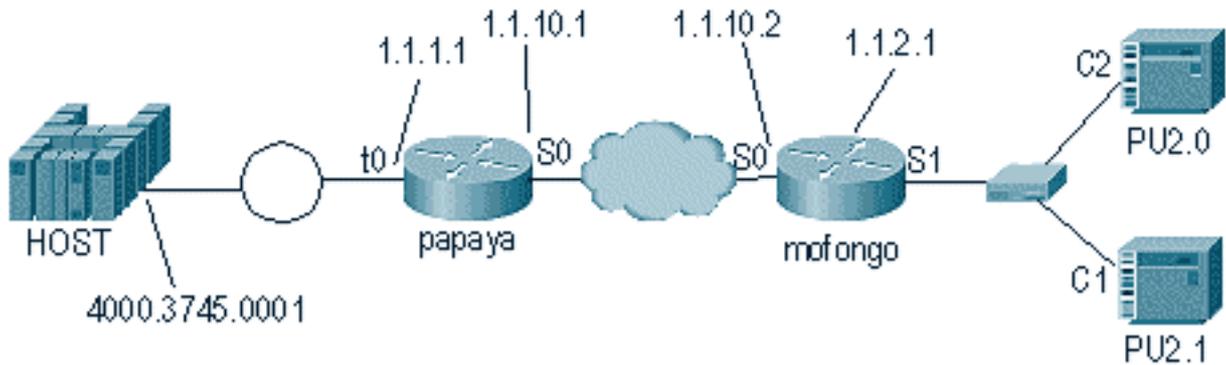
DLSw SDLC PU2.0 コントローラの設定例を示します。



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>
---	--

マルチドロップをコーディングする場合は、通常の PU2.0 デバイスよりも PU2.1 が優秀で交換する情報が多いことを念頭においてください。回線を PU2.0 デバイスに対してプライマリとしてコーディングする必要があるため、これはマルチドロップ環境を設定する際に重要になります。また、PU2.1 デバイスの SDLC アドレスの xid-poll を追加する必要があり、これによってコードがコントローラそれぞれで実行される内容が解釈されます。次に設定の例を示します。



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>

show コマンド

DLSw メディア変換に使用する show コマンドの詳細については、『[データリンク スイッチング プラス](#)』を参照してください。

PU2.1 用 DLSw/SDLC 実行時の SDLC パケットのデバッグ

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

最初に起きることは XID、つまり BF から FF の SDLC ブロードキャスト アドレスへの送信です。

```
Serial2 SDLC output      FFBF
```

次に、XID を 5494 から受信します。これは、この debug sdhc packet コマンドの出力に表示される XID 形式 2 タイプ 3 です：

```
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
```

次はこのコマンドのいくつかのフィールドの説明です。

- **073000DD**：このフィールドは、5494で設定されているブロックIDとID番号です。ブロック ID および ID 番号は XID と呼ばれ、セッション ネゴシエーション中に 5494 によってピアに送信されます。
- **NETA**：このフィールドは、使用されているAdvanced Peer-to-Peer Networking(APPN)Network Identifier(NETID)です。通常、このフィールドはピアに設定されている NETID に一致します。この場合、ピアは AS/400 です。
- **CP5494**：このフィールドは、5494のコントロールポイント(CP)名です。
- **DD**：このフィールドはSDLCアドレスです。

次に、XID を AS/400 から受信します。

```
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** : このフィールドは、AS/400のブロックIDとID番号です。
- **RTP400A** : このフィールドは、AS/400のCP名です。CP名は、AS/400のDisplay Network Attributes(DSPNETA)ファイルにあります。

次に、SNRM (93) および UA (73) がその行に表示されます。SNRM の前は、常にルータはブロードキャストアドレスを使用します。ここから先は、ルータは常に DD の実際のポーリングアドレスを使用します。

```

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

```

この時点で、ルータと 5494 間の Receiver Ready (RR) の状態が安定しているのでこの接続を停止します。

注：デバッグを実行する必要があるルータに他のSDLCインターフェイスがあり、ロギングがバッファされていない場合は、ルータが一時停止することがあります。端末でデバッグを実行する時期とロギングについての理解は経験によって身に付きます。確信が持てない場合は、常にバッファされたロギングおよび show log コマンドを使用して SDLC のデバッグを表示します。

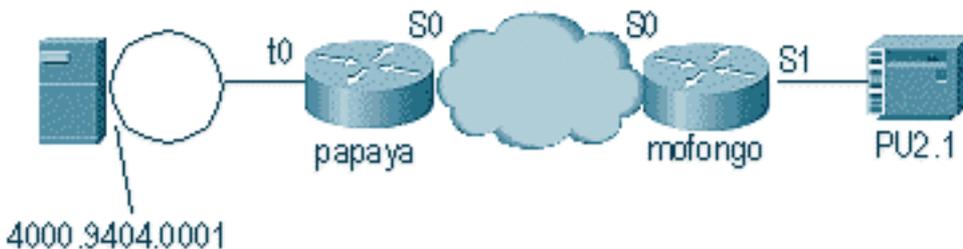
AS/400でコントローラをオフにします。これにより、セッションのSDLC側で発生する DISK(53)とUA(73)を確認できます。

```

Serial2 SDLC output      DD53
Serial2 SDLC input      DD73

```

DLSw メディア変換例



インターフェイスが起動し終わった後、ルータはリモートコントローラの場所を決定してプロセスを開始します。

```

%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 のフレームを受信した後、DLswはこのセッションに対する有限状態マシン (FSM) を起動します。これは、DLsw と Cisco Link Services Interface (CLSI) 間の REQ_OPNSTN.Reg および REQ_OPNSTN.Cfm のメッセージで実行されます。

```
DLsw: START-FSM (488636): event:DLC-Id state:DISCONNECTED
DLsw: core: dlsw_action_a()
DISP Sent : CLSI Msg : REQ_OPNSTN.Reg 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
```

CLSI との通信後、DLsw がリモート ルータにセッション起動の CUR フレームを送信します。これらは、この 2 台のルータ間でのみ発生します。

```
%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
```

一度回線が確立されると、ルータが保存される XID を送信し XID 交換を開始します。XID がある場所を理解することが重要です。この例では、XID がローカル DLC ステーションからのもので WAN-XID がリモート ルータまたはリモート ステーションからのものであることをデータリンク制御 (DLC) -ID が意味します。

```
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.Reg 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
```

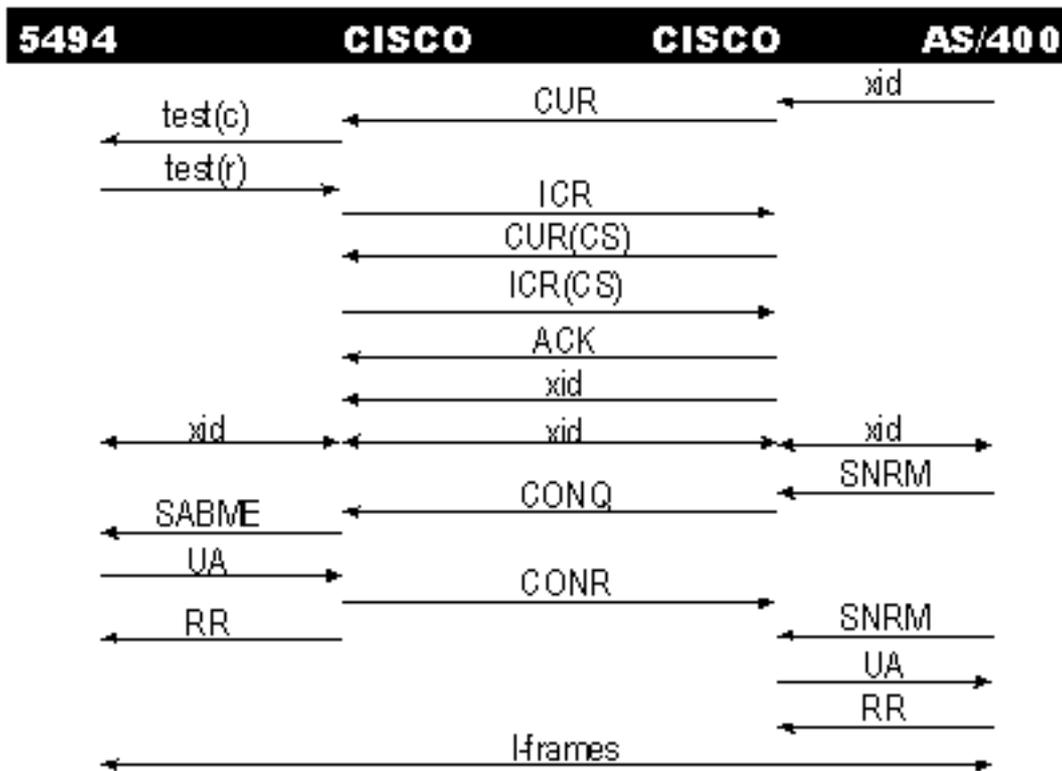
ルータは AS400 (SABME) から **CONQ** を受信します。これは、SNRM としてシリアル回線に変換されます。次に、ルータはこのシリアル回線での UA (**CONNECT.Cfm**) を待ち、次に **CONR** をもう一方の側に送信します。これによってセッションの状態が **CONNECTED** に変わります。

```
%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
```

[逆メディア変換を実行する DLSw](#)

もう一つの一般的な設定は逆 SDLLC です。逆 SDLLC では、プライマリステーションがルータへの SDLC 回線を介して接続されます。これは、ユーザがホストをトークンリング接続に移行する際のホスト環境で通常見られます。SDLC 回線ではリモート PU がアクティブかどうか不明なことが頻繁にあるため、逆 SDLLC は DLSw が SDLC 回線を処理する方法を変更します。



最初に、AS/400 は、この場合ではプライマリであるか、またはこのロールでネゴシエート可能に設定するので、セッションを開始する必要があります。シリアル回線が動作可能になった後に AS/400 が最初の XID を送信すると、ルータはリモート コントローラの検索プロセスを開始します。回線が設定された後、XID ネゴシエーションをその回線で開始できます。

XID ネゴシエーションが終了すると、AS/400 が SNRM をルータに送信します。これにより、ルータは CONQ を送信しリモート ルータからの CONR を待ちます。ルータは、SNRM を確認し CONR を受信するまで UA に応答できません。コードのほとんどすべてのバージョンでは、ルータはセッションをタイムアウトするまで 30 秒待ちます。これは、プライマリ デバイスがいったんリモート ホストから CONR を受信したときのプライマリ デバイスからの SNRM の受信に関係しています。

最新の Cisco IOS 11.1 のコードでは、デフォルトは 30 秒から 1 分に変更されました。AS/400 では、このタイムアウトは非生産的応答タイマーと呼ばれ、デフォルトは 32 秒です。

ローカル DLSw メディア変換



%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 のローカルで最初に注意することは、シリアル側からのXIDです。このXIDは、ルータがLLCテストフレーム/応答通過を送信するまで保存される必要があります。

```
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
```

次に、テストステーションがルータを離れ、応答がAS/400から返されます。これで、ルータはローカルFSMを作成できます。

注：これはローカルセッションであることに注意してください。

```
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
```

FSMが使用可能であることをルータがローカルに確認した後、ルータはパートナーにXIDを送信できます。この例では、パートナーはAS400 (ID.Req)です。

```
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
```

次に、XIDをトークンリングから受信します。ID.Indの長さは108です。ルータはこのXIDをパートナーに転送します。これはSDLC回線です。これは、送信されたID.Reqで識別されます。ルー

タがパケットを受信するたびに、線形有限状態マシン (LFSM) を開始する必要があります。LFSM は開始する場所および実行中のポイントの情報を提供するので、このデバッグを理解するために重要です。

```
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
```

次に、XID 応答がシリアル回線から受信されてパートナー (この例ではトークンリングステーション) へ転送されます。これは、XID 交換がこの PU2.1 デバイスに対して終了するまで続行されます。

```
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 交換の後、ルータは **CONNECT.Ind** を介して AS/400 から SABME を受信します。これはルータに対し、SDLC 回線に **CONNECT.Req** (SNRM) を送信するように指示します。そのあとシリアル回線から **CONNECT.Cfm** (UA) を受信し、それによって DLSw コードは AS/400 に **CONNECT.Rsp** (UA) を送信します。

```
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
```

コントローラ (SDLC) をシャットダウンするときのセッションが表示されます。

```
%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
```

次に、ルータは AS400 (DISCONNECT.Rsp) に DISC を送信します。次に、ルータはローカル回線の切断を開始します。

```
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
```

ルータが AS/400 から DISCONNECT.Ind (UA) を受信した後、ルータはセッションの削除を終了し、非接続ステートに移行します。

関連情報

- [IBM テクノロジー](#)
- [テクニカル サポートとドキュメント – Cisco Systems](#)