



# 1-Port Channelized OC-12/STM-4 Line Card の設定

この章では、Cisco 10000 シリーズ 1-Port Channelized OC-12/STM-4 Line Card（以下、1 ポート チャネライズド OC-12/STM-4 ラインカード）の設定手順について説明します。

1 ポート チャネライズド OC-12/STM-4 ラインカードはフラクショナル T1 ( $n \times$  DS0)、DS-1、E1、フラクショナル E1 ( $n \times$  DS0)、および DS3 IP サービスを使用して、ISP カスタマーとネットワークバックボーンを接続します。このラインカードは、(シングルモード) 光ファイバポートを使用して、OC-12 帯域幅速度での全二重動作を実現します。その結果、Cisco 10000 シリーズルータは、1 つのインターフェイスで最大 768 の個別チャネライズドオプションに接続できるようになります。

具体的な内容は次のとおりです。

- [ソフトウェア サポート \(p.6-2\)](#)
- [SONET のチャネル化 \(p.6-3\)](#)
- [SDH のチャネル化 \(p.6-4\)](#)
- [ラインカードのデフォルト値 \(p.6-5\)](#)
- [ラインカードのインターフェイス構文 \(p.6-9\)](#)
- [SONET フレーミングインターフェイスの設定例 \(p.6-10\)](#)
- [SDH フレーミングインターフェイスの設定例 \(p.6-13\)](#)
- [SONET コントローラの SONET または SDH フレーミング用の設定 \(p.6-17\)](#)
- [T3、VT、または AUG コントローラの作成 \(p.6-19\)](#)

## ソフトウェア サポート

表 6-1 に、1 ポート チャネライズド OC-12/STM-4 ライン カードをサポートするリリース系統ごとに、最低限必要な Cisco IOS リリースを示します。

表 6-1 1 ポート チャネライズド OC-12/STM-4 ライン カードのソフトウェア サポート

必要な PRE	最低限必要な Cisco IOS リリース
PRE1	Cisco IOS Release 12.0(9)SL 以降の Cisco IOS Release 12.0SL リリース Cisco IOS Release 12.0(17)ST 以降の Cisco IOS Release 12.0ST リリース Cisco IOS Release 12.0(22)S 以降の Cisco IOS Release 12.0S リリース Cisco IOS Release 12.2(8)BZ 以降の Cisco IOS Release 12.2(8)BZ リリース (SONET モード専用) SONET および SDH モードの両方に対応するには、Cisco IOS Release 12.0(23)S 以降の Cisco IOS 12.0S リリースを使用します。
PRE2	Cisco IOS Release 12.3(7)XI 以降の Cisco IOS 12.3XI リリース SONET および SDH モードの両方に対応するには、Cisco IOS Release 12.2(15)BX 以降の Cisco IOS 12.2BX リリースを使用します。 Cisco IOS Release 12.2(28)SB 以降の Cisco IOS 12.2SB リリース
PRE3	Cisco IOS Release 12.2(31)SB2 以降の Cisco IOS 12.2SB リリース

## ハードウェアとソフトウェアの互換性の確認

Cisco 10000 シリーズ ルータ シャーシに搭載された PRE は、ルータで稼働する Cisco IOS ソフトウェアをサポートしていなければなりません。show version コマンドを使用して、搭載されている PRE のバージョンを確認してください。

機能が Cisco IOS リリースでサポートされているかどうかを確認したり、この機能に関するソフトウェア マニュアルの入手先を調べたり、ルータに搭載されたハードウェアに対する Cisco IOS ソフトウェアの最低限のソフトウェア要件を調べるには、シスコが提供している Cisco.com の Software Advisor ツールを使用してください (<http://www.cisco.com/cgi-bin/Support/CompNav/Index.pl>)。

このツールを使用しても、システム内のライン カードの互換性は検証されません。各ハードウェア ライン カード、モジュール、またはオプションに必要な最低限の Cisco IOS 要件が示されます。

このツールにアクセスするには、Cisco.com に登録する必要があります。

## SONET のチャネル化

1 ポート チャネライズド OC-12/STM-4 ライン カードは、次の SONET チャネル化モードをサポートします。

- STS-1 > DS3/T3
- STS-1 > DS3/T3 > DS1
- STS-1 > DS3/T3 > DS3 サブレート
- STS-1 > VT1.5 > DS1
- STS-1 > VT2 > E1

表 6-2 に、1 ポート チャネライズド OC-12/STM-4 ライン カードに設定された SONET フレーミング コントローラで SONET チャネル化を実現するためのシリアル インターフェイス例を示します。

表 6-2 SONET フレーミング コントローラの SONET インターフェイス例

インターフェイス タイプ	構文
STS-1、 DS3/T3 経由	Router(config)# <b>interface serial</b> <slot>/<subslot>/<port>.<T3#>
STS-1、 DS3/T3 経由、 DS1 にチャネル化	Router(config)# <b>interface serial</b> <slot>/<subslot>/<port>.<T3#>/<ds1#>:[chan]
STS-1、 サブレート DS3/T3 経由	Router(config)# <b>interface serial</b> <slot>/<subslot>/<port>.<T3#>
STS-1、 VT1.5 経由、 DS1 にチャネル化	Router(config)# <b>interface serial</b> <slot>/<subslot>/<port>.<VT#>/<ds1#>:[chan]
STS-1、 VT2 経由、 E1 にチャネル化	Router(config)# <b>interface serial</b> <slot>/<subslot>/<port>.<VT#>/<E1#>:[chan]

## SDH のチャネル化

1 ポート チャネライズド OC-12/STM-4 ライン カードは、次の SDH チャネル化モードをサポートします。

- STM-1 > AU-3 > DS3/T3
- STM-1 > AU-3 > DS3/T3 > DS3 サブレート
- STM-1 > AU-3 > TUG-2 > C-11 > DS1/T1
- STM-1 > AU-3 > TUG-2 > C-12 > E1
- STM-1 > AU-4 > TUG-3 > TUG-2 > C-11 > DS1/T1
- STM-1 > AU-4 > TUG-3 > TUG-2 > C-12 > E1

表 6-3 に、1 ポート チャネライズド OC-12/STM-4 ライン カードに設定された SDH フレーミング コントローラで SDH チャネル化を実現するためのシリアル インターフェイス例を示します。

表 6-3 SDH フレーミング コントローラの SDH インターフェイス例

インターフェイス タイプ	構文
STM-4、 STM-1 経由、 AU-3 経由、 DS3/T3 経由	Router (config)# <b>interface serial</b> <slot>/<subslot>/<port>.<T3#>
STM-4、 STM-1 経由、 AU-3 経由、 サブレート DS3/T3 経由	Router (config)# <b>interface serial</b> <slot>/<subslot>/<port>.<T3#>
STM-4、 STM-1 経由 AU-3 経由、 TUG-2 経由、 C-11 (DS1/T1) 経由	Router (config)# <b>interface serial</b> <slot>/<subslot>/<port>.<AU-3#>/<TUG-2>/<C-11>:[chan]
STM-4、 STM-1 経由 AU-3 経由、 TUG-2 経由、 C-12 (E1) 経由	Router (config)# <b>interface serial</b> <slot>/<subslot>/<port>.<AU-3#>/<TUG-2>/<C-12>:[chan]
STM-4、 STM-1 経由 AU-4 経由、 TUG-3 経由、 TUG-2 経由、 C-11 (DS1/T1) 経由	Router (config)# <b>interface serial</b> <slot>/<subslot>/<port>.<AU-4#>/<TUG-3>/<TUG-2>/<C-11>:[chan]
STM-4、 STM-1 経由 AU-4 経由、 TUG-3 経由、 TUG-2 経由、 C-12 (E1) 経由	Router (config)# <b>interface serial</b> <slot>/<subslot>/<port>.<AU-4#>/<TUG-3>/<TUG-2>/<C-12>:[chan]

## ラインカードのデフォルト値

1 ポート チャネライズド OC-12/STM-4 ラインカードには、コマンドのデフォルト値が 2 セットあります。使用中のデフォルト値セットが SONET フレーミング用であるか、または SDH フレーミング用であるかを判別するには、**framing** コマンドを使用します。

次の表に、コマンドのデフォルト値、およびコマンドを入力する場合の前提となるコンフィギュレーションモードを示します。必要に応じて、どのコマンドがどの回線タイプに適用されるかも示します。また、デフォルト値を変更するために使用するコマンド、および接続のリモートエンドで値を同じ（または反対）に設定する必要があるかどうかを示します。

### SONET フレーミング用のラインカード デフォルト値

ここに記載された表では、1 ポート チャネライズド OC-12/STM-4 ラインカードが SONET フレーミング用に設定されている場合に適用されるコマンドのデフォルト値を示します。

表 6-4 に、1 ポート チャネライズド OC-12/STM-4 ラインカードが SONET フレーミング用に設定されている場合に、SONET コントローラ コンフィギュレーションモードで使用可能なコマンドのデフォルト値を示します。

表 6-5 に、1 ポート チャネライズド OC-12/STM-4 ラインカードが SONET フレーミング用に設定されている場合に、T3 コントローラ コンフィギュレーションモードで使用可能なコマンドのデフォルト値を示します。

表 6-7 に、1 ポート チャネライズド OC-12/STM-4 ラインカードが SONET フレーミング用に設定されている場合に、T3 コントローラについて T1 コントローラ コンフィギュレーションモードで使用可能なコマンドのデフォルト値を示します。

表 6-6 に、1 ポート チャネライズド OC-12/STM-4 ラインカードが SONET フレーミング用に設定されている場合に、Virtual Tributary (VT; 仮想トリビュタリ) コントローラ コンフィギュレーションモードで使用可能なコマンドのデフォルト値を示します。

**表 6-4 SONET フレーミングの場合の SONET コントローラ コンフィギュレーション コマンドのデフォルト値**

コマンド名	デフォルト設定	コマンド構文	リモートエンドの設定
SONET コントローラ コンフィギュレーションモード			
clock source	line	<code>clock source {internal   line}</code>	片側を <b>internal</b> に設定
framing	sonet	<code>framing {sonet   sdh}</code>	同じ
loopback	no loopback	<code>[no] loopback {internal   line}</code>	—
overhead	J0 の場合は 1 S1S0 の場合は 0	<code>overhead {j0   s1s0}</code>	—

表 6-5 SONET フレーミングの場合の T3 コントローラ コンフィギュレーション コマンドのデフォルト値

コマンド名	デフォルト設定	コマンド構文	リモートエンドの設定
T3 コントローラ コンフィギュレーション モード			
cablelength	0	<code>cablelength length</code>	—
channelized	channelized	<code>[no] channelized</code>	—
clock source <sup>1</sup>	internal	<code>clock source {line   internal}</code>	片側を <b>internal</b> に設定
framing	auto-detect	<code>framing {auto-detect   c-bit   m23}</code>	同じ
idle pattern	0X7e (flags)	<code>idle pattern {0x0 to 0xFF}</code>	同じ
overhead	c2 の場合は 4 j1 の場合は 64	<code>overhead {c2   j1}</code>	—

1. SONET クロック ソースのデフォルト値は **line** です。少なくとも接続の片側を **internal** に設定する必要があります。

表 6-6 SONET フレーミングの場合の VT コントローラ コンフィギュレーション コマンドのデフォルト値

コマンド名	デフォルト設定	コマンド構文	リモートエンドの設定
VT コントローラ コンフィギュレーション モード			
overhead	c2 の場合は 2 j1 の場合は 64	<code>overhead {c2   j1}</code>	—

表 6-7 SONET フレーミングの場合の T1 および E1 コンフィギュレーション コマンドのデフォルト値

コマンド名	デフォルト設定	コマンド構文	リモートエンドの設定
t1 clock source	internal	<code>t1 ch_group_number clock source {line   internal}</code>	片側を <b>internal</b> に設定
t1 framing	esf	<code>t1 t1-number framing {esf   sf}</code>	同じ
e1 clock source	internal	<code>e1 ch_group_number clock source {line   internal}</code>	片側を <b>internal</b> に設定
e1 framing	crc4	<code>e1 e1-number framing [crc4   no-crc4]</code>	同じ

## SDH フレーミング用のラインカード デフォルト値

ここに記載された表では、1 ポート チャネライズド OC-12/STM-4 ラインカードが SDH フレーミング用に設定されている場合に適用されるコマンドのデフォルト値を示します。

表 6-8 に、1 ポート チャネライズド OC-12/STM-4 ラインカードが SDH フレーミング用に設定されている場合に、SONET コントローラ コンフィギュレーション モードで使用可能なコマンドのデフォルト値を示します。

表 6-9 に、1 ポート チャネライズド OC-12/STM-4 ラインカードが SDH フレーミング用に設定されている場合に、AU-3 と AU-4-TUG-3 の両方について、AU-3 コントローラ コンフィギュレーション モードで使用可能なコマンドのデフォルト値を示します。

表 6-10 に、1 ポート チャネライズド OC-12/STM-4 ラインカードが SDH フレーミング用に設定されている場合の T3 コマンドのデフォルト値を示します。

表 6-11 に、1 ポート チャネライズド OC-12/STM-4 ラインカードが SONET または SDH フレーミング用に設定されている場合の T1 および E1 コマンドのデフォルト値を示します。

**表 6-8 SDH フレーミングの場合の SONET コントローラ コンフィギュレーション コマンドのデフォルト値**

コマンド名	デフォルト設定	コマンド構文	リモートエンドの設定
SDH の場合の SONET コントローラ コンフィギュレーション モード			
clock source <sup>1</sup>	line	[no] clock source {internal   line}	片側を <b>internal</b> に設定
framing	sonet	framing {sonet   sdh}	同じ
loopback	no loopback	[no] loopback {internal   line}	—
overhead	j0 の場合は 1 S1S0 の場合は 2	overhead {j0   s1s0}	—
aug controller	no aug controller	[no] aug controller <au-3   au-4-tug-3>	—

1. SDH クロック ソースのデフォルト値は **line** です。少なくとも接続の片側を **internal** に設定する必要があります。

**表 6-9 SDH フレーミングの場合の、AU-3 および AU-4-TUG-3 に対応する AU-3/AU-4-TUG-3 コントローラ コンフィギュレーション コマンドのデフォルト値**

コマンド名	デフォルト設定	コマンド構文	リモートエンドの設定
SDH の場合の AU-3/AU-4-TUG-3 コントローラ コンフィギュレーション モード			
clock source	internal	clock source {line   internal}	片側を <b>internal</b> に設定
overhead	c2 の場合は 0 j1 の場合は 16	overhead {c2   j1}	—

表 6-10 SDH フレーミングの場合の T3 コンフィギュレーション コマンドのデフォルト値

コマンド名	デフォルト設定	コマンド構文	リモートエンドの設定
bandwidth		<code>bandwidth bandwidth</code>	—
crc	16	<code>crc {16   32}</code>	—
encapsulation	HDLC	<code>encapsulation {frame-relay   hdlc   ppp}</code>	同じ
idle-character	flags (0x7e)	<code>idle-character {flags   marks}</code>	
keepalive	10 秒	<code>keepalive period</code>	—

表 6-11 T1 および E1 インターフェイス コンフィギュレーション コマンドのデフォルト値

コマンド名	デフォルト設定	コマンド構文	リモートエンドの設定
<b>T1 インターフェイス コンフィギュレーション モード</b>			
crc	16	<code>[no] crc [16   32]</code>	同じ
dsu bandwidth	44210	<code>dsu bandwidth bandwidth</code>	同じ
dsu mode	cisco	<code>dsu mode mode</code>	同じ
encapsulation	HDLC	<code>encapsulation [hdlc   ppp]</code>	同じ
framing	c-bit	<code>framing [c-bit   m13]</code>	同じ
idle character	flags (0x7e)	<code>idle character [flags   marks]</code>	同じ
keepalive	10 秒	<code>keepalive seconds</code>	同じ
mtu	4470	<code>mtu size</code>	同じ
<b>E1 インターフェイス コンフィギュレーション モード</b>			
crc	16	<code>[no] crc [16   32]</code>	同じ
encapsulation	HDLC	<code>encapsulation [hdlc   ppp]</code>	同じ
framing	c-bit	<code>framing [c-bit   m13]</code>	同じ
idle character	flags (0x7e)	<code>idle character [flags   marks]</code>	同じ
keepalive	10 秒	<code>keepalive</code>	同じ
mtu	4470	<code>mtu size</code>	同じ



## ラインカードのインターフェイス構文

コンフィギュレーション コマンド内で非チャネライズドまたは 1 ポート チャネライズド OC-12/STM-4 ラインカードインターフェイスを指定するには、表 6-12 に示された構文を使用します。

表 6-12 1 ポート チャネライズド OC-12/STM-4 ラインカードのインターフェイス構文

インターフェイスのタイプ	スロット	サブスロット	ポート	STS-1 パス (T3)	T1 または E1 番号	チャンネルグループ番号
非チャネライズド	1 ~ 8/	0/	0.	1 ~ 12	—	—
チャネライズド	1 ~ 8/	0/	0.	1 ~ 12/	1 ~ 28 : T1 の場合 1 ~ 32 : E1 の場合	0 ~ 23 (T1 の場合) 0 ~ 31 (E1 の場合)

例 :

- コントローラ コンフィギュレーション モードで T1 インターフェイス 6 を変更 :  

```
Router(config)# controller t3 2/0/0.1
Router(config-controller)# t1 6 command
```
- インターフェイス コンフィギュレーション モードで T1 インターフェイス 6、チャンネルグループ 8 を変更 :  

```
Router(config)# interface serial 2/0/0.1/6:8
Router(config-if)
```

## SONET フレーミング インターフェイスの設定例

1 ポート チャネライズド OC-12/STM-4 ライン カードには STS-1 接続を最大で 12 設定できます。各 STS-1 接続は T3 コントローラまたは VT コントローラとして設定できます。

次に、フルレートおよびサブレート T3 インターフェイス、および T1 インターフェイスとフラクショナル T1 インターフェイスを作成するための基本手順を示します。各 T3 コントローラは単一の T3 インターフェイス（フルまたはサブレート）、28 の T1 インターフェイス、またはそれ以上の個数のフラクショナル T1 として設定できます。

**ステップ 1** framing コマンドを使用して、フレーミング タイプを SONET に設定します。

```
Router(config)# controller sonet 2/0/0
Router(config-controller)# framing sonet
```

**ステップ 2** T3 インターフェイスに STS-1 パスを割り当てます。次の例では、path コマンドを使用して 4 つの T3 インターフェイスを設定します。

```
Router(config-controller)# path 1 controller t3
Router(config-controller)# path 2 controller t3
Router(config-controller)# path 3 controller t3
Router(config-controller)# path 4 controller t3
Router(config-controller)# exit
```

**ステップ 3** インターフェイスを作成します。次の例では、T3 コントローラ（4/0/0.1 ~ 4/0/0.4）ごとに異なるタイプのインターフェイスが作成されます。

### フルレート T3 インターフェイス

a. コントローラ コンフィギュレーション モードを開始します。

```
Router(config)# controller t3 2/0/0.1
Router(config-controller)#
```

b. フルレート T3 インターフェイスを作成するには、**no channelized** コマンドを入力して、T1 インターフェイスを除去します。

```
Router(config-controller)# no channelized
Router(config-controller)# exit
Router(config)#
```

c. インターフェイス コンフィギュレーション モードを開始します。

```
Router(config)# interface serial 2/0/0.1
```

d. ステップ 4 に進みます。

### サブレート T3 インターフェイス

a. コントローラ コンフィギュレーション モードを開始します。

```
Router(config)# controller t3 2/0/0.2
Router(config-controller)#
```

b. サブレート T3 インターフェイスを作成するには、まずフルレート T3 インターフェイスを作成します。

```
Router(config-controller)# no channelized
Router(config-controller)# exit
Router(config)#
```

- c. インターフェイス コンフィギュレーション モードを開始します。このモードでは、**dsu bandwidth** コマンドを使用して、サブレート T3 インターフェイスを作成できます。次の例では、帯域幅が 16000 kbps のサブレート T3 インターフェイスが作成されます。

```
Router(config)# interface serial 2/0/0.2
Router(config-if)# dsu bandwidth 16000
```

- d. ステップ 4 に進みます。

#### フル T1 を含むチャネライズド T3 インターフェイス

- a. コントローラ コンフィギュレーション モードを開始します。

```
Router(config)# controller t3 2/0/0.3
Router(config-controller)#
```

- b. **t1 channel group** コマンドを使用して、T1 インターフェイスを作成します。次の例では、T1 インターフェイス 1 (28 ある中の 1 つ) が、番号が 20 (0 ~ 23 の任意の番号) の単一チャンネルグループで構成されるように定義されます。このチャンネルグループには、24 のすべての DS0 タイムスロットが含まれます。

```
Router(config-controller)# t1 1 channel-group 20 timeslots 1-24
```

- c. 作成したチャンネルグループに対して、インターフェイス コンフィギュレーション モードを開始します。

```
Router(config)# interface serial 2/0/0.3/1:20
```

- d. ステップ 4 に進みます。

#### フラクショナル T1 を含むチャネライズド T3 インターフェイス

- a. コントローラ コンフィギュレーション モードを開始します。

```
Router(config)# controller t3 2/0/0.4
Router(config-controller)#
```

- b. **t1 channel group** コマンドを使用して、フラクショナル T1 インターフェイスを作成します。次の例では、T1 インターフェイス 3 (28 ある中の 1 つ) が、番号が 19、20、および 21 の 3 つのチャンネルグループ (0 ~ 23 の任意の番号) で構成されるように定義されます。これらのチャンネルグループには、合計 24 の DS0 タイムスロットが含まれています。各チャンネルグループは個別のインターフェイスを表します。

```
Router(config-controller) t1 3 channel-group 19 timeslots 1-6, 10
Router(config-controller) t1 3 channel-group 20 timeslots 7,8,9
Router(config-controller) t1 3 channel-group 21 timeslots 11-24
```

- c. 1 つのチャンネルグループに対して、インターフェイス コンフィギュレーション モードを開始します。次に例を示します。

```
Router(config)# interface serial 2/0/0.4/3:19
```

- d. ステップ 4 に進みます。

- ステップ 4** カプセル化方式を入力します。次に、フレームリレー カプセル化を使用する場合のコマンドを示します。PPP または HDLC も選択できます。

```
Router(config-if)# encapsulation frame-relay
```

- ステップ 5** システム上で IP ルーティングがイネーブルである場合は、IP アドレスおよびサブネット マスクを割り当てることができます。次に例を示します。

```
Router(config-if)# ip address 172.16.32.49 255.255.0.0
```

**ステップ 6** ルーティング プロトコルをイネーブルにするために必要なコンフィギュレーション サブコマンドをすべて追加し、インターフェイス回線の特性を設定します。

**ステップ 7** シャットダウン ステートをアップに変更して、インターフェイスをイネーブルにします。

```
Router(config-if)# no shutdown
```

**ステップ 8** コンフィギュレーション サブコマンドをすべて入力して、設定を完了したら、**Ctrl-Z** を押して、コンフィギュレーション モードを終了します。

**ステップ 9** 新しい設定を NVRAM に書き込みます。

```
Router# copy running-config startup-config
```

---

作成したインターフェイス設定は、該当する Cisco IOS コンフィギュレーション コマンドを使用して、いつでも変更できます。

## SDH フレーミング インターフェイスの設定例

1 ポート チャネライズド OC-12/STM-4 ラインカードが SDH フレーミング用に設定されている場合は、最大で次に示すコントローラを設定できます。

- 12 の STM-0 コントローラ。各 STM-0 コントローラを 1 つの AU-3 コントローラとして設定し、それぞれを 1 つの T3 インターフェイス、28 の T1 インターフェイス、または 21 の E1 インターフェイスとして設定できます。これらのインターフェイスはフラクショナル T1 または E1 にチャンネル化できます。
- 4 つの STM-1 コントローラ。各 STM-1 コントローラを 3 つの AU-4-TUG-3 コントローラとして設定し、それぞれを 28 の T1 インターフェイス、または最大 21 の E1 インターフェイスとして設定できます。これらのインターフェイスはフラクショナル T1 または E1 にチャンネル化できます。



**(注)** AU-3 および AU-4-TUG-3 は高次パスに基づいています。独立して動作せずに、3 つがまとまって動作します。いずれかの AU-3 または AU-4-TUG-3 コントローラをシャットダウンすると、同じ SONET ポートの残りの 2 つのコントローラもシャットダウンします。

SONET **controller** コマンドはモードを指定し、該当するタイプの (サブ) コントローラを 3 つ作成します。これらのコントローラは、ほかのコントローラと同様に設定できます。

作成されたフォーマットに応じて、コントローラ SONET 名にサブポート拡張子が追加されます。サブポート拡張子の形式は、AU-3 と AU-4-TUG-3 コントローラで異なります (次の例を参照)。

### AU-3 コントローラの構文

```
controller au-3 <slot>/<subslot>/<port>.<AU-3#>
```

スロット 2 に搭載された 4 ポート チャネライズド STM-1 ラインカードのポート 0 の AU-3 コントローラ #3 に関する **controller** コマンドの例は、次のとおりです。

```
controller au-3 2/0/0.3
```

### AU-4-TUG-3 コントローラの構文

```
controller au-4-tug-3 <slot>/<subslot>/<port>.<AU-4#>/<TUG-3#>
```

スロット 2 に搭載された 1 ポート チャネライズド OC-12/STM-4 ラインカードのポート 0 の AU-4-TUG-3 コントローラ #3 に関するコマンド例は、次のとおりです。

```
controller au-4-tug-3 2/0/0.1/3
```

## show controller コマンドの構文

AU-3 および AU-4-TUG-3 コントローラの固有の設定を表示するには、**show controller** コマンドを使用します。

AU-3 の場合：

```
show controller au-3 <slot>/<subslot>/<port>.<AU-3#>[/<tug-2>/<t1>]
```

AU-4-TUG-3 の場合 :

```
show controller au-4-tug-3
<slot>/<subslot>/<port>.<AU-4#>/<TUG-3#>[/<tug-2>/<t1 or e1>]
```

各値は次のとおりです。

- *slot* — 1 ~ 8
- *subslot* — 0
- *port* — 常に 0
- *AU-3#* — 1 ~ 12
- *AU-4#* — 常に 1
- *TUG-3#* — 1 ~ 12
- *tug-2* — 1 ~ 7
- *t1* — 1 ~ 4 (*e1* の場合は 1 ~ 3)



(注)

[/<tug-2>/<t1>] オプションを使用できるのは、モード (C-11 または C-12) がコントローラに設定されている場合のみです。

たとえば、AU-3 を介した T1 の場合、スロット 2 に搭載されたラインカードの AU-3 コントローラ #3 の設定を表示するコマンドは、次のとおりです。

```
show controller au-3 2/0/0.3
```

AU-4-TUG-3 の場合、スロット 2 に搭載された 1 ポート チャネライズド OC-12/STM-4 ラインカードの AU-4-TUG-3 コントローラ #3 に関する設定を表示するコマンドは、次のとおりです。

```
show controller au-4-tug-3 2/0/0.1/3
```

同じ AU-4-TUG-3 の特定の TUG-2 および E1 を表示するには、次のコマンドを使用します。

```
show controller au-4-tug-3 2/0/0-1/3/1/3
```

## AU-3 コントローラ インターフェイス

AU-3 コントローラを使用して T3 および T1 インターフェイスを作成する手順は、次のとおりです。

- ステップ 1** グローバル コンフィギュレーション モードで、SONET コントローラ コンフィギュレーション モードを開始します。

次の例では、1 ポート チャネライズド OC-12/STM-4 ラインカードがスロット 1 に搭載されています。

```
Router(config)# controller sonet 2/0/0
Router(config-controller)#
```

- ステップ 2** **aug controller** コマンドを使用して、STM-4 ポートの動作モードを指定します (AU-3 の場合、12 のコントローラ 2/0/0.1 ~ 2/0/0.12 が作成されます)。

```
Router(config-controller)# aug controller au-3
Router(config-controller)#
```

## フルレート T3 インターフェイス



(注) T3 回線が動作するには、AU-3 コントローラを経由する必要があります。

- a. ポートにコントローラを設定します。

次に、スロット 2 に搭載されたラインカードに AU-3 コントローラ #3 を設定する例を示します。

```
Router(config-controller)# controller au-3 2/0/0.3
Router(config-controller)#
```

- b. コンテナモードを C-3 に設定して、T3 インターフェイスを確立します。

```
Router(config-controller)# mode c-3
Router(config-controller)# exit
Router(config)#
```

C-3 モードでは、シリアルインターフェイスが非チャネライズド T3 レベルで作成されます。

## T1 チャンネル グループ インターフェイス

- a. 単一の OC-12 ポートにコントローラを設定します。

次に、スロット 2 に搭載されたラインカードの OC-12 ポートに AU-3 コントローラ #2 を設定する例を示します。

```
Router(config)# controller au-3 2/0/0.2
Router(config-controller)#
```

- b. コンテナモードを C-11 に設定して、T1 インターフェイスを確立します。

```
Router(config-controller)# mode c-11
Router(config-controller)#
```

- c. T1 インターフェイスを作成します。

次の例では、TUG-2 インターフェイスは 1 (1 ~ 7 の任意の番号)、T1 インターフェイスは 2 (1 ~ 4 の任意の番号) で、単一のチャンネルグループ 5 (0 ~ 23 の任意の番号) で構成されます。このチャンネルグループには、24 のすべての DS0 タイムスロットが含まれます。

```
Router(config-controller)# tug-2 1 t1 2 channel-group 5 timeslots 1-24
Router(config-controller)#
```

## AU-4-TUG-3 コントローラ インターフェイス

AU-4-TUG-3 コントローラを使用して T1 および E1 インターフェイスを作成する手順は、次のとおりです。



### 注意

ポートに作成された 3 つの AU-4-TUG-3 コントローラは同じ高次パスに基づいて動作し、互いに独立した動作は行いません。つまり、いずれかの AU-4-TUG-3 コントローラを「シャットダウン」すると、同じ SONET ポートの残りの 2 つのコントローラもシャットダウンします。

- ステップ 1** SONET コントローラ コンフィギュレーション モードを開始します。

次の例では、ラインカードはスロット 2 に搭載されています。

```
Router(config)# controller sonet 2/0/0
Router(config-controller)#
```

- ステップ 2** **aug controller** コマンドを使用して、OC-12 ポートの動作モードを指定します (AU-4 の場合、3 つのコントローラが作成されます)。

```
Router(config-controller)# aug controller au-4-tug-3
Router(config-controller)#
```

### AU-4-TUG-3/TUG-2 を介した T1 チャネル グループ インターフェイス

- a. STM-4 ポートにコントローラを設定します。次に、スロット 2 に搭載されたラインカードのポート 0 に AU-4-TUG-3 コントローラ #2 を設定する例を示します。

```
Router(config-controller)# controller au-4-tug-3 2/0/0.1/2
Router(config-controller)#
```

- b. コンテナ モードを C-11 に設定して、T1 インターフェイスを確立します。

```
Router(config-controller)# mode c-11
Router(config-controller)#
```

- c. T1 インターフェイスを作成します。

次の例では、TUG-2 インターフェイスは 1 (1 ~ 7 の任意の番号)、T1 インターフェイスは 2 (1 ~ 4 の任意の番号) で、単一のチャネルグループ 5 (0 ~ 23 の任意の番号) として定義されます。このチャネルグループには、24 のすべての DS0 タイムスロットが含まれます。

```
Router(config-controller)# tug-2 1 t1 2 channel-group 5 timeslots 1-24
Router(config-controller)#
```

### AU-4-TUG-3/TUG-2 を介した E1 チャネル グループ インターフェイス

- a. STM-4 ポートにコントローラを設定します。

次に、スロット 2 に搭載されたラインカードのポート 0 に AU-4-TUG-3 コントローラ #2 を設定する例を示します。

```
Router(config-controller)# controller au-4-tug-3 2/0/0.1/2
Router(config-controller)#
```

- b. コンテナ モードを C-12 に設定して、E1 インターフェイスを確立します。

```
Router(config-controller)# mode c-12
Router(config-controller)#
```

- c. E1 インターフェイスを作成します。

次の例では、TUG-2 インターフェイスは 1 (1 ~ 7 の任意の番号)、E1 インターフェイスは 2 (1 ~ 3 の任意の番号) で、単一のチャネルグループ 5 (0 ~ 31 の任意の番号) として定義されます。このチャネルグループには、31 の DS0 タイムスロットすべてが含まれます。

```
Router(config-controller)# tug-2 1 e1 2 channel-group 5 timeslots 1-31
Router(config-controller)#
```



## SONET コントローラの SONET または SDH フレーミング用の設定

デフォルトでは、1 ポート チャネライズド OC-12/STM-4 ラインカードは、12 の STS-1 接続で構成されます。1 ポート チャネライズド OC-12/STM-4 ラインカードを設定するには、各 STS-1 を T3 または VT インターフェイスとして設定する必要があります。ここでは、T3 および VT インターフェイスを作成して STS-1 接続をテストする場合に使用するコマンドについて説明します。内容は次のとおりです。

- [コントローラ コンフィギュレーションモードの開始 \(p.6-17\)](#)
- [SONET または SDH フレーミングの選択 \(p.6-17\)](#)
- [SONET コントローラ ループバック モードの設定 \(p.6-18\)](#)

### コントローラ コンフィギュレーションモードの開始

コントローラ コンフィギュレーションモードを開始して、コントローラを設定できるようにするには、**controller sonet** コマンドを使用します。

```
controller sonet slot/subslot/port
```

ここで、*slot* は 1～8、*subslot* および *port* は両方とも 0 です。

次に、スロット 2 に搭載されたカードに対して SONET コントローラ コンフィギュレーションモードを開始する例を示します。

```
Router(config)# controller sonet 2/0/0
Router(config-controller)#
```

### SONET または SDH フレーミングの選択

1 ポート チャネライズド OC-12/STM-4 ラインカードを SONET 用に設定するには、まず **framing** コマンドを使用してフレーミングタイプを SONET または SDH に設定する必要があります。

1 ポート チャネライズド OC-12/STM-4 ラインカードのフレーミングが SONET または SDH に設定してある場合に、フレーミングを変更するには、**no framing** コマンドを入力して、ラインカードをデフォルト値に戻す必要があります。そのあとで、**framing <framing type>** コマンドを使用して、新しいフレーミングを設定します。



注意

**no framing** コマンドを入力すると、ラインカードの設定が消去され、ラインカードのすべてのデフォルト値が復元されます。このコマンドを使用するときは注意してください。

```
framing {sonet | sdh}
[no] framing
```

デフォルトは SONET です。

フレーミングを削除して、ラインカードをデフォルト値に戻すには、このコマンドの **no** 形式を使用します。

次に、スロット 2 に搭載された 1 ポート チャネライズド OC-12/STM-4 ラインカードに SONET フレーミングを指定する例を示します。

```
Router(config)# controller sonet 2/0/0
Router(config-controller)# framing sonet
```

## SONET コントローラ ループバック モードの設定

SONET コントローラ ループバックをイネーブルにするには、**loopback** コマンドを使用します。

```
loopback [internal | line]
[no] loopback [internal | line]
```

デフォルトはループバックなしです。

ループバックを停止する場合は、このコマンドの **no** 形式を使用します。

**loopback** コマンドの詳細については、オンラインで『*Cisco 10000 Series ESR Troubleshooting Guide*』を参照してください。

次のコマンド例では、スロット 2 に搭載された 1 ポート チャネライズド OC-12/STM-4 ラインカードで内部ループバックを実行します。

```
Router(config)# controller sonet 2/0/0
Router(config-controller)# loopback internal
```

## T3、VT、または AUG コントローラの作成

ここでは、T3、VT、または AUG コントローラを作成および設定する例を示します。内容は次のとおりです。

- SONET フレーミングの場合の STS-1 パスの T3 または VT としての指定 (p.6-19)
- SDH フレーミングの場合の STM-4 ポートの AU-3 または AU-4-TUG-3 コントローラとしての指定 (p.6-19)
- SONET コントローラ ループバック モードの設定 (p.6-18)
- SONET フレーミングの場合の VT コマンド (p.6-20)
- SONET または SDH フレーミングの場合の非チャネライズド T3 コマンド (p.6-21)
- SONET または SDH フレーミングの場合のチャネライズド T3 コマンド (p.6-25)
- SONET または SDH フレーミングの場合の T1 または E1 チャンネル グループの作成 (p.6-30)
- SONET または SDH フレーミングの場合のチャネライズド T1 コマンド (p.6-33)

### SONET フレーミングの場合の STS-1 パスの T3 または VT としての指定

1 ポート チャネライズド OC-12/STM-4 ライン カードが SONET フレーミング用に設定されている場合は、**path** コマンドを使用して、STS-1 パスを T3 または VT コントローラとして指定できます。最大 12 の STS-1 チャンネルを指定できます (図 6-1 に、OC-12、T3 インターフェイス、VT インターフェイス、および T1 インターフェイスの関係を示します)。

```
path STS_number controller {t3 | vt}
```

ここで、*STS\_number* は仮想 T3 または VT コントローラです。1 ~ 12 の値を入力します。

T3 コントローラを削除するには、このコマンドの **no** 形式を使用します。

次の例では、番号が 1 の STS-1 が T3 回線として定義され、T3 コントローラ 2/0/0.1 が生成されます。

```
Router(config)# controller sonet 2/0/0
Router(config-controller)# path 1 controller t3
```

次の例では、番号が 1 の STS-1 が VT 回線として定義され、VT コントローラ 2/0/0.1 が生成されます。

```
Router(config)# controller sonet 2/0/0
Router(config-controller)# path 1 controller vt
```

### SDH フレーミングの場合の STM-4 ポートの AU-3 または AU-4-TUG-3 コントローラとしての指定

1 ポート チャネライズド OC-12/STM-4 ライン カードが SDH フレーミング用に設定されている場合は、**aug controller** コマンドを使用して、STM-4 ポートの動作モードを AU-3 コントローラまたは AU-4-TUG-3 コントローラに指定できます。このコマンドは、指定タイプのサブコントローラを 12 作成します。作成されたサブコントローラは、ほかのコントローラと同様に設定できます。

```
[no] aug controller <au-3/au-4-tug-3>
```

ここで、*au-3* および *au-4-tug-3* はコントローラのタイプを指定します。1 ~ 12 の値を入力します。

AU-3 または AU-4-TUG-3 コントローラを削除するには、このコマンドの **no** 形式を使用します。

これらのコントローラの名前フォーマットでは、SONET コントローラ名にサブポート拡張子を追加する必要があります。サブポート拡張子は、AU-3 と AU-4-TUG-3 コントローラで異なります。

### AU-3 コントローラ

AU-3 コントローラの構文は、次のとおりです。

```
controller au-3 <slot>/<subslot>/<port>.<AU-3#>
```

たとえば、次のコマンドでは、スロット 2 に搭載されたラインカードのポート 0 の AU-3 コントローラ #3 を使用します。

```
Router(config-controller)# controller au-3 2/0/0.3
```

### AU-4-TUG-3 コントローラ

AU-4-TUG-3 コントローラの構文は、次のとおりです。

```
Router(config-controller)# controller au-4-tug-3
<slot>/<subslot>/<port>.<AU-4#>/<TUG-3#>
```



(注)

SDH フレーミング SONET ポートに作成された 12 の AU-4-TUG-3 コントローラは同じ高次パスに基づいて動作し、互いに独立した動作は行いません。つまり、いずれかの AU-4-TUG-3 コントローラを「シャットダウン」すると、同じ SDH フレーミング SONET ポートの残りの 2 つのコントローラもシャットダウンします。

たとえば、次のコマンドでは、スロット 2 に搭載された 1 ポート チャネライズド OC-12/STM-4 ラインカードのポート 0 の AU-4-TUG-3 コントローラ #3 を使用します。

```
Router(config-controller)# controller au-4-tug-3 2/0/0.1/3
```

## SONET フレーミングの場合の VT コマンド

ここでは、**controller vt** コマンドを使用して、1 ポート チャネライズド OC-12/STM-4 ラインカードの SONET フレーミング STS-1 コントローラをさらにチャネル化する方法について説明します。

### VT のコントローラ コンフィギュレーション モードの開始

1 ポート チャネライズド OC-12/STM-4 ラインカードの VT コントローラは、28 の T1 (VT1.5) インターフェイスまたは 21 の E1 (VT2) インターフェイスにチャネル化できます。**controller vt** コマンドを使用すると、VT リンクをシャットダウンしたり、T1 または E1 インターフェイスの設定を変更したりできます。

T1 設定の詳細については、「[SONET または SDH フレーミングの場合の T1 または E1 チャネルグループの作成](#)」(p.6-30)、および「[SONET または SDH フレーミングの場合のチャネライズド T1 コマンド](#)」(p.6-33) を参照してください。

VT リンクを設定するには、**controller vt** コマンドを入力します。

```
slot/sub-slot/port.path
```

ここで、*path* は 1 ~ 12 の値です。各値は 28 の T1 回線または 21 の E1 回線を収容する VT を表します。

次に例を示します。

```
Router# configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
Router(config)# controller vt 1/0/0.1
Router(config-controller)#
```

## SONET または SDH フレーミングの場合の非チャネライズド T3 コマンド

ここでは、SONET フレーミングまたは SDH フレーミング コントローラで非チャネライズド T3 を設定する方法について説明します。

1 ポート チャネライズド OC-12/STM-4 ライン カードが SONET フレーミング用に設定されている場合、T3 インターフェイスはデフォルトで 28 の T1 インターフェイスにチャネル化されます。フルレートまたはサブレート T3 インターフェイスを作成するには、T3 インターフェイスを非チャネル化する必要があります。

1 ポート チャネライズド OC-12/STM-4 ラインカードが SDH フレーミング用に設定されている場合は、C-3 コンテナ モードによって、非チャネライズド T3 シリアル インターフェイスが自動的に作成されます。

ここでは、フルレートおよびサブレート T3 インターフェイスを作成、カスタマイズ、およびテストする場合に使用するコマンドについて説明します。内容は次のとおりです。

- [T3 のコントローラ コンフィギュレーション モードの開始 \(p.6-21\)](#)
- [SONET フレーミング コントローラの T3 インターフェイスの非チャネライズドとしての設定 \(p.6-22\)](#)
- [サブレート T3 の実装 \(p.6-22\)](#)
- [フレーミング タイプの設定 \(p.6-23\)](#)
- [DSU モードの指定 \(p.6-23\)](#)
- [スクランプリングのイネーブル化 \(p.6-23\)](#)
- [アイドル文字の指定 \(p.6-23\)](#)
- [SONET または SDH フレーミングの場合の T3 BER テストの実行 \(p.6-24\)](#)

非チャネライズド T3 を設定する場合は、「[SONET または SDH フレーミングの場合のチャネライズド T3 コマンド](#)」(p.6-25) に記載されている **clock source**、**mdl**、**equipment**、および **loopback** コマンドも使用できます。

### T3 のコントローラ コンフィギュレーション モードの開始

非チャネライズド T3 インターフェイスを作成するには、まず、設定する T3 コントローラに対して、コントローラ コンフィギュレーション モードを開始する必要があります。

```
controller T3 slot/sub-slot/port.path
```

ここで、*path* は T3 インターフェイス番号を指定します。

次に、コントローラ コンフィギュレーション モードを開始する例を示します。

```
Router(config)# controller T3 1/0/0.1
Router(config-controller)#
```

## SONET フレーミング コントローラの T3 インターフェイスの非チャネライズドとしての設定

1 ポート チャネライズド OC-12/STM-4 ライン カードが SONET フレーミング用に設定されている場合は、T3 インターフェイスを非チャネル化して、フルレートまたはサブレート T3 インターフェイスを作成する必要があります。

T3 インターフェイスを非チャネライズド (クリア チャネル) として設定するには、**no channelized** コマンドを入力します。

```
[no] channelized
```



### 注意

**no channelized** コマンドを使用すると、チャネライズド T3 インターフェイスからすべてのチャンネルグループが削除されます。チャンネルグループが設定されている場合は、このコマンドを使用するときに注意してください。

インターフェイスをデフォルトに戻すには、**channelized** コマンドを使用します。T3 インターフェイスのデフォルト値は、チャネライズドです。

次の例では、非チャネライズド T3 インターフェイスが作成されます。

```
Router(config)# controller T3 1/0/0.1
Router(config-controller)# no channelized
```

## サブレート T3 の実装

サブレート T3 を実装するには、非チャネライズド T3 インターフェイスの帯域幅を指定します。そのためには、インターフェイス コンフィギュレーション モードで **dsu bandwidth bandwidth** コマンドを使用します。

```
dsu bandwidth bandwidth
[no] bandwidth bandwidth
```

ここで、*bandwidth* は 0 ~ 44,210 kbps の数値です。

デフォルト帯域幅は 44,210 kbps です。

デフォルトの帯域幅に戻すには、このコマンドの **no** 形式を使用します。

値を指定すると、現在の DSU (データ サービス ユニット) モードのタイムスロット サイズに基づいて、許容される最も近い値に帯域幅が設定されます。

このコマンドを適切に使用するには、接続のリモート エンドに、**dsu mode** コマンドでサポートされている PA-T3 または PA-2T3 ポート アダプタまたは T3 DSU が搭載された Cisco 7200 ルータまたは Cisco 7500 ルータを配置する必要があります。

次の例では、帯域幅として 16,000 kbps が指定されます。

```
Router(config)# interface serial 1/0/0.1
Router(config-if)# dsu bandwidth 16000
```

## フレーミング タイプの設定

非チャネライズド T3 コントローラのフレーミング タイプを指定するには、**framing** コマンドを使用します。

```
framing [c-bit | m13]
[no] framing [c-bit | m13]
```

デフォルトのフレーミング タイプは C ビットです。

デフォルトのフレーミング タイプに戻すには、このコマンドの **no** 形式を使用します。

次の例では、フレーミングが m13 に設定されます。

```
Router(config)# interface serial 1/0/0.1
Router(config-if)# framing m13
```

## DSU モードの指定

選択した T3 インターフェイスに対して DSU モードを指定するには、インターフェイス コンフィギュレーション モードで **dsu mode** コマンドを使用します。このコマンドは、製造元独自の多重化方式をエミュレートするようにライン カードを設定します。

```
dsu mode [Adtran | cisco | Digital-link | Kentrox | Larscom | verilink-highbit |
verilink-lowbit]
[no] dsu mode
```

デフォルトの DSU モードは **cisco** です。

DSU モードをデフォルトに戻すには、このコマンドの **no** 形式を使用します。

次の例では、DSU モードが **cisco** に設定されます。

```
Router(config)# interface serial 1/0/0.1
Router(config-if)# dsu mode cisco
```

## スクランブリングのイネーブル化

非チャネライズド T3 インターフェイスでスクランブリングをイネーブルにするには、インターフェイス コンフィギュレーション モードで **scramble** コマンドを使用します。

```
scramble
[no] scramble
```

このコマンドのデフォルト設定は、**no scramble**（スクランブリングがディセーブル）です。

リンクの両側で、スクランブリング設定を同じにする必要があります。

次の例では、指定された T3 インターフェイスでスクランブリングがイネーブルになります。

```
Router(config)# interface serial 1/0/0.1
Router(config-if)# scramble
```

## アイドル文字の指定

非チャネライズド T3 インターフェイスで、HDLC パケット間に送信する特定の文字を設定するには、インターフェイス コンフィギュレーション モードで **idle character** コマンドを使用します。

```
idle-character [flags | marks]
[no] idle-character [flags | marks]
```

各値は次のとおりです。

- **flags** — アイドル文字に 0x7e を設定します。
- **marks** — アイドル文字に 0xff (すべて) を設定します。

デフォルトのアイドル文字は 0x7e です。

アイドル文字をデフォルトに戻すには、このコマンドの **no** 形式を使用します。

次の例では、アイドル文字が flags に設定されます。

```
Router(config)# interface serial 1/0/0.1
Router(config-if)# idle-character flags
```



(注) marks (0xff) が中断信号として解釈されるシステムがあるため、flags (0x7e) を推奨します。

## SONET または SDH フレーミングの場合の T3 BER テストの実行

Bit Error Rate (BER; ビット誤り率) テストを実行するように、非チャネライズド T3 インターフェイスを設定できます。BER テストは、現場でケーブルを調べて、信号問題を解決する場合に使用します。非チャネライズド T3 インターフェイスから BER テストパターンを送信するには、次のインターフェイス コンフィギュレーション コマンドを使用します。

```
bert [errors number | pattern pattern] interval time
[no] bert
```

各値は次のとおりです。

- **errors number** — 1 ~ 255
- **pattern pattern** — 値は次のとおり
  - 0s — すべてゼロの反復テストパターン (00000...)
  - 1s — すべて 1 の反復テストパターン (11111...)
  - 2^15 — 疑似ランダム O.151 テストパターン (32,768 ビット長)
  - 2^20-O153 — 疑似ランダム O.153 テストパターン (1,048,575 ビット長)
  - QRSS-2^20 — 疑似ランダム QRSS O.151 テストパターン (1,048,575 ビット長)
  - 2^23 — 疑似ランダム O.151 テストパターン (8,388,607 ビット長)
- **interval time** — 1 ~ 1440 分

**no bert** コマンドを使用すると、いつでも BER テストを終了できます。

詳細については、オンラインで『Cisco 10000 Series Internet Router Troubleshooting Guide』を参照してください。

例：

- T3 回線 1 を介して、BER テスト疑似ランダムパターン 2^20 を 5 分間送信します。

```
Router(config)# interface serial 1/0/0.1
Router(config-if)# bert pattern 2^20 interval 5
```

- T3 回線 1 を介して、すべて 1 の反復パターンを 1440 分間送信します。

```
Router(config)# interface serial 1/0/0.1
Router(config-if)# bert pattern 1s interval 1440
```



## SONET または SDH フレーミングの場合のチャネライズド T3 コマンド

ここでは、SONET フレーミングまたは SDH フレーミング コントローラでチャネライズド T3 を設定する方法について説明します。

デフォルトでは、1 ポート チャネライズド OC-12/STM-4 ライン カードの T3 インターフェイスは、28 個の T1 インターフェイスにチャネル化されます。ここでは、チャネライズド T3 インターフェイスをカスタマイズして、テストする場合に使用するコマンドについて説明します。ここでは、次の手順について説明します。

- [T3 のコントローラ コンフィギュレーション モードの開始 \(p.6-25\)](#)
- [T3 インターフェイスのチャネライズドとしての設定 \(p.6-26\)](#)
- [フレーミング タイプの設定 \(p.6-26\)](#)
- [MDL メッセージの入力 \(p.6-27\)](#)
- [アイドル パターンの指定 \(p.6-28\)](#)
- [T3 クロック ソースの設定 \(p.6-28\)](#)
- [T3 コントローラのループバック モードの設定 \(p.6-29\)](#)
- [リモート ループバック コマンドに応答するための T3 コントローラの設定 \(p.6-29\)](#)

加入者トラフィックにインターフェイスを使用するには、T1 および DS0 コンポーネントを設定する必要があります。詳細については、「[SONET または SDH フレーミングの場合の T1 または E1 チャネル グループの作成](#)」(p.6-30) を参照してください。

図 6-1 に、チャネル化によって実現される、設定可能なインターフェイス帯域幅レベルを示します。

### T3 のコントローラ コンフィギュレーション モードの開始

T3 インターフェイスに対してコントローラ コンフィギュレーション モードを開始しなければならない場合があります。たとえば、非チャネライズド T3 インターフェイスをチャネライズドに戻したり、Maintenance Data Link (MDL) メッセージを設定したりするには、このモードを開始します。

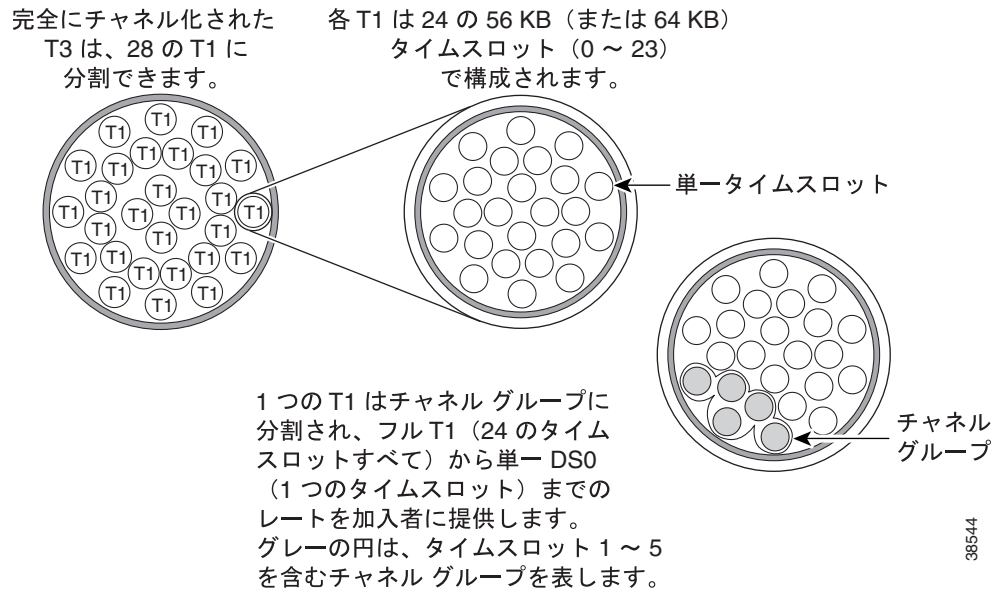
```
controller T3 slot/sub-slot/port.path
```

*Path* は T3 インターフェイス番号を指定します。

次に、コントローラ コンフィギュレーション モードを開始する例を示します。

```
Router(config)# controller T3 1/0/0.1
Router(config-controller)#
```

図 6-1 T3 のチャンネル化



## T3 インターフェイスのチャネライズドとしての設定

T3 インターフェイスはデフォルトでチャンネル化されます。インターフェイスが非チャネライズドに設定されている場合は、**channelized** コマンドを使用します。

```
[no] channelized
channelized
```



### 注意

**no channelized** コマンドを使用すると、チャネライズド T3 インターフェイスからすべてのチャンネルグループが削除されます。チャンネルグループが設定されている場合は、このコマンドを使用するときに注意してください。

次に、チャネライズド T3 インターフェイスを作成する例を示します。

```
Router(config)# controller T3 1/0/0.1
Router(config-controller)# channelized
```

## フレーミングタイプの設定

チャネライズド T3 コントローラのフレーミングタイプを指定するには、**framing** コマンドを使用します。

```
framing [c-bit | m23 | auto-detect]
[no] framing [c-bit | m23 | auto-detect]
```

デフォルトは auto-detect です。

フレーミングタイプをデフォルトに戻すには、このコマンドの **no** 形式を使用します。

遠端からフレーミングタイプを検出して、同じフレーミングタイプを送信するように、1 ポートチャネライズド OC-12/STM-4 ラインカードに指示することができます。

```
Router(config)# controller T3 1/0/0.1
Router(config-controller)# framing auto-detect
```

## MDL メッセージの入力

チャネライズド T3 インターフェイスに MDL メッセージ (ANSI T1.107a-1990 仕様で定義) を設定できます。



(注) MDL メッセージがサポートされるのは、T3 フレーミングが C ビット パリティに設定されている場合のみです (「[フレーミング タイプの設定](#)」 [p.6-26] を参照)。

MDL メッセージを設定するには、**mdl** コマンドを使用します。

```
mdl {transmit {path | idle-signal | test-signal} | string {eic | lic | fic | unit | pfi |
port | generator} id_string}
[no] mdl {transmit {path | idle-signal | test-signal} | string {eic | lic | fic | unit |
pfi | port | generator} id_string}
```

各値は次のとおりです。

- **transmit path** — MDL パス メッセージの送信をイネーブルにします。
- **transmit idle-signal** — MDL アイドル信号メッセージの送信をイネーブルにします。
- **eic** — 機器 ID コード (最大 10 文字)
- **lic** — 場所 ID コード (最大 11 文字)
- **fic** — フレーム ID コード (最大 10 文字)
- **unit** — ユニット ID コード (最大 6 文字)
- **pfi** — MDL パス メッセージに含まれるファシリティ ID コード (最大 38 文字)
- **port** — MDL アイドル信号メッセージに含まれる (アイドル信号の開始元となる) 機器のポート (最大 38 文字)
- **generator** — MDL テスト信号メッセージに含まれるジェネレータ番号 (最大 38 文字)

デフォルトでは、MDL メッセージは設定されません。

MDL メッセージを削除するには、このコマンドの **no** 形式を使用します。

次に、MDL メッセージの設定例を示します。

- MDL パス メッセージの送信をイネーブルにします。  
Router(config)# controller T3 1/0/0.1  
Router(config-controller)# mdl transmit path
- MDL アイドル信号メッセージの送信をイネーブルにします。  
Router(config-controller)# mdl transmit idle-signal
- MDL テスト信号メッセージの送信をイネーブルにします。  
Router(config-controller)# mdl transmit test-signal
- 機器 ID コードを入力します。  
Router(config-controller)# mdl string eic router A
- 場所 ID コードを入力します。  
Router(config-controller)# mdl string lic test network
- フレーム ID コードを入力します。  
Router(config-controller)# mdl string fic building b

## ■ T3、VT、または AUG コントローラの作成

- ユニット ID コードを入力します。  
Router(config-controller)# mdl string unit *abc*
- ファシリティ ID コードを入力します。  
Router(config-controller)# mdl string pfi *string*
- MDL アイドル信号メッセージに格納して送信するポート番号を入力します。  
Router(config-controller)# mdl string port *string*
- MDL テスト信号メッセージに格納して送信するジェネレータ番号を入力します。  
Router(config-controller)# mdl string generator *string*

## アイドルパターンの指定

チャネライズド T3 インターフェイスに属する未設定のすべてのタイムスロットインターフェイスについて、HDLC パケット間に送信される特定のパターンを設定できます。そのためには、**idle pattern** コマンドを使用します。

```
idle pattern patterns
[no] idle pattern patterns
```

ここで、*patterns* は 0x0 ~ 0xff (16 進数) または 0 ~ 255 (10 進数) の値です。この値は 16 進数または 10 進数で入力できます。0 ~ 254 の値を指定すると、アイドルパターンは HDLC flags (0x7e) に設定され、255 の値を指定すると、0xff (すべて 1) に設定されます。



(注)

---

marks (0xff) が中断信号として解釈されるシステムがあるため、flags (0x7e) を推奨します。

---

デフォルトのアイドルパターンは 0x7e です。

アイドルパターンをデフォルト値に戻すには、このコマンドの **no** 形式を使用します。

例：

- 16 進数のアイドルパターンを設定します。  
Router(config)# controller T3 1/0/0.1  
Router(config-controller)# idle pattern 0x10
- 10 進数のアイドルパターンを設定します。  
Router(config)# controller T3 1/0/0.1  
Router(config-controller)# idle pattern 23

## T3 クロック ソースの設定

選択した T3 コントローラに内部クロック ソースまたは回線クロック ソースを設定するには、プロンプトで **clock source** コマンドを使用します。このコマンドは、コントローラ コンフィギュレーション モードで設定します。

```
clock source {internal | line}
```

各値は次のとおりです。

- **internal** — 内部クロック ソースを使用するように指定します。
- **line** — ネットワーク クロック ソースを使用するように指定します。

デフォルトは内部クロック ソースです。

次の例では、回線クロック ソースを使用するように T3 コントローラに指示します。

```
Router(config)# controller T3 1/0/0.1
Router(config-controller)# clock source line
```

## T3 コントローラのループバック モードの設定

T3 コントローラにループバック モードを設定するには、**loopback** コマンドを使用します。

```
loopback {local | network | remote}
[no] loopback {local | network | remote}
```

デフォルトはループバックなしです。

T3 コントローラのループバックをディセーブルにするには、このコマンドの **no** 形式を使用します。

例：

- T3 コントローラにローカル ループバックを設定します。

```
Router(config)# controller T3 1/0/0
Router(config-controller)# loopback local
```

ローカル ループバックの場合、すべてのチャネルがルータにループされると同時に、ネットワークに T3 Alarm Indication Signal (AIS; アラーム表示信号) が送信されます。ローカル ループバックを使用すると、ポートにネットワーク ケーブルが接続されていない場合に、ポートの問題を診断できます。

- T3 ポートにネットワーク ループバックを設定します。

```
Router(config)# controller T3 1/0/0
Router(config-controller)# loopback network
```

ネットワーク ループバックの場合、T3 回線はネットワーク方向にループバックされ、中央のスイッチング オフィスとポートを接続するケーブルの問題を診断できます。

- T3 ポートにリモート ループバックを設定します。

```
Router(config)# controller T3 1/0/0
Router(config-controller)# loopback remote
```

リモート ループバックの場合、遠端 (セントラル オフィス) で T3 回線をループするコマンドが送信されます。ポート アダプタとスイッチング オフィスを接続するケーブルの問題を診断する場合に使用できます。

## リモート ループバック コマンドに応答するための T3 コントローラの設定

リモート機器と組み合わせてループバックを実行するには、**equipment loopback** コマンドを使用します。

```
equipment [customer | network] loopback
```

各値は次のとおりです。

- **customer** — リモート T3 機器からのループバック コマンドにポートが応答できるようにします。
- **network** — リモート T3 ループバック コマンドを無視するように、コントローラを設定します。

## ■ T3、VT、または AUG コントローラの作成

例：

リモート ループバック 要求に応答するためのコントローラ機能をイネーブルにするには、次のように入力します。

```
Router(config)# controller T3 1/0/0
Router(config-controller)# equipment customer loopback
```

コントローラがリモート ループバック コマンドに応答しないように設定するには、次のように入力します。

```
Router(config)# controller T3 1/0/0
Router(config-controller)# equipment network loopback
```



(注)

リモート ループバックを使用できるのは、C ビット パリティ フレーミングを使用する場合のみです。

**loopback** コマンドの詳細については、オンラインで『[Cisco 10000 Series Internet Router Troubleshooting Guide](#)』を参照してください。

## SONET または SDH フレーミングの場合の T1 または E1 チャネル グループの作成

加入者トラフィックをサポートするインターフェイスを作成するには、タイムスロットをチャネルグループに結合する必要があります。チャネルグループの設定を容易にするために、T1 および E1 に関する次の特性を検討してください。

### SONET フレーミング T3 コントローラの場合

- 1 つのチャネライズド T3 は 28 の T1 で構成されます。
- 1 つの T1 は 24 のタイムスロット (DS0) で構成されます。

### SONET フレーミング VT コントローラの場合

T1 の場合：

- 1 つの VT コントローラは 28 の T1 (VT1.5) で構成されます。
- 1 つの T1 は 24 のタイムスロット (DS0) で構成されます。

E1 の場合：

- 1 つの VT コントローラは 21 の E1 (VT2) で構成されます。
- 1 つの E1 は 31 のタイムスロット (DS0) で構成されます。

### SDH フレーミング T3 コントローラの場合

- 1 つの T3 は 7 つの TUG-2 で構成されます。
- 1 つの TUG-2 は次のいずれかで構成されます。
  - それぞれ 24 のタイムスロット (DS0) で構成可能な 4 つの T1
  - 31 のタイムスロット (DS0) で構成可能な 3 つの E1

## T1 SONET フレーミング インターフェイス用のチャンネル グループの作成

SONET フレーミング インターフェイス用のチャンネル グループを作成するには、**t1 channel-group** コマンドを使用します。

図 6-1 に、1 ポート チャネライズド OC-12/STM-4 ライン カードのチャンネル化の仕組みを示します。

次のコマンドを使用して、論理チャンネル グループを作成します。

```
t1 t1-number channel-group channel-group-number timeslots list-of-timeslots [speed {56
| 64}]
[no] t1 t1-number channel-group channel-group-number
```

各値は次のとおりです。

- **t1-number** — 1 ~ 28 の T1 インターフェイス番号
- **channel-group channel-group-number** — 0 ~ 23 の任意の番号を持つチャンネル グループを指定します。
- **timeslots list-of-timeslots** — 1 ~ 24、または 1 ~ 24 内のサブ範囲の組み合わせを指定できます。範囲を指定するには、ハイフン、カンマ、または両方の組み合わせを使用します。1 つのタイムスロットが 1 つの DS0 に相当します (次に示す例を参照)。
- **speed {56 | 64}** — タイムスロットの速度を 56 kbps または 64 kbps として指定するオプションの引数。デフォルトは 64 です (56 kbps のタイムスロットは SF [スーパー フレーム] フレーミングに対応し、通常、B8ZS ラインコーディングをサポートしない古い T1 機器で使用します)。

論理チャンネル グループを削除するには、このコマンドの **no** 形式を使用します。

次に、**t1 channel-group** コマンドの使用例を示します。

- 次の例では、T1 インターフェイス 3 にチャンネル グループ 20 が含まれ、9 つのチャネライズドタイムスロットで構成されています。

```
Router(config)# controller t3 1/0/0.1
Router(config-controller)# t1 3 channel-group 20 timeslots 1-8, 10
```

T1 インターフェイス 3 のチャンネル グループ 20 に対してインターフェイス コンフィギュレーション モードを開始するには、次のように入力します。

```
Router(config)# interface serial 1/0/0.1/3:20
```

- 次の例では、T1 インターフェイス 4 にチャンネル グループ 18 が含まれ、24 のすべてのタイムスロットで構成されていて、フル T1 インターフェイスを形成しています。

```
Router(config)# controller t3 1/0/0.1
Router(config-controller)# t1 4 channel-group 18 timeslots 1-24
```

T1 インターフェイス 4 のチャンネル グループ 18 に対してインターフェイス コンフィギュレーション モードを開始するには、次のように入力します。

```
Router(config)# interface serial 1/0/0.1/4:18
```

- 次の例では、T1 インターフェイス 5 に 3 つのチャンネル グループ、および合計 11 のタイムスロットが含まれています。

```
Router(config)# controller t3 1/0/0.1
Router(config-controller)# t1 5 channel-group 19 timeslots 1-6
Router(config-controller)# t1 5 channel-group 20 timeslots 10
Router(config-controller)# t1 5 channel-group 21 timeslots 7-9, 24
```

- 次の例では、T1 1 のチャンネル グループ 20 が削除されます。

```
Router(config)# controller T3 1/0/0.1
Router(config-controller)# no t1 1 channel-group 20
```

## SDH フレーミング インターフェイス用のチャネル グループの作成

## チャネルに関する制限

SDH コントローラごとのチャネル数の上限は 128、1 ポート チャネライズド OC-12/STM-4 ラインカードの単一の OC-12 ポートの最大チャネル数は 192 です。この制限は、コントローラにチャネルグループを作成する場合に、T1 と E1 の両方のインターフェイスに適用されます。

たとえば、ポート 2 の T1 コントローラ #1 に 128 のチャネル、ポート 2 の T1 コントローラ #2 に 64 のチャネルを作成した場合、T1 コントローラ #3 または T1 コントローラ #4 には追加チャネルを作成できません。単一ポート（この例ではポート 2）のチャネル数の上限 192 に達したためです。

論理チャネルグループを作成するには、次のいずれかのコマンドを使用します。

## T1 の場合：

```
[no] tug-2 <tug-2#> t1 <t1#> channel-group <channel#> timeslots <range> [speed 64|56]
```

## E1（フレーミング）の場合：

```
[no] tug-2 <tug-2#> e1 <e1#> channel-group <channel#> timeslots <range>
```

## E1（非フレーミング）の場合：

```
[no] tug-2 <tug-2#> e1 <e1#> unframed
```

各値は次のとおりです。

- *tug-2* — 1 ～ 7
- *t1#* — 1 ～ 4
- *e1#* — 1 ～ 3
- **channel-group** *channel#* — 0 ～ 23 の任意の番号を持つチャネルグループを指定します。
- **timeslots** *range* — 1 ～ 24（T1 の場合）、1 ～ 31（E1 の場合）、またはこれらの範囲内のサブ範囲の組み合わせを指定できます。範囲を指定するには、ハイフン、カンマ、または両方の組み合わせを使用します。1 つのタイムスロットが 1 つの DS0 に相当します（次に示す例を参照）。
- **speed** {56|64} — タイムスロットの速度を 56 kbps または 64 kbps として指定するオプションの T1 引数。デフォルトは 64 です（56 kbps のタイムスロットは SF フレーミングに対応し、通常、B8ZS をサポートしない古い T1 機器で使用します）。



(注) **speed** 引数は T1 チャネルグループにのみ適用され、E1 では無効です。

- **unframed** — 非フレーミング E1 チャネルを作成します。これにより、チャネルグループ 0 を含むインターフェイスが作成されます。



(注) **unframed** 引数は、E1 チャネルグループにのみ適用され、T1 では無効です。

論理チャネルグループを削除するには、このコマンドの **no** 形式を使用します。

次に、**channel-group** コマンドの使用例を示します。



- 次の例では、T1 インターフェイス 3 にチャンネル グループ 20 が含まれ、9 つのチャネライズド タイムスロットで構成されています。

```
Router(config-controller)# controller au-3 2/0/0.2
Router(config-controller)# mode c-11
Router(config-controller)# tug-2 1 t1 3 channel-group 20 timeslots 1-9
```

T1 インターフェイス 3 のチャンネル グループ 20 に対してインターフェイス コンフィギュレーション モードを開始するには、次のように入力します。

```
Router(config)# interface serial 2/0/0.2/3:20
```

- 次の例では、T1 インターフェイス 4 にチャンネル グループ 18 が含まれ、24 のすべてのタイムスロットで構成されていて、フル T1 インターフェイスを形成しています。

```
Router(config-controller)# controller au-3 2/0/0.2
Router(config-controller)# mode c-11
Router(config-controller)# tug-2 1 t1 4 channel-group 18 timeslots 1-24
```

T1 インターフェイス 4 のチャンネル グループ 18 に対してインターフェイス コンフィギュレーション モードを開始するには、次のように入力します。

```
Router(config)# interface serial 2/0/0.2/4:18
```

- 次の例では、T1 インターフェイス 5 に 3 つのチャンネル グループ、および合計 11 のタイムスロットが含まれています。

```
Router(config-controller)# controller au3 2/0/0.2
Router(config-controller)# mode c-11
Router(config-controller)# tug-2 1 t1 5 channel-group 19 timeslots 1-6
Router(config-controller)# tug-2 1 t1 5 channel-group 20 timeslots 10
Router(config-controller)# tug-2 1 t1 5 channel-group 21 timeslots 7-9, 24
```

- 次の例では、チャンネル グループ 20 が削除されます。

```
Router(config-controller)# controller au3 2/0/0.2
Router(config-controller)# no tug-2 1 t1 5 channel-group 20
```

## SONET または SDH フレーミングの場合のチャネライズド T1 コマンド

T3 インターフェイスのコントローラ コンフィギュレーション モードから、T1 インターフェイスの特性を変更するコマンドを入力できます。ここで示されるコマンドの用途は、次のとおりです。

- [フレーミング フォーマットの設定 \(p.6-34\)](#)
- [イエロー アラームの制御 \(p.6-34\)](#)
- [T1 クロック ソースの設定 \(p.6-35\)](#)
- [FDL の設定 \(p.6-35\)](#)
- [T1 BER テストの設定 \(p.6-35\)](#)
- [ループバック モードの設定 \(p.6-36\)](#)

T1 インターフェイスを設定したら、インターフェイス コンフィギュレーション モードを開始して、カプセル化、ルーティング、およびその他の命令を追加できます。次に例を示します。

```
Router(config)# interface serial 1/0/0.1/4:18
```

## フレーミング フォーマットの設定

T1 インターフェイス フレーミング フォーマットを指定するには、次のコマンドを使用します。

```
t1 t1-number framing {esf | sf [hdlc-idle {0x7E | 0xFF}]}
[no] t1 t1-number framing {esf | sf [hdlc-idle {0x7E | 0xFF}]}
```

各値は次のとおりです。

- *t1-number* — 1～28 の T1 インターフェイス
- **framing** — 拡張スーパーフレーム (Extended Super Frame) または SF。SF の hdlc-idle は 0x7E または 0xFF に設定できます。
- **hdlc-idle** オプションを使用すると、T1 インターフェイスのアイドル パターンを 0x7e (デフォルト) または 0xFF に設定できます。

デフォルトのフレーミング フォーマットは拡張スーパーフレームです。

フレーミングをデフォルト値に戻すには、このコマンドの **no** 形式を使用します。

例：

- T1 インターフェイス 6 に SF フレーミング フォーマットを設定します。  

```
Router(config)# controller T3 1/0/0.1
Router(config-controller)# t1 6 framing sf
```
- T1 インターフェイス 16 に拡張スーパーフレーム フレーミング フォーマットを設定します。  

```
Router(config)# controller T3 1/0/0.1
Router(config-controller)# t1 16 framing esf
```

## イエロー アラームの制御

イエロー アラームの検出または生成をオンまたはオフにするには、**t1 yellow** コマンドを使用します。

```
t1 t1-number yellow {detection | generation}
[no] t1 t1-number yellow {detection | generation}
```

各値は次のとおりです。

- *t1-number* — 1～28 の T1 インターフェイス
- **detection** — 障害がある場合は、リモート デバイスからインターフェイスに通知され、Cisco IOS ソフトウェアによってコンソールにメッセージが送信されます。
- **generation** — 障害がある場合は、インターフェイスからリモート デバイスに通知され、Cisco IOS ソフトウェアによってコンソールにメッセージが送信されます。

デフォルト速度に 64 を使用している フル T1 インターフェイス (24 のタイムスロット) に SF フレーミングを選択した場合は、**no t1-number yellow detection** コマンドを使用して、イエロー アラーム検出をオフにすることを検討してください。SF フレーミングの場合、イエロー アラームが誤って検出されることがあります。

次の例では、T1 インターフェイス 1 にイエロー検出が設定されます。

```
Router(config)# controller t3 2/0/0.1
Router(config-controller)# t1 1 yellow detection
```

## T1 クロック ソースの設定

T1 インターフェイスに内部クロック ソースまたは回線（ネットワーク）クロック ソースを設定するには、**controller** コマンドを使用します。

```
t1 t1-number clock source {internal | line}
```

各値は次のとおりです。

- *t1-number* — 1 ～ 28 の T1 インターフェイス
- **internal** — 内部クロック ソースを使用するように指定します。
- **line** — ネットワーク クロック ソースを使用するように指定します。

デフォルトは内部クロック ソースです。

次の例では、回線からクロック ソースを取得するようにインターフェイスに指示します。

```
Router(config)# controller T3 1/0/0.1
Router(config-controller)# t1 1 clock source line
```

## FDL の設定

**fdl ansi** コマンドを使用すると、T1 接続の両端で、Facility Data Link (FDL; ファシリティ データ リンク) を介して ANSI T1.403 パフォーマンス レポートを 1 秒間送信するように設定できます。

```
t1 t1-number fdl ansi
[no] t1 t1-number fdl ansi
```

ここで、*t1-number* は 1 ～ 28 の T1 インターフェイスです。

この機能をディセーブルにするには、このコマンドの **no** 形式を使用します。



(注)

---

このコマンドを使用できるのは、T1 フレーミングが拡張スーパーフレームの場合のみです。

---

次の例では、FDL がイネーブル化されます。

```
Router(config)# controller t3 1/0/0.1
Router(config-controller)# t1 2 fdl ansi
```

## T1 BER テストの設定

独立した BER テストを実行するように、各 T1 インターフェイスを設定できます。BER テストは、現場でケーブルを調べて、信号問題を解決する場合に使用します。T1 インターフェイスから BER テスト パターンを送信するには、次のコマンドを使用します。

```
t1 t1-number bert pattern pattern interval time [unframed]
[no] t1 t1-number bert
```

各値は次のとおりです。

- *t1-number* — 1 ～ 28 の T1 インターフェイス番号
- *time* — 1 ～ 14,400 分
- *pattern* — 値は次のとおり
  - 2<sup>11</sup> — 疑似ランダム テスト パターン (2048 ビット長)
  - 2<sup>15</sup> — 疑似ランダム O.151 テスト パターン (32,768 ビット長)
  - 2<sup>20</sup>-O153 — 疑似ランダム O.153 テスト パターン (1,048,575 ビット長)

- **unframed** — T1 フレーミングおよびペイロード ビットを含む T1 帯域幅全体を BER テストパターンで使用します。**unframed** を省略すると、T1 フレーミング コマンドの設定に応じて T1 は SF または拡張スーパーフレーム フレーミングになり、BER テストパターンは T1 ペイロード ビットのみを占有します。



(注) 各 T3 で実行できる BER テストは、一度に 1 つのみです。

このコマンドの **no** 形式を使用すると、いつでも BER テストを終了できます。

詳細については、オンラインで『[Cisco 10000 Series Internet Router Troubleshooting Guide](#)』を参照してください。

例：

- T1 インターフェイス 10 を介して、BER テスト疑似ランダム パターン 2<sup>15</sup> を 5 分間送信します。  

```
Router(config)# controller T3 1/0/0.1
Router(config-controller)# t1 10 bert pattern 2^15 interval 5 unframed
```

## ループバック モードの設定

T1 インターフェイスを設定するときに問題が発生した場合は、コントローラ コンフィギュレーション モードで次のコマンドを使用して、ラインカードのトラブルシューティングを実行できます。

```
t1 t1-number loopback [local | network {line | payload} | remote [line [fdl {ansi | bellcore}]] | payload [fdl | ansi]]]
[no] t1 t1-number loopback [local | network {line | payload} | remote [line [fdl {ansi | bellcore}]] | payload [fdl | ansi]]]
```

各値は次のとおりです。

- **t1-number** — 1 ~ 28 の T1 インターフェイス
- **local** — T1 フレームでルータ出力データをルータ方向にループバックさせて、Alarm Indication Signal (AIS; アラーム表示信号) をネットワークに送信します。
- **network {line | payload}** — データをネットワーク方向にループバックさせて、HDLC コントローラにローカル ループバックを自動設定するか (**line**)、またはペイロードデータをネットワーク方向にループバックさせて、HDLC コントローラにローカルループバックを自動設定します (**payload**)。
- **remote line fdl {ansi | bellcore}** — 反復する 16 ビットの拡張スーパーフレーム データ リンク キーワードをリモートエンドに送信して、リモートエンドでネットワーク ラインループバックを開始するように要求します。ANSI またはベルコア キーワードを指定できます。
- **remote payload [fdl] [ansi]** — 反復する 16 ビット拡張スーパーフレーム データ リンク コードワードをリモートエンドに送信して、リモートエンドでネットワーク ペイロードループバックを開始するように要求します。**fdl** および **ansi** を使用すると、T1 チャンネルでリモート ペイロード FDL ANSI ビットループバックがイネーブルになります。

ループバックを終了する場合は、このコマンドの **no** 形式を使用します。

このコマンドの詳細については、オンラインで『[Cisco 10000 Series Internet Router Troubleshooting Guide](#)』を参照してください。

例：

- T3 コントローラに T1 インターフェイス 1 でのローカル ループバックを設定します。

```
Router(config)# controller t3 1/0/0.1
Router(config-controller)# t1 1 loopback local
```

- T3 コントローラに、T1 インターフェイス 1 でのリモート FDL ANSI ループバックを設定します。

```
Router(config)# controller t3 1/0/0.1
Router(config-controller)# t1 1 loop remote line fdl ansi
```

■ T3、VT、または AUG コントローラの作成