# Cisco 10000 Series Router Line Card Configuration Guide

November, 2009


**Americas Headquarters**
Cisco Systems, Inc.
170 West Tasman Drive
San Jose, CA 95134-1706
USA
http://www.cisco.com
Tel: 408 526-4000
800 553-NETS (6387)
Fax: 408 527-0883



Text Part Number: OL-8834-04

# CONTENTS

# About This Guide

This guide describes how to configure Cisco 10000 series router line cards. For information about features that are supported on the Cisco 10000 series router using the line cards, such as ATM PVC autoprovisioning, automatic protection switching (APS), or quality of service (QoS), see the following configuration guides:

- *Cisco 10000 Series Router Software Configuration Guide* at the following URL: http://www.cisco.com/en/US/docs/routers/10000/10008/configuration/guides/broadband/bba.html
- *Cisco 10000 Series Router Quality of Service Configuration Guide* at the following URL: http://www.cisco.com/en/US/docs/routers/10000/10008/configuration/guides/qos/qoscf.html

This document describes the following topics:



# Guide Revision History

| Cisco IOS Release | Part Number | Publication Date |
|---|---|---|
| **Release 12.2(31)SB2** | OL-8834-04 | November, 2006 |

**Description**

Added Performance Routing Engine 3 (PRE3) support information for the 1-Port Gigabit Ethernet, 1-Port Gigabit Ethernet Half-Height, 1-Port OC-12, and 1-Port Channelized OC-12/STM-4 line cards.

Added VC limitations per priority level per port for the 1-port OC-12, 4-port OC-3, and 8-port E3/DS3 ATM line cards.

| Cisco IOS Release | Part Number | Publication Date |
|---|---|---|
| **Release 12.2(28)SB2** | OL-8834-03 | July, 2006 |

**Description**

Added the "Management Port" section on page 9-2 of Chapter 9, "8-Port Fast Ethernet Half-Height Line Card Configuration" to indicate that the Fast Ethernet interface 0/0/0 is only used for management purposes. Subinterfaces cannot be configured on this interface.

Changed the format of the Guide Revision History.

| Cisco IOS Release | Part Number | Publication Date |
|---|---|---|
| **Release 12.2(28)SB** | OL-8834-02 | March, 2006 |

**Description**

Added Chapter 4, "4-Port Channelized T3 Half-Height Line Card Configuration".

Added the 4-Port Channelized T3 Half-Height line card to the list of supported line cards in Chapter 17, "Provisioning a Subrate E3 or T3 Interface".

Added the "Controlling the S1 SONET Overhead Byte" section to the configuration chapters for the: 4-Port OC-3/STM-1 ATM line card, 1-Port OC-12 ATM line card, 6-Port OC-3/STM-1 Packet over SONET line card, and 1-Port OC-12 Packet over SONET line card.

Removed a chapter on configuring MLP connections. Refer to the MLP information in the *Cisco 10000 Series Router Software Configuration Guide* and the *Cisco 10000 Series Quality of Service Configuration Guide*.

| Cisco IOS Release | Part Number | Publication Date |
|---|---|---|
| **Release 12.3(7)XI7** | OL-8834-01 | December, 2005 |

**Description**

Retired OL-0716-10 and created this guide with the same organization as the *Cisco 10000 Series Router Line Card Hardware Installation Guide*, using a book part for each type of line card, such as ATM.

Because more up-to-date and complete information is included in the *Cisco 10000 Series Router Software Configuration Guide* and the *Cisco 10000 Series Quality of Service Configuration Guide*, the Qos and APS chapters were not ported to the new guide.

Removed several sections that have been moved to more appropriate locations in other guides:

- Configuring the router at startup (moved to the *Cisco 10008 Quick Start Guide* and the *Cisco 10008 Router Hardware Installation Guide*)
- Managing file systems (moved to the *Cisco 10008 Quick Start Guide* and the *Cisco 10008 Router Hardware Installation Guide*)
- Managing PRE redundancy (moved to the *Cisco 10000 Series Router Performance Routing Engine Installation*)
- Upgrading software (moved to the *Cisco 10000 Series Router Performance Routing Engine Installation*)
- Managing system boot parameters (moved to the *Cisco 10008 Router Hardware Installation Guide*)

Incorporated information on provisioning a subrate E3 or T3 interface, formerly documented in a feature guide, as Chapter 17, "Provisioning a Subrate E3 or T3 Interface" in this guide.

Added the minimum IOS releases that are supported by each line card.

Revised information for preprovisioning a line card.

# Audience

The Cisco 10000 Series Router Line Card Configuration Guide is designed for the person who will configure and maintain the line cards on a Cisco 10000 series router. To benefit from this guide, this person must be experienced using Cisco IOS.

# Document Organization

The *Cisco 10000 Series Router Line Card Configuration Guide* is organized as follows:

| Chapter | Title | Description |
|---|---|---|
| Chapter 1 -15 | Line card configuration | Each chapter provides configuration information for a specific Cisco 10000 series router line card, including minimum software release information, hardware and software compatibility information, and the commands you use to configure the line card. |
| Chapter 16 | Preparing for Line Card Installation and Configuration | Describes the treatment of a new line card, preprovisioning a line card, and resetting a line card. |
| Chapter 17 | Provisioning a Subrate E3 or T3 Interface | Describes provisioning a subrate E3 or T3 interface. |
| Chapter 18 | Configuration Examples | Provides several large configuration examples. |

# Document Conventions

This document uses the following conventions:

| Convention | Description |
|---|---|
| ^ or Ctrl | The ^ and Ctrl symbols represent the Control key. For example, the key combination ^D or Ctrl-D means hold down the Control key while you press the D key. Keys are indicated in capital letters but are not case sensitive. |
| *string* | A string is a nonquoted set of characters shown in italics. For example, when setting an SNMP community string to public, do not use quotation marks around the string or the string will include the quotation marks. |

Command syntax descriptions use the following conventions:

| Convention | Description |
|---|---|
| **bold** | Bold text indicates commands and keywords that you enter literally as shown. |
| *italics* | Italic text indicates arguments for which you supply values. |
| [x] | Square brackets enclose an optional element (keyword or argument). |
| \| | A vertical line indicates a choice within an optional or required set of keywords or arguments. |

| Convention | Description |
|---|---|
| [x \| y] | Square brackets enclosing keywords or arguments separated by a vertical line indicate an optional choice |
| {x \| y} | Braces enclosing keywords or arguments separated by a vertical line indicate a required choice. |

Nested sets of square brackets or braces indicate optional or required choices within optional or required elements. For example:

| Convention | Description |
|---|---|
| [x {y \| z}] | Braces and a vertical line within square brackets indicate a required choice within an optional element. |

Examples use the following conventions:

| Convention | Description |
|---|---|
| `screen` | Examples of information displayed on the screen are set in Courier font. |
| `bold screen` | Examples of text that you must enter are set in Courier bold font. |
| `< >` | Angle brackets enclose text that is not printed to the screen, such as passwords. |
| ! | An exclamation point at the beginning of a line indicates a comment line. (Exclamation points are also displayed by the Cisco IOS software for certain processes.) |
| `[ ]` | Square brackets enclose default responses to system prompts. |

The following conventions are used to attract the attention of the reader:

**Caution** Means reader be careful. In this situation, you might do something that could result in equipment damage or loss of data.

**Note** Means reader take note. Notes contain helpful suggestions or references to materials not contained in this manual.

# Related Documentation

For more information about the Cisco 10000 series router, its features, and hardware, go to the Cisco 10000 series router documentation roadmap, located at the following URL:

http://www.cisco.com/en/US/products/hw/routers/ps133/products_documentation_roadmap09186a00804ba4f3.html

For information about Cisco IOS Release 12.2, including command reference and system error messages, go to the Cisco IOS Release 12.2 documentation web page, located at the following URL:

http://www.cisco.com/en/US/products/sw/iosswrel/ps1835/tsd_products_support_series_home.html

# Obtaining Documentation and Submitting a Service Request

For information on obtaining documentation, submitting a service request, and gathering additional information, see the monthly *What's New in Cisco Product Documentation*, which also lists all new and revised Cisco technical documentation, at:

http://www.cisco.com/en/US/docs/general/whatsnew/whatsnew.html

Subscribe to the *What's New in Cisco Product Documentation* as a Really Simple Syndication (RSS) feed and set content to be delivered directly to your desktop using a reader application. The RSS feeds are a free service and Cisco currently supports RSS Version 2.0.

# Part 1

## ATM Line Cards

CHAPTER 1

# 4-Port OC-3/STM-1 ATM Line Card Configuration

This chapter describes the procedures for configuring the Cisco 10000 series 4-Port OC-3/STM-1c ATM line card, hereafter known as the 4-Port OC-3/STM-1 ATM line card.

The 4-Port OC-3/STM-1 ATM line card is a NEBS-compliant device that performs Layer 2 Asynchronous Transfer Mode (ATM) functions. This line card receives and transmits ATM cells on each network physical interface connected to a line card port, and simultaneously transmits and receives packets from the Cisco 10000 series router backplane.

This chapter contains the following sections:


- Software Support, page 1-1
- Default Values, page 1-2
- Line Card VC Limitations, page 1-2
- Interface Syntax, page 1-3
- Interface Configuration Examples, page 1-4
- ATM Commands, page 1-6


# Software Support

Table 1-1 shows the minimum Cisco IOS release on each release train that supports the 4-Port OC-3/STM-1 ATM line card.

*Table 1-1 4-Port OC-3/STM-1 ATM Line Card Software Support*

| Required PRE | Minimum Cisco IOS Releases |
|---|---|
| PRE1 | Cisco IOS Release 12.0(21)SX and later releases of Cisco IOS Release 12.0SX<br>Cisco IOS Release 12.0(20)ST and later releases of Cisco IOS Release 12.0ST<br>Cisco IOS Release 12.0(22)S and later releases of Cisco IOS Release 12.0S<br>Cisco IOS Release 12.2(8)BZ and later releases of Cisco IOS Release 12.2BZ |
| PRE2 | Cisco IOS Release 12.3(7)XI and later releases of Cisco IOS 12.3XI<br>Cisco IOS Release 12.2(15)BX and later releases of Cisco IOS Release 12.2BX<br>Cisco IOS Release 12.2(28)SB and later releases of Cisco IOS 12.2SB |

## Checking Hardware and Software Compatibility

The PRE installed in the Cisco 10000 series router chassis must support the Cisco IOS software running on the router. Use the **show version** command to check the PRE version installed.

To see if a feature is supported by a Cisco IOS release, to locate the software document for that feature, or to check the minimum software requirements of Cisco IOS software with the hardware installed on your router, Cisco maintains the Software Advisor tool on Cisco.com at http://www.cisco.com/cgi-bin/Support/CompNav/Index.pl

This tool does not verify whether line cards within a system are compatible, but does provide the minimum Cisco IOS requirements for individual hardware line cards, modules, or options.

You must be a registered user on Cisco.com to access this tool.

# Default Values

Table 1-2 lists default configuration values for the 4-Port OC-3/STM-1 ATM line card. This table also includes the command used to modify a default value, and provides information about values to set on the remote end of the connection.

***Table 1-2 4-Port OC-3/STM-1 ATM line card Defaults***

| Command Name | Default Setting | Command Syntax | Remote Side Setting |
|---|---|---|---|
| mtu (maximum transmission unit)* | 4470 bytes | **[no] mtu** *bytes* | same |
| atm clock internal | no atm clock internal | **[no] atm clock internal** | opposite |
| loopback | loopback none | **[no] loopback [line \| diagnostic {parallel \| path \| serial}]** | — |

# Line Card VC Limitations

The Cisco 10000 series router supports four ATM service categories for virtual circuits (VCs):

- Constant Bit Rate (CBR)
- Variable Bit Rate-non-real-time (VBR-nrt)
- Unspecified Bit Rate (UBR) with a peak cell rate (PCR), referred to as shaped UBR
- UBR without a PCR, referred to as unshaped UBR

The segmentation and reassembly (SAR) mechanism configures priority and additional traffic management parameters for the various ATM service categories. Table 1-3 lists the priority levels the SAR sets for the service categories.

***Table 1-3 ATM Service Categories***

| Parameter | CBR | VBR-rt | VBR-nrt | Shaped UBR | Unshaped UBR |
|---|---|---|---|---|---|
| Priority | 0 | 1 | 2 | 3 | None |

The number of SAR priority levels and the service categories supported at each priority level vary from line card to line card. For example, the 4-port OC-3 line card supports the two levels of priority and the service categories listed in Table 1-4.

The ATM line cards support a maximum number of VCs per priority. That VC limit depends on the VC limit of the SAR (SAR limit) and the number of priority levels configured. Table 1-4 describes how to determine the VC limit per priority level per port for the 4-port OC-3 line card.

*Table 1-4 Maximum Number of VCs per Priority*

| ATM Line Card | SAR Priority Levels | VC Rate | Maximum Number of VCs per Priority |
|---|---|---|---|
| 4-Port OC-3 | 0 = CBR, VBR-nrt VCs<br>1 = UBR VCs | Half line rate and below | SAR limit / number of PHYs / number of priority levels<br>2 priority system:<br>65,536 / 4 / 2 = 8192 VCs per priority level per port |

Configuring more channels or VCs than there are available priority locations can cause random channels or VCs to get stuck in the SAR. This occurs when an active channel tries to reschedule itself, but no priority locations are available. Therefore, the channel cannot find a place to reschedule itself, which results in a lost event for the channel, and the channel becomes stuck in the SAR.

On the PRE2, when a VC becomes stuck in the SAR, the PRE2 scheduler stops forwarding traffic on only the VC that is stuck in the SAR; the other VCs still carry traffic. On the PRE3, the PRE3 scheduler stops forwarding traffic on all the VCs configured on that ATM line card.

# Interface Syntax

To specify an interface number in a configuration command, use the syntax in Table 1-5 to identify main interfaces and subinterfaces on the 4-Port OC-3/STM-1 ATM line card in Cisco Series 10000 routers.

*Table 1-5 4-Port OC-3 ATM Interface Syntax*

| Type of Interface | Router | Slot | Subslot | Port | Subinterface |
|---|---|---|---|---|---|
| Main interface | 10008 | 1 through 8/ | 0/ | 0 to 3 | — |
| Subinterface | 10008 | 1 through 8/ | 0/ | 0.n to 3.n | n = 1 to 4294967295 |
| Main interface | 10005 | 1 through 5/ | 0/ | 0 to 3 | — |
| Subinterface | 10005 | 1 through 5/ | 0/ | 0.n to 3.n | n = 1 to 4294967295 |

Examples:

- Modify a PVC associated with the main interface.

```
Router(config)# interface atm 2/0/0
Router(config-if)# pvc 0/200
Router(config-if-atm-vc)#
```

- Modify a permanent virtual circuit (PVC) associated with a subinterface.

```
Router(config)# interface atm 7/0/0.1
Router(config-subif)# pvc 0/101
Router(config-if-atm-vc)#
```

# Interface Configuration Examples

This section provides sample procedures for creating ATM subinterfaces, permanent virtual circuits (PVCs), and virtual circuit (VC) classes, and procedures for enabling Integrated Local Management Interface (ILMI).

## Creating a Subinterface

Use the following procedure to create a subinterface.

**Step 1** Separate the ATM interface into subinterfaces using the **interface** command. You can create either a point-to-point or multipoint subinterface.

In the following example, multipoint subinterface number 1 is created on port 0 of the 4-Port OC-3/STM-1 ATM line card in slot 2.

```
Router(config)# interface atm 2/0/0.1 multipoint
Router(config-subif)#
```

**Step 2** Assign an IP address and subnet mask to the created interface using the **ip address** configuration subcommand, as shown in the following example:

```
Router(config-subif)# ip address 172.27.48.209 255.255.0.0
Router(config-subif)#
```

You have created ATM subinterface 2/0/0.1. To configure or modify this interface, use the following command:

```
Router(config)# interface atm 2/0/0.1
Router(config-subif)#
```

## Creating a PVC

You can create multiple PVCs on a 4-Port OC-3/STM-1 ATM line card interface. You can create PVCs on the main interface or on a subinterface.

To create a PVC

**Step 1** Enter interface or subinterface configuration mode.

Use the **pvc** command to specify a virtual path identifier (VPI) value between 0 and 255 and a virtual channel identifier (VCI) value between 0 and 65535. The following example shows how to create a PVC with a VPI value of 0 and VCI value of 100 on a subinterface.

```
Router(config-subif)# pvc 0/100
Router(config-if-atm-vc)#
```

**Step 2** Use the **protocol ip** configuration subcommand to assign a peer IP address to the PVC, as shown in the following example:

```
Router(config-if-atm-vc)# protocol ip 172.16.32.49
Router(config-if-atm-vc)#
```

## Creating a VC Class

This procedure demonstrates how to create an ATM VC class. An ATM VC class is a PVC boilerplate—a PVC description that you can apply to one or more PVCs.

To create a PVC boilerplate:

**Step 1** Use the global configuration mode **vc-class atm** *name* command, where *name* is the name value you assign. The following example shows how to create the ATM VC class named *boston*.

```
Router(config)# vc-class atm boston
Router(config-vc-class)#
```

**Step 2** Enter commands to describe the ATM VC class you named *boston*. This example shows how to specify that the *boston* class uses AAL5+MUX encapsulation with a variable bit-rate non real-time (VBR-NRT) PVC.

```
Router(config-vc-class)# encaps aal5mux ip
Router(config-vc-class)# vbr-nrt 30000 20000 128
Router(config-vc-class)# exit
Router(config)#
```

You have created a VC class named *boston*. The next procedure describes how to apply this class to a PVC or subinterface.

## Applying a VC Class

You can apply a VC class (created in the previous procedure) to a PVC or an interface.

- In the following example, the class named *boston* is applied to subinterface 5/0/0.1.

```
Router(config)# interface atm 5/0/0.1
Router(config-subif)# class-int boston
Router(config-subif)#
```

- In the following example, the class named *boston* is applied to a new PVC (0/102) in subinterface 5/0/0.2.

```
Router(config)# interface atm 5/0/0.2
Router(config-subif)# pvc 0/102
Router(config-if-atm-vc)# class-vc boston
Router(config-if-atm-vc)#
```

You have completed the steps for assigning a VC class to a PVC.

## Enabling ILMI PVC Discovery

You can enable ILMI to automatically discover PVCs on neighboring switches and duplicate those PVC entries on the 4-Port OC-3/STM-1 ATM line card.

To enable ILMI:

**Step 1** Create PVC 0/16 on the main interface as shown in the following example:

```
Router(config)# interface atm 5/0/0
Router(config-if)# pvc 0/16 ilmi
Router(config-if-atm-vc)# exit
Router(config-if)#
```

**Step 2** In the following example, ILMI PVC discovery is enabled for the selected port on the 4-Port OC-3/STM-1 ATM line card (step 1 references port 0).

```
Router(config-if)# atm ilmi-pvc-discovery
```

You have completed the steps required for running ILMI PVC discovery. You can use the **show atm pvc** command to display the PVCs on the Cisco 10000 series router.

## Completing a Configuration

This section offers general information on creating and completing a configuration of an 4-Port OC-3/STM-1 ATM line card.

To configure and interface:

**Step 1** Enter interface configuration mode and then specify necessary parameters, such as the IP address and subnet mask.

**Step 2** After you enter all of the configuration subcommands to complete the configuration, enter **Ctrl-Z** (hold down the **Control** key while you press **Z**) to exit configuration mode.

**Step 3** Write the new configuration to memory:

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

The system displays an OK message when the configuration is stored. After you have completed your configuration, you can check it by using **show** commands.

# ATM Commands

Each 4-Port OC-3/STM-1 ATM line card permits you to specify up to four OC-3 interfaces. You can put all the PVCs on the main interfaces, or you can create associated subinterfaces. This section describes the principal commands for customizing interfaces and PVCs:

- Global Configuration Command, page 1-7
- Interface and Subinterface Commands, page 1-7
- ATM PVC Commands, page 1-15
- Useful show Commands, page 1-19

# Global Configuration Command

A PVC boilerplate is a PVC description that you can apply to one or more PVCs or interfaces.

To create a PVC boilerplate, use the global configuration mode **vc-class atm** command:

**vc-class atm** *class_name*

Where *class_name* is any value that describes the VC class.

After you enter the **vc-class atm** command, you are placed in VC class configuration mode. In this mode, you describe the action you want the class to take by entering commands and their arguments. These commands and arguments are described in the "ATM PVC Commands" section on page 1-15.

In the following example, an ATM VC class named *cambridge* is created and defined. This example shows how to specify that the class uses AAL5+MUX encapsulation and a VBR-NRT PVC.

```
Router(config)# vc-class atm cambridge
Router(config-vc-class)# encaps aal5mux ip
Router(config-vc-class)# vbr-nrt 30000 20000 128
Router(config-vc-class)# exit
Router(config)#
```

For information on applying a VC class name, see the "Attaching an ATM VC Class to an Interface" section on page 1-9 and the "Attaching an ATM VC Class to a PVC" section on page 1-17.

# Interface and Subinterface Commands

This section describes principal commands for configuring ATM interfaces and subinterfaces. This section describes the following global configuration commands:

## Creating and Entering Subinterfaces

Use the **interface** command to segment an OC-3 ATM main interface into multiple subinterfaces to simplify line card management, to create interfaces with different MTU sizes, and to create connections to different networks.

**interface atm** *slot/subslot/port.subinterface type*
**no interface atm** *slot/subslot/port.subinterface type*

Where:

- *slot* is 1 to 5 for a Cisco 10005 router
- *slot* is 1 to 8 for a Cisco 10008 router
- *subslot* is always 0 for a full-height line card
- *port* is 0 to 3
- *subinterface* is a value from 1 to 4294967295
- *type* is always **point-to-point** or **multipoint**

To enter the subinterface at a later time, you do not need to specify the type.

To remove a subinterface and its PVCs, use the **no interface** command. To change a subinterface type, you must first remove the subinterface.

Examples:

- Create subinterface number 1 at port 3 for 4-Port OC-3/STM-1 ATM line card in slot 1.

```
Router(config)# interface atm 1/0/3.1 point-to-point
Router(config-subif)#
```

- Enter an existing subinterface.

```
Router(config)# interface atm 1/0/3.1
Router(config-subif)#
```

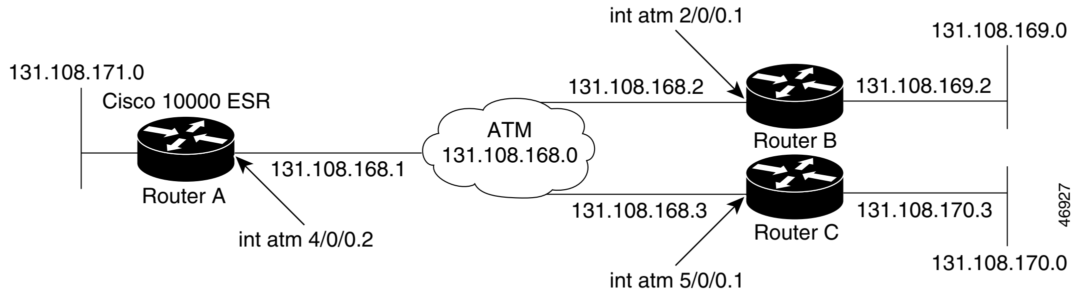
- Use a multipoint subinterface.

  If you want multiple PVCs to go to the same network, you must create a multipoint subinterface. For example:

```
Router(config)# interface atm 4/0/2.2 multipoint
Router(config-subif)#
```

  After you create the subinterface, you can create PVCs that are attached to the same network. Figure 1-1 shows a multipoint subinterface on a fully meshed network. Fully meshed indicates that any workstation can communicate with any other workstation.

*Figure 1-1 Multipoint ATM Configuration*




The following example commands for routers A, B, and C show how to configure the ATM interfaces for each router shown in Figure 1-1. These examples show the configuration commands you must enter to configure the multipoint connections in that network:

Router A

```
Router(config) # interface atm 4/0/0.2 multi
Router(config-subif) # ip address 131.108.168.1 255.255.255.0
Router(config-subif) # pvc 0/35
Router(config-if-atm-vc) # protocol ip 131.108.168.2 broadcast
Router(config-if-atm-vc)# en
Router(config-subif) # pvc 0/36
Router(config-if-atm-vc) # protocol ip 131.108.168.3 broadcast
Router(config-if-atm-vc)# en
Router(config-subif) # en
```

Router B

```
Router(config) # interface atm 2/0/0.1 multi
Router(config-subif) # ip address 131.108.168.2 255.255.255.0
Router(config-subif) # pvc 0/35
Router(config-if-atm-vc) # protocol ip 131.108.168.1 broadcast
Router(config-if-atm-vc)# en
Router(config-subif) # pvc 0/37
Router(config-if-atm-vc) # protocol ip 131.108.168.3 broadcast
Router(config-if-atm-vc)# en
Router(config-subif) # en
```

Router C

```
Router(config) # interface atm 5/0/0.1 multi
Router(config-subif) # ip address 131.108.168.3 255.255.255.0
Router(config-subif) # pvc 0/36
Router(config-if-atm-vc) # protocol ip 131.108.168.1 broadcast
Router(config-if-atm-vc)# en
Router(config-subif) # pvc 0/37
Router(config-if-atm-vc) # protocol ip 131.108.168.2 broadcast
Router(config-if-atm-vc)# ex
Router #
```

## Attaching an ATM VC Class to an Interface

Use the **class-int** command to attach an ATM VC class to an interface. If you customize a PVC, its customization takes precedence over the interface class.

**class-int** *class_name*

Where *class_name* is the name of the class created using the global configuration **class-vc atm** command.

In the following example, a VC class named *cambridge* is created and attached to subinterface 3/0/0.1.

```
Router(config)# vc-class atm cambridge
Router(config-vc-class)# encaps aal5mux ip
Router(config-vc-class)# vbr-nrt 30000 20000 128
Router(config-vc-class)# exit
Router(config)# interface atm 3/0/0.1
Router(config-subif)# class-int cambridge
```

## Creating ATM PVPs

To create a permanent virtual path (PVP) used to multiplex one or more VBR-NRT VCs, use the **atm pvp** interface configuration command.

**atm pvp** *vpi peak-rate* [**no-f4-oam**]
**no atm pvp** *vpi*

Where:

- **vpi** is the ATM network virtual path identifier (VPI) of the VC used to multiplex the permanent virtual path. The range is 0 to 255. You must use a VPI value that is not already in use (by a VC).
- *peak-rate* is the maximum rate in Kbps at which the PVP can transmit data. You can enter values from 84 Kbps to 74,880 Kbps, and you can also enter 149,760 Kbps. The PVP peak rate value supersedes all rate values set for VBR-NRT PVCs associated with the PVP.
- **no-f4-oam** (optional) restricts the PVP from passing operations/administration/maintenance (OAM) packets. When you create a PVP, the system creates (by default) PVCs with VCI values of 3 and 4 for each PVP, which pass OAM packets.

To verify the configuration of a PVP, use the **show atm vp** exec command.

The following example shows how to create a PVP with a peak rate of 50,000 Kbps. The subsequent created VCs are multiplexed onto this virtual path.

```
Router(config)# interface atm 7/0/0
Router(config-if)# atm pvp 25 50000
Router(config-if)# pvc 25/100
Router(config-if-atm-vc)# vbr-nrt 10000 5000 16
Router(config-if-atm-vc)# exit
Router(config-if)# pvc 25/101
Router(config-if-atm-vc)# vbr-nrt 10000 5000 16
Router(config-if-atm-vc)# exit
Router(config-if)# pvc 25/102
Router(config-if-atm-vc)# vbr-nrt 10000 5000 16
Router(config-if-atm-vc)# exit
Router(config)#
```

## Creating a PVC

This section describes how to create a permanent virtual circuit.

**Note** You can only create one PVC on a point-to-point interface. Multiple PVCs can be created on a multipoint interface.

You can establish an unspecified bit rate (UBR) PVC by entering the **pvc** command:

**pvc** [*word*] [*vpi_value*/]{*vci_value*} [**ilmi**]

Where:

- *vpi_value* is in the range from 0 to 255. If you do not specify a VPI value, the system assigns the value 0.
- *vci_value* is in the range 1 to 65535. The VCI value should be 33 or greater because all lower-numbered PVCs are already assigned.
- *word* is an optional name referring to this connection.
- **ilmi** parameter maps the ILMI channel to the PVC for this interface. You can only use this argument for PVCs created on the main interface. We recommend that you use this argument with PVC 0/16. For more information about activating ILMI, see the "Activating ATM ILMI PVC Discovery" section on page 1-11.

By default, the **pvc** command creates a UBR PVC. To specify a VBR-NRT PVC, see the "Configuring VBR-NRT" section on page 1-17.

Examples:

- Create PVC 0/105 on the main interface.

```
Router(config)# interface atm 5/0/0
Router(config-if)# pvc 105
Router(config-if-atm-vc)
```

- Create PVC 2/102 on a subinterface.

```
Router(config)# interface atm 5/0/0.1
Router(config-subif)# pvc 2/102
Router(config-if-atm-vc)#
```

## Enabling ATM ILMI

Use the **atm ilmi-enable** interface configuration command to enable ILMI on a port.

**atm ilmi-enable**
**no atm ilmi-enable**

The default is ILMI is enabled, but you should disable the ILMI if the peer does not support ILMI. For peers to be able to exchange ILMI information, you must create PVC 0/16 using the **ilmi** argument.

The following example shows how to disable ILMI:

```
Router(config)# interface atm 5/0/0
Router(config-if)# no atm ilmi-enable
```

## Activating ATM ILMI PVC Discovery

This command causes ILMI-compliant devices to propagate PVCs. Use the **atm ilmi-pvc-discovery** interface configuration command to activate ATM PVC discovery.

**atm ilmi-pvc-discovery** [**subinterface**]
**no atm ilmi-pvc-discovery** [**subinterface**]

Where:

- **subinterface** instructs the software to associate all PVCs with existing subinterface numbers equal to their VPI numbers. For example: PVC 2/102 would be listed under subinterface 7/0/0.2, PVC 12/156 would be listed under 7/0/0.12, and so on.

**Note** The **subinterface** argument associates PVCs only with subinterfaces that have already been created. If there is no subinterface for a VPI value, the system associates the PVC with the main interface.

The following example shows how to enable PVC discovery on the ATM main interface 7/0/0, for port 0, on a 4-Port OC-3/STM-1 ATM line card that has ILMI enabled.

```
Router(config)# interface atm 7/0/0
Router(config-if)# pvc 0/16 ilmi
Router(config-if-atm-vc)# exit
Router(config-if)# atm ilmi-pvc-discovery
```

## Specifying the ATM ILMI Keepalive Rate

Use the **atm ilmi-keepalive** command to specify the ILMI keepalive rate.

**atm ilmi-keepalive [***seconds* **[retry** *counts***]]**

Where:

- *seconds* is a value from 1 to 65535
- **retry** *counts* is a value from 2 to 5

The default value for seconds is 5000 and for retry counts is 4.

The following example shows how to enable ILMI keepalives for the ATM interface 5/0/0:

```
Router(config)# interface atm 5/0/0
Router(config-if)# atm ilmi-keepalive 10000 retry 3
```

## Configuring the ATM Clock

Use the **atm clock internal** command to configure the clock source as internal.

**atm clock internal**
**no atm clock internal**

The default clock setting is **no atm clock internal**, which means that clocking is derived from the line.

In the following example, clocking is set from the router.

```
Router(config)# interface atm 5/0/0
Router(config-if)# atm clock internal
```

## Specifying the ATM Flag

This command is typically used to meet a standards requirement or to ensure interoperability with another vendor's equipment. Use the **atm flag s1s0** command to specify the ATM flag value for the s1and s0 bits.

**atm flag s1s0** *value*

Where:

- **s1s0** is part of the payload pointer byte
- *value* is from 0 to 3

The default s1s0 value is 0.

The following example shows how to assign a value of 2 to the ATM flag:

```
Router(config)# interface atm 5/0/0
Router(config-if)# atm flag s1s0 2
```

## Controlling ATM Alarm Reporting

To control selected SONET alarms so that they are logged to the console for an ATM interface, use the **atm report** interface configuration command.

**atm report** {**b1-tca** | b**2-tca** | **b3-tca** | **lais** | **lcd** | **lrdi** | **pais** | **plop** | **prdi** | **rdool** | **sd-ber** | **sf-ber** | **slof** | **slos**}

**no atm report** {**b1-tca** | b**2-tca** | **b3-tca** | **lais** | **lcd** | **lrdi** | **pais** | **plop** | **prdi** | **rdool** | **sd-ber** | **sf-ber** | **slof** | **slos**}

Where:

- **b1-tca**—B1 bit error rate (BER) threshold crossing alarm
- **b2-tca**—B2 BER threshold crossing alarm
- **b3-tca**—B3 BER threshold crossing alarm
- **lais**—Line Alarm Indication Signal
- **lcd**—Loss of cell delineation
- **lrdi**—Line Remote Defect Indication
- **pais**—Path Alarm Indication Signal
- **plop**—Path Loss of Pointer
- **prdi**—Path Remote Defect Indication
- **rdool**—Receive Data Out Of Lock
- **sd-ber**—Line bit interleave parity error (LBIP) BER in excess of signal degrade (SD) threshold
- **sf-ber**—LBIP BER in excess of signal fail (SF) threshold
- **slof**—Section Loss of Frame
- **slos**—Section Loss of Signal

To disable logging of select SONET alarms, use the **no** form of this command.

Reporting an alarm means that the alarm can be logged to the console. Not all alarms are logged. SONET alarm hierarchy rules dictate that only the most severe alarm of an alarm group is reported. Whether an alarm is reported or not, you can view the current state of a defect by checking the Active Defects line from the **show controllers atm** command output.

The following example shows how to enable reporting of SD-BER and LAIS alarms on the interface:

```
Router(config)# interface atm 3/0/0
Router(config-if)# atm report sd-ber
Router(config-if)# atm report lais
Router(config-if)# end
Router#
```

## Specifying the ATM Alarm Thresholds

Specify the bit error rate (BER) threshold by using the **atm threshold** command:

**atm threshold** {**b1-tca** | **b2-tca** | **b3-tca** | **sd-ber** | **sf-ber**} *value*

Where:

- **b1-tca**—B1 BER threshold crossing alarm
- **b2-tca**—B2 BER threshold crossing alarm
- **b3-tca**—B3 BER threshold crossing alarm
- **sd-ber—**Set Signal Degrade BER threshold
- **sf-ber—**Set Signal Fail BER threshold
- *value* is an exponential value from $10^{-3}$ to $10^{-9}$ representing the BER at which an alarm occurs.

The default for all thresholds, except **sf-ber**, is $10^{-6}$. The default for **sf-ber** is $10^{-3}$.

The following example shows how to specify the B1 BER threshold crossing alarm value of 4:

```
Router(config)# interface atm 5/0/0
Router(config-if)# atm threshold b1-tca 4
```

## Controlling the S1 SONET Overhead Byte

On Cisco 10000 series routers, ATM line cards run over SONET. In most situations, the default value for the S1 SONET overhead byte (0x0) does not need to be changed. Refer to the SONET standards for information about the possible values for the S1 SONET overhead byte and the definition of each value.

### Controlling a Transmitted S1 Overhead Byte

In Cisco IOS Release 12.2(28)SB, use the **pos flag s1-byte tx** command in interface configuration mode to control the transmission of the S1 SONET overhead byte.

**pos flag s1-byte tx** *value*

Where:

- *value* is in the range of 0x0 to 0xF
- 0x0 is the default value

In the following example the S1 SONET overhead byte is set to 0xF:

```
pos flag s1-byte tx 0xF
```

### Reacting to a Received S1 Overhead Byte

In Cisco IOS Release 12.2(28)SB, use the **pos flag s1-byte rx-communicate** command to direct the router to switch the clock source to internal when it receives an S1 SONET overhead byte with a value of 0xF. When the S1 SONET overhead byte changes from 0xF to any other value, the clock source reverts back to the clock source specified in the user configuration.

The S1 overhead byte is ignored by the receiving router unless the **pos flag s1-byte rx-communicate** command is issued.

**pos flag s1-byte rx-communicate**

**no pos flag s1-byte rx-communicate**

The following example directs the router to switch to internal clocking when it receives an S1 SONET overhead byte with a value of 0xF:

```
pos flag s1-byte rx-communicate
```

The default for the **pos flag s1-byte rx-communicate** command is disabled or off.

## Running Loopbacks

Use the **loopback** command to run a loopback diagnostic tests.

```
loopback {line | diagnostic {parallel | path | serial}}
no loopback {line | diagnostic {parallel | path | serial}}
```

Where:

- **line** is the line loopback
- **diagnostic** starts an internal diagnostic loopback
- **parallel** is the internal diagnostic parallel loopback
- **path** is the internal diagnostic path loopback
- **serial** is the internal diagnostic serial loopback

The following example shows hot to run the diagnostic serial loopback:

```
Router(config)# interface atm 5/0/0
Router(config-if)# loopback diagnostic serial
```

# ATM PVC Commands

After you create a PVC using the **pvc** command, you can customize the PVC or a VC class by using the commands described in this section.



## Specifying a Protocol

Use the **protocol ip** command in interface-ATM-VC configuration mode or VC-class configuration mode to do one or both of the following:

- Configure a static map for an ATM PVC or VC class.

- Enable Inverse ARP or Inverse ARP broadcasts on an ATM PVC by either configuring Inverse ARP directly on the PVC or in a VC class (applies to IP protocols only).

```
protocol ip {protocol-address | inarp} [[no] broadcast]
no protocol ip {protocol-address | inarp} [[no] broadcast]
```

Where:

- *protocol-address* is the peer destination address that is being mapped to a PVC.
- **inarp** (valid only for IP protocols on PVCs) enables Inverse ARP on an ATM PVC. If you specify a protocol-address instead of inarp, Inverse ARP is automatically disabled for that protocol.
- [**no**] **broadcast** (optional) indicates that this PVC sends out broadcast packets (for example, IGRP updates). Pseudo broadcasting is supported. The broadcast keyword of the protocol ip command takes precedence if you previously configured the broadcast command on the ATM PVC.

  For PVCs created under point-to-point subinterfaces, broadcast is enabled by default. For PVCs created under multipoint subinterfaces, use the **broadcast** argument to propagate IP routes.

Use the **no** form of this command to remove a static map or disable Inverse ARP.

**Note** Use the **inarp** command to configure Inverse ARP frequency.

The following example shows how to specify IP protocol on an ATM PVC:

```
Router(config)# interface atm 5/0/0
Router(config-if)# pvc 0/105
Router(config-if-atm-vc)# protocol ip 172.16.32.49
```

## Configuring a Broadcast

Use the **broadcast** command to configure broadcast packet duplication and transmission for an ATM PVC or VC class.

The **broadcast** command is not used to enable ATM *cell-level* multicast, broadcast, replication, or to set up the broadcast of user level traffic. The **broadcast** command indicates which PVC (or PVCs) sends out broadcast traffic. This is typically limited to traffic associated with routing protocols and routing updates (for example, OSPF hello packets).

**Note** The **broadcast** argument within the **protocol ip** command takes precedence over the **broadcast** command. See the "Specifying a Protocol" section on page 1-15 for more information.

Use the default form of this command to restore the default behavior described below.

```
broadcast
no broadcast
```

The default is broadcast.

Use the **no** form of this command to disable transmission of broadcast packets.

For PVCs created under point-to-point subinterfaces, **broadcast** is enabled by default. For PVCs created under multipoint subinterfaces, you should use the **broadcast** command if you want to propagate IP routes (only the first PVC on a multipoint interface receives broadcast traffic).

The following example shows how to use the **broadcast** command to restore the default behavior:

```
Router(config)# interface atm 5/0/0.4
Router(config-subif)# pvc 0/105
Router(config-if-atm-vc)# broadcast
Router(config-if-atm-vc)#
```

## Configuring Inverse ARP

Use the **inarp** command in interface-ATM-VC configuration mode or VC-class configuration mode to configure the Inverse ARP time period for an ATM PVC or VC class.

**inarp** *minutes*
**no inarp** *minutes*

Where *minutes* is the inverse ARP frequency from 1 to 60 minutes.

The default frequency is 15 minutes.

Use the **no** form of this command to restore the default inverse ARP time period behavior.

**Note** This command is supported only for AAL5+SNAP encapsulation (the default) when Inverse ARP is enabled. Use the **encapsulation** command to configure AAL5+SNAP encapsulation and the **protocol** command to enable Inverse ARP.

The following example shows how to specify an inverse ARP frequency of 40 minutes on an ATM PVC:

```
Router(config)# interface atm 5/0/0
Router(config-if)# pvc 0/105
Router(config-if-atm-vc)# inarp 40
```

## Attaching an ATM VC Class to a PVC

Use the **class-vc** command to attach an ATM VC class to a PVC.

**class-vc** *name*

Where:

- *name* is the name of the class created with the global configuration **class-vc atm** command.

The following example shows how to assign an ATM VC class named *boston* to an ATM PVC:

```
Router(config)# interface atm 5/0/0.4
Router(config-subif)# pvc 2/100
Router(config-if-atm-vc)# class-vc boston
```

## Configuring VBR-NRT

Use the **vbr-nrt** command to configure the variable bit rate non real-time (VBR-NRT) traffic management type and specify output peak cell rate, output sustainable cell rate, and output maximum burst cell size for an ATM PVC or VC class.

You can use the **vbr-nrtv** command in PVC configuration mode or VC-class configuration mode.

**vbr-nrt** *peak_cell_rate sustainable_cell_rate maximum_burst_size*
**no vbr-nrt** *peak_cell_rate sustainable_cell_rate maximum_burst_size*

Where:

- *peak* is the peak cell rate (PCR) from 38 Kbps to 149,760 Kbps. The PCR must be at least equal to the sustainable cell rate (SCR)
- *sustainable* is the sustainable cell rate (SCR) from 38 Kbps to the PCR
- *maximum* is a number from 1 to 65,535, which represents maximum burst size (MBS) in cells

The default class of service is unspecified bit rate (UBR) running at the maximum line rate of the physical interface.

Use the **no** form of this command to remove the VBR-NRT parameters and return the PVC to its default of unspecified bit rate (UBR).

The following example shows how to configure the VBR-NRT traffic parameters on an ATM PVC:

```
Router(config)# interface atm 5/0/0
Router(config-if)# pvc 0/105
Router(config-if-atm-vc)# vbr-nrt 50000 20000 200
```

## Specifying Encapsulation

Use the **encapsulation** command in PVC configuration mode or VC-class configuration mode to specify the ATM adaptation layer (AAL) and encapsulation type for an ATM PVC or VC class.

```
encapsulation {aal5mux ip | aal5snap}
no encapsulation {aal5mux ip | aal5snap}
```

Where:

- **aal5mux ip** is AAL5+MUX encapsulation
- **aal5snap** is AAL5+LLC/SNAP encapsulation (the default)

Use the **no** form of this command to return an encapsulation to the default SNAP.

The following example shows how to specify **aal5mux ip** encapsulation for an ATM PVC:

```
Router(config)# interface atm 5/0/0.4
Router(config-subif)# pvc 0/105
Router(config-if-atm-vc)# encaps aal5mux ip
```

## Enabling ILMI Management

Use the **ilmi manage** command in PVC configuration mode or VC-class configuration mode to enable ILMI management on an ATM PVC. This command changes the convergence of higher-level protocols based on link-state changes.

```
ilmi manage
no ilmi manage
```

Use the **no** form of this command to disable ILMI management.

The following example shows how to enable ILMI management on an ATM PVC:

```
Router(config)# interface atm 5/0/0
Router(config-if)# pvc 0/105
Router(config-if-atm-vc)# ilmi manage
```

## Configuring OAM Retry

Use the **oam retry** command in PVC configuration mode or VC-class configuration mode to configure OAM retry.

**oam retry** *up_value* [*down_value frequency*]
**no oam retry** *up_value* [*down_value frequency*]

Where:

- *up_value* is a number from 1 to 600 that represents the OAM retry count before declaring that a VC is up.

  The default is 3 retries.
- *down_value* is a number from 1 to 600 that represents the OAM retry count before declaring a VC is down.

  The default is 5 retries.
- *frequency* is a number from 1 to 1000 that represents the OAM retry polling frequency, in seconds.

  The default is 1 second.

Use the **no** form of this command to return OAM retry to its default values.

The following example shows how to configure OAM retry for an ATM PVC:

```
Router(config)# interface atm 5/0/0
Router(config-if)# pvc 0/105
Router(config-if-atm-vc)# oam retry 10 10 5
```

## Enabling OAM Loopback Cell Generation and Management

To enable end-to-end F5 OAM loopback cell generation and OAM management for an ATM PVC or VC class, use the **oam-pvc** command in PVC configuration mode or VC-class configuration mode.

**oam-pvc** [**manage**] [*frequency*]
**no oam-pvc** [**manage**] [*frequency*]

Where:

- **manage** is an optional keyword that brings down the interface or subinterface if the PVC loopback fails.
- *frequency* (optional) is the number of seconds between transmitting OAM loopback cells. Values range from 0 to 600 seconds.

The default value is 10 seconds.

Use the **no** form of this command to disable generation of OAM loopback cells and OAM management.

The following example shows how to enable OAM loopback cell generation for an ATM PVC:

```
Router(config)# interface atm 5/0/0
Router(config-if)# pvc 0/105
Router(config-if-atm-vc)# oam-pvc 300
```

# Useful show Commands

### show atm vc

Use the **show atm vc** command to display information about the VCs on the interface.

```
Router#show atm vc
               VCD /                                              Peak Avg/Min Burst
Interface      Name         VPI   VCI  Type   Encaps     Kbps   Kbps  Cells Sts
2/0/0          1              0    16   PVC    ILMI149760UP
2/0/0          9              0   100   PVC    MUX 149760UP
2/0/0.2        7              2    32   PVC    SNAP149760UP
2/0/0          8              2    33   PVC    SNAP149760UP
2/0/0          18             2   100   PVC    SNAP149760UP
2/0/0.2        6              4    24   PVC    SNAP149760UP
2/0/0          2             25     3   PVC    F4-OAM 50000UP
2/0/0          3             25     4   PVC    F4-OAM 50000UP
2/0/0          14            25   100   PVC    SNAP 50000 50000    0UP
2/0/0          16            25   101   PVC    SNAP 50000 50000    0UP
2/0/0          17            25   102   PVC    SNAP 50000 50000    0UP
2/0/0          10            26     3   PVC    F4-OAM 50000UP
2/0/0          11            26     4   PVC    F4-OAM50000UP
2/0/0          12            27     3   PVC    F4-OAM 50000UP
2/0/0          13            27     4   PVC    F4-OAM 50000UP
2/0/0          19            33   100   PVC    SNAP 10000 8000   10UP
Router#
```

## show atm vp

Use the **show atm vp** command to display information about the VPs on the interface.

```
Router#show atm vp
                    Data  CES    Peak    CES
Interface    VPI    VCs   VCs    Kbps    Kbps     Status
ATM2/0/0     25     3     0      50000   0        ACTIVE
ATM2/0/0     26     0     0      50000   0        ACTIVE
ATM2/0/0     27     0     0      50000   0        ACTIVE
Router#
```

## show atm pvc

Use the **show atm pvc** *vpi_number/vci_number* command to display detailed information about a specific PVC.

```
Router#show atm pvc 0/100

ATM2/0/0: VCD: 9, VPI: 0, VCI: 100
UBR, PeakRate: 149760
AAL5-MUX, etype:0x800, Flags: 0xC23, VCmode: 0x0
OAM frequency: 0 second(s), OAM retry frequency: 1 second(s)
OAM up retry count: 3, OAM down retry count: 5
OAM Loopback status: OAM Disabled
OAM VC state: Not Managed
ILMI VC state: Not Managed
InARP DISABLED
InPkts: 0, OutPkts: 0, InBytes: 0, OutBytes: 0
InPRoc: 0, OutPRoc: 0, Broadcasts: 0
InFast: 0, OutFast: 0, InAS: 0, OutAS: 0
InPktDrops: 0, OutPktDrops: 0
Out CLP=1 Pkts: 0
OAM cells received: 0
F5 InEndloop: 0, F5 InSegloop: 0, F5 InAIS: 0, F5 InRDI: 0
OAM cells sent: 0
F5 OutEndloop: 0, F5 OutSegloop: 0, F5 OutRDI: 0
OAM cell drops: 0
PVC Discovery: NOT_VERIFIED
Status: UP
Router#
```

CHAPTER 2

# 8-Port E3/DS3 ATM Line Card Configuration

This chapter describes procedures for configuring the Cisco 10000 series 8-Port E3/DS3 ATM line card, hereafter known as the 8-Port E3/DS3 ATM line card.

The 8-Port E3/DS3 ATM line card has eight E3 or DS3 (T3) copper interface terminations that provide Asynchronous Transfer Mode (ATM) serial interfaces and perform Layer 2 ATM functions.

This chapter contains the following sections:



# Software Support

Table 2-1 shows the minimum Cisco IOS release on each release train that supports the 8-Port E3/DS3 ATM line card.

***Table 2-1 8-Port E3/DS3 ATM Line Card Software Support***

| Required PRE | Minimum Cisco IOS Releases |
|---|---|
| PRE2 | Cisco IOS Release 12.2(16)BX and later releases of Cisco IOS Release 12.2BX<br>Cisco IOS Release 12.3(7)XI and later releases of Cisco IOS Release 12.3XI<br>Cisco IOS Release 12.2(28)SB and later releases of Cisco IOS 12.2SB |

## Checking Hardware and Software Compatibility

The PRE installed in the Cisco 10000 series router chassis must support the Cisco IOS software running on the router. Use the **show version** command to check the PRE version installed.

To see if a feature is supported by a Cisco IOS release, to locate the software document for that feature, or to check the minimum software requirements of Cisco IOS software with the hardware installed on your router, Cisco maintains the Software Advisor tool on Cisco.com at http://www.cisco.com/cgi-bin/Support/CompNav/Index.pl

This tool does not verify whether line cards within a system are compatible, but does provide the minimum Cisco IOS requirements for individual hardware line cards, modules, or options.

You must be a registered user on Cisco.com to access this tool.

# Default Values

Table 2-2 lists default configuration values for the 8-Port E3/DS3 ATM line card. This table also includes the commands used to modify a default values, and provides information about values to set on the remote end of the connection.

***Table 2-2 8-Port E3/DS3 ATM Line Card Defaults***

| Command Name | Default Setting | Command Syntax | Remote Side Setting |
|---|---|---|---|
| mtu (maximum transmission unit) | 4470 bytes | **[no] mtu** *bytes* | same |
| atm clock internal | line | **[no] atm clock internal** | opposite |
| loopback | loopback none | **[no] loopback [line \| diagnostic \| payload]** | — |

# Line Card VC Limitations

The Cisco 10000 series router supports four ATM service categories for virtual circuits (VCs):

- Constant Bit Rate (CBR)
- Variable Bit Rate-non-real-time (VBR-nrt)
- Unspecified Bit Rate (UBR) with a peak cell rate (PCR), referred to as shaped UBR
- UBR without a PCR, referred to as unshaped UBR

The segmentation and reassembly (SAR) mechanism configures priority and additional traffic management parameters for the various ATM service categories. Table 2-3 lists the priority levels the SAR sets for the service categories.

***Table 2-3 ATM Service Categories***

| Parameter | CBR | VBR-rt | VBR-nrt | Shaped UBR | Unshaped UBR |
|---|---|---|---|---|---|
| Priority | 0 | 1 | 2 | 3 | None |

The number of SAR priority levels and the service categories supported at each priority level vary from line card to line card. For example, the 8-port E3/DS3 supports the two levels of priority and the service categories listed in the Table 2-4.

The ATM line cards support a maximum number of VCs per priority. That VC limit depends on the VC limit of the SAR (SAR limit) and the number of priority levels configured. Table 2-4 describes how to determine the VC limit per priority level per port for the 8-port E3/DS3 line card.

*Table 2-4 Maximum Number of VCs per Priority per Port*

| ATM Line Card | SAR Priority Levels | VC Rate | Maximum Number of VCs per Priority |
|---|---|---|---|
| 8-Port E3/DS3 | 0 = CBR VCs<br>0 = VBR-nrt VCs<br>1 = UBR VCs | Half line rate and below | SAR limit / number of PHYs / number of priority levels<br>2 priority system:<br>65,536 / 8 / 2 = 4096 VCs per priority level per port |

Configuring more channels or VCs than there are available priority locations can cause random channels or VCs to get stuck in the SAR. This occurs when an active channel tries to reschedule itself, but no priority locations are available. Therefore, the channel cannot find a place to reschedule itself, which results in a lost event for the channel, and the channel becomes stuck in the SAR.

On the PRE2, when a VC becomes stuck in the SAR, the PRE2 scheduler stops forwarding traffic on only the VC that is stuck in the SAR; the other VCs still carry traffic. On the PRE3, the PRE3 scheduler stops forwarding traffic on all the VCs configured on that ATM line card.

# Interface Syntax

To specify an interface number in a configuration command, use the syntax in Table 2-5 to identify interfaces and subinterfaces on the 8-Port E3/DS3 ATM line card in the Cisco series 10000 router.

*Table 2-5 8-Port E3/DS3 ATM Line Card Interface Syntax*

| Type of Interface | Router | Slot | Subslot | Port | Subinterface |
|---|---|---|---|---|---|
| 8-Port E3/DS3 interface | 10008 | 1 through 8/ | 0/ | 0 to 7 | — |
| 8-Port E3/DS3 subinterface | 10008 | 1 through 8/ | 0/ | 0.n to 7.n | n = 1 to 4294967295 |
| 8-Port E3/DS3 interface | 10005 | 1 through 5/ | 0/ | 0 to 7 | — |
| 8-Port E3/DS3 subinterface | 10005 | 1 through 5/ | 0/ | 0.n to 7.n | n = 1 to 4294967295 |

Examples:

- Modify a permanent virtual circuit (PVC) associated with an interface.

```
Router(config)# interface atm 2/0/0
Router(config-if)# pvc 0/200
Router(config-if-atm-vc)#
```

- Modify a PVC associated with a subinterface.

```
Router(config)# interface atm 7/0/0.1
Router(config-subif)# pvc 0/101
Router(config-if-atm-vc)#
```

# Line Card Interface Configuration Examples

This section provides sample procedures for configuring the 8-Port E3/DS3 ATM line card for E3 or T3 connections, as well as creating ATM subinterfaces, permanent virtual circuits (PVCs), and virtual circuit (VC) classes. Procedures for enabling Integrated Local Management Interface (ILMI) are also included.

## Creating a Subinterface

Use the following procedure to create a subinterface on an 8-Port E3/DS3 ATM line card interface.

**Step 1** Separate the ATM interface into subinterfaces using the **interface** command. You can create either a point-to-point or multipoint subinterface.

In the following example, multipoint subinterface number 1 is created on port 0 of the 8-Port E3/DS3 ATM line card in slot 2.

```
Router(config)# interface atm 2/0/0.1 multipoint
Router(config-subif)#
```

**Step 2** Assign an IP address and subnet mask to the subinterface using the **ip address** command, as shown in the following example:

```
Router(config-subif)# ip address 172.27.48.209 255.255.0.0
```

You created ATM subinterface 2/0/0.1. To configure or modify this interface, use the following command:

```
Router(config)# interface atm 2/0/0.1
Router(config-subif)#
```

## Creating a PVC

You can create multiple PVCs on an 8-Port E3/DS3 ATM line card interface. You can create PVCs on an interface or subinterface.

To create a PVC, perform the following steps:

**Step 1** Enter interface or subinterface configuration mode.

Use the **pvc** command to specify a virtual path identifier (VPI) value between 0 and 255 and a virtual channel identifier (VCI) value between 0 and 65535. The following example shows how to create a PVC with a VPI value of 0 and VCI value of 100 on a subinterface.

```
Router(config-subif)# pvc 0/100
Router(config-if-atm-vc)#
```

**Step 2** Use the **protocol ip** command to assign a peer IP address to the PVC, as shown in the following example:

```
Router(config-if-atm-vc)# protocol ip 172.16.32.49
Router(config-if-atm-vc)#
```

# Creating an ATM VC Class

This procedure demonstrates how to create an ATM VC class. An ATM VC class is a PVC boilerplate—a PVC description that you can apply to one or more PVCs.

To create an ATM VC class, perform the following steps:

**Step 1** From global configuration mode, enter the **vc-class atm** *name* command, where *name* is the name value you assign. The following example shows how to create the ATM VC class named *boston*.

```
Router(config)# vc-class atm boston
Router(config-vc-class)#
```

**Step 2** Enter commands to describe the ATM VC class you named boston. This example shows how to specify that the boston class uses AAL5+MUX encapsulation with a variable bit-rate non real-time (VBR-NRT) PVC.

```
Router(config-vc-class)# encapsulation aal5mux ip
Router(config-vc-class)# vbr-nrt 30000 20000 128
Router(config-vc-class)# exit
Router(config)#
```

You have created a VC class named boston. The next procedure describes steps for applying this class to a PVC or subinterface.

# Applying a VC Class

You can apply a VC class (created in the previous procedure) to a PVC or a subinterface.

- In the following example, the class named boston is applied to a new PVC (0/102) in subinterface 5/0/0.2.

```
Router(config)# interface atm 5/0/0.2
Router(config-subif)# pvc 0/102
Router(config-if-atm-vc)# class-vc boston
Router(config-if-atm-vc)#
```

- In the following example, the class named boston is applied to subinterface 5/0/0.1.

```
Router(config)# interface atm 5/0/0.1
Router(config-subif)# class-int boston
Router(config-subif)#
```

You have completed the steps for assigning a VC class to a PVC.

# Enabling ILMI PVC Discovery

You can enable ILMI to automatically discover PVCs on neighboring switches and duplicate those PVC entries on the 8-Port E3/DS3 ATM line card.

To enable ILMI, perform the following steps:

**Step 1** Create PVC 0/16 on the main interface as shown in the following example:

```
Router(config)# interface atm 5/0/0
Router(config-if)# pvc 0/16 ilmi
Router(config-if-atm-vc)# exit
Router(config-if)#
```

**Step 2** Enable ILMI PVC discovery for the selected port (step 1 references port 0).

```
Router(config-if)# atm ilmi-pvc-discovery
```

You have completed the steps required to run ILMI PVC discovery. You can use the **show atm pvc** command to display the PVCs on the Cisco 10000 series router.

## Completing a Configuration

This section offers general information on creating and completing a configuration of an 8-Port E3/DS3 ATM line card.

To configure an interface, perform the following steps:

**Step 1** Enter interface configuration mode and specify necessary parameters, such as the IP address and subnet mask.

**Step 2** After you enter all of the commands to complete the configuration, press **Ctrl-Z** (hold down the **Control** key while you press **Z**) to exit configuration mode.

**Step 3** Write the new configuration to memory:

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

The system displays an OK message when the configuration is stored.

To check your completed configuration, use the **show** commands. For more information about show commands, see Useful show Commands, page 2-21.

# Line Card Commands

The 8-Port E3/DS3 ATM line card allows you to specify eight E3 or eight T3 interfaces. You can put all the PVCs on the interfaces, or you can create associated subinterfaces. This section describes the principal commands for customizing interfaces and PVCs:


- Global Configuration Commands, page 2-7
- ATM Interface and Subinterface Commands, page 2-7
- ATM PVC Commands, page 2-16
- Useful show Commands, page 2-21

# Global Configuration Commands

A PVC boilerplate is a PVC description that you can apply to one or more PVCs or interfaces.

To create a PVC boilerplate, use the **vc-class atm** command in global configuration mode:

**vc-class atm** *class_name*

Where *class_name* is any value that describes the VC class.

After you enter the **vc-class atm** command, you enter VC class configuration mode. In this mode, you describe the action you want the class to take by entering commands and their arguments. These commands and arguments are described in the "ATM PVC Commands" section on page 2-16.

In the following example, an ATM VC class named *paris* is created and defined. This example shows how to specify that the class uses AAL5+MUX encapsulation and a VBR-NRT PVC.

```
Router(config)# vc-class atm paris
Router(config-vc-class)# encapsulation aal5mux ip
Router(config-vc-class)# vbr-nrt 30000 20000 128
Router(config-vc-class)# exit
Router(config)#
```

For information on applying a VC class name, see the "Attaching an ATM VC Class to an Interface or Subinterface" section on page 2-12 and the "Attaching an ATM VC Class to a PVC" section on page 2-18.

# ATM Interface and Subinterface Commands

This section describes how to configure ATM interfaces and subinterfaces using the principle commands described in the following sections:

## Setting the Line Card to E3 or T3 mode

You must configure the 8-Port E3/DS3 ATM line card for either E3 or T3 from interface configuration mode.

**Note** When you configure the 8-Port E3/DS3 ATM line card for E3 or T3, all 8 ports of the line card operate in the mode you have selected.

**atm dsx3mode** {**e3** | **t3**}{**adm** | **plcp**}

Where:

**adm** means ATM direct mapping

**plcp** means physical layer convergence procedure (PLCP)—specifications that map ATM cells into physical media, such as T3 or E3, and define certain management information.

The default is t3 adm.

In the following example, the 8-Port E3/DS3 ATM line card located in slot 2 is configured for E3 mode with PLCP:

```
Router(config)# interface atm 2/0/0
Router(config-if)# atm dsx3mode e3 plcp
```

## Changing the Mode of the Line Card

The 8-Port E3/DS3 ATM line card has four modes (e3 adm, e3 plcp, t3 adm and t3 plcp), the default mode being t3 adm. Each mode has it own bandwidth and cell delay variation tolerance (CDVT) values.

When a line card is freshly inserted, the default mode of the card can be changed. However, if VCs, VPs or subinterfaces have already been configured on the line card, the line card must first be removed and reprovisioned prior to the mode change.

To change the mode of an already provisioned card use the following commands in the configuration mode:

**hw-module slot <> shut**

**no card <>**

**no hw-module slot <> shut**

After the line card is restarted you can change the mode, and re-apply the configurations that were lost during module shutdown.

The following example shows how to change the mode of an already provisioned card, from E3 mode with ADM to E3 mode with PLCP.

```
Router(config)# hw-module slot 2/0 shut
Router(config)# no card 2/0
Router(config)# no hw-module slot 2/0 shut
```

To change the mode of the line card:

```
Router(config)# interface atm 2/0/0
Router(config-if)# atm dsx3mode e3 plcp
```

To re-apply the configurations to the line card:

```
Router(config)#interface atm 2/0/0.1 point-to-point
Router(config-subif)#pvc 1/100
Router(config-if-atm-vc)#vbr-nrt 4000 4000
```

```
Router(config-if-atm-vc)#end
```

## Setting the Line Card Framing

The 8-Port E3/DS3 ATM line card supports ADM and PLCP framing on E3 and T3 connections. The following sections describe how to set T3 ADM, T3 PLCP, E3 ADM, and E3 PLCP framing.

### Setting T3 ADM Framing

If you configured the 8-Port E3/DS3 ATM line card for T3 ADM using the **atm dsx3mode ds3 adm** command, configure the framing using the **atm framing** command in interface configuration mode.

[**no**] **atm framing** {**cbitadm** | **m23adm**}

Where:

**cbitadm** is C-bit ADM T3 Framing

**m23adm** is M23 ADM T3 Framing

The default is cbitadm.

In the following example, M23 ADM T3 framing is set:

```
Router(config)# interface atm 2/0/0
Router(config-if)# atm framing m23adm
```

### Setting T3 PLCP Framing

If you configured the 8-Port E3/DS3 ATM line card for T3 PLCP using the **atm dsx3mode ds3 plcp** command, configure the framing using the **atm framing** command from interface configuration mode.

[**no**] **atm framing** {**cbitplcp** | **m23plcp**}

Where:

**cbitplcp** is C-bit PLCP T3 Framing

**m23plcp** is M23 PLCP T3 Framing

The default cbitplcp

In the following example, C-bit PLCP T3 framing is set:

```
Router(config)# interface atm 2/0/0
Router(config-if) atm framing cbitplcp
```

### Setting E3 ADM Framing

If you configured the 8-Port E3/DS3 ATM line card for E3 ADM using the **atm dsx3mode e3 adm** command, configure the framing using the **atm framing** command in interface configuration mode.

[**no**] **atm framing** {**g751adm** | **g832adm**}

Where:

**g751adm** is G.751 ADM E3 framing

**g832adm** is G.832 ADM E3 framing

The default is g751adm.

In the following example, G.751 ADM E3 framing is set:

```
Router(config)# interface atm 2/0/0
```

```
Router(config-if)# atm framing g751adm
```

### Setting E3 PLCP Framing

If you configured the 8-Port E3/DS3 ATM line card for E3 PLCP using the **atm dsx3mode e3 plcp** command, configure the framing using the **atm framing** command in interface configuration mode.

[**no**] **atm framing g751plcp**

Where **g751plcp** is G.751 PLCP E3 framing

The default is g751plcp.

In the following example, G.751 PLCP E3 framing is set:

```
Router(config)# interface atm 2/0/0
Router(config-if)# atm framing g751plcp
```

## Creating and Entering Subinterfaces

Use the **interface** command to segment an E3 or T3 ATM main interface into multiple subinterfaces to simplify line card management, to create interfaces with different MTU sizes, and to create connections to different networks.

**interface atm** *slot/subslot/port.subinterface type*
**no interface atm** *slot/subslot/port.subinterface type*

Where:

- *slot* is 1 to 5 for a Cisco 10005 router.
- *slot* is 1 to 8 for a Cisco 10008 router.
- *subslot* is always 0 for a full-height line card.
- *port* is 0 to 3.
- *subinterface* is a value from 1 to 4294967295.
- *type* is always **point-to-point** or **multipoint**.

To remove a subinterface and its PVCs, use the **no interface** command. To change a subinterface type, you must first remove the subinterface.

Examples:

- Create subinterface number 1 at port 3 for 8-Port E3/DS3 ATM line card in slot 1.

```
Router(config)# interface atm 1/0/3.1 point-to-point
Router(config-subif)#
```

- Enter an existing subinterface.

```
Router(config)# interface atm 1/0/3.1
Router(config-subif)#
```

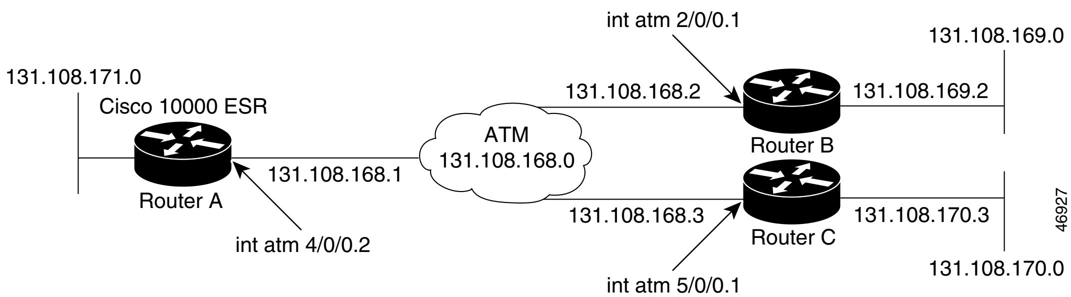
- Use a multipoint subinterface.

  If you want multiple PVCs to go to the same network, you must create a multipoint subinterface. For example:

```
Router(config)# interface atm 4/0/2.2 multipoint
Router(config-subif)#
```

After you create the subinterface, you can create PVCs that are attached to the same network. Figure 2-1 shows a multipoint subinterface on a fully meshed network. Fully meshed indicates that any workstation can communicate with any other workstation.

*Figure 2-1 Multipoint ATM Configuration*




The following examples show the commands used to configure the ATM interfaces for routers A, B, and C (see Figure 2-1), including the configuration commands you must enter to configure the multipoint connections in that network:

Router A

```
Router(config)# interface atm 4/0/0.2 multi
Router(config-subif)# ip address 131.108.168.1 255.255.255.0
Router(config-subif)# pvc 0/35
Router(config-if-atm-vc)# protocol ip 131.108.168.2 broadcast
Router(config-if-atm-vc)# exit
Router(config-subif)# pvc 0/36
Router(config-if-atm-vc)# protocol ip 131.108.168.3 broadcast
Router(config-if-atm-vc)# exit
Router(config-subif)# exit
Router(config)#
```

Router B

```
Router(config)# interface atm 2/0/0.1 multi
Router(config-subif)# ip address 131.108.168.2 255.255.255.0
Router(config-subif)# pvc 0/35
Router(config-if-atm-vc)# protocol ip 131.108.168.1 broadcast
Router(config-if-atm-vc)# exit
Router(config-subif)# pvc 0/37
Router(config-if-atm-vc)# protocol ip 131.108.168.3 broadcast
Router(config-if-atm-vc)# exit
Router(config-subif)# exit
Router(config)#
```

Router C

```
Router(config)# interface atm 5/0/0.1 multi
Router(config-subif)# ip address 131.108.168.3 255.255.255.0
Router(config-subif)# pvc 0/36
Router(config-if-atm-vc)# protocol ip 131.108.168.1 broadcast
Router(config-if-atm-vc)# exit
Router(config-subif)# pvc 0/37
Router(config-if-atm-vc)# protocol ip 131.108.168.2 broadcast
Router(config-if-atm-vc)# exit
Router(config)#
```

## Attaching an ATM VC Class to an Interface or Subinterface

Use the **class-int** command to attach an ATM VC class to an interface or subinterface. If you customize a PVC, that customization takes precedence over the interface class.

**class-int** *class_name*

Where *class_name* is the name of the class created using the global configuration **class-vc atm** command.

In the following example, a VC class named *paris* is created and attached to subinterface 3/0/0.1.

```
Router(config)# vc-class atm paris
Router(config-vc-class)# encapsulation aal5mux ip
Router(config-vc-class)# vbr-nrt 30000 20000 128
Router(config-vc-class)# exit
Router(config)# interface atm 3/0/0.1
Router(config-subif)# class-int paris
```

## Creating ATM PVPs

To create a permanent virtual path (PVP) used to multiplex one or more VBR-NRT VCs, use the **atm pvp** interface configuration command.

**atm pvp** *vpi peak-rate* [**no-f4-oam**]
**no atm pvp** *vpi*

Where:

- *vpi* is the ATM network virtual path identifier (VPI) of the VC used to multiplex the permanent virtual path. The range is 0 to 255. You must use a VPI value that is not already in use by a VC.
- *peak-rate* is the maximum rate in kilobits per second (kbps) at which the PVP can transmit data. For an E3 interface, you can enter values from 38 kbps to 34,368 kbps. For a T3 interface, you can enter values from 38 kbps to 44,200 kbps. The PVP peak rate value supersedes all rate values set for VBR-NRT PVCs associated with the PVP.

| Interface Type | Allowable Kbps Range for *peak-rate* |
|---|---|
| E3 | 38 to 34,368 |
| T3 | 38 to 44,200 |

- **no-f4-oam** (optional) restricts the PVP from passing operations/administration/maintenance (OAM) packets. When you create a PVP, the system creates (by default) PVCs with VCI values of 3 and 4 for each PVP which passes OAM packets.

To verify the configuration of a PVP, use the **show atm vp** exec command.

The following example shows how to create a PVP with a peak rate of 30,000 kbps. Subsequently, the created VCs are multiplexed onto this virtual path.

```
Router(config)# interface atm 7/0/0
Router(config-if)# atm pvp 25 30000
Router(config-if)# pvc 25/100
Router(config-if-atm-vc)# vbr-nrt 10000 5000 16
Router(config-if-atm-vc)# exit
Router(config-if)# pvc 25/101
Router(config-if-atm-vc)# vbr-nrt 10000 5000 16
Router(config-if-atm-vc)# exit
Router(config-if)# pvc 25/102
Router(config-if-atm-vc)# vbr-nrt 10000 5000 16
Router(config-if-atm-vc)# exit
Router(config)#
```

## Creating a PVC

This section describes how to create a permanent virtual circuit.

**Note** You can only create one PVC on a point-to-point interface. Multiple PVCs can be created on a multipoint interface.

You can establish an unspecified bit rate (UBR) PVC by entering the **pvc** command:

**pvc** [*word*] [*vpi_value*/]{*vci_value*} [**ilmi**]

- Where:
- *word* is an optional name referring to this connection.
- *vpi_value* is in the range from 0 to 255. If you do not specify a VPI value, the system assigns the value 0.
- *vci_value* is in the range 1 to 65535. The VCI value should be 33 or greater because all lower-numbered PVCs are already assigned.
- **ilmi** parameter maps the ILMI channel to the PVC for this interface. You can only use this argument for PVCs created on the main interface. We recommend that you use this argument with PVC 0/16. For more information about activating ILMI, see the "Activating ATM ILMI PVC Discovery" section on page 2-14.

By default, the **pvc** command creates a UBR PVC. To specify a VBR-NRT PVC, see the "Configuring VBR-NRT" section on page 2-19.

Examples:

- Create PVC 0/105 on the interface.

```
Router(config)# interface atm 5/0/0
Router(config-if)# pvc 105
Router(config-if-atm-vc)#
```

- Create PVC 2/102 on a subinterface.

```
Router(config)# interface atm 5/0/0.1
Router(config-subif)# pvc 2/102
Router(config-if-atm-vc)#
```

## Enabling ATM ILMI

Use the **atm ilmi-enable** interface configuration command to enable ILMI on a port.

**`atm ilmi-enable`**
**`no atm ilmi-enable`**

The default is ILMI is enabled, but you should disable the ILMI if the peer does not support ILMI. For peers to be able to exchange ILMI information, you must create PVC 0/16 using the **ilmi** argument.

The following example shows how to disable ILMI:

```
Router(config)# interface atm 5/0/0
Router(config-if)# no atm ilmi-enable
```

## Activating ATM ILMI PVC Discovery

This command causes ILMI-compliant devices to propagate PVCs. Use the **atm ilmi-pvc-discovery** command in interface configuration mode to activate ATM PVC discovery.

**`atm ilmi-pvc-discovery`** [**`subinterface`**]
**`no atm ilmi-pvc-discovery`** [**`subinterface`**]

Where:

- **subinterface** instructs the software to associate all PVCs with existing subinterface numbers equal to their VPI numbers. For example: PVC 2/102 is listed under subinterface 7/0/0.2, PVC 12/156 is listed under 7/0/0.12, and so on.

**Note** The **subinterface** argument associates PVCs only with subinterfaces that have already been created. If there is no subinterface for a VPI value, the system associates the PVC with the main interface.

The following example shows how to enable PVC discovery on the ATM interface 7/0/0, for port 0, on an 8-Port E3/DS3 ATM line card that has ILMI enabled.

```
Router(config)# interface atm 7/0/0
Router(config-if)# pvc 0/16 ilmi
Router(config-if-atm-vc)# exit
Router(config-if)# atm ilmi-pvc-discovery
```

## Specifying the ATM ILMI Keep-Alive Rate

Use the **atm ilmi-keepalive** command to specify the ILMI keep-alive rate.

**`atm ilmi-keepalive [`**_`seconds`_ **`[retry`** _`counts`_**`]]`**

Where:

- *seconds* is a value from 1 to 65535
- **retry** *counts* is a value from 2 to 5

The default value for seconds is 5000 and for retry counts is 4.

The following example shows how to enable ILMI keepalives for the ATM interface 5/0/0:

```
Router(config)# interface atm 5/0/0
Router(config-if)# atm ilmi-keepalive 10000 retry 3
```

## Configuring the ATM Clock

Use the **atm clock internal** command to configure the clock source as internal.

```
atm clock {internal | line}
no atm clock internal
```

The default clock setting is **no atm clock internal**, which means that clocking is derived from the line.

In the following example, clocking is set to internal.

```
Router(config)# interface atm 5/0/0
Router(config-if)# atm clock internal
```

## Configuring Cable Length

The **atm lbo** command improves signal strength for losses associated with lengthy cables. To specify whether the cable attached to the interface is short or long, use the **atm lbo** command in the following format:

```
atm lbo {short | long}
```

Where:

**short** is for a cable that is less than 225 feet.

**long** is for a cable that is 225 feet or greater.

The default is short.

In the following example, the cable length is set to short:

```
Router(config)# interface 2/0/0
Router(config-controller)# atm lbo short
```

## Controlling ATM Alarm Reporting

To control alarms so that they are logged to the console for an ATM interface, use the **atm report** command in interface configuration mode.

```
atm report {los | oof | ais | ferf | lcd}
no atm report {los | oof | ais | ferf | lcd}
```

Where:

- **los**—Loss of Signal
- **oof**—Out of Frame
- **ais**—Alarm Indication Signal
- **ferf**—Far End Receive Failure
- **lcd**—Loss of cell delineation

To disable logging of alarms, use the **no** form of this command.

Reporting an alarm means that the alarm can be logged to the console. Not all alarms are logged. The alarm hierarchy rules dictate that only the most severe alarm of an alarm group is reported. Whether an alarm is reported or not, you can view the current state of a defect by checking the Active Defects line from the **show controllers atm** command output.

The following example shows how to enable reporting of SD-BER and AIS alarms on the interface:

```
Router(config)# interface atm 3/0/0
Router(config-if)# atm report los
Router(config-if)# atm report ais
Router(config-if)# end
Router#
```

## Running Loopbacks

Use the **loopback** command to run a loopback tests.

**loopback** {**diagnostic** | **payload** | **line**}
**no loopback** {**diagnostic** | **payload** | **line**}

Where:

- **diagnostic** means transmit data is looped to receive data.
- **payload** means the DS3 or E3 overhead bits are regenerated and inserted into the received DS3 or E3 data stream, and the resulting stream is transmitted.
- **line** means the received data stream is looped to the transmit data stream.

The following example shows how to run the diagnostic loopback:

```
Router(config)# interface atm 5/0/0
Router(config-if)# loopback diagnostic
```

# ATM PVC Commands

After you create a PVC using the **pvc** command, you can customize the PVC or a VC class by using the commands described in this section.

- Specifying a Protocol, page 2-17
- Configuring a Broadcast, page 2-17
- Configuring Inverse ARP, page 2-18
- Attaching an ATM VC Class to a PVC, page 2-18
- Configuring VBR-NRT, page 2-19
- Specifying Encapsulation, page 2-19
- Enabling ILMI Management, page 2-20
- Configuring OAM Retry, page 2-20
- Enabling OAM Loopback Cell Generation and Management, page 2-20

## Specifying a Protocol

Use the **protocol ip** command in interface-ATM-VC configuration mode or VC-class configuration mode to configure a static map for an ATM PVC or VC class.

**protocol ip** {*protocol-address* | **arp** | **cdp** | **clns** | **clns_es** | **clns_is** | **cmns** | **compressedtcp** | {**ip** {*address*} | **ppp** | **pppoe** | **snapshot**}} [[**no**] **broadcast**]}

**no protocol ip** {*protocol-address* | **arp** | **cdp** | **clns** | **clns_es** | **clns_is** | **cmns** | **compressedtcp** | {**ip** {*address*} | **ppp** | **pppoe** | **snapshot**}} [[**no**] **broadcast**]}

Where:

- *protocol-address* is the peer destination address that is being mapped to a PVC.
- **arp** (valid only for IP protocols on PVCs) enables ARP on an ATM PVC.
- **cdp** is the Cisco Discovery Protocol.
- **clns** is ISO Connectionless Network Service (CLNS).
- **clns_es** is ISO CLNS end system.
- **clns_is** is ISO CLNS intermediate system.
- **cmns** is ISO Connection-Mode Network Service (CMNS).
- **compressedtcp** is compressed TCP.
- **ip** is IP.
- **ppp** is LLC PPP over AAL5 encapsulation.
- **pppoe** is PPP over Ethernet.
- **snapshot** is Snapshot routing support.
- [**no**] **broadcast** (optional) indicates that this PVC sends out broadcast packets (for example, IGRP updates). Pseudo broadcasting is supported. The broadcast keyword of the **protocol ip** command takes precedence if you previously configured the **broadcast** command on the ATM PVC.

  For PVCs created under point-to-point subinterfaces, **broadcast** is enabled by default. For PVCs created under multipoint subinterfaces, use the **broadcast** argument to propagate IP routes.

Use the **no** form of this command to remove a static map.

**Note** Use the **inarp** command to configure Inverse ARP frequency.

The following example shows how to specify IP protocol on an ATM PVC:

```
Router(config)# interface atm 5/0/0
Router(config-if)# pvc 0/105
Router(config-if-atm-vc)# protocol ip 172.16.32.49
```

## Configuring a Broadcast

Use the **broadcast** command to configure broadcast packet duplication and transmission for an ATM PVC or VC class.

The **broadcast** command is not used to enable ATM *cell-level* multicast, broadcast, replication, or to set up the broadcast of user level traffic. The **broadcast** command indicates which PVC (or PVCs) sends out broadcast traffic. This is typically limited to traffic associated with routing protocols and routing updates (for example, OSPF hello packets).

**Note** The **broadcast** argument within the **protocol ip** command takes precedence over the **broadcast** command. See the "Specifying a Protocol" section on page 2-17 for more information.

Use the default form of this command to restore the default behavior described below.

```
broadcast
no broadcast
```

The default is broadcast.

Use the **no** form of this command to disable transmission of broadcast packets.

For PVCs created under point-to-point subinterfaces, **broadcast** is enabled by default. For PVCs created under multipoint subinterfaces, you should use the **broadcast** command if you want to propagate IP routes (only the first PVC on a multipoint interface receives broadcast traffic).

The following example shows how to use the **broadcast** command to restore default behavior:

```
Router(config)# interface atm 5/0/0.4
Router(config-subif)# pvc 0/105
Router(config-if-atm-vc)# broadcast
Router(config-if-atm-vc)#
```

## Configuring Inverse ARP

Use the **inarp** command in interface-ATM-VC configuration mode or VC-class configuration mode to configure the inverse ARP time period for an ATM PVC or VC class.

```
inarp minutes
no inarp minutes
```

Where *minutes* is the inverse ARP frequency from 1 to 60 minutes.

The default frequency is 15 minutes.

Use the **no** form of this command to restore the default inverse ARP time period behavior.

**Note** This command is supported only for AAL5+SNAP encapsulation (the default) when Inverse ARP is enabled. Use the **encapsulation** command to configure AAL5+SNAP encapsulation and the **protocol** command to enable Inverse ARP.

The following example shows how to specify an inverse ARP frequency of 40 minutes on an ATM PVC:

```
Router(config)# interface atm 5/0/0
Router(config-if)# pvc 0/105
Router(config-if-atm-vc)# inarp 40
```

## Attaching an ATM VC Class to a PVC

Use the **class-vc** command to attach an ATM VC class to a PVC.

```
class-vc name
```

Where:

- *name* is the name of the class created with the global configuration **class-vc atm** command.

The following example shows how to assign an ATM VC class named *boston* to an ATM PVC:

```
Router(config)# interface atm 5/0/0.4
Router(config-subif)# pvc 2/100
Router(config-if-atm-vc)# class-vc boston
```

## Configuring VBR-NRT

Use the **vbr-nrt** command to configure the variable bit rate non real-time (VBR-NRT) traffic management type and specify output peak cell rate, output sustainable cell rate, and output maximum burst cell size for an ATM PVC or VC class.

You can use the **vbr-nrt** command in PVC configuration mode or VC-class configuration mode.

**vbr-nrt** *peak_cell_rate sustainable_cell_rate maximum_burst_size*
**no vbr-nrt** *peak_cell_rate sustainable_cell_rate maximum_burst_size*

Where:

- *peak_cell_rate* is the peak cell rate (PCR). For an E3 interface, you can enter values from 38 kbps to 34,368 kbps. For a T3 interface, you can enter values from 38 kbps to 44,200 kbps. The PCR must be at least equal to the sustainable cell rate (SCR).
- *sustainable_cell_rate* is the sustainable cell rate (SCR) from 38 kbps to the PCR.
- *maximum_burst_size* is a number from 1 to 65,535, which represents maximum burst size (MBS) in cells.

The default class of service is unspecified bit rate (UBR) running at the maximum line rate of the physical interface.

Use the **no** form of this command to remove the VBR-NRT parameters and return the PVC to its default of UBR.

The following example shows how to configure the VBR-NRT traffic parameters on an ATM PVC:

```
Router(config)# interface atm 5/0/0
Router(config-if)# pvc 0/105
Router(config-if-atm-vc)# vbr-nrt 10000 5000 200
```

## Specifying Encapsulation

Use the **encapsulation** command in PVC configuration mode or VC-class configuration mode to specify the ATM adaptation layer (AAL) and encapsulation type for an ATM PVC or VC class.

**encapsulation** {**aal5mux ip** | **aal5snap**}
**no encapsulation** {**aal5mux ip** | **aal5snap**}

Where:

- **aal5mux ip** is AAL5+MUX encapsulation.
- **aal5snap** is AAL5+LLC/SNAP encapsulation (the default).

Use the **no** form of this command to return an encapsulation to the default SNAP.

The following example shows how to specify **aal5mux ip** encapsulation for an ATM PVC:

```
Router(config)# interface atm 5/0/0.4
Router(config-subif)# pvc 0/105
Router(config-if-atm-vc)# encapsulation aal5mux ip
```

## Enabling ILMI Management

Use the **ilmi manage** command in PVC configuration mode or VC-class configuration mode to enable ILMI management on an ATM PVC. This command changes the convergence of higher-level protocols based on link-state changes.

**ilmi manage**
**no ilmi manage**

Use the **no** form of this command to disable ILMI management.

The following example shows how to enable ILMI management on an ATM PVC:

```
Router(config)# interface atm 5/0/0
Router(config-if)# pvc 0/105
Router(config-if-atm-vc)# ilmi manage
```

## Configuring OAM Retry

Use the **oam retry** command in PVC configuration mode or VC-class configuration mode to configure OAM retry.

**oam retry** *up_value* [*down_value frequency*]
**no oam retry** *up_value* [*down_value frequency*]

Where:

- *up_value* is a number from 1 to 600 that represents the OAM retry count before declaring that a VC is up.

  The default is 3 retries.
- *down_value* is a number from 1 to 600 that represents the OAM retry count before declaring a VC is down.

  The default is 5 retries.
- *frequency* is a number from 1 to 1000 that represents the OAM retry polling frequency, in seconds.

  The default is 1 second.

Use the **no** form of this command to return OAM retry to its default values.

The following example shows how to configure OAM retry for an ATM PVC:

```
Router(config)# interface atm 5/0/0
Router(config-if)# pvc 0/105
Router(config-if-atm-vc)# oam retry 10 10 5
```

## Enabling OAM Loopback Cell Generation and Management

To enable end-to-end F5 OAM loopback cell generation and OAM management for an ATM PVC or VC class, use the **oam-pvc** command in PVC configuration mode or VC-class configuration mode.

**oam-pvc** [**manage**] [*frequency*]
**no oam-pvc** [**manage**] [*frequency*]

Where:

- **manage** is an optional keyword that brings down the interface or subinterface if the PVC loopback fails.

- *frequency* (optional) is the number of seconds between transmitting OAM loopback cells. Values range from 0 to 600 seconds.

The default value is 10 seconds.

Use the **no** form of this command to disable generation of OAM loopback cells and OAM management.

The following example shows how to enable OAM loopback cell generation for an ATM PVC:

```
Router(config)# interface atm 5/0/0
Router(config-if)# pvc 0/105
Router(config-if-atm-vc)# oam-pvc 300
```

# Useful show Commands

You can use the following **show** commands to view ATM interfaces and subinterfaces, and to troubleshoot ATM problems.

### show atm vc

Use the **show atm vc** command to display information about the VCs on the interface.

```
Router# show atm vc
               VCD /                                     Peak Avg/Min Burst
Interface      Name         VPI   VCI  Type   Encaps     Kbps   Kbps  Cells Sts
2/0/0          1              0    16   PVC    ILMI149760UP
2/0/0          9              0   100   PVC    MUX149760UP
2/0/0.2        7              2    32   PVC    SNAP149760UP
2/0/0          8              2    33   PVC    SNAP149760UP
2/0/0          18             2   100   PVC    SNAP149760UP
2/0/0.2        6              4    24   PVC    SNAP149760UP
2/0/0          2             25     3   PVC    F4-OAM 50000UP
2/0/0          3             25     4   PVC    F4-OAM 50000UP
2/0/0          14            25   100   PVC    SNAP 50000 50000    0UP
2/0/0          16            25   101   PVC    SNAP 50000 50000    0UP
2/0/0          17            25   102   PVC    SNAP 50000 50000    0UP
2/0/0          10            26     3   PVC    F4-OAM 50000UP
2/0/0          11            26     4   PVC    F4-OAM50000UP
2/0/0          12            27     3   PVC    F4-OAM 50000UP
2/0/0          13            27     4   PVC    F4-OAM 50000UP
2/0/0          19            33   100   PVC    SNAP 10000 8000   10UP
Router#
```

### show atm vp

Use the **show atm vp** command to display information about the VPs on the interface.

```
Router# show atm vp
                    Data  CES    Peak    CES
Interface     VPI    VCs   VCs    Kbps    Kbps      Status
ATM2/0/0      25     3     0      50000   0         ACTIVE
ATM2/0/0      26     0     0      50000   0         ACTIVE
ATM2/0/0      27     0     0      50000   0         ACTIVE
Router#
```

### show atm pvc

Use the **show atm pvc** *vpi_number/vci_number* command to display detailed information about a specific PVC.

```
Router# show atm pvc 0/100
```

```
ATM2/0/0: VCD: 9, VPI: 0, VCI: 100
UBR, PeakRate: 149760
AAL5-MUX, etype:0x800, Flags: 0xC23, VCmode: 0x0
OAM frequency: 0 second(s), OAM retry frequency: 1 second(s)
OAM up retry count: 3, OAM down retry count: 5
OAM Loopback status: OAM Disabled
OAM VC state: Not Managed
ILMI VC state: Not Managed
InARP DISABLED
InPkts: 0, OutPkts: 0, InBytes: 0, OutBytes: 0
InPRoc: 0, OutPRoc: 0, Broadcasts: 0
InFast: 0, OutFast: 0, InAS: 0, OutAS: 0
InPktDrops: 0, OutPktDrops: 0
Out CLP=1 Pkts: 0
OAM cells received: 0
F5 InEndloop: 0, F5 InSegloop: 0, F5 InAIS: 0, F5 InRDI: 0
OAM cells sent: 0
F5 OutEndloop: 0, F5 OutSegloop: 0, F5 OutRDI: 0
OAM cell drops: 0
PVC Discovery: NOT_VERIFIED
Status: UP
Router#
```

CHAPTER 3

# 1-Port OC-12 ATM Line Card Configuration

This chapter describes procedures for configuring the Cisco 10000 series 1-Port OC-12 ATM line card, hereafter known as the 1-Port OC-12 ATM line card.

The 1-Port OC-12 ATM line card is a trunk uplink for the Cisco 10000series routers that provides IP packet routing over ATM virtual circuit connections using a single-mode fiber intermediate reach SC connector.

This chapter contains the following sections:



## Software Support

Table 3-1 shows the minimum Cisco IOS release on each release train that supports the 1-Port OC-12 ATM line card.

*Table 3-1 1-Port OC-12 ATM Line Card Software Support*

| Required PRE | Minimum Cisco IOS Releases |
|---|---|
| PRE1 | Cisco IOS Release 12.0(22)S and later releases of Cisco IOS Release 12.0S |
| PRE2 | Cisco IOS Release 12.0(10)SL and later releases of Cisco IOS Release 12.0SL<br>Cisco IOS Release 12.0(22)S and later releases of Cisco IOS Release 12.0S<br>Cisco IOS Release 12.3(7)XI and later releases of Cisco IOS 12.3XI<br>Cisco IOS Release 12.2(28)SB and later releases of Cisco IOS 12.2SB |
| PRE3 | Cisco IOS Release 12.2(31)SB2 and later releases of Cisco IOS 12.2SB |

## Checking Hardware and Software Compatibility

The PRE installed in the Cisco 10000 series router chassis must support the Cisco IOS software running on the router. Use the **show version** command to check the PRE version installed.

To see if a feature is supported by a Cisco IOS release, to locate the software document for that feature, or to check the minimum software requirements of Cisco IOS software with the hardware installed on your router, Cisco maintains the Software Advisor tool on Cisco.com at
http://www.cisco.com/cgi-bin/Support/CompNav/Index.pl

This tool does not verify whether line cards within a system are compatible, but does provide the minimum Cisco IOS requirements for individual hardware line cards, modules, or options.

You must be a registered user on Cisco.com to access this tool.

# Default Values

Table 3-2 lists default configuration values for the 1-Port OC-12 ATM line card. This table also includes the command used to modify a default value, and provides information about values to set on the remote end of the connection.

*Table 3-2 1-Port OC-12 ATM Line Card Defaults*

| Command Name | Default Setting | Command Syntax | Remote Side Setting |
|---|---|---|---|
| mtu (maximum transmission unit) | 9180 bytes | **[no] mtu** *bytes* | same |
| atm clock internal | no atm clock internal | **[no] atm clock internal** | opposite |
| loopback | loopback none | **[no] loopback [line \| diagnostic {parallel \| path \| serial}]** | — |
| pvc encapsulation | aal5snap | **encapsulation [aal5snap \| aal5mux ip]** | same |

# Line Card VC Limitations

The Cisco 10000 series router supports four ATM service categories for virtual circuits (VCs):

- Constant Bit Rate (CBR)
- Variable Bit Rate-non-real-time (VBR-nrt)
- Unspecified Bit Rate (UBR) with a peak cell rate (PCR), referred to as shaped UBR
- UBR without a PCR, referred to as unshaped UBR

The segmentation and reassembly (SAR) mechanism configures priority and additional traffic management parameters for the various ATM service categories. Table 3-3 lists the priority levels the SAR sets for the service categories.

*Table 3-3 ATM Service Categories*

| Parameter | CBR | VBR-rt | VBR-nrt | Shaped UBR | Unshaped UBR |
|---|---|---|---|---|---|
| Priority | 0 | 1 | 2 | 3 | None |

The number of SAR priority levels and the service categories supported at each priority level vary from line card to line card. For example, the 1-port OC-12/STM-1 line card supports the four levels of priority and the service categories listed in Table 3-4.

The ATM line cards support a maximum number of VCs per priority. That VC limit depends on the VC limit of the SAR (SAR limit) and the number of priority levels configured. Table 3-4 describes how to determine the VC limit per priority level for the 1-port OC-12/STM-1 line card.

*Table 3-4 Maximum Number of VCs per Priority*

| ATM Line Card | SAR Priority Levels | VC Rate | Maximum Number of VCs per Priority |
|---|---|---|---|
| 1-Port OC-12/ STM-1 | 0 = CBR VCs<br>1 = VBR-rt VCs<br>2 = VBR-nrt VCs<br>3 = UBR VCs | Full line rate | SAR limit / 2 / number of priority levels<br>With 4 priority system:<br>65,536 / 2 / 4 = 8192 VCs per priority level |
| | | Half line rate and below | SAR limit / number of priority levels<br>4 priority system:<br>65,536 / 4 = 16,384 VCs per priority level |

Configuring more channels or VCs than there are available priority locations can cause random channels or VCs to get stuck in the SAR. This occurs when an active channel tries to reschedule itself, but no priority locations are available. Therefore, the channel cannot find a place to reschedule itself, which results in a lost event for the channel, and the channel becomes stuck in the SAR.

On the PRE2, when a VC becomes stuck in the SAR, the PRE2 scheduler stops forwarding traffic on only the VC that is stuck in the SAR; the other VCs still carry traffic. On the PRE3, the PRE3 scheduler stops forwarding traffic on all the VCs configured on that ATM line card.

# Shaped UBRs on the OC-12 ATM Line Card

On an OC-12 ATM line card, when you configure UBR PVCs with a shaped value (UBR-PCR) and the shaped value is greater than one-half of the line rate (for example, 299,520 Kbps), the following limitations apply:

- The number of VCs the OC-12 line card supports is up to one-half of the VC scaling limit of 16,384 VCs. Cisco IOS software counts each UBR-PVC above 299,520 as two VCs. Therefore, the active VC count must be maintained at the following:

  16,384 > (number of VCs at 299,520 and above * 2) + (number of VCs below 299,520)

  At any time, if more VCs are active than the allowed number above, the SAR on the line card leaks buffers, which results in a reduced buffer pool for active VCs and the SAR might fail if enough buffers are lost. To recover the lost buffers, reboot the system.
- The router allows you to enter shaping values between 299,520 and 599,040, which the SAR does not support. The SAR performs shaping in the range of 599,040 and 299,520 to 299,538. If you configure a shaping value between 299,528 and 399,032, the shape rate the SAR returns is unclear.
- If you initially set a shaping rate of 599,040 and then change to another rate, or you initially configure a shape rate and change to a rate of 599,040, the router accepts the command and the **show** commands display the new rate. However, the SAR does not perform shaping correctly until the next reload.

If you change the shaping rate from 599,040 to a lower rate, the LP shaper in VTMS allows the average rate to be met. However, during traffic bursts, ATM-level shaping is not accurate.

# Interface Syntax

To specify an interface number in a configuration command, use the syntax in Table 3-5 to identify interfaces on the 1-Port OC-12 ATM line card.

*Table 3-5 1-Port OC-12 ATM Interface Syntax*

| Type of Interface | Slot | Subslot | Port | Subinterface |
|---|---|---|---|---|
| Main interface | 1 to 8/ | 0/ | 0. | — |
| Subinterface | 1 to 8/ | 0/ | 0. | 1 to 4294967295 |

Examples:

- Modify a PVC associated with the main interface.

```
Router(config)# interface atm 2/0/0
Router(config-if)# pvc 0/200
Router(config-if-atm-vc)#
```

- Modify a permanent virtual circuit (PVC) associated with a subinterface.

```
Router(config)# interface atm 7/0/0.1
Router(config-subif)# pvc 0/101
Router(config-if-atm-vc)#
```

# Interface Configuration Samples

This section provides sample procedures for creating subinterfaces, permanent virtual circuits (PVCs), virtual circuit (VC) classes, and for enabling Integrated Local Management Interface (ILMI).

## Creating a Subinterface

Use the following procedure to create a subinterface.

**Step 1** Divide the ATM interface into subinterfaces using the **interface** command. You can create either a point-to-point or multipoint subinterface.

In the following example, multipoint subinterface number 1 is created on an 1-Port OC-12 ATM line card in slot 2.

```
Router(config)# interface atm 2/0/0.1 multipoint
Router(config-subif)#
```

**Step 2** Assign an IP address and subnet mask to the interface using the **ip address** configuration subcommand, as shown in the following example:

```
Router(config-subif)# ip address 172.27.48.209 255.255.0.0
Router(config-subif)#
```

You have created interface 2/0/0.1. To enter this interface, use the following command:

```
Router(config)# interface atm 2/0/0.1
Router(config-subif)#
```

## Creating a PVC

You can create up multiple PVCs on the 1-Port OC-12 ATM line card interface. You can create PVCs on the main interface or on a subinterface.

To create a PVC

**Step 1** Enter interface or subinterface configuration mode.

Use the **pvc** command to specify a virtual path identifier (VPI) value between 0 and 255 and a virtual channel identifier (VCI) value between 0 and 65535. The following example creates a PVC with a VPI value of 0 and VCI value of 100 on a subinterface.

```
Router(config-subif)# pvc 0/100
Router(config-if-atm-vc)#
```

**Step 2** Assign a peer IP address to the PVC using the **protocol ip** configuration subcommand, as in the following example:

```
Router(config-if-atm-vc)# protocol ip 172.16.32.49
Router(config-if-atm-vc)#
```

## Creating a VC Class

This procedure demonstrates how to create an ATM VC class. An ATM VC class is a PVC boilerplate—a PVC description that you can apply to one or more PVCs.

**Step 1** To create a PVC boilerplate, use the global configuration mode **vc-class atm** command. The following example creates the ATM VC class named *boston*.

```
Router(config)# vc-class atm boston
Router(config-vc-class)#
```

**Step 2** Enter commands to describe the ATM VC class named *boston*. This example shows how to specify that the class uses AAL5+MUX encapsulation and configure a variable bit rate-nonreal time (VBR-NRT) PVC.

```
Router(config-vc-class)# encaps aal5mux ip
Router(config-vc-class)# vbr-nrt 30000 20000 128
Router(config-vc-class)# exit
Router(config)#
```

You have created a VC class named *boston*. The next procedure describes how to apply this class to a PVC or subinterface.

## Applying a VC Class

You can apply a VC class (created in the previous procedure) to a PVC or an interface.

- In the following example, the class named *boston* is applied to subinterface 5/0/0.1.

```
Router(config)# interface atm 5/0/0.1
Router(config-subif)# class-int boston
Router(config-subif)#
```

- In the following example, the class named *boston* is applied to a new PVC (0/102) in subinterface 5/0/0.2.

```
Router(config)# interface atm 5/0/0.2
Router(config-subif)# pvc 0/102
Router(config-if-atm-vc)# class-vc boston
Router(config-if-atm-vc)#
```

You have completed the steps for assigning a VC class to a PVC.

## Enabling ILMI PVC Discovery

You can enable ILMI to automatically discover PVCs on neighboring switches and duplicate those PVC entries on the 1-Port OC-12 ATM line card.

To enable ILMI

---

**Step 1** Create PVC 0/16 on the main interface, as shown in the following example:

```
Router(config)# interface atm 5/0/0
Router(config-if)# pvc 0/16 ilmi
Router(config-if-atm-vc)# exit
Router(config-if)#
```

**Step 2** In the following example, ILMI PVC discovery is enabled for the 1-Port OC-12 ATM line card.

```
Router(config-if)# atm ilmi-pvc-discovery
```

You have completed the steps required for running ILMI PVC discovery. You can use the **show atm pvc** command to display the PVCs on the Cisco 10000 series router.

---

## Completing a Configuration

This section offers general information on completing a configuration of an 1-Port OC-12 ATM line card.

---

**Step 1** After you configure the interfaces for ATM, you may need to enter interface configuration mode and specify routing protocols, network addresses, and so on.

**Step 2** After you have included all of the configuration subcommands to complete the configuration, enter **Ctrl-Z** (hold down the **Control** key while you press **Z**) to exit configuration mode.

**Step 3** Write the new configuration to memory:

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

The system displays an OK message when the configuration is stored. After you have completed your configuration, you can check it by using **show** commands.

# Commands

The 1-Port OC-12 ATM line card consists of a single OC-12 interface. You can put all the PVCs on this interface, or you can create subinterfaces. This section describes the principal commands for customizing interfaces and PVCs:



## Global Configuration Command

A PVC boilerplate is a PVC description that you can apply to one or more PVCs or interfaces

To create a PVC boilerplate, use the global configuration mode **vc-class atm** command.

**vc-class atm** *class_name*

Where *class_name* is any word that describes the class.

After you create the class, you enter VC class configuration mode. In this mode, you describe the action you want the class to take by entering commands and arguments. These commands and arguments are described in the "ATM PVC Commands" section on page 3-15.

In the following example, an ATM VC class named *cambridge* is created and defined. This example shows how to specify that the class uses AAL5+MUX encapsulation and a VBR-NRT PVC.

```
Router(config)# vc-class atm cambridge
Router(config-vc-class)# encaps aal5mux ip
Router(config-vc-class)# vbr-nrt 30000 20000 128
Router(config-vc-class)# exit
Router(config)#
```

For information on applying a VC class name, see the "Attaching an ATM VC Class to an Interface" section on page 3-9 and the "Attaching an ATM VC Class to a PVC" section on page 3-17.

## Interface and Subinterface Commands

This section describes principal commands for configuring ATM interfaces and subinterfaces. This section describes the following global configuration commands:

## Creating and Entering Subinterfaces

Use the **interface** command to divide the 1-Port OC-12 ATM line card interface into multiple subinterfaces to simplify management of the card, create interfaces with different MTU sizes, and create connections to different networks.

**`interface atm`** *`slot/subslot/port.subinterface type`*
**`no interface atm`** *`slot/subslot/port.subinterface type`*

Where:

- *slot* is 1 to 8
- *subslot* and *port* are both 0
- *subinterface* is a number from 1 to 4,294,967,295
- *type* is **point-to-point** or **multipoint**

To enter the subinterface at a later time, you do not need to specify the type.

To remove a subinterface and its PVCs, use the **no interface** command. To change a subinterface type, you must first remove the subinterface.

Examples:

- Create subinterface number 1 for an 1-Port OC-12 ATM line card in slot 1.

  ```
  Router(config)# interface atm 1/0/0.1 point-to-point
  Router(config-subif)#
  ```

- Enter an existing subinterface.

  ```
  Router(config)# interface atm 1/0/0.1
  Router(config-subif)#
  ```

- Use a multipoint subinterface.

  If you want multiple PVCs to go to the same network, you must create a multipoint subinterface. For example:

  ```
  Router(config)# interface atm 4/0/0.2 multipoint
  Router(config-subif)#
  ```

  After creating the subinterface, you can create PVCs that go to the same network. Figure 3-1 shows a multipoint subinterface on a fully meshed network. Fully meshed indicates that any workstation can communicate with any other workstation.

*Figure 3-1 Multipoint ATM Configuration*

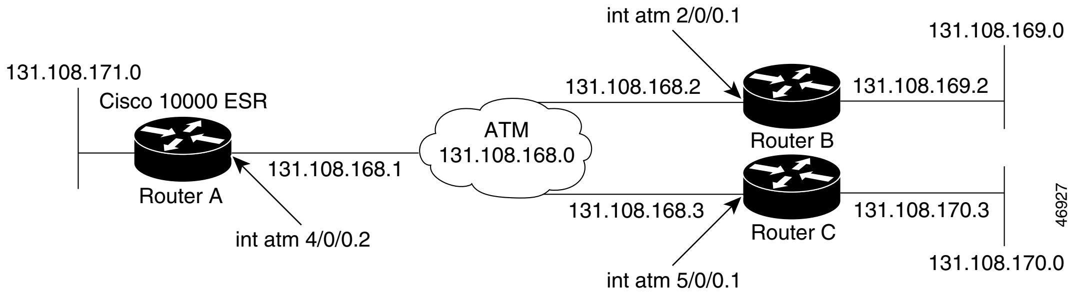



Table 3-6 provides the configuration commands used to configure the multipoint connections in the network shown in Figure 3-1.

*Table 3-6 Multipoint Configuration Example*

| Router A | Router B | Router C |
|---|---|---|
| `interface atm 4/0/0.2 multi`<br>`ip address 131.108.168.1 255.255.255.0`<br>`pvc 0/35`<br>`protocol ip 131.108.168.2 broadcast`<br>`pvc 0/36`<br>`protocol ip 131.108.168.3 broadcast` | `interface atm 2/0/0.1 multi`<br>`ip address 131.108.168.2 255.255.255.0`<br>`pvc 0/35`<br>`protocol ip 131.108.168.1 broadcast`<br>`pvc 0/37`<br>`protocol ip 131.108.168.3 broadcast` | `interface atm 5/0/0.1 multi`<br>`ip address 131.108.168.3 255.255.255.0`<br>`pvc 0/36`<br>`protocol ip 131.108.168.1 broadcast`<br>`pvc 0/37`<br>`protocol ip 131.108.168.2 broadcast` |

## Attaching an ATM VC Class to an Interface

To attach an ATM VC class to an interface, use the **class-int** command. If you customize a PVC, its customization takes precedence over the interface class.

**`class-int`** *`class_name`*

Where *class_name* is the name of the class created using the global configuration **class-vc atm** command.

In the following example, a VC class named *cambridge* is created and attached to subinterface 3/0/0.1.

```
Router(config)# vc-class atm cambridge
Router(config-vc-class)# encaps aal5mux ip
Router(config-vc-class)# vbr-nrt 30000 20000 128
Router(config-vc-class)# exit
Router(config)# interface atm 3/0/0.1
Router(config-subif)# class-int cambridge
```

## Creating ATM PVPs

To create a permanent virtual path (PVP) used to multiplex one or more VBR-NRT VCs, use the **atm pvp** interface configuration command.

**`atm pvp`** *`vpi peak-rate`* `[`**`no-f4-oam`**`]`
**`no atm pvp`** *`vpi`*

Where:

- **vpi** is the ATM network virtual path identifier (VPI) of the VC used to multiplex the permanent virtual path. The range is 0 to 255. You must use a VPI value that is not already in use (by a VC).
- *peak-rate* is the maximum rate in Kbps at which the PVP can transmit data. You can enter values from 84 Kbps to 299,520 Kbps and you can also enter 599,040 Kbps. The PVP peak rate value supersedes all cell rate values set for VBR-NRT PVCs.
- **no-f4-oam** (optional) restricts the PVP from passing operations/administration/maintenance (OAM) packets. When you create a PVP, the system creates PVCs with VCI values of 3 and 4 for each PVP, which pass OAM packets.

To verify the configuration of a PVP, use the **show atm vp** EXEC command.

The following example shows how to create a PVP with a peak rate of 50000 Kbps. The subsequent VCs created are multiplexed onto this virtual path.

```
Router(config)# interface atm 7/0/0
Router(config-if)# atm pvp 25 50000
Router(config-if)# pvc 25/100
Router(config-if-atm-vc)# vbr-nrt 10000 5000 16
Router(config-if-atm-vc)# exit
Router(config-if)# pvc 25/101
Router(config-if-atm-vc)# vbr-nrt 10000 5000 16
Router(config-if-atm-vc)# exit
Router(config-if)# pvc 25/102
Router(config-if-atm-vc)# vbr-nrt 10000 5000 16
Router(config-if-atm-vc)# exit
```

## Creating a PVC

You can create a unspecified bit rate (UBR) PVC by entering the **pvc** command:

**pvc** [*word*] [*vpi_value*/]{*vci_value*} [**ilmi**]

Where:

- *vpi_value* is in the range from 0 to 255. If you do not specify a VPI value, the system assigns the value of 0.
- *vci_value* is in the range from 1 to 65535. The VCI value should be at least 33, because all of the lower-numbered PVCs are generally used for specific purposes.
- *word* is a name referring to this connection.
- the **ilmi** parameter maps the ILMI channel to the PVC for this interface. You can only use this argument for PVCs created on the main interface. It is highly recommended that you use this argument with PVC 0/16. For more information about activating ILMI, see the section "Activating ATM ILMI PVC Discovery" section on page 3-11.

By default, the **pvc** command creates a UBR PVC; to create a VBR-NRT PVC, see the "Configuring VBR-NRT" section on page 3-17.

**Note** You can only create one PVC on a point-to-point interface. Multiple PVCs can be created on a multipoint interface.

Examples:

- Create PVC 0/105 on the main interface.

```
Router(config)# interface atm 5/0/0
Router(config-if)# pvc 105
Router(config-if-atm-vc)
```

- Create PVC 2/102 on a subinterface.

```
Router(config)# interface atm 5/0/0.1
Router(config-subif)# pvc 2/102
Router(config-if-atm-vc)#
```

## Enabling ATM ILMI

Use the **atm ilmi-enable** interface configuration command to enable the ILMI on a port.

```
atm ilmi-enable
no atm ilmi-enable
```

The default is ILMI is enabled, but you should disable the ILMI if the peer does not support ILMI. In order for peers to exchange ILMI information, you must create PVC 0/16, using the **ilmi** argument.

The following example disables ILMI:

```
Router(config)# interface atm 5/0/0
Router(config-if)# no atm ilmi-enable
```

## Activating ATM ILMI PVC Discovery

To activate ATM PVC discovery, use the **atm ilmi-pvc-discovery** interface configuration command. This command causes ILMI-compliant devices to propagate PVCs.

```
atm ilmi-pvc-discovery [subinterface]
no atm ilmi-pvc-discovery [subinterface]
```

Where:

- **subinterface** instructs the software to associate all PVCs with existing subinterface numbers equal to their VPI numbers. For example: PVC 2/102 would be listed under subinterface 7/0/0.2, PVC 12/156 would be listed under 7/0/0.12, and so on.

**Note** The **subinterface** argument associates PVCs only with subinterfaces that have already been created. If there is no subinterface for a VPI value, the system associates the PVC with the main interface.

The following example shows how to enable PVC Discovery on the ATM main interface 7/0/0 on an 1-Port OC-12 ATM line card that has ILMI enabled.

```
Router(config)# interface atm 7/0/0
Router(config-if)# pvc 0/16 ilmi
Router(config-if-atm-vc)# exit
Router(config-if)# atm ilmi-pvc-discovery
```

## Specifying the ATM ILMI Keepalive Rate

To specify the ILMI keepalive rate, enter the **atm ilmi-keepalive** command.

**atm ilmi-keepalive [***seconds* **[retry** *counts***]]**

Where:

- *seconds* is a value from 1 to 65535
- **retry** *counts* is a value from 2 to 5

The default value for seconds is 5000 and for retry counts is 4.

The following example shows how to enable ILMI keepalives for the ATM interface 5/0/0:

```
Router(config)# interface atm 5/0/0
Router(config-if)# atm ilmi-keepalive 10000 retry 3
```

## Configuring ATM Clock

To configure the clock source as internal, use the **atm clock internal** command:

**atm clock internal**
**no atm clock internal**

The default clock setting is **no atm clock internal**, which means that clocking is derived from the line.

In the following example, clocking is set from the router.

```
Router(config)# interface atm 5/0/0
Router(config-if)# atm clock internal
```

## Specifying the ATM Flag

To specify the ATM flag value for the s1s0 bit, use the **atm flag s1s0** command. This command is typically used to meet a standards requirement or to ensure interoperability with another vendor's equipment.

**atm flag s1s0** *value*

Where:

- **s1s0** is part of the payload pointer byte
- *value* is from 0 to 3

The default s1s0 value is 0.

The following example assigns a value of 2 to the ATM flag:

```
Router(config)# interface atm 5/0/0
Router(config-if)# atm flag s1s0 2
```

## Controlling ATM Alarm Reporting

To control selected SONET alarms so that they are logged to the console for an ATM interface, use the **atm report** interface configuration command.

**atm report** {**b1-tca** | b**2-tca** | **b3-tca** | **lais** | **lcd** | **lrdi** | **pais** | **plop** | **prdi** | **rdool** | **sd-ber** | **sf-ber** | **slof** | **slos**}

**no atm report** {**b1-tca** | b**2-tca** | **b3-tca** | **lais** | **lcd** | **lrdi** | **pais** | **plop** | **prdi** | **rdool** | **sd-ber** | **sf-ber** | **slof** | **slos**}

Where:

- **b1-tca**—B1 bit error rate (BEF) threshold crossing alarm
- **b2-tca**—B2 BER threshold crossing alarm
- **b3-tca**—B3 BER threshold crossing alarm
- **lais**—Line Alarm Indication Signal
- **lcd**—Loss of cell delineation
- **lrdi**—Line Remote Defect Indication
- **pais**—Path Alarm Indication Signal
- **plop**—Path Loss of Pointer
- **prdi**—Path Remote Defect Indication
- **rdool**—Receive Data Out Of Lock
- **sd-ber**—Line bit interleave parity error (LBIP) BER in excess of signal degrade (SD) threshold
- **sf-ber**—LBIP BER in excess of signal fail (SF) threshold
- **slof**—Section Loss of Frame
- **slos**—Section Loss of Signal

To disable logging of select SONET alarms, use the **no** form of this command.

Reporting an alarm means that the alarm can be logged to the console. Not all alarms are logged. SONET alarm hierarchy rules dictate that only the most severe alarm of an alarm group is reported. Whether an alarm is reported or not, you can view the current state of a defect by checking the Active Defects line from the **show controllers atm** command output. A defect is a problem indication that is a candidate for an alarm.

The following example shows how to enable reporting of SD-BER and LAIS alarms on the interface:

```
Router(config)# interface atm 3/0/0
Router(config-if)# atm report sd-ber
Router(config-if)# atm report lais
Router(config-if)# end
Router#
```

## Specifying the ATM Alarm Thresholds

Specify the bit error rate (BER) threshold by using the **atm threshold** command:

**atm threshold** {**b1-tca** | **b2-tca** | **b3-tca** | **sd-ber** | **sf-ber**} *value*

Where:

- **b1-tca**—B1 BER threshold crossing alarm
- **b2-tca**—B2 BER threshold crossing alarm
- **b3-tca**—B3 BER threshold crossing alarm
- **sd-ber**—Set Signal Degrade BER threshold
- **sf-ber**—Set Signal Fail BER threshold
- *value* is an exponential value from $10^{-3}$ to $10^{-9}$ representing the BER at which an alarm occurs.

The default for all thresholds, except **sf-ber**, is $10^{-6}$. The default for **sf-ber** is $10^{-3}$.

The following example shows how to specify the B1 BER threshold crossing alarm value of 4:

```
Router(config)# interface atm 5/0/0
Router(config-if)# atm threshold b1-tca 4
```

## Controlling the S1 SONET Overhead Byte

On Cisco 10000 series routers, ATM line cards run over SONET. In most situations, the default value for the S1 SONET overhead byte (0x0) does not need to be changed. Refer to the SONET standards for information about the possible values for the S1 SONET overhead byte and the definition of each value.

### Controlling a Transmitted S1 Overhead Byte

In Cisco IOS Release 12.2(28)SB, use the **pos flag s1-byte tx** command in interface configuration mode to control the transmission of the S1 SONET overhead byte.

**pos flag s1-byte tx** *value*

Where:

- *value* is in the range of 0x0 to 0xF
- 0x0 is the default value

In the following example the S1 SONET overhead byte is set to 0xF:

```
pos flag s1-byte tx 0xF
```

### Reacting to a Received S1 Overhead Byte

In Cisco IOS Release 12.2(28)SB, use the **pos flag s1-byte rx-communicate** command to direct the router to switch the clock source to internal when it receives an S1 SONET overhead byte with a value of 0xF. When the S1 SONET overhead byte changes from 0xF to any other value, the clock source reverts back to the clock source specified in the user configuration.

The S1 overhead byte is ignored by the receiving router unless the **pos flag s1-byte rx-communicate** command is issued.

**pos flag s1-byte rx-communicate**

**no pos flag s1-byte rx-communicate**

The following example directs the router to switch to internal clocking when it receives an S1 SONET overhead byte with a value of 0xF:

```
pos flag s1-byte rx-communicate
```

The default for the **pos flag s1-byte rx-communicate** command is disabled or off.

## Running Loopbacks

You can run a loopback by using the **loopback** command:

```
loopback {line | diagnostic {parallel | path | serial}}
no loopback {line | diagnostic {parallel | path | serial}}
```

Where:

- **line** is the line loopback
- **diagnostic** starts an internal diagnostic loopback
- **parallel** is the internal diagnostic parallel loopback

- **path** is the internal diagnostic path loopback
- **serial** is the internal diagnostic serial loopback

The following example shows hot to run the diagnostic serial loopback:

```
Router(config)# interface atm 5/0/0
Router(config-if)# loopback diagnostic serial
```

# ATM PVC Commands

After you create a PVC using the **pvc** command, you can customize the PVC or a VC class by using the commands described in this section.



## Specifying a Protocol

Use the **protocol ip** command in interface-ATM-VC configuration mode or VC-class configuration mode to do one or both of the following:

- Configure a static map for an ATM PVC or VC class.
- Enable Inverse ARP or Inverse ARP broadcasts on an ATM PVC by either configuring Inverse ARP directly on the PVC or in a VC class (applies to IP protocols only).

```
protocol ip {protocol-address | inarp} [[no] broadcast]
no protocol ip {protocol-address | inarp} [[no] broadcast]
```

Where:

- *protocol-address* is the peer destination address that is being mapped to a PVC.
- **inarp** (valid only for IP protocols on PVCs) enables Inverse ARP on an ATM PVC. If you specify a protocol-address instead of inarp, Inverse ARP is automatically disabled for that protocol.
- [**no**] **broadcast** (optional) indicates that this PVC sends out broadcast packets (for example, IGRP updates). Pseudo broadcasting is supported. The **broadcast** keyword of the protocol ip command takes precedence if you previously configured the broadcast command on the ATM PVC.

  For PVCs created under point-to-point subinterfaces, broadcast is enabled by default. For PVCs created under multipoint subinterfaces, you should use the **broadcast** argument if you want to propagate IP routes.

Use the **no** form of this command to remove a static map or disable Inverse ARP.

**Note** Use the **inarp** command to configure Inverse ARP frequency.

The following example shows how to specify IP protocol on an ATM PVC:

```
Router(config)# interface atm 5/0/0
Router(config-if)# pvc 0/105
Router(config-if-atm-vc)# protocol ip 172.16.32.49
```

## Configuring a Broadcast

To configure broadcast packet duplication and transmission for an ATM PVC or VC class, use the **broadcast** command.

The **broadcast** command is not used to enable ATM *cell-level* multicast, broadcast, replication, or to set up the broadcast of user level traffic. The **broadcast** command indicates which PVC (or PVCs) sends out broadcast traffic. This is typically limited to traffic associated with routing protocols and routing updates (for example, OSPF hello packets).

**Note** The **broadcast** argument within the **protocol ip** command takes precedence over the **broadcast** command. See the "Specifying a Protocol" section on page 3-15 for additional information about the **protocol ip** command.

Use the default form of this command to restore the default behavior described below.

**broadcast**
**no broadcast**

The default is broadcast.
Use the **no** form of this command to disable transmission of broadcast packets.

For PVCs created under point-to-point subinterfaces, **broadcast** is enabled by default. For PVCs created under multipoint subinterfaces, you should use the **broadcast** command if you want to propagate IP routes (only the first PVC on a multipoint interface receives broadcast traffic).

The following example shows how to use the **broadcast** command to restore the default behavior:

```
Router(config)# interface atm 5/0/0.4
Router(config-subif)# pvc 0/105
Router(config-if-atm-vc)# broadcast
Router(config-if-atm-vc)#
```

## Configuring Inverse ARP

To configure the Inverse ARP time period for an ATM PVC or VC class, use the **inarp** command in interface-ATM-VC configuration mode or VC-class configuration mode.

**inarp** *minutes*
**no inarp** *minutes*

Where *minutes* is the Inverse ARP frequency from 1 to 60 minutes.

The default frequency is 15 minutes.

Use the **no** form of this command to restore the default Inverse ARP time period behavior.

**Note** This command is supported only for AAL5+SNAP encapsulation (the default) when Inverse ARP is enabled. Use the **encapsulation** command to configure AAL5+SNAP encapsulation and the **protocol** command to enable Inverse ARP.

The following example shows how to specify an Inverse ARP frequency of 40 minutes on an ATM PVC:

```
Router(config)# interface atm 5/0/0
Router(config-if)# pvc 0/105
Router(config-if-atm-vc)# inarp 40
```

## Attaching an ATM VC Class to a PVC

To attach an ATM VC class to a PVC, use the **class-vc** command.

**class-vc** *name*

Where:

- *name* is the name of the class created with the global configuration **class-vc atm** command.

The following example shows how to assign an ATM VC class named *boston* to an ATM PVC:

```
Router(config)# interface atm 5/0/0.4
Router(config-subif)# pvc 2/100
Router(config-if-atm-vc)# class-vc boston
```

## Configuring VBR-NRT

To configure the variable bit rate-nonreal time (VBR-NRT) traffic management type and specify output peak cell rate, output sustainable cell rate, and output maximum burst cell size for an ATM PVC or VC class, use the **vbr-nrt** command. Use **vbr-nrt** in interface-ATM-VC configuration mode or VC-class configuration mode.

**vbr-nrt** *peak_cell_rate sustainable_cell_rate maximum_burst_size*
**no vbr-nrt** *peak_cell_rate sustainable_cell_rate maximum_burst_size*

Where:

- *peak* is the peak cell rate (PCR) from 84 Kbps to 299,520 Kbps and 599,040 Kbps.The PCR must be at least equal to the sustainable cell rate (SCR)
- *sustainable* is the sustainable cell rate (SCR) from 84 Kbps to 299,520 Kbps and 599,040 Kbps.
- *maximum* is a number from 1 to 256 that represents Maximum Burst Size (MBS) in cells

The default class of service is unspecified bit rate (UBR) running at the maximum line rate of the physical interface.

Use the **no** form of this command to remove the VBR-NRT parameters and return the PVC to its default of unspecified bit rate (UBR).

You can create up to 254 VBR-NRT PVCs on an 1-Port OC-12 ATM line card.

The following example shows how to configure the VBR-NRT traffic parameters on an ATM PVC:

```
Router(config)# interface atm 5/0/0
Router(config-if)# pvc 0/105
Router(config-if-atm-vc)# vbr-nrt 50000 20000 200
```

## Specifying Encapsulation

To specify the ATM adaptation layer (AAL) and encapsulation type for an ATM PVC or VC class, use the **encapsulation** command in interface-ATM-VC configuration mode or VC-class configuration mode.

**encapsulation** {**aal5mux ip** | **aal5snap**}
**no encapsulation** {**aal5mux ip** | **aal5snap**}

Where:

- **aal5mux ip** is AAL5+MUX encapsulation
- **aal5snap** is AAL5+LLC/SNAP encapsulation (the default)

Use the **no** form of this command to remove an encapsulation from a PVC or VC class.

The following example shows how to specify **aal5mux ip** encapsulation for an ATM PVC:

```
Router(config)# interface atm 5/0/0.4
Router(config-subif)# pvc 0/105
Router(config-if-atm-vc)# encaps aal5mux ip
```

## Enabling ILMI Management

To enable ILMI management on an ATM PVC, use the **ilmi manage** command in interface-ATM-VC configuration mode or VC-class configuration mode. This command changes the convergence of higher-level protocols based on link-state changes.

**ilmi manage**
**no ilmi manage**

Use the **no** form of this command to disable ILMI management.

The following example shows how to enable ILMI management on an ATM PVC:

```
Router(config)# interface atm 5/0/0
Router(config-if)# pvc 0/105
Router(config-if-atm-vc)# ilmi manage
```

## Configuring OAM Retry

To configure OAM retry, use the **oam retry** command in interface-ATM-VC configuration mode or VC-class configuration mode.

**oam retry** *up_value* [*down_value frequency*]
**no oam retry** *up_value* [*down_value frequency*]

Where:

- *up_value* is a number from 1 to 600 that represents the OAM retry count before declaring a VC is up.

  Default is 3 retries.
- *down_value* is a number from 1 to 600 that represents the OAM retry count before declaring a VC is down.

  Default is 5 retries.
- *frequency* is a number from 1 to 1000 that represents the OAM retry polling frequency, in seconds.

  Default is 1 second.

Use the **no** form of the command to remove oam retry parameters.

The following example shows how to configure OAM retry to an ATM PVC:

```
Router(config)# interface atm 5/0/0
Router(config-if)# pvc 0/105
Router(config-if-atm-vc)# oam retry 10 10 5
```

## Enabling OAM Loopback Cell Generation and Management

To enable end-to-end F5 OAM loopback cell generation and OAM management for an ATM PVC or VC class, use the **oam-pvc** command in interface-ATM-VC configuration mode or VC-class configuration mode.

**oam-pvc** [**manage**] [*frequency*]
**no oam-pvc** [**manage**] [*frequency*]

Where:

- **manage** is an optional keyword that brings down the line if the PVC loopback fails.
- *frequency* (optional) is the number of seconds between transmitting OAM loopback cells. Values range from 0 to 600 seconds.

The default value is 10 seconds.

Use the **no** form of this command to disable generation of OAM loopback cells and OAM management.

The following example enables OAM loopback cell and OAM management to an ATM PVC:

```
Router(config)# interface atm 5/0/0
Router(config-if)# pvc 0/105
Router(config-if-atm-vc)# oam-pvc 300
```

# Useful show Commands

### show atm vc

Use the **show atm vc** command to display information about the VCs on the interface.

```
Router# show atm vc
               VCD /                                      Peak Avg/Min Burst
Interface      Name         VPI   VCI  Type   Encaps     Kbps   Kbps  Cells Sts
2/0/0          1              0    16   PVC    ILMI      599040              UP
2/0/0          9              0   100   PVC    MUX       599040              UP
2/0/0.2        7              2    32   PVC    SNAP      599040              UP
2/0/0          8              2    33   PVC    SNAP      599040              UP
2/0/0          18             2   100   PVC    SNAP      599040              UP
2/0/0.2        6              4    24   PVC    SNAP      599040              UP
2/0/0          2             25     3   PVC    F4-OAM     50000              UP
2/0/0          3             25     4   PVC    F4-OAM     50000              UP
2/0/0          14            25   100   PVC    SNAP       50000  50000    0   UP
2/0/0          16            25   101   PVC    SNAP       50000  50000    0   UP
2/0/0          17            25   102   PVC    SNAP       50000  50000    0   UP
2/0/0          10            26     3   PVC    F4-OAM     50000              UP
2/0/0          11            26     4   PVC    F4-OAM     50000              UP
2/0/0          12            27     3   PVC    F4-OAM     50000              UP
2/0/0          13            27     4   PVC    F4-OAM     50000              UP
2/0/0          19            33   100   PVC    SNAP       10000   8000   10   UP
Router#
```

### show atm vp

Use the **show atm vp** command to display information about the VPs on the interface.

```
Router# show atm vp
                    Data  CES    Peak    CES
Interface    VPI    VCs   VCs    Kbps    Kbps     Status
ATM2/0/0     25     3     0      50000   0        ACTIVE
ATM2/0/0     26     0     0      50000   0        ACTIVE
ATM2/0/0     27     0     0      50000   0        ACTIVE
Router#
```

### show atm pvc

Use the **show atm pvc** *vpi_number/vci_number* command to display detailed information about a specific PVC.

```
Router# show atm pvc 0/100

ATM2/0/0: VCD: 9, VPI: 0, VCI: 100
UBR, PeakRate: 599040
AAL5-MUX, etype:0x800, Flags: 0xC23, VCmode: 0x0
OAM frequency: 0 second(s), OAM retry frequency: 1 second(s)
OAM up retry count: 3, OAM down retry count: 5
OAM Loopback status: OAM Disabled
OAM VC state: Not Managed
ILMI VC state: Not Managed
InARP DISABLED
InPkts: 0, OutPkts: 0, InBytes: 0, OutBytes: 0
InPRoc: 0, OutPRoc: 0, Broadcasts: 0
InFast: 0, OutFast: 0, InAS: 0, OutAS: 0
InPktDrops: 0, OutPktDrops: 0
Out CLP=1 Pkts: 0
OAM cells received: 0
F5 InEndloop: 0, F5 InSegloop: 0, F5 InAIS: 0, F5 InRDI: 0
OAM cells sent: 0
F5 OutEndloop: 0, F5 OutSegloop: 0, F5 OutRDI: 0
OAM cell drops: 0
PVC Discovery: NOT_VERIFIED
Status: UP
Router#
```

# PART 2

## Channelized Line Cards

CHAPTER 4

# 4-Port Channelized T3 Half-Height Line Card Configuration

This chapter describes the Cisco 10000 series 4-Port Channelized T3 Half-Height line card, hereafter known as the 4-Port Channelized T3 Half-Height line card.

The 4-Port Channelized T3 Half-Height line card provides the Cisco 10000 router with four DS3 ports of high-density T3 service (eight T3 ports per slot).

This chapter contains the following sections:



## Software Support

Table 4-1 shows the minimum Cisco IOS release on each release train that supports the 4-Port OC-3/STM-1 ATM line card.

*Table 4-1 4-Port OC-3/STM-1 ATM Line Card Software Support*

| Required PRE | Minimum Cisco IOS Releases |
|---|---|
| PRE2 | Cisco IOS Release 12.2(28)SB and later releases of Cisco IOS 12.2SB |

## Checking Hardware and Software Compatibility

The PRE installed in the Cisco 10000 series router chassis must support the Cisco IOS software running on the router. Use the **show version** command to check the PRE version installed.

To see if a feature is supported by a Cisco IOS release, to locate the software document for that feature, or to check the minimum software requirements of Cisco IOS software with the hardware installed on your router, Cisco maintains the Software Advisor tool on Cisco.com at
http://www.cisco.com/cgi-bin/Support/CompNav/Index.pl

This tool does not verify whether line cards within a system are compatible, but does provide the minimum Cisco IOS requirements for individual hardware line cards, modules, or options.

You must be a registered user on Cisco.com to access this tool.

# Feature Overview

The 4-Port Channelized T3 Half-Height line card is very similar in function to the 6-Port Channelized T3 line card. The differences are summarized in the following features list for the 4-Port Channelized T3 Half-Height line card:

- Provides the Cisco 10000 series router with four DS3 ports of high-density T3 service (eight T3 ports per slot)
- Supports full line rate transfers of packet sizes from 64 bytes to 8000 bytes
- Supports 256 serial data channels for the first three ports, and 255 serial data channels for the last port, or 1023 channels per half-height module
- The 256 (255) serial data channels per port are configurable up to:
  - 1 DS3 interface
  - 28 DS1 interfaces
  - 256 (255) N x DS0 interfaces
  - Or any combination of these interfaces that does not exceed the bandwidth of the port

  **Note** Port 3 supports only 255 channels.

- Provides high availability when two 4-Port Channelized T3 Half-Height line cards are installed and interconnected with a Y-cable in a redundant configuration
- Supports six T1 BER tests simultaneously for each set of two ports
- Adds new BER test patterns (3in24, 1in8, and 2in8) and the **show bert** command to display BER test statistics
- Adds remote loopback inband looping codes (2in5 and 3in5) for T1 SF and ESF framing

**Note** For information on installing half-height line cards in subslots, refer to the *Cisco 10000 Series Routers Line Card Hardware Installation Guide.*

# Default Values

Table 4-2 lists default values for the 4-Port Channelized T3 Half-Height line card and indicates which values apply to which line speeds. The table includes the command used for modifying a default value and indicates whether a value needs to be the same (or opposite) on the remote end of the connection.

***Table 4-2 4-Port Channelized T3 Half-Height Line Card Default Values***

| Command Name | Default Setting | Command Syntax | Remote Side Setting | DS0 | T1 | Ch T3 | Unch T3 |
|---|---|---|---|---|---|---|---|
| **Controller Configuration Mode** | | | | | | | |
| cable length | 249 | **cablelength** *feet* | — | | | x | x |
| channelized | channelized | [**no**] **channelized** | — | | | x | x |
| clock source | internal | **clock source** [**line** \| **internal**] | At least one side set to **internal** | | | x | x |
| T1 clock source | internal | **t1** *t1-number* **clock source** [**line** \| **internal**] | At least one side set to **internal** | | x | | |
| framing | auto-detect | **framing** [**c-bit** \| **m23** \| **auto-detect**] | Same | | | x | |
| T1 framing | esf | **t1** *t1-number* **framing** [**esf** \| **sf** [**hdlc-idle** {**0x7e** \| **0xff**}]] | Same | | x | | |
| idle pattern | 0x7e (flags) | **idle pattern** [**0x0 to 0xff**] | Same | | x | x | |
| **Interface Configuration Mode** | | | | | | | |
| crc (cyclical redundancy check) | 16 | [**no**] **crc** [**16** \| **32**] | Same | x | x | | x |
| dsu bandwidth | 44,210 | **dsu bandwidth** *bandwidth* | Same | | | | x |
| dsu mode | cisco | **dsu mode** *mode* | Same | | | | x |
| encapsulation | HDLC | **encapsulation** [**hdlc** \| **ppp** \| **frame-relay**] | Same | x | x | | x |
| framing | c-bit | **framing** [**c-bit** \| **m13**] | Same | | | | x |
| idle character | flags (0x7e) | **idle character** [**flags** \| **marks**] | Same | | | | x |
| keepalive | keepalive (10 sec) | **keepalive** | Same | x | x | | x |
| mtu (maximum transmission unit) | 1500 | **mtu** *size* | Same | x | x | | x |
| scramble | No scrambling | [**no**] **scramble** | Same | | | | x |

# Interface Syntax

To specify an interface number in a configuration command, use the syntax in Table 4-3 to identify interfaces on the 4-Port Channelized T3 Half-Height line card.

*Table 4-3 4-Port Channelized T3 Half-Height line card Interface Syntax*

| Type of Interface | Slot | Subslot | Port (T3 Number) | T1 Number | Channel Group Number |
|---|---|---|---|---|---|
| Unchannelized | 1 to 8/ | 0 or 1/ | 0 to 3 | — | — |
| Channelized | 1 to 8/ | 0 or 1/ | 0 to 3/ | 1 to 28 | 0 to 23 |

Examples:

- Modifying T1 interface 6 in controller configuration mode:

```
Router(config)# controller t3 2/0/0
Router(config-controller)# t1 6
```

- Modifying T1 interface 6, channel group number 8 in interface configuration mode:

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

# Interface Configuration Sample

Each T3 controller can be configured as a single T3 interface (full or subrate), as 28 T1 interfaces, or as an even larger number of fractional T1s. The following procedure walks you through the basic steps for creating full-rate and subrate T3 interfaces, as well as T1 and fractional T1 interfaces:

**Step 1** Create an interface. In the following examples, each type of interface is created in a different T3 controller (2/0/0 through 2/0/3).

**Full-Rate T3 Interface**

**a.** Enter controller configuration mode.

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

**b.** To create a full-rate T3 interface, you must eliminate the T1 interfaces by entering the **no channelized** command.

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

**c.** Go to interface configuration mode:

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

**d.** You can now continue to Step 2.

#### Subrate T3 Interface

a. Enter controller configuration mode.

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

b. To create a subrate T3 interface, first create a full-rate one.

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

c. Then go into interface configuration mode, where you can use the **dsu bandwidth** command to create a subrate T3 interface. In this example, a subrate T3 interface is created that has a bandwidth of 16,000 kbps.

```
Router(config)# interface serial 2/0/1
Router(config-if)# dsu mode digital-link
Router(config-if)# dsu bandwidth 16000
```

d. You can now continue to Step 2.

#### Full T1 Interface

a. Enter controller configuration mode.

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

b. Use the **t1 channel group** command to create a T1 interface. In the following example, T1 interface 1 (of 28) is defined as being made up of a single channel group, number 20 (any number between 0 and 23). This channel group consists of all 24 DS0 time slots.

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

c. Go to interface configuration mode for the channel group you just created.

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

d. You can now continue to Step 2.

#### Fractional T1 Interface

a. Enter controller configuration mode.

```
Router(config)# controller t3 2/0/3
Router(config-controller)#
```

b. Use the **t1 channel group** command to create fractional T1 interfaces. In the following example, T1 interface 3 (of 28) is defined as being made up of three channel groups, numbers 19, 20, and 21. (Numbers between 0 and 23 are allowed.) The channel groups consist of a total of 24 DS0 time slots. Each channel group represents a separate interface.

```
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. Go to interface configuration mode for one of the channel groups; for example:

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

d. You can now continue to Step 2.

**Step 2** Enter the encapsulation method. This example shows the command for using Frame Relay encapsulation. You can also choose PPP or HDLC.

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

**Step 3** If IP routing is enabled on the system, assign an IP address and subnet mask; for example:

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

**Step 4** Add any configuration subcommands required to enable routing protocols and set the interface line characteristics.

**Step 5** Change the shutdown state to up, which enables the interface.

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

**Step 6** When you have entered all of the configuration subcommands to complete the configuration, press **Ctrl-Z** to exit configuration mode.

**Step 7** To write the new configuration to NVRAM, type

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

After you create an interface configuration, you can modify it at any time by using the appropriate Cisco IOS configuration commands.

# Unchannelized T3 Commands

By default, a T3 interface on a 4-Port Channelized T3 Half-Height line card is channelized into 28 T1 interfaces. You must unchannelize the T3 interface in order to create a full-rate or subrate T3 interface. This section describes the commands you use to create, customize, and test full-rate and subrate T3 interfaces. This section describes the following:


- Configuring a T3 Interface as Unchannelized, page 4-7
- Specifying the DSU Mode, page 4-7
- Specifying Subrate T3 Bandwidth, page 4-7
- Setting the Framing Type, page 4-8
- Enabling Scrambling, page 4-8
- Specifying an Idle Character, page 4-8
- Configuring a BER Test, page 4-9
- Specifying the Cable Length, page 4-10
- Entering MDL Messages, page 4-10
- Setting the Clock Source, page 4-11
- Configuring Loopback Mode, page 4-12
- Running Equipment Loopbacks, page 4-12


**Note** Configuring a T3 interface as unchannelized must occur in controller configuration mode. All other configuration of an unchannelized T3 interface must occur in *interface* configuration mode.

## Configuring a T3 Interface as Unchannelized

The default value for a T3 interface is channelized. To create an unchannelized T3 interface, you must first enter controller configuration mode for the T3 controller you want to configure.

**controller T3** *slot/sub-slot/port*

After entering controller configuration mode, you can configure the T3 interface as unchannelized (clear channel) by entering the **no channelized** command. Use the **channelized** command to return the interface to its default (channelized).

**[no] channelized**

**Caution** The **no channelized** command removes all channel groups from a channelized T3 interface. If you have already configured channel groups, use this command with caution.

In the following example, an unchannelized T3 interface is created:

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

## Specifying the DSU Mode

To specify a DSU mode for a selected T3 interface, use the **dsu mode** command from interface configuration mode. This command configures the line card to emulate a manufacturer's proprietary multiplexing scheme.

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

The default DSU mode is **cisco**.

Use the **no** form of the command to return the DSU mode to its default.

In the following example, the DSU mode is set to **cisco**:

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

## Specifying Subrate T3 Bandwidth

To specify subrate T3 bandwidth, use the **dsu bandwidth** *bandwidth* command from interface configuration mode.

**[no] dsu bandwidth** *bandwidth*

Where *bandwidth* is a numeric value between 0 and 44210 kbps.

The default bandwidth is 44210 kbps.

To return to the default bandwidth, use the **no** form of this command.

When you specify a value, the software sets the bandwidth to the closest acceptable bandwidth, based on the time slot size for the current DSU mode.

To use the **dsu bandwidth** command, the remote side of the connection must also support the same DSU modes.

In the following example, a bandwidth of 16000 kbps is specified:

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

# Setting the Framing Type

To specify a framing type for the unchannelized T3 controller, use the **framing** command.

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

The default framing type is C-bit.

Use the **no** form of this command to restore the default framing type.

In the following example, framing is set to m13:

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

# Enabling Scrambling

To enable scrambling on an unchannelized T3 interface, use the **scramble** command from interface configuration mode.

**[no] scramble**

The default setting for this command is **no scramble** (scrambling disabled).

Both sides of the link should have the same scrambling setting.

In the following example, scrambling is enabled on the specified T3 interface:

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

# Specifying an Idle Character

To set a specific character on the unchannelized T3 interface to be transmitted between HDLC packets, use the **idle character** command from interface configuration mode.

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

Where:

- **flags** sets an idle character of 0x7e.
- **marks** sets an idle character of all 0xff.

The default idle character is 0x7e.

**Note** Because flags is the default, the output of the **show running-config** command does not display the flags idle character setting.

Use the **no** form of the command to return the idle character to its default.

In the following example, the idle character is set to **flags**:

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

**Note** Some systems interpret marks (or 0xff) as an abort signal. Therefore, flags (or 0x7e) is preferred.

# Configuring a BER Test

You can configure an unchannelized T3 interface to run a bit error rate (BER) test. The test can be used in checking cables and solving signal problems in the field. To send a BER test pattern on an unchannelized T3 interface, use the following interface configuration command:

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

Where:

- **errors** *number* is 1 to 255.
- **pattern** *pattern* is
  - 0s—Repetitive test pattern of all zeros (00000..)
  - 1s—Repetitive test pattern of all ones (11111..)
  - 2^15—Pseudorandom O.151 test pattern (32,768 bits long)
  - 2^20-O153—Pseudorandom O.153 test pattern (1,048,575 bits long)
  - QRSS-2^20—Pseudorandom QRSS O.151 test pattern (1,048,575 bits long)
  - 2^23—Pseudorandom O.151 test pattern (8,388,607 bits long)
  - alt-0-1—Repetitive alternating test pattern of zeros (0s) and ones (1s), for example 01010101
- **interval** *time* is 1 to 1440 minutes.

You can terminate a BER test at any time using the **no bert** command.

For more information, refer to the online *Cisco 10000 Series Internet Router Troubleshooting Guide*.

The following are example of configuring a BER test:

- Send a BER test pseudorandom pattern of 2^20 through T3 interface 1/0/0 for 5 minutes.

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

- Send a repetitive pattern of all 1s through T3 interface 1/0/0 for 1440 minutes.

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

To show BER test statistics, use the **show controllers t3** *slot*/*subslot*/*port* **bert** command in EXEC or privileged EXEC mode:

```
Router# show controllers t3 6/1/0 bert
T3 6/1/0 is up.
BERT test result (done)
Test Pattern : 2^15, Status : Not Sync, Sync Detected : 1
Interval : 5 minute(s), Time Remain : 0 minute(s)
Bit Errors (since BERT started): 0 bits,
Bits Received (since BERT started): 13025 Mbits
Bit Errors (since last sync): 0 bits
Bits Received (since last sync): 13025 Mbits
```

## Specifying the Cable Length

To specify the cable length for the T3 controller, use the **cablelength** command.

**[no] cablelength** *feet*

Where *feet* is a number from 0 to 450.

The default value is 249 feet.

Use the **no** form of this command to restore the default cable length.

In the following example, the cable length value is set to 40 feet.

```
Router(config)# interface serial 1/0/0
Router(config-if)# cablelength 40
```

**Note** This command causes the system to use one of two settings for impedance matching and pulse shaping, one setting for any cable length between 0 and 249 feet and another setting for any cable length greater than 250 feet. The exact value you enter is stored in the configuration file.

## Entering MDL Messages

You can configure maintenance data link (MDL) messages (as defined in the ANSI T1.107a-1990 specification) on the unchannelized T3 interface.

**Note** MDL messages are supported only when the T3 framing is set for C-bit parity. (See the "Setting the Framing Type" section on page 4-8.)

To configure MDL messages, use the **mdl** command.

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

Where:

- **transmit path** enables transmission of the MDL path message.
- **transmit idle-signal** enables transmission of the MDL idle signal message.
- **eic** is the equipment identification code (up to 10 characters).
- **lic** is the location identification code (up to 11 characters).
- **fic** is the frame identification code (up to 10 characters).
- **unit** is the unit identification code (up to 6 characters).
- **pfi** is the facility identification code to include in the MDL path message (up to 38 characters).
- **port** is the equipment port (which initiates the idle signal) to include in the MDL idle signal message (up to 38 characters).
- **generator** is the generator number to include in the MDL test signal message (up to 38 characters).

The default is that no MDL message is configured.

Use the **no** form of the command to remove an MDL message.

Examples of configuring MDL messages follow:

- Enable the MDL path message transmission.

```
Router(config)# interface serial 1/0/0
Router(config-if)# mdl transmit path
```

- Enable the MDL idle signal message transmission.

```
Router(config-if)# mdl transmit idle-signal
```

- Enable the MDL test signal message transmission.

```
Router(config-if)# mdl transmit test-signal
```

- Enter the equipment identification code.

```
Router(config-if)# mdl string eic router A
```

- Enter the location identification code.

```
Router(config-if)# mdl string lic test network
```

- Enter the frame identification code.

```
Router(config-if)# mdl string fic building b
```

- Enter the unit identification code.

```
Router(config-if)# mdl string unit abc
```

- Enter the facility identification code.

```
Router(config-if)# mdl string pfi string
```

- Enter the port number to send in the MDL idle signal message.

```
Router(config-if)# mdl string port string
```

- Enter the generator number to send in the MDL test signal message.

```
Router(config-if)# mdl string generator string
```

# Setting the Clock Source

At the prompt, set the internal or line clock source for the selected T3 controller using the **clock source** command. This command is set in controller configuration mode.

**clock source** {**internal** | **line**}

Where:

- **internal** specifies that the internal clock source is used.
- **line** specifies that the network clock source is used.

The default is clock source internal.

In this example, a T3 controller is instructed to use a line clock source.

```
Router(config)# interface serial 1/0/0
Router(config-if)# clock source line
```

**Note** The clock source cannot be specified as **line** on both ends of the connection.

## Configuring Loopback Mode

You can configure the T3 controller for loopback mode using the **loopback** command.

**[no] loopback** [**local** | **network** | **remote**]

Local and network loopbacks are the same.

To cancel a loopback, use the **no** form of the command.

For more information on the **loopback** command, refer to the online *Cisco 10000 Series Internet Router Troubleshooting Guide*.

Examples:

- Configure the T3 controller for local loopback.

```
Router(config)# interface serial 1/0/0
Router(config-if)# loopback local
```

- Configure the T3 controller for remote loopback.

```
Router(config)# interface serial 1/0/0
Router(config-if)# framing c-bit
Router(config-if)# loopback remote
```

**Note** Remote loopback works only when framing is set to **c-bit**.

## Running Equipment Loopbacks

Use the **equipment loopback** command to run loopbacks in conjunction with remote equipment.

**[no] equipment** [**customer** | **network**] **loopback**

Where:

- **customer** enables the line card to respond to remote T3 loopback commands from the remote T3 equipment.
- **network** causes the line card to ignore remote T3 loopback commands.

Use the **no** form of the command to terminate the loopback.

For more information on the **loopback** command, refer to the online *Cisco 10000 Series Internet Router Troubleshooting Guide*.

In the following example, an equipment network loopback is configured:

```
Router(config)# interface serial 1/0/0
Router(config-if)# equipment network loopback
```

# Channelized T3 Commands

By default, a T3 interface on a 4-Port Channelized T3 Half-Height line card is channelized into 28 T1 interfaces. This section describes the commands you use to customize and test a channelized T3 interface. This section describes:

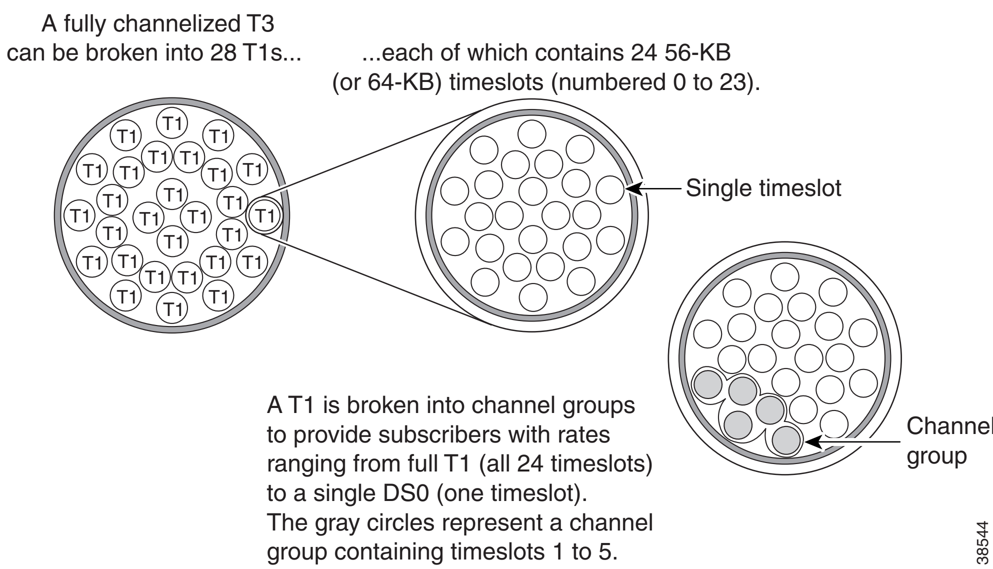
- Configuring a T3 Interface as Channelized, page 4-13
- Specifying the Cable Length, page 4-14

- Setting the Framing Type, page 4-14
- Entering MDL Messages, page 4-14
- Specifying the Idle Pattern, page 4-15
- Setting the Clock Source, page 4-16
- Configuring Loopback Mode, page 4-16
- Running Equipment Loopbacks, page 4-17


To use the channels for subscriber traffic, you must configure the T1 and DS0 components. For more information, see the "Channel-Group Command for DS0 Time Slots and T1s" section on page 4-17.

Figure 4-1 shows the levels of configurable interface bandwidth that channelization offers.

***Figure 4-1 Channelization of T3s***




# Configuring a T3 Interface as Channelized

All configuration of a T3 channelized interface must occur in controller configuration mode.

**controller T3** *slot/sub-slot/port*

A T3 interface is channelized by default. Use the **channelized** command if you had previously made the interface unchannelized and want to change the setting.

**[no] channelized**

**Caution** The **no channelized** command removes all channel groups from a channelized T3 interface. If you have already configured channel groups, use this command with caution.

The following example shows the creation of a channelized T3 interface:

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

# Specifying the Cable Length

To specify the cable length for the T3 controller, use the **cablelength** command.

**[no] cablelength** *feet*

Where *feet* is a number from 0 to 450.

The default value is 249 feet.

Use the **no** form of this command to restore the default cable length.

In the following example, the cable length value is set to 40 feet.

```
Router(config)# controller T3 1/0/0
Router(config-controller)# cablelength 40
```

**Note** This command causes the system to use one of two settings for impedance matching and pulse shaping, one setting for any cable length between 0 and 249 feet and another setting for any cable length greater than 250 feet. The exact value you enter is stored in the configuration file.

# Setting the Framing Type

To specify a framing type for the channelized T3 controller, use the **framing** command.

**[no] framing** [**c-bit** | **m23** | **f**]

The default is **auto-detect**.

Use the **no** form of the command to return the framing type to its default.

You can instruct the 4-Port Channelized T3 Half-Height line card to detect the framing type from the far end and transmit that same framing type as follows:

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

# Entering MDL Messages

You can configure maintenance data link (MDL) messages (as defined in the ANSI T1.107a-1990 specification) on the channelized T3 interface.

**Note** MDL messages are supported only when the T3 framing is set for C-bit parity. (See the "Setting the Framing Type" section on page 4-14.)

To configure MDL messages, use the **mdl** command.

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

Where:

- **transmit path** enables transmission of the MDL path message.
- **transmit idle-signal** enables transmission of the MDL idle signal message.
- **eic** is the equipment identification code (up to 10 characters).

- **lic** is the location identification code (up to 11 characters).
- **fic** is the frame identification code (up to 10 characters).
- **unit** is the unit identification code (up to 6 characters).
- **pfi** is the facility identification code to include in the MDL path message (up to 38 characters).
- **port** is the equipment port (which initiates the idle signal) to include in the MDL idle signal message (up to 38 characters).
- **generator** is the generator number to include in the MDL test signal message (up to 38 characters).

The default is that no MDL message is configured.

Use the **no** form of the command to remove an MDL message.

Examples of configuring MDL messages follow:

- Enable the MDL path message transmission.

```
Router(config)# controller T3 1/0/0
Router(config-controller)# mdl transmit path
```

- Enable the MDL idle signal message transmission.

```
Router(config-controller)# mdl transmit idle-signal
```

- Enable the MDL test signal message transmission.

```
Router(config-controller)# mdl transmit test-signal
```

- Enter the equipment identification code.

```
Router(config-controller)# mdl string eic router A
```

- Enter the location identification code.

```
Router(config-controller)# mdl string lic test network
```

- Enter the frame identification code.

```
Router(config-controller)# mdl string fic building b
```

- Enter the unit identification code.

```
Router(config-controller)# mdl string unit abc
```

- Enter the facility identification code.

```
Router(config-controller)# mdl string pfi string
```

- Enter the port number to send in the MDL idle signal message.

```
Router(config-controller)# mdl string port string
```

- Enter the generator number to send in the MDL test signal message.

```
Router(config-controller)# mdl string generator string
```

## Specifying the Idle Pattern

You can set a specific pattern to be transmitted between HDLC packets on all unconfigured time slots that belong to a channelized T3 interface. To do so, use the **idle pattern** command.

**[no] idle pattern** *patterns*

Where *patterns* is a number in the range of 0x0 to 0xff (hexadecimal) or 0 to 255 (decimal). You can enter this value in either hexadecimal or decimal form. Values of 0 to 254 set the idle pattern to HDLC flags (0x7e); a value of 255 sets the pattern to 0xff (all ones).

**Note** Some systems interpret marks (or 0xf) as an abort signal. Therefore, flags (or 0x7e) is preferred.

The default idle pattern is 0x7e.

Use the **no** form of the command to return the idle pattern to its default value.

Examples:

- Set a hexadecimal idle pattern.

```
Router(config)# controller T3 1/0/0
Router(config-controller)# idle pattern 0x10
```

- Set a decimal idle pattern.

```
Router(config)# controller T3 1/0/0
Router(config-controller)# idle pattern 23
```

# Setting the Clock Source

At the prompt, set the internal or line clock source for the selected T3 controller using the **clock source** command. This command is set in controller configuration mode.

**clock source** {**internal** | **line**}

Where:

- **internal** specifies that the internal clock source is used
- **line** specifies that the network clock source is used

The default is clock source internal.

In this example, a T3 controller is instructed to use a line clock source.

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

**Note** The clock source cannot be specified as **line** on both ends of the connection.

# Configuring Loopback Mode

You can configure the T3 controller for loopback mode using the **loopback** command.

**[no] loopback** [**local** | **network** | **remote**]

Local and network loopbacks are the same.

To cancel a loopback, use the **no** form of the command.

**Note** When you configure loopback under a channelized T3 controller, all the T1 interfaces are configured to loopback mode.

For more information on the **loopback** command, refer to the online *Cisco 10000 Series Internet Router Troubleshooting Guide*.

Examples:

- Configure the T3 controller for local loopback.

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

- Configure the T3 controller for remote loopback.

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

**Note** Remote loopback works only when framing is set to **c-bit**.

## Running Equipment Loopbacks

Use the **equipment loopback** command to run loopbacks in conjunction with remote equipment.

**[no] equipment [customer | network] loopback**

Where:

- **customer** enables the line card to respond to remote T3 loopback commands from the remote T3 equipment.
- **network** causes the line card to ignore remote T3 loopback commands.

Use the **no** form of the command to terminate the loopback.

For more information on the **loopback** command, refer to the online *Cisco 10000 Series Internet Router Troubleshooting Guide*.

In the following example, an equipment network loopback is configured:

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

# Channel-Group Command for DS0 Time Slots and T1s

Each channelized T3 consists of 28 T1s, each of which is divided into 24 time slots (or DS0s). To create an interface that supports subscriber traffic, you must combine the time slots into channel groups. Use the **t1 channel-group** command to create a channel group.

Channel groups have these characteristics:

- A channel group is an interface.
- A channel group can be one time slot to 24 time slots in size.
  - A channel group that consists of 24 time slots is a T1 interface.
  - A channel group that consists of fewer than 24 time slots can be described as a fractional T1 interface.
- Each group of 24 time slots can be divided into multiple fractional T1 interfaces.
- A channel group cannot be part of more than one T1.

Figure 4-1 illustrates channelization possibilities on the 4-Port Channelized T3 Half-Height line card.

You create a logical channel group using the following command:

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

Where:

- *t1-number* is T1 interface number 1 to 28.
- **channel-group** *channel-group-number* identifies the channel group with any number from 0 to 23.
- **timeslots** *list-of-timeslots* can be 1 to 24 or a combination of subranges within 1 to 24. You can indicate a range using a hyphen, commas, or a combination of both. One time slot equals one DS0. Refer to the examples below.
- **speed** {**56 | 64**} is an optional argument that specifies the speed of a time slot as either 56 or 64 kbps. The default is 64. (The 56-kbps time slots are generally used with older T1 equipment that does not support B8ZS and are associated with SF framing.)

Use the **no** form of the command to remove a logical channel group.

The following examples show how to use the **t1 channel-group** command:

- In this example, T1 interface 3 includes channel group 20 and consists of nine channelized time slots:

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

  To enter interface configuration mode for this channel group, enter the following:

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

- In the following example, T1 interface 4 includes channel group 18 and consists of all 24 time slots, creating a full T1 interface:

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

  To enter interface configuration mode for this channel group, you enter the following:

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

- In the following example, T1 interface number 5 is divided into three channel groups, which total 11 time slots:

```
Router(config)# controller t3 1/0/0
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
```

- In the following example, channel group 20 is removed:

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

# Channelized T1 Commands

From controller configuration mode for a T3 interface, you can enter commands to modify aspects of a T1 interface. This section describes the following:


- Setting the Framing Format, page 4-19
- Controlling Yellow Alarms, page 4-19
- Setting the Clock Source, page 4-20
- Configuring FDL, page 4-20
- Configuring a BER Test, page 4-21
- Configuring T1 Loopback Mode, page 4-22


After you configure a T1 interface, you can add encapsulation, routing, and other instructions by entering interface configuration mode; for example:

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

## Setting the Framing Format

You can specify the T1 interface framing format using the following command:

**[no] t1** *t1-number* **framing** {**esf** | **sf** [**hdlc-idle** {**0x7e** | **0xff**}]}

Where:

- *t1-number* is T1 interface number 1 to 28.
- **framing** is either extended super frame (ESF) or super frame (SF). You can set SF hdlc-idle to 0x7e or 0xff.
- **hdlc-idle** options allow you to set the idle pattern for the T1 interface to either 0x7e (the default) or 0xff.

The default framing format is extended super frame (ESF).

Use the **no** form of the command to return framing to its default value.

The following are examples of setting the framing format:

- Set SF framing format for T1 interface 6.

```
Router(config)# controller T3 1/0/0
Router(config-controller)# t1 6 framing sf
```

- Set ESF framing format for T1 interface 16.

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

## Controlling Yellow Alarms

Use the **t1 yellow** command to turn the detection or generation of a yellow alarm on or off.

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

Where:

- *t1-number* is T1 interface 1 to 28.

- **detection** means that the interface is told it is failing by the remote device, causing Cisco IOS to send a message to the console.
- **generation** means that the interface notifies the remote device if it is failing, causing Cisco IOS to send a message to the console.

When you select SF framing for a full T1 interface (24 time slots) that uses the default speed of 64, consider using the **no** *t1-number* **yellow detection** command to turn off yellow alarm detection, because the yellow alarm can be incorrectly detected with SF framing.

In the following example, T1 interface 1 is set to yellow detection:

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

# Setting the Clock Source

You can set the internal or line (network) clock source for a T1 interface using the **controller** command.

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

Where:

- *t1-number* is T1 interface 1 to 28.
- **internal** specifies that the internal clock source is used.
- **line** specifies that the network clock source is used.

The default is clock source internal.

In the following example, the interface is instructed to get its clock source from the line:

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

# Configuring FDL

You can enable 1-second transmissions of performance reports through the facility data link (FDL) according to the ANSI T1.403 specification, on both ends of the T1 connection. To do so, use the following command:

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

Where *t1-number* is T1 interface 1 to 28.

Use the **no** form of the command to disable this feature.

**Note** You can use this command only when the T1 framing is ESF.

In the following example, FDL is enabled:

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

## Configuring a BER Test

You can configure an individual T1 interface to run an independent BER test. The test is used to check cables and solve signal problems in the field. To send a BER test pattern on a T1 interface, use the following command in controller configuration mode:

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

Where:

- *t1-number* is T1 interface number 1 to 28.
- *time* is 1 to 14,400 minutes.
- *pattern* is
  - 0s—Repetitive test pattern of all zeros (00000..)
  - 1s—Repetitive test pattern of all ones (11111..)
  - 2^11—Pseudorandom test pattern (2048 bits long)
  - 2^15—Pseudorandom O.151 test pattern (32,768 bits long)
  - 2^20-O153—Pseudorandom O.153 test pattern (1,048,575 bits long)
  - 2^20-QRSS—Pseudorandom QRSS O.151 test pattern (1,048,575 bits long)
  - 2^23—Pseudorandom O.151 test pattern (8,388,607 bits long)
  - alt-0-1—Repetitive alternating test pattern of zeros (0s) and ones (1s), for example 01010101
  - 3in24—Fixed test pattern of F0100 0100 0000 0000 0000 0100 (3 bits are 1s in 24 bits)
  - 1in8—Fixed pattern of F0100 0000 (1 bit is a 1 in 8 bits)
  - 2in8—Fixed pattern of F0100 0010 (2 bits are 1s in 8 bits)
- **unframed** causes the BER test pattern to use the entire T1 bandwidth, including the T1 framing and payload bits. If **unframed** is omitted, the T1 is either SF or ESF framed as configured by the T1 framing command, and the BER test pattern occupies only the T1 payload bits.

**Note** For each T3, you can run only one BER test at a time.

You can terminate a BER test at any time using the **no** form of the command.

For more information, refer to the online *Cisco 10000 Series Internet Router Troubleshooting Guide*.

The following are examples of configuring a BER test:

- Send a BER test pseudorandom pattern of 2^20 through T1 interface 10 for 5 minutes.

```
Router(config)# controller T3 1/0/0
Router(config-controller)# t1 10 bert pattern 2^20 interval 5 unframed
```

- Send a repetitive pattern of all 1s through T1 interface 10 for 14,400 minutes (240 hours).

```
Router(config)# controller T3 1/0/0
Router(config-controller)# t1 10 bert pattern 1s interval 14400 unframed
```

To show BER test statistics, use the **show controllers t3** *slot/subslot/port t1-number* **bert** command in EXEC or privileged EXEC mode:

```
Router# show controllers t3 6/1/1/1 bert
T3 6/1/1/1 is up. Hardware is C10K Half Height CT3 line card
T1 1
```

```
BERT test result (running)
Test Pattern : 2^15, Status : Sync, Sync Detected : 1
Interval : 5 minute(s), Time Remain : 5 minute(s)
Bit Errors (since BERT started): 0 bits,
Bits Received (since BERT started): 36 Mbits
Bit Errors (since last sync): 0 bits
Bits Received (since last sync): 36 Mbits
```

The 4-Port Channelized T3 Half-Height line card supports any combination of six T1 BER tests simultaneously for each set of two ports. That is, six T1 BER tests for ports 0 and 1, and six T1 BER tests for ports 2 and 3. Table 4-4 shows a sample BER test distribution for the four ports of the 4-Port Channelized T3 Half-Height line card.

***Table 4-4 Sample BER Test Distribution***

| Port Number | BER Tests |
|---|---|
| Port 0 | 6 T1 BER tests |
| Port 1 | 1 T3 BER test |
| Port 2 | 3 T1 BER tests |
| Port 3 | 3 T1 BER tests |

# Configuring T1 Loopback Mode

If problems occur when you configure a T1 interface, you can troubleshoot the line card by using the following command from controller configuration mode:

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

Where:

- *t1-number* is T1 interface 1 to 28.
- **local** loops the router output data back toward the router at the T1 framer and sends an alarm indication signal (AIS) out toward the network.
- **network** {**line** | **payload**} loops the data back toward the network and automatically sets a local loopback at the HDLC controllers (line) or loops the payload data back toward the network and automatically sets a local loopback at the HDLC controller (payload).
- **remote line fdl** {**ansi** | **bellcore**} sends a repeating, 16-bit ESF data link keyword to the remote end, requesting that it enter into a network line loopback. You can specify an ANSI or Bellcore keyword.

**Note** Loopback codes are defined in ANSI T1.404-1989 Table 3 (Assigned Bit-Oriented ESF Data-Link Messages).

The 4-Port Channelized T3 Half-Height line card sends the following code for remote line FDL ANSI loopback:

- Line Loopback Active Code: 0 000111 01111111
- Line Loopback Deactivate Code: 0 010010 01111111

The 4-Port Channelized T3 Half-Height line card sends the following code for remote line FDL Bellcore (SmartJack) loopback:

- Network Use (loopback active) Code: 0 001001 01111111
- Network Use (loopback deactivate) Code: 0 010010 01111111

- **remote line inband** [**maintenance**] sends a repeating inband pattern to the remote end, requesting entry into a network line loopback. The inband loopback request overwrites all data in the T1 with the loop request pattern. The remote end responds to this code only after it receives the pattern continuously for at least five seconds.

  The **maintenance** keyword is new in 12.2(28)SB. The **maintenance** keyword sends loopback codes to an interface on the remote end of the connection. This sets loopback on the remote end interface. (Note that the 4-Port Channelized T3 Half-Height line card does not *respond* to inband maintenance codes.)

  - Repeating Inband Loopback Activate Code (SF framing): 00001 loop up code
  - Repeating Inband Loopback Deactivate Code (SF framing): 001 loop down code
  - Repeating Inband Loopback Activate Code (SF and ESF framing): 2 in 5 loop up code
  - Repeating Inband Loopback Deactivate Code (SF and ESF framing): 3 in 5 loop down code

- **remote payload** [**fdl** | **ansi**] sends a repeating, 16-bit ESF data link code word to the remote end, requesting entry into a network payload loopback. Using **fdl** and **ansi** enables the remote payload facility data link (FDL) ANSI bit loopback on the T1 channel. Loopback codes are defined in ANSI T1.404-1989 Table 3 (Assigned Bit-Oriented ESF Data-Link Messages).

**Note** Due to hardware constraints, when you execute the **t1 loopback remote payload command,** the type of loopback that is implemented is line loopback.

The 4-Port Channelized T3 Half-Height line card sends the following code for remote payload FDL ANSI loopback:

- Payload Loopback Active Code: 0 001010 011111111
- Payload Loopback Deactivate Code: 0 011001 011111111

Use the **no** form of the command to terminate a loopback.

For more information on this command, refer to the online *Cisco 10000 Series Internet Router Troubleshooting Guide*.

The following are examples of configuring loopback mode:

- Configure the T3 controller for local loopback on T1 interface 1.

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

- Configure the T3 controller for remote FDL ANSI loopback on T1 interface 1.

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

# High Availability Using Line Card Redundancy

You can configure one-to-one line card redundancy using the optional Y-cables and configuring 4-Port Channelized T3 Half-Height line cards for redundancy. This redundancy provides automatic failover protection at the line card level for the 4-Port Channelized T3 Half-Height line card. Because the redundancy is at the line card level, all ports are switched over during a failover.

The redundant line cards occupy the two subslots within the same chassis line card slot. There is one Y-cable for receive and one for transmit for each of the four ports for the two redundant line cards. For information about installing redundant 4-Port Channelized T3 Half-Height line cards and the Y-cables, see the *Cisco 10000 Series Router Line Card Hardware Installation Guide*.

This section contains the following topics:



## Prerequisites for Line Card Redundancy

Before configuring line card redundancy, the Y-cables must be installed; before deconfiguring redundancy the Y-cables must be removed.

## Restrictions for Line Card Redundancy

The following restrictions apply to line card redundancy:

- Port-level redundancy is not supported.
- Redundant cards must occupy the two subslots within the same physical line card slot.
- The line card that will act as the primary line card must be the first line card configured, and it must occupy subslot 1.

## Configuring Line Card Redundancy

After verifying that the Y-cables are installed, you are ready to configure one-to-one redundancy between two 4-Port Channelized T3 Half-Height line cards. To configure line card redundancy, enter the following commands beginning in global configuration mode:

| | Command | Purpose |
|---|---|---|
| **Step 1** | `Router(config)# redundancy` | Enters redundancy mode. |
| **Step 2** | `Router(config-red)# linecard-group linecard-groupId y-cable` | Enters line card redundancy group mode and creates a line card redundancy group.<br>*linecard-groupId*—An unsigned integer in the range 0 to the (maximum number of chassis line card subslots/2) –1. For the 4-Port Channelized T3 Half-Height line card, the range is 0 to 7.<br>**y-cable**—Specifies Y-cable as the link protection type for the line card group. |
| **Step 3** | `Router(config-red-lc)# description description string` | (Optional) Adds a description for the line card group. |
| **Step 4** | `Router(config-red-lc)# member subslot slot/subslot primary` | Configures the redundancy role of the primary (active) line card.<br>The line card that will act as the primary line card must be the first line card configured, and it must occupy subslot 1.<br>**Note** Configuring redundancy on the primary line card interrupts traffic and applies the default configuration to the line card. |
| **Step 5** | `Router(config-red-lc)# member subslot slot/subslot secondary` | Configures the redundancy role of the secondary (standby) line card.<br>The line card that will act as the secondary line card must be the second line card configured, and it must occupy subslot 0.<br>**Note** Configuring redundancy on the secondary line card resets the line card, interrupts traffic, and applies the default configuration to the line card. |

Example 4-1 configures two 4-Port Channelized T3 Half-Height line cards that are installed in line card slot 2 for one-to-one redundancy.

***Example 4-1 Configuring Line Card Redundancy***

```
Router(config)# redundancy
Router(config-red)# linecard-group 1 y-cable
Router(config-red-lc)# member subslot 2/1 primary
Router(config-red-lc)# member subslot 2/0 secondary
```

# Verifying and Monitoring Line Card Redundancy

To verify line card redundancy and to display information about a line card or a line card group, use the **show redundancy linecard** command. The following table describes the type of information provided by some of the command options.

| Command | Description |
|---|---|
| **show redundancy linecard all** | Displays the redundancy state of all line card slots |
| **show redundancy linecard subslot** | Displays the redundancy state of the line card in a subslot |
| **show redundancy linecard group** | Displays the redundancy state of line cards in a line card group |
| **show redundancy linecard history** | Displays recent activity in line card redundancy |
| **show redundancy linecard sub-block** | Displays troubleshooting information |

The output examples in this section use the following configuration:

```
!
redundancy
 linecard-group 0 y-cable
  member subslot 8/1 primary
  member subslot 8/0 secondary
 mode sso
!
controller T3-RED 8/1/0
controller T3-RED 8/1/1
controller T3-RED 8/1/2
controller T3-RED 8/1/3
!
```

Example 4-2 shows output from the **show redundancy linecard all** command. It shows the redundancy state of all line card slots. The line card group is indicated as "None" if the slot is empty, or if it contains a line card that has not been configured for redundancy.

For a redundant line card, the mode can be primary or secondary and is established when redundancy is configured. The mode of a line card does not change. For the 4-Port Channelized T3 Half-Height line card, the line card in subslot 0 is always secondary, and the line card in subslot 1 is always primary.

The line card role indicates which line card in a redundant configuration is sending and receiving traffic. The line card with the active role is sending and receiving traffic. The other line card in a redundant configuration has the role of standby. Unlike the line card mode, the role of a line card can change.

In Example 4-2 you can see there has been a switchover at some point, because the line card in slot 8, subslot 0 has the active role, even though its mode is secondary.

***Example 4-2 show redundancy linecard all Command Output***

```
Router# show redundancy linecard all
              LC     My          Peer        Peer   Peer
Slot  Subslot  Group  State       State       Slot   Subslot  Role      Mode
-----------------------------------------------------------------------------
 1    0        None   Init        -           None   None     None      None
 1    1        None   Init        -           None   None     None      None
 2    0        None   Init        -           None   None     None      None
 2    1        None   Init        -           None   None     None      None
 3    0        None   Init        -           None   None     None      None
 3    1        None   Init        -           None   None     None      None
 4    0        None   Init        -           None   None     None      None
 4    1        None   Init        -           None   None     None      None
 5    0        None   Init        -           None   None     None      None
 5    1        None   Init        -           None   None     None      None
 6    0        None   Init        -           None   None     None      None
 6    1        None   Init        -           None   None     None      None
```

```
 7   0        None   Init       -          None   None     None     None
 7   1        None   Init       -          None   None     None     None
 8   0        0      Active     Stdby Hot  8      1        Active   Secondary
 8   1        0      Stdby Hot  -          8      0        Standby  Primary
```

Example 4-3 shows output from the **show redundancy linecard subslot** command. It displays information for a particular line card.

#### *Example 4-3 show redundancy linecard subslot Command Output*

```
Router# show redundancy linecard subslot 8/1
 Redundant LC Group Number: 0
 LC Slot: 8
 LC Subslot: 1
 Redundant LC Name: 8/1
 Redundant LC Mode: Primary
 Redundant LC Role: Standby
 Redundant LC My State: Stdby Hot
 Redundant LC Peer State: -

Router# show redundancy linecard subslot 8/0
 Redundant LC Group Number: 0
 LC Slot: 8
 LC Subslot: 0
 Redundant LC Name: 8/0
 Redundant LC Mode: Secondary
 Redundant LC Role: Active
 Redundant LC My State: Active
 Redundant LC Peer State: Stdby Hot
```

Example 4-4 shows output from the **show redundancy linecard group** command. It shows line card information for a specific line card group, which in this example is group 0. Note that the **show redundancy linecard group** command displays only the static line card configuration information. For example, it shows the line card mode (primary or secondary) but not the line card role, which changes in a switchover. The line card in slot 8, subslot 1, is primary even though it currently has the standby role.

Because there is only one line card group in the sample configuration, the **show redundancy linecard group all** command would display the same output as shown in Example 4-4.

#### *Example 4-4 show redundancy linecard group Command Output*

```
Router# show redundancy linecard group 0
Group Identifier: 0
NON-revertive
Group Redundancy Type: Y_CABLE
Group Redundancy Class: 1:1
Group Redundancy Configuration Type: LINECARD GROUP
Primary: Slot 8
         Subslot 1
Secondary: Slot 8
           Subslot 0
```

The **show redundancy linecard history** command and the **show redundancy linecard sub-block** command provide information that can be helpful to Cisco technical support personnel when they are assisting you in troubleshooting a redundancy problem. The output in Example 4-5 was collected after a line card switchover.

**Note** Example 4-5 and Example 4-6 identify the slot/subslot by an integer called a slot unit. The slot unit is a platform-defined number that uniquely identifies the slot/subslot as an integer. For example, a line card in slot 1, subslot 0 is identified as slot unit 2, and a line card in slot 1, subslot 1, is slot unit 3. The line card in slot 8, subslot 0, is slot unit 16; the line card in slot 8, subslot 1, is slot unit 17.

### *Example 4-5 show redundancy linecard history Command Output*

```
Router# show redundancy linecard history
Aug 29 2005 06:03:16 46400671 - Slot(17): MY State Change, (Stdby Wait) -> (Stdby Hot)
Aug 29 2005 06:03:16 46400671 - Slot(16): PEER State Change, (Stdby Wait) -> (Stdby Hot)
Aug 29 2005 06:03:16 46400671 - Slot(16): PEER FSM Execution , Active:Stdby Wait:State
Ntfy
Aug 29 2005 06:03:16 46400671 - Slot(17): MY State Change, (Stdby LC Feat Sync) -> (Stdby
Wait)
Aug 29 2005 06:03:16 46400671 - Slot(16): PEER State Change, (Stdby LC Feat Sync) ->
(Stdby Wait)
Aug 29 2005 06:03:16 46400671 - Slot(16): PEER FSM Execution , Active:Stdby LC Feat
Sync:Feat Sync Done
Aug 29 2005 06:03:16 46400671 - Slot(17): MY State Change, (Stdby FIB Dnld) -> (Stdby LC
Feat Sync)
Aug 29 2005 06:03:16 46400671 - Slot(16): PEER State Change, (Stdby FIB Dnld) -> (Stdby LC
Feat Sync)
Aug 29 2005 06:03:16 46400671 - Slot(16): PEER FSM Execution , Active:Stdby FIB Dnld:Feat
Sync Req
Aug 29 2005 06:03:16 46400671 - Slot(17): MY State Change, (Stdby LC Cfg Dnld) -> (Stdby
FIB Dnld)
Aug 29 2005 06:03:16 46400671 - Slot(16): PEER State Change, (Stdby LC Cfg Dnld) -> (Stdby
FIB Dnld)
Aug 29 2005 06:03:16 46400671 - Slot(16): PEER FSM Execution , Active:Stdby LC Cfg
Dnld:Cfg Dnld Done
Aug 29 2005 06:03:16 46400671 - Slot(17): MY State Change, (Stdby Cold) -> (Stdby LC Cfg
Dnld)
Aug 29 2005 06:03:16 46400671 - Slot(16): PEER State Change, (Stdby Cold) -> (Stdby LC Cfg
Dnld)
Aug 29 2005 06:03:16 46400644 - Slot(16): PEER FSM Execution , Active:Stdby Cold:Cfg Dnld
Aug 29 2005 06:03:16 46400644 - Slot(17): MY State Change, (Init) -> (Stdby Cold)
Aug 29 2005 06:03:16 46400644 - Slot(16): PEER State Change, (Init) -> (Stdby Cold)
Aug 29 2005 06:03:16 46400644 - Slot(16): PEER FSM Execution , Active:Init:Up
Aug 23 2005 21:18:03 49333 - Slot(16:-1): Checkpoint State Update, (Active:Init)
Aug 23 2005 21:18:03 49333 - Slot(17:-1): Checkpoint State Update, (Init:Init)
Aug 23 2005 21:18:03 49332 - Slot(16:-1): Checkpoint State Update, (Active Cold:Init)
Aug 23 2005 21:18:03 49332 - Slot(17:16): Checkpoint State Update, (Active:Active Cold)
Aug 23 2005 21:11:07 7881 - Slot(17:-1): Checkpoint State Update, (Active:Stdby Hot)
Aug 23 2005 21:11:07 7881 - Slot(16:-1): Checkpoint State Update, (Stdby Hot:Init)
Aug 23 2005 21:10:42 6393 - Slot(16): MY FSM execution, Init:Init:Reset
```

### *Example 4-6 show redundancy linecard sub-block Command Output*

```
Router# show redundancy linecard sub-block all
SlotIndex  Port     H-IDB M-HW A-HW  Type     Role     Flag
-----------------------------------------------------------
 16         0         5     21   13    Physical None       0x15
 16         1         7     22   15    Physical None       0x15
 16         2         9     23   17    Physical None       0x15
 16         3         11    24   19    Physical None       0x15
 17         0         21    13   5     Virtual  -          0x3
 17         0         13    21   5     Physical None       0x5
 17         1         22    15   7     Virtual  -          0x3
```

```
17          1          15    22    7     Physical None       0x5
17          2          23    17    9     Virtual  -          0x3
17          2          17    23    9     Physical None       0x5
17          3          24    19    11    Virtual  -          0x3
17          3          19    24    11    Physical None       0x5
```

# Performing a Manual Line Card Switchover

To initiate a manual switchover of the active 4-Port Channelized T3 Half-Height line card to the standby 4-Port Channelized T3 Half-Height line card, enter the following command in privileged Exec mode:

| Command | Purpose |
|---|---|
| Router# **redundancy linecard-group switchover from subslot** *slot/subslot* | Forces the switchover of the active line card to the standby line card.<br><br>Performing a switchover does not change the existing line card configuration, and traffic is not interrupted. |

Example 4-7 shows how you can force the switchover of the active line card in slot 2, subslot 1, to the standby line card in slot 2, subslot 0.

***Example 4-7 Manual Line Card Switchover***

```
Router# redundancy linecard-group switchover from subslot 2/1
```

# Removing Line Card Redundancy

After verifying that the Y-cables are removed, you are ready to remove one-to-one redundancy between two 4-Port Channelized T3 Half-Height line cards. To remove line card redundancy, enter the following commands beginning in global configuration mode:

| | Command | Purpose |
|---|---|---|
| **Step 1** | Router(config)# **redundancy** | Enters redundancy mode. |
| **Step 2** | Router(config-red)# **linecard-group** *linecard-groupId* **y-cable** | Enters line card redundancy group configuration mode for the specified line card redundancy group.<br><br>*linecard-groupId*—An unsigned integer in the range 0 to the (maximum number of chassis line card subslots/2) –1. For the 4-Port Channelized T3 Half-Height line card, the range is 0 to 7.<br><br>**y-cable**—Specifies Y-cable as the link protection type for the line card group. |

| | Command | Purpose |
|---|---|---|
| **Step 3** | `Router(config-red-lc)#` **no member subslot** *slot/subslot* **secondary** | Removes the redundancy role of the secondary (standby) line card.<br>You must remove redundancy from the secondary line card before removing redundancy from the primary line card.<br>**Note** Removing redundancy resets the line card (traffic is interrupted) and applies the default configuration to the line card. If you remove redundancy from the secondary line card only, traffic is not interrupted through the primary line card. |
| **Step 4** | `Router(config-red-lc)#` **no member subslot** *slot/subslot* **primary** | (Optional) Removes the redundancy role of the primary (active) line card.<br>You must remove redundancy from the secondary line card before removing redundancy from the primary line card.<br>**Note** Removing redundancy resets the line card (traffic is interrupted) and applies the default configuration to the line card. If you remove redundancy from the secondary line card only, traffic is not interrupted through the primary line card. |
| **Step 5** | `Router(config-red-lc)#` **exit** | Returns to redundancy mode. |
| **Step 6** | `Router(config-red)#` **no linecard-group** *linecard-groupId* **y-cable** | Removes the line card redundancy group. |

Example 4-8 removes one-to-one redundancy between two 4-Port Channelized T3 Half-Height line cards that are installed in slot 2 for one-to-one redundancy.

***Example 4-8 Removing Line Card Redundancy***

```
Router(config)# redundancy
Router(config-red)# linecard-group 1 y-cable
Router(config-red-lc)# no member subslot 2/0 secondary
Router(config-red-lc)# no member subslot 2/1 primary
Router(config-red-lc)# exit
Router(config-red)# no linecard-group 1 y-cable
```

# Failover Conditions

The following describes the conditions under which the primary line card will fail over to the secondary line card when line card redundancy is configured.

- User-initiated switchover—You have performed a manual switchover using the **redundancy linecard-group switchover from subslot** command.
- Line card-initiated switchover—The line card detects a failover condition and sends a cutover interrupt to the performance routing engine (PRE).
- PRE-initiated OIR switchover—The PRE detects the removal of the primary line card.
- PRE-initiated keepalive timeout switchover—More than 20 seconds have elapsed since the PRE received a keepalive message from the line card.

# Command Reference

The following commands are new in Cisco IOS Release 12.2(28)SB:

- **show controllers t3 bert**
- **linecard-group y-cable**
- **member subslot**

## show controllers t3 bert Command

To display BER test statistics, use the **show controllers t3 bert** command in EXEC or privileged EXEC mode.

**show controllers t3** {*slot/subslot/port*} [*/t1-number*] **bert**

### Syntax Description

| | |
|---|---|
| *slot* | Chassis line card slot number. |
| *subslot* | Chassis line card subslot number. |
| *port* | Interface number on the line card. |
| *t1-number* | (Optional) Logical T1 interface number. |
| **bert** | Displays BER test statistics. |

### Command Modes

EXEC

Privileged EXEC

### Command History

| Release | Modification |
|---|---|
| 12.2(28)SB | This command was introduced on the Cisco 10000 series routers. |

### Examples

The following example shows BER test statistics for an unchannelized T3 interface:

```
Router# show controllers t3 6/1/0 bert
T3 6/1/0 is up.
BERT test result (done)
Test Pattern : 2^15, Status : Not Sync, Sync Detected : 1
Interval : 5 minute(s), Time Remain : 0 minute(s)
Bit Errors (since BERT started): 0 bits,
Bits Received (since BERT started): 13025 Mbits
Bit Errors (since last sync): 0 bits
Bits Received (since last sync): 13025 Mbits
```

The following example shows BER test statistics for a channelized T3 interface:

```
Router# show controllers t3 6/1/0 bert
T3 6/1/0 is up.
BERT test result (running)
Test Pattern : 2^15, Status : Sync, Sync Detected : 1
Interval : 3 minute(s), Time Remain : 1 minute(s)
Bit Errors (since BERT started): 0 bits,
```

```
Bits Received (since BERT started): 5493 Mbits
Bit Errors (since last sync): 0 bits
Bits Received (since last sync): 5493 Mbits
```

The following example shows BER test statistics for a T1 interface:

```
Router# show controllers t3 6/1/1/1 bert
T3 6/1/1/1 is up. Hardware is C10K Half Height CT3 line card
T1 1
BERT test result (running)
Test Pattern : 2^15, Status : Sync, Sync Detected : 1
Interval : 5 minute(s), Time Remain : 5 minute(s)
Bit Errors (since BERT started): 0 bits,
Bits Received (since BERT started): 36 Mbits
Bit Errors (since last sync): 0 bits
Bits Received (since last sync): 36 Mbits
```

### Related Commands

| Command | Description |
|---|---|
| **bert** | Configures a BER test for an unchannelized or channelized T3 interface. |
| **t1 bert pattern** | Configures a BER test for a T1 interface. |

# linecard-group y-cable Command

To create a line card group for one-to-one line card redundancy, use the **linecard-group y-cable** command in redundancy mode. To remove the line card redundancy group, use the **no** form of this command.

**linecard-group** *linecard-groupId* **y-cable**

**no linecard-group** *linecard-groupId* **y-cable**

### Syntax Description

| | |
|---|---|
| *linecard-groupId* | An unsigned integer in the range 0 to the (maximum number of chassis line card subslots/2) –1. |
| **y-cable** | The link protection type for the line card group. |

### Defaults

No default behavior or values.

### Command Modes

Redundancy

### Command History

| Release | Modification |
|---|---|
| 12.2(28)SB | This command was introduced on the Cisco 10000 series router. |

**Usage Guidelines**

The **no linecard-group y-cable** command removes the line card redundancy group and frees the *linecard-groupId* for reuse. The **no linecard-group y-cable** command succeeds only if there are no subslot members in the line card redundancy group.

**Examples**

The following example creates line card group number 1 for one-to-one line card redundancy:

```
Router(config)# redundancy
Router(config-red)# linecard-group 1 y-cable
```

**Related Commands**

| Command | Description |
|---|---|
| **member subslot** | Configures the redundancy role of a line card in the line card group. |
| **redundancy** | Enters redundancy mode. |
| **show redundancy linecard** | Displays information about a redundant line card or line card group. |

# member subslot Command

To configure the redundancy role of a line card, use the **member subslot** command in line card redundancy group mode. To remove the redundancy role of a line card, use the **no** form of this command.

**member subslot** *slot/subslot* {**primary** | **secondary**}

**no member subslot** *slot/subslot* {**primary** | **secondary**}

**Syntax Description**

| | |
|---|---|
| *slot* | Chassis line card slot number. |
| *subslot* | Chassis line card subslot number. |
| **primary** \| **secondary** | Configures the redundancy role of the line card.<br>• **primary**—Active line card.<br>• **secondary**—Standby line card. |

**Defaults**

No default behavior or values

**Command Modes**

Line card redundancy group

**Command History**

| Release | Modification |
|---|---|
| 12.2(28)SB | This command was introduced on the Cisco 10000 series routers. |

## Usage Guidelines

The primary line card must be the first line card configured and must occupy subslot 1. The secondary line card must be the second line card configured and must occupy subslot 0. Only one primary line card and one secondary line card can be configured.

## Examples

The following creates line card group number 1 for one-to-one line card redundancy. It also specifies the line card in subslot 1 as the primary (active) line card, and the line card in subslot 0 as the secondary (standby) line card:

```
Router(config)# redundancy
Router(config-red)# linecard-group 1 y-cable
Router(config-red-lc)# member subslot 2/1 primary
Router(config-red-lc)# member subslot 2/0 secondary
```

## Related Commands

| Command | Description |
|---|---|
| **linecard-group y-cable** | Creates a line card group for one-to-one line card redundancy. |
| **redundancy** | Enters redundancy mode. |
| **show redundancy linecard** | Displays information about a redundant line card or line card group. |

CHAPTER 5

# 24-Port Channelized E1/T1 Line Card Configuration

This chapter describes the procedures for configuring the Cisco 10000 series 24-Port Channelized E1/T1 line card, hereafter known as the 24-Port Channelized E1/T1 line card.

The 24-Port Channelized E1/T1 line card provides Cisco 10000 routers with 24 copper channelized or unchannelized interface ports that you can configure as E1 or T1 interfaces.

This chapter contains the following sections:



## Software Support

Table 5-1 shows the minimum Cisco IOS release on each release train that supports the 24-Port Channelized E1/T1 line card.

*Table 5-1 24-Port Channelized E1/T1 Line Card Software Support*

| Required PRE | Minimum Cisco IOS Releases |
|---|---|
| PRE1 | Cisco IOS Release 12.0(22)S and later releases of Cisco IOS Release 12.0S |
| PRE2 | Cisco IOS Release 12.2(15)BX and later releases of Cisco IOS Release 12.2BX<br>Cisco IOS Release 12.3(7)XI and later releases of Cisco IOS 12.3XI<br>Cisco IOS Release 12.2(28)SB and later releases of Cisco IOS 12.2SB |

## Checking Hardware and Software Compatibility

The PRE installed in the Cisco 10000 series router chassis must support the Cisco IOS software running on the router. Use the **show version** command to check the PRE version installed.

To see if a feature is supported by a Cisco IOS release, to locate the software document for that feature, or to check the minimum software requirements of Cisco IOS software with the hardware installed on your router, Cisco maintains the Software Advisor tool on Cisco.com at http://www.cisco.com/cgi-bin/Support/CompNav/Index.pl

This tool does not verify whether line cards within a system are compatible, but does provide the minimum Cisco IOS requirements for individual hardware line cards, modules, or options.

You must be a registered user on Cisco.com to access this tool.

# Commands and Default Values

You can operate all the line card ports to accept either E1 or T1 lines, but not both. Once you configure a line card in E1 or T1 mode, you configure all ports on that line card as the same type controller. The commands for configuring either E1 or T1 interfaces are described in the following sections.

## E1 Commands

Table 5-2 lists line card commands, default values, and specifies the command syntax you use. This table also includes the commands to use to modify a default value, and indicates whether a similar value (on the remote end of the connection) must be the same (or opposite).

*Table 5-2 E1 Line Card Commands and Default Values*

| Command Name | Default Value | Command Syntax | Remote Device Setting |
|---|---|---|---|
| **Enable Commands** | | | |
| hw-module (reset) | — | **hw-module slot** *slot* **reset** \| **subslot** *slot*/*subslot* **reset** | — |
| show controllers | — | **show controllers e1** *slot*/*subslot*/*port* | — |
| show interfaces | — | **show interface serial** *slot*/*subslot*/*port*:*channel-group* {**accounting** \| **controller** \| **crb** \| **description** \| **fair-queue** \| **irb** \| **mac-accounting** \| **mpls-exp** \| **precedence** \| **random-detect** \| **rate-limit** \| **shape**} | — |
| **Configure Commands** | | | |
| card | mode e1 | [**no**] **card** *slot*/*subslot* **24che1t1-1 mode** {**e1** \| **t1**} | — |
| hw-module (shutdown) | no hw-module | [**no**] **hw-module** {**slot** *slot* **shutdown** \| **subslot** *slot*/*subslot* **shutdown**} | — |
| controller | — | **controller** {**e1** \| **t1**} *slot*/*subslot*/*port* | — |
| interface | — | [**no**] **interface serial** *slot*/*subslot*/*port*:*channel-group* | — |

*Table 5-2 E1 Line Card Commands and Default Values (continued)*

| Command Name | Default Value | Command Syntax | Remote Device Setting |
|---|---|---|---|
| **Controller Configuration Commands** | | | |
| bert | No bert | [**no**] **bert pattern** {**2^11** \| **2^15** \| **2^20-0153** \| **2^20-QRSS**} **interval** *minutes* | — |
| channel-group | No channel-group | [**no**] **channel-group** *channel number* {**timeslots** *timeslots* [**speed 64**] \| **unframed**} | — |
| clock source | line | [**no**] **clock source** {**internal** \| **line**} | At least one side set to internal |
| description | No description | [**no**] **description** *text* | — |
| framing | crc4 | [**no**] **framing** {**crc4** [**australia**]\| **no-crc4** [**australia**]} | — |
| linecode | hdb3 | [**no**] **linecode hdb3** | — |
| loopback | No loopback | [**no**] **loopback** {**local** \| **network** {**line** \| **payload**}} | — |
| national reserve | No national reserve (0) | [**no**] **national reserve** *int* sa4 *sa5 sa6 sa7 sa8* | — |
| shutdown | no shutdown | [**no**] **shutdown** | — |

# E1 Interface Numbering Syntax

You can use the line card in different Cisco series 10000 router chassis. To specify line card E1 interfaces, use the syntax listed in Table 5-3 for the router chassis in use.

*Table 5-3 E1 Line Card Interface Syntax*

| Router | Con troller | Slot | Subslot | Port Number | Channel Group Number |
|---|---|---|---|---|---|
| Cisco 10008 | E1 | 1 to 8 | /0 | /0 to /23 | :1 to:31 |
| Cisco 10005 | E1 | 1 to 5 | /0 | /0 to /23 | :1 to:31 |

**Note** Because this is a full-height line card, its subslot value is always 0.

Examples:

- Modify E1 controller 0 in slot 2 in controller configuration mode:

```
Router(config)# controller e1 2/0/0
Router(config-controller)# command
Router(config-controller)#
```

- Modifying E1 controller 0, channel-group number 8 in interface configuration mode:

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

You use the syntax in Table 5-3 to specify slot, subslot, port numbers, and channel-group numbers when configuring ports for E1 operation on a 24-Port Channelized E1/T1 line card.

You can configure a maximum of 744 channelized E1 interfaces on the 24 line card ports (24 ports x 31 E1 timeslots). To do this, you assign a channel-group number to the timeslots of the E1 interface you configure at each port (see the "Specifying E1 Channel Groups" section on page 5-21).

# T1 Commands

Table 5-4 lists T1 configuration default values, line card commands, and specifies the values you use. The table includes the commands you use for modifying a default value and indicates whether a similar value, on the remote end of the connection, needs to be the same (or opposite).

***Table 5-4 T1 Line Card Commands and Default Values***

| Command Name | Default Values | Command Syntax | Remote Device Setting |
|---|---|---|---|
| **Enable Commands** | | | |
| hw-module (reset) | — | **hw-module slot** *slot* **reset** \| **subslot** *slot***/***subslot* **reset** | — |
| show controllers | — | **show controllers t1** *slot***/***subslot***/***port* [**remote performance**] | — |
| show interface | — | **show interface serial** *slot***/***subslot***/***port***:***channel-group* {**accounting** \| **controller** \| **crb** \| **description** \| **fair-queue** \| **irb** \| **mac-accounting** \| **mpls-exp** \| **precedence** \| **random-detect** \| **rate-limit** \| **shape**} | — |
| **Configure Commands** | | | |
| card | mode e1 | [**no**] **card** *slot***/***subslot* **24che1t1-1** [**mode** {**e1** \| **t1**}] | — |
| hw-module (shutdown) | no hw-module | [**no**] **hw-module** {**slot** *slot* **shutdown** \| **subslot** *slot***/***subslot* **shutdown**} | — |
| controller | — | **controller** {**e1** \| **t1**} *slot***/***subslot***/***port* | — |
| interface | — | [**no**] **interface serial** *slot***/***subslot***/***port***:***channel-group* | — |
| **Controller Configuration Commands** | | | |
| bert | No bert | [**no**] **bert pattern** {**2^11** \| **2^15** \| **2^20-0153** \| **2^20-QRSS**} **interval** *minutes* | — |
| cablelength | No cablelength | [**no**] **cablelength short** {**110** \| **220** \| **330** \| **440** \| **550** \| **660**} | |
| channel-group | No channel-group | [**no**] **channel-group** *channel number* **timeslots** *timeslots* [**speed** {**56** \| **64**}] | — |

*Table 5-4 T1 Line Card Commands and Default Values (continued)*

| Command Name | Default Values | Command Syntax | Remote Device Setting |
|---|---|---|---|
| clock source | line | [no] **clock source** {**internal** \| **line**} | At least one side set to internal |
| description | No description | [no] **description** *text* | — |
| fdl | No fdl | [no] **fdl** {**ansi** \| **att**} | — |
| framing | esf | [no] **framing** {**esf** \| **sf**} | — |
| linecode | b8zs | [no] **linecode** {**ami** \| **b8zs**} | — |
| loopback | No loopback | [no] **loop** {**local** \| **network** {**line** \| **payload**} \| **remote** {**esf** {**line** \| **payload**} \| **iboc**}} | — |
| shutdown | no shutdown | [no] **shutdown** | — |
| yellow | yellow | [no] **yellow** {**generation** \| **detection**} | — |

# T1 Interface Numbering Syntax

You can use the line card in different Cisco series 10000 router chassis. To specify line card T1 interfaces, use the syntax listed in Table 5-5 for the router chassis in use.

*Table 5-5 T1 Line Card Interface Syntax*

| Router | Line Interface | Slot | Subslot | Port Number | Channel Group Number |
|---|---|---|---|---|---|
| Cisco 10008 | T1 | 1 to 8 | /0 | /0 to /23 | :1 to:24 |
| Cisco 10005 | T1 | 1 to 5 | /0 | /0 to /23 | :1 to:24 |

Examples:

- Modify T1 controller 0 on slot 2 in controller configuration mode:

```
Router(config)# controller T1 2/0/0
Router(config-controller)# command
Router(config-controller)#
```

- Modifying T1 controller 0, channel-group number 8 in interface configuration mode:

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

You use the syntax in Table 5-4 to specify slot, subslot, port numbers, and channel-group numbers when configuring a port as an T1 controller on a 24-Port Channelized E1/T1 line card.

You can configure a maximum of 576 channelized T1 interfaces on the 24 line card ports (24 ports x 24 T1 timeslots). To do this, you assign a T1 channel-group number to the timeslots of each interface you configure at each port (see the “Specifying T1 Channel Groups” section on page 5-22).

# Configuration Task Overview

You can configure the ports of the 24-Port Channelized E1/T1 line card as channelized or unframed E1 interfaces, or as channelized T1 interfaces. To configure an E1 or T1 line card interface, perform the following procedure:

**Step 1** Designate the card mode as T1 or E1.

**Step 2** Configure the controller parameters and save the configuration to NVRAM.

**Step 3** Create one or more serial interfaces at each active port.

**Step 4** Specify the interface parameters for each port channel-group and save the configuration to NVRAM.

**Note** Each time you change the card mode (from E1 to T1 or from T1 to E1), the existing *running* configuration for the card interface is erased. If you want to keep a line card configuration, you must save it before you change the card mode.

# E1 Configuration Examples

The examples in this section familiarize you with the line card configuration tasks and typify how you apply the commands listed in Table 5-2 and the syntax in Table 5-3 to configure a port as an E1 controller. These examples also show how to configure E1 channel-group bandwidth at any of the 24 available ports on the line card. The example steps are intended to quickly familiarize you with the procedures you use to set up an E1 port as an E1 controller.

**Note** You must be in privileged EXEC mode to specify and configure a port as an E1 controller on any line card port.

## Configuring an E1 Controller

From the IOS global configuration mode, you use the **config terminal** command to enter the controller configuration mode and to configure a card as E1 mode (see the "Configuring a Controller" section on page 5-16 for details).

## Configuring Full-Rate Unchannelized E1 Bandwidths

To configure the full-rate unchannelized E1 bandwidth at a port, you assign the maximum bandwidth (2048 kbps) when you include **unframed** on the command line in controller configuration mode. The following example shows the steps you use to create full-rate unchannelized E1 operation at any port.

**Step 1** **Specify a Controller:** To configure the available full E1 bandwidth on an unchannelized port, you select the E1 controller from the global configuration mode:

```
Router> enable
Password: (you enter your administrative password)
Router# config terminal
Enter Configuration commands, one per line. End with CNTRL/Z.
Router(config)# controller e1 2/0/0
```

**Step 2** **Specify a Channel Group:** To specify a channel group number as a full-rate unchannelized E1 port, you assign a channel-group number from 1 to 31 and specify **unframed**:

```
Router(config-controller)# channel-group 6 unframed
Router(config-controller)
```

**Note** You can not make timeslot assignments on unframed E1 ports.

The **channel group** command assigns channel-group number 6 to timeslots 0 to 31 (the entire available E1 unframed bandwidth) and subslot 0 at port 0 of the 24-Port Channelized E1/T1 line card in router chassis slot 2.

Whenever you configure an E1 channel-group as **unframed,** you assign all 32 timeslots (the entire available 2048 kbps E1 bandwidth) to that port and channel group.

**Note** When you specify **unframed** for an E1 channel, you only configure a single channel on a port.

**Step 3** **Specify Additional Controller Parameters:** Specify any other E1 controller parameters for this port and other ports (see Table 5-2 on page 5-2 and the "Controller Configuration Commands" section on page 5-17).

**Step 4** Save the controller configuration to NVRAM

```
Router(config-if)# end
Router# write
```

**Step 5** **Specify Interface Parameters:** Go to interface configuration mode for any channel group you created in the controller configuration mode:

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

**Step 6** Enter a value for protocol encapsulation. This example shows the command for using frame relay. You can also choose PPP or HDLC (default).

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

**Step 7** If IP routing is enabled on the system, assign an IP address and subnet mask. For example:

```
Router(config-if)# ip address 10.16.32.49 255.255.240.0
```

**Step 8** Add any other serial interface configuration commands required to enable other routing protocols and to set other interface characteristics.

**Step 9** Be sure all physical port connections have been made on the 24-Port Channelized E1/T1 line card in the Cisco series 10000 router chassis (see the *Cisco 10000 Series Router Line Card Hardware Installation Guide*).

**Step 10** When you have entered all of the necessary interface configuration commands (see the *Cisco IOS Interface and Hardware Component Configuration Guide)*, to complete the configuration, return to the privileged EXEC mode.

```
Router(config-if)# end
Router#
```

**Step 11** Write the new configuration to NVRAM:

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

## Configuring Full-Rate Channelized E1 Bandwidths

The following example shows you the steps you use to create a channelized full-rate interface:

**Step 1** **Specify a Controller:** To configure the available channelized full-rate bandwidth of an interface on port 1 of a line card in router chassis slot 2, you must first select the E1 controller from the global configuration mode:

```
Router(config)# controller e1 2/0/1
Router(config-controller)
```

**Step 2** **Specify a Channel Group:** To specify a channel group number for a full-rate channelized E1 port, you assign a channel-group number from 1 to 31 from the controller configuration mode:

```
Router(config-controller)# channel-group 7 timeslots 1-31
Router(config-controller)
```

This command assigns channel-group number 7 and timeslots 1 through 31 (the maximum available 1984 kbps channelized E1 bandwidth) to port 1 of the line card in router chassis slot 2.

**Step 3** **Specify Controller Parameters:** to specify any other controller parameters you enter the controller parameters for the port you have selected (see Table 5-2 on page 5-2 and the "Controller Configuration Commands" section on page 5-17).

**Step 4** **Specify Interface Parameters:** Go to interface configuration mode for any channel group you created and specify interface parameters:

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

## Configuring Fractional Channelized E1 Bandwidths

To configure an available channelized (framed) fractional E1 bandwidth on a port, you must select the E1 controller from the global configuration mode.

**Step 1** The following two examples in this step show you how to configure all the available port 2 E1 timeslots into two interfaces:

Example 1:

```
Router(config)# controller e1 2/0/2
Router(config-controller)# channel-group 8 timeslots 1, 2, 5, 11-31
Router(config-controller)#
```

These commands configure port 2 time slots 1, 2, 5, and 11 to 31, of the line card in router chassis slot 2 as channel-group (interface) number 8.

Example 2:

```
Router(config-controller)# channel-group 9 timeslots 3, 4, 6-10
Router(config-controller)#
```

Because you are still in the controller configuration mode, you use this command to configure channel-group (interface) number 9 and remaining timeslots 3, 4, and 6 to 10 on port 2 of the 24-Port Channelized E1/T1 line card in router chassis slot 2.

**Step 2** **Specify Controller Parameters:** to specify any other controller parameters you enter the controller parameters for the port you have selected (see Table 5-2 on page 5-2 and the "Controller Configuration Commands" section on page 5-17).

To view the available commands you use to configure E1 controller parameters, type a question mark on the command line from the controller configuration mode (see step 2 of the "Configuring Full-Rate Unchannelized E1 Bandwidths" section on page 5-6).

## Configuring E1 Serial Interfaces

For a detailed description of how to configure, monitor, and troubleshoot an E1 serial interface, see the *Configuration Fundamentals Configuration Guide* and the *Cisco IOS Interface and Hardware Component Configuration Guide*. This section describes the general procedure you use for configuring an E1 controller and an E1 serial interface on any E1/T1 line-card port from the interface configuration mode.

**Step 1** Enter the controller configuration mode to configure the controller (see the "Configuring a Controller" section on page 5-16 for details).

**Step 2** Use the **channel-group** command to assign framed or unframed E1 bandwidth allotments to the line card ports (see the "Specifying Channel Groups" section on page 5-21 for details).

**Step 3** Go to interface configuration mode for any channel group you created in the controller configuration mode (see the "Configuring a Serial Interface" section on page 5-17 for details.

**Step 4** Enter a value for protocol encapsulation. This example shows the command for using frame relay. You can also choose PPP or HDLC (default).

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

**Step 5** If IP routing is enabled on the system, assign an IP address and subnet mask. For example:

```
Router(config-if)# ip address 10.16.32.49 255.255.240.0
```

**Step 6** Add any other serial interface configuration commands required to enable other routing protocols and to set other interface characteristics.

**Step 7** Be sure all physical port connections have been made on the 24-Port Channelized E1/T1 line card in the Cisco series 10000 router chassis (see the *Cisco 10000 Series Router Line Card Hardware Installation Guide*).

**Step 8** When you have entered all of the necessary interface configuration commands (see the *Cisco IOS Interface and Hardware Component Configuration Guide)*, to complete the configuration, return to the privileged EXEC mode.

```
Router(config-if)# end
Router#
```

**Step 9** Write the new configuration to NVRAM:

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

After you create the port serial interface configurations, you can modify them at any time using the appropriate Cisco IOS commands described in the following sections and in the *Cisco IOS Interface and Hardware Component Configuration Guide* and the *Cisco IOS Command References*.

# T1 Configuration Examples

The examples in this section familiarize you with the line card configuration tasks and typify how you apply the commands listed in Table 5-4 and the syntax in Table 5-5 to configure a T1 controller. These examples also show how to configure T1 channel-group bandwidth at any of the 24 available ports on the line card. The example steps are intended to quickly familiarize you with the procedures you use to set up a T1 port.

**Note** You must be in privileged EXEC mode to specify and configure a controller for T1 operation on any line card port.

## Configuring a T1 Controller

From the IOS global configuration mode, you use the **config terminal** command to enter the global configuration mode (see the "Configuring a Controller" section on page 5-16 for details).

## Configuring Full-Rate Channelized T1 Bandwidths

To configure the available channelized (framed) full-rate T1 bandwidth on a line card port, you must select a T1 controller from the global configuration mode. The following example shows you the steps you use to create a channelized full-rate T1 controller:

**Step 1** **Specify a Controller:** To configure the available T1 channelized full-rate bandwidth of an interface on port 1 of a line card in router chassis slot 2, you select a T1 controller from the global configuration mode and then specify all available timeslots from the controller configuration mode:

```
Router(config)# controller t1 2/0/1
Router(config-controller)# channel-group 7 timeslots 1-24
Router(config-controller)
```

These commands assign channel-group number 7 and timeslots 1 through 24 (the maximum available framed T1 bandwidth) to port 1 of the line card in router chassis slot 2.

**Step 2** **Specify Controller Parameters:** to specify any other controller parameters you enter the controller parameters for the controller you have selected (see Table 5-4 on page 5-4 and the "Controller Configuration Commands" section on page 5-17).

## Configuring Fractional Channelized T1 Bandwidths

To configure an available channelized (framed) fractional T1 bandwidth on a port, you must select a T1 controller from the global configuration mode. The following example shows the procedure you use to create a fractional channelized T1 interface.

**Step 1** The following two examples in this step show you how to configure all the available T1 mode port 2 timeslots as interfaces:

Example 1:

```
Router(config)# controller t1 2/0/2
Router(config-controller)# channel-group 8 timeslots 1, 2, 5, 11-24
Router(config-controller)#
```

These commands configure port 2 time slots 1, 2, 5, and 11 to 24, of the line card in router chassis slot 2 as T1 channel-group (interface) number 8.

Example 2:

```
Router(config-controller)# channel-group 9 timeslots 3, 4, 6-10
Router(config-controller)#
```

Because you are still in the controller configuration mode, you use this command to configure T1 channel-group (interface) number 9 and remaining timeslots 3, 4, and 6 to 10 on port 2 of the line card in router chassis slot 2.

**Step 2** **Specify Controller Parameters:** to specify any other controller parameters you enter the controller parameters for the controller you have selected (see Table 5-4 on page 5-4 and the "Controller Configuration Commands" section on page 5-17).

## Configuring T1 Serial Interfaces

For a detailed description of how to configure, monitor, and troubleshoot a T1 serial interface, see the *Configuration Fundamentals Configuration Guide* and the *Cisco IOS Interface and Hardware Component Configuration Guide*. This section describes the general procedure you use for configuring a T1 serial interface on any line-card port from the interface configuration mode.

**Step 1** Enter the controller configuration mode to configure the T1 controller (see the "Configuring a T1 Controller" section on page 5-10).

**Step 2** Use the **channel-group** command to assign framed T1 bandwidths to the line card ports.

**Step 3** Go to interface configuration mode for any T1 channel group you created in the controller configuration mode and specify the serial interface channel-group number:

```
Router(config)# interface serial 2/0/2:20
Router(config-if)#
```

**Step 4** Enter a value for protocol encapsulation. This example shows the command for using frame relay. You can also choose PPP or HDLC (default).

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

**Step 5** If IP routing is enabled on the system, assign an IP address and subnet mask. For example:

```
Router(config-if)# ip address 10.16.32.49 255.255.240.0
```

**Step 6** Add any other configuration commands required to enable other routing protocols and to set other interface characteristics.

**Step 7** Be sure all physical port connections have been made on the line card in the Cisco series 10000 router chassis (see the *Cisco 10000 Series Router Line Card Hardware Installation Guide*).

**Step 8** When you have entered all of the necessary interface configuration commands (see the *Cisco IOS Interface and Hardware Component Configuration Guide* publication), to complete the configuration, return to the privileged EXEC mode.

```
Router(config-if)# end
Router#
```

**Step 9** Write the new configuration to NVRAM:

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

After you create the port interface configurations, you can modify them at any time using the appropriate Cisco IOS commands described in the following sections and in the *Cisco IOS Interface and Hardware Component Configuration Guide* and the *Cisco IOS Command References*.

# Command Descriptions

You invoke the various IOS commands using the following Cisco IOS command modes (see Table 5-2 on page 5-2):

- Privileged EXEC
- Global Configuration
- Controller Configuration
- Interface Configuration

You must begin configuring the 24-Port Channelized E1/T1 line card from the privileged EXEC command mode.

## Privileged EXEC Commands

The Cisco IOS privileged EXEC mode commands are:

- hw-module
- show controllers
- show interfaces

You enter the privileged EXEC configuration mode from the user EXEC mode:

```
Router> enable
Password: (you enter your administrative password)
Router#
```

When your prompt changes to `Router#` you are in the privileged EXEC mode.

## Simulating Line Card Installation and Removal

You use this **hw-module** command to simulate installation and removal of a line card at any router chassis slot you previously referenced using the **card** command (see the "Preconfiguring a Line Card" section on page 5-15):

```
Router# hw-module {slot slot reset | subslot slot/subslot reset}
Router#
```

Example:

```
Router# hw-module slot 5 reset
```

This command simulates removal and installation of any line card in router chassis slot 5.

This command does not have a **no** form.

## Displaying Controller Information

You use the **show controllers** command to display information about an E1 or T1 controller configuration at any of the 24 line card ports. Use the following syntax for an E1 controller:

```
show controllers e1 slot/subslot/port
```

Example 1: The following example shows the output displayed on the system console for line card E1 operation in router chassis slot 1, subslot 0, port 0 when this command executes:

```
Router# show controller e1 1/0/0
1 1/0/0 is down.
Applique type is Channelized E1 - balanced
Receiver has loss of signal.
Framing is CRC4, Line Code is HDB3, Clock Source is Line.
International Bit: 1, National Bits: 11111
Data in current interval (12 seconds elapsed):
0 Line Code Violations, 0 Path Code Violations
0 Slip Secs, 0 Fr Loss Secs, 0 Line Err Secs, 0 Degraded Mins
0 Errored Secs, 0 Bursty Err Secs, 0 Severely Err Secs, 12 Unavail Secs
Hardware is C10K CHE1T1 line card
Router#
```

Example 2: The following example shows the output displayed on the system console for a line card operating in T1 mode in router chassis slot 1, subslot 0, port 0 when this command executes:

```
Router# show controller t1 1/0/0
1 1/0/0 is down.
Applique type is Channelized T1 - balanced
Receiver has loss of signal.
Framing is esf, Line Code is ami, Clock Source is Line.
Data in current interval (12 seconds elapsed):
0 Line Code Violations, 0 Path Code Violations
0 Slip Secs, 0 Fr Loss Secs, 0 Line Err Secs, 0 Degraded Mins
0 Errored Secs, 0 Bursty Err Secs, 0 Severely Err Secs, 12 Unavail Secs
Hardware is C10K CHE1T1 line card
Router#
```

## Displaying Interface Information

You use the **show interface** command to display information about the current E1/T1 channel-group (interface) at any of the 24 24-Port Channelized E1/T1 line card ports using the following syntax:

```
show interfaces serial slot/subslot/port:channel-group [subcommand]
```

The Following example shows the output displayed on the system console for the serial interface configuration on router chassis slot 1, subslot 0, port 0, and channel-group 0 when this command executes:

```
Router# show interfaces serial 1/0/0:0
Serial1/0/0:0 is down, line protocol is down
Hardware is Channelized E1T1 controller
MTU 1500 bytes, BW 64 Kbit, DLY 20000 usec, rely 255/255, load 1/255
Encapsulation HDLC, crc 16, loopback not set
Keepalive set (10 sec)
Last input never, output never, output hang never
Last clearing of "show interface" counters 00:00:18
Queueing strategy: PXF First-In-First-Out
Output queue 0/32, 0 drops; input queue 0/75, 0 drops
5 minute input rate 0 bits/sec, 0 packets/sec
5 minute output rate 0 bits/sec, 0 packets/sec
0 packets input, 0 bytes, 0 no buffer
Received 0 broadcasts, 0 runts, 0 giants, 0 throttles
0 input errors, 0 CRC, 0 frame, 0 overrun, 0 ignored, 0 abort
0 packets output, 0 bytes, 0 underruns
0 output errors, 0 collisions, 0 interface resets
0 output buffer failures, 0 output buffers swapped out
0 carrier transitions alarm present
Timeslot(s) Used: 1, subrate: 64Kb/s, transmit delay is 0 flags
non-inverted data
Router#
```

This command does not have a **no** form. You must specify at least one channel group when you use this command. You can also specify a subcommand from a number of associated subcommands (see Table 5-2 on page 5-2 and Table 5-4 on page 5-4).

# Global Configuration Commands

The line card Cisco IOS global configuration commands are:

- card
- hw-module (shutdown)
- controller
- interface

You use the **configure terminal** command to enter the global mode from the privileged EXEC mode:

Example:

```
Router> enable
Password: (you enter your administrative password)
Router# config terminal
Enter Configuration commands, one per line. End with CNTRL/Z.
Router(config)#
```

When your prompt changes to `Router(config)#` you are in the global configuration mode.

## Preconfiguring a Line Card

Using the **card** command you can preconfigure a router chassis slot for a specific line card and operating mode in the absence of such a line card in that slot. After you have preconfigured a router chassis slot for a 24-Port Channelized E1/T1 line card and E1 or T1 operation, when you install a 24-Port Channelized E1/T1 line card into the preconfigured slot, that line card is ready to operate in the preconfigured mode.

When you install a 24-Port Channelized E1/T1 line card in the router chassis slot you have preconfigured, a power-on self test (POST) should immediately take place and the line card should then be ready for operation.

You use the **card** command in the global configuration mode to preprovision any Cisco 10000 series router chassis slot for line card E1 or T1 operation when no line card is available, or to change modes for a line card that is already installed:

```
Router(config)# [no] card slot/subslot 24che1t1-1 [mode {e1 | t1}]
Router(config)#
```

Where:

- **24che1t1-1** is the Cisco E1/T1 line-card Cisco product reference, which you select from a list of optional product references for any line card you can install and configure in a Cisco series 10000 router chassis.

**Note** The -1 number specifies the current 24-Port Channelized E1/T1 line card revision and may change.

- **mode e1** configures all the line card ports for the E1 framing format.
- **mode t1** configures the line card ports for the T1 framing format.

The following examples show how to use the **card** command to simulate the line card controller E1 mode and E1 operation:

- Example 1: In the following example, you configure router chassis slot 3 for E1 operation and use the **hw-module** and **card** commands to simulate a 24-Port Channelized E1/T1 line card installation:

```
Router> enable
Password: (you enter your administrative password)
Router# hw-module slot 3 reset
Router# config terminal
Enter Configuration commands, one per line. End with CNTRL/Z.
Router(config)# card 3/0 24che1t1-1 mode e1
Router(config)#
```

- Example 2: In the following example, you configure router chassis slot 3 for the T1 operation and use the **hw-module** and **card** commands to simulate a 24-Port Channelized E1/T1 line card installation:

```
Router> enable
Password: (you enter your administrative password)
Router# hw-module slot 3 reset
Router# config terminal
Enter Configuration commands, one per line. End with CNTRL/Z.
Router(config)# card 3/0 24che1t1-1 mode t1
Router(config)#
```

The following relationships exist between the Cisco series 10000 router on-line insertion (OIR) feature and the **card** command:

  - When you install a 24-Port Channelized E1/T1 line card into a router chassis slot for which a **24che1t1-1** entry already exists in the running configuration, the OIR feature invokes the existing E1 or T1 controller configuration.
  - When you install a 24-Port Channelized E1/T1 line card into a router chassis slot for which no **24che1t1-1** entry exists in the running configuration, the OIR feature invokes the default configuration.
  - When you install a 24-Port Channelized E1/T1 line card into a router chassis slot for which a product reference other than **24che1t1-1** exists, the OIR feature invokes the default configuration.

- In this example you simulate the removal of a 24-Port Channelized E1/T1 line card from router chassis slot 3, which you previously specified with the **card** command.

**Note** To use the “no” form of card command, the line card must be physically removed from the chassis.

```
Router(config)# hw-module slot 3 shutdown
Router(config)# no card 3/0
Router(config)#
```

## Shutting Down Line Card Simulation

You use this command to turn on and shut down simulated 24-Port Channelized E1/T1 line card operation using the following syntax:

Router (config)#[**no**] **hw-module** {**slot** *slot* **shutdown** | **subslot** *subslot*/*slot* **shutdown**}

The following examples show how you use the **hw-module shutdown** command to shut down or turn on the E1/T1 line-card simulated running configuration:

- In this example, you shut down a simulated running configuration in router chassis slot 5:

```
Router(config)# hw-module slot 5 shutdown
Router(config)#
```

- In this example, you turn on the simulated running configuration of the line card in router chassis slot 5:

```
Router(config)# no hw-module slot 5 shutdown
Router(config)#
```

## Configuring a Controller

You use the **configure terminal** command to enter the global configuration mode and to configure the E1 or T1 controller. To select a chassis slot, subslot, and port E1 controller, you use the following syntax from the global configuration mode:

Router(config)# **controller** {**t1** | **e1**} *slot*/*subslot*/*port*

Example 1: In the following example, you select chassis slot 2 E1 controller:

```
Router> enable
Password: (you enter your administrative password)
Router# config terminal
Enter Configuration commands, one per line. End with CNTRL/Z.
Router(config)# controller e1 2/0/0
Router(config-controller)#
```

Example 2: In the following example, you select chassis slot 2 T1 controller:

```
Router> enable
Password: (you enter your administrative password)
Router# config terminal
Enter Configuration commands, one per line. End with CNTRL/Z.
Router(config)# controller t1 2/0/0
Router(config-controller)#
```

## Configuring a Serial Interface

For each port you configure at the controller configuration level, you must also configure various associated serial interface parameters for each channel group you previously created at that port. To configure 24-Port Channelized E1/T1 line card serial interfaces for their serial interface parameters, you must be in the interface configuration mode. To specify interface parameters for any channel group you previously created, use the following syntax:

[**no**] **interface serial** *slot*/*subslot*/*port*:*channel-group*

Use the **no** form of the channel-group command (see the "Specifying E1 Channel Groups" section on page 5-21 and the "Specifying T1 Channel Groups" section on page 5-22) to disable a serial interface at an E1 or T1 port, respectively, and a channel group you previously provisioned.

Example:

```
Router> enable
Password: (you enter your administrative Password)
Router# config terminal
Enter Configuration commands, one per line. End with CNTRL/Z.
Router(config)# interface serial 2/0/0:20
Router(config-if)#
```

These commands establishes the interface configuration mode for the 24-Port Channelized E1/T1 line card in router chassis slot 2, subslot 0, port 0 and channel-group 20. When your prompt changes to `Router(config-if)#` you can configure the channel group serial interface parameters.

**Note** You can not specify a channel group until you have first selected an E1 or T1 controller.

# Controller Configuration Commands

After installing a 24-Port Channelized E1/T1 line card into a router chassis slot, you configure it in controller configuration mode. You can also preconfigure a router chassis slot and the line-card ports for E1 or T1 mode, and as serial interfaces, before you install a line card into the preconfigured slot (see the "Preconfiguring a Line Card" section on page 5-15).

The controller configuration commands listed in Table 5-2 and in Table 5-4 are those you use to preconfigure or configure a 24-Port Channelized E1/T1 line card controller. You also use these commands to provide the line card with its system functionality. To configure the line card at the E1 or T1 controller level, you perform the following general procedure:

**Step 1** Configure or preconfigure a chassis slot for E1 or T1 mode.

**Step 2** Configure or preconfigure controller timeslots as channel groups (interfaces).

**Step 3** Specify any other E1 or T1 controller parameters.

**Step 4** Save the controller configuration by writing it to NVRAM.

## E1 Port Interface Density

Using all 24 line card ports, you can configure as many as 744 **C**E1 interfaces (channels) of any 24-Port Channelized E1/T1 line card you install in a router chassis slot. You can also preconfigure as many channels for that same chassis slot even though the slot does not contain a line card. You use the **card** command (see the "Preconfiguring a Line Card" section on page 5-15) to preconfigure any router chassis slot, subslot, and port for E1 controller mode that does not contain a line card.

## T1 Port Interface Density

Using all 24 line card ports, you can configure as many as 576 CT1 interfaces (channels) of any 24-Port Channelized E1/T1 line card you install in a router chassis slot. You can also preconfigure as many channels for that same chassis slot even though the slot does not contain a T1 line card. You use the **card** command (see the "Preconfiguring a Line Card" section on page 5-15) to preconfigure any router chassis slot, subslot, and port for T1 controller mode that does not contain a line card.

# Controller Commands

This section describes the commands you use to create, customize, and test an E1 or T1 interface at the system level.

After you configure the line card, slot, subslot, port, and channel-group numbers, you can configure other E1 or T1 controller parameters (see the "E1 Configuration Examples" section on page 5-6 and the "T1 Configuration Examples" section on page 5-10). You configure the E1 or T1 controller configuration parameters using the commands described in this section:

## Running a BER Test

You can configure an interface to run a BERT (Bit Error Rate Test). You use a BERT to isolate cable signal problems in the field, to check network cabling, and to isolate apparent signaling problems in the field.

Before you begin a network BERT, you need to set up a remote loopback from the far-end (remote) DSU/CSU (see the "Specifying Loopbacks" section on page 5-27).

Use the following loopback commands to set the near end devices (CSU/DSUs) to perform loopbacks:

- **loopback local line–**Use this loopback submode on both E1 and T1 ports. This submode configures the line card to loop transmitted packets back as received packets after the framer. The line card also receives all incoming packets and outgoing packets are also transmitted to the remote network device.
- **loopback network**—Use this loopback mode on both E1 and T1 ports. This mode configures the line card to loop packets transmitted from the line card to a remote device back to the line card as received packets. This loopback mode has the following submodes:
  - l**oopback network line**—Use this loopback submode on both E1 and T1 ports. This submode configures the line card to loop received packets back to the remote device without first framing them.
  - **loopback network payload**—Use this loopback submode on both E1 and T1 ports. This submode configures the line card to loop received packets back to the remote device after first framing them.
- **loopback remote** — Operates in T1 mode only (see the "Specifying T1 Loopbacks" section on page 5-27). This loopback mode has the following submodes:
  - **loopback remote line—**Use this loopback submode on T1 ports to configures the far-end device to loop received packets back to the line card without first framing them.
  - **loopback remote payload**—Use this loopback submode on T1 ports to configures the far-end device to loop received packets back to the line card after framing them.

### Running an E1 BERT

You can only run a BERT (bit-rate error test) over the full bandwidth of an E1 controller in **unframed** mode. To send a BERT pattern on a line card E1 interface, use the following controller configuration command syntax:

```
[no] bert pattern pattern interval time [unframed]
```

Where:

- **pattern** *pattern* is:
  - **2^11**—pseudorandom test pattern (32,768 bits long)
  - **2^15**—pseudorandom O.151 test pattern (32,768 bits long)
  - **2^20-O153**—pseudorandom O.153 test pattern (1,048,575 bits long)
  - **2^20-QRSS**—pseudorandom QRSS O.151 test pattern (1,048,575 bits long)
- **interval** *time* is 1 to 1440 minutes.

The **no** form of the bert command is the default. You can terminate a BERT at any time using the **no bert** command.

**Note** The **no bert** command terminates a running BERT and, whether running or stopped, inhibits the display of BERT information on the console when you invoke the **show controllers** command (see the "Displaying Controller Information" section on page 5-13).

Example:

To send a BERT pseudorandom pattern of 2^20-O.153 through E1 interface 1/0/0 for 5 minutes:

```
Router(config)# controller e1 1/0/0
Router(config-controller)#bert pattern 2^20-O153 interval 5 unframed
Router(config-controller)#
```

For more information, see the *Cisco IOS Interface and Hardware Component Configuration Guide* and the *Cisco 10000 Series Internet Router Troubleshooting Guide*.

### Running a T1 BERT

In addition to the **local** and **network loopback** configurations described in the "Running a BER Test" section on page 5-19, you can also perform **remote** loopback testing on T1 interfaces.

**Note** You can only perform **remote** loopback testing on T1 ports that also have C-bit parity framing configured.

To send a BERT pattern on a line card T1 interface configured for remote loopback, use the following controller configuration command syntax:

**[no] bert pattern** *pattern* **interval** *time*

Use the following command to send a BERT pattern to the far-end device:

Example:

To send a BERT pseudorandom pattern of 2^20-O.153 through T1 interface 1/0/0 to the far-end device for 5 minutes:

```
Router(config)# controller t1 1/0/0
Router(config-controller)# bert pattern 2^20-O153 interval 5
Router(config-controller)#
```

## Specifying T1 Short-Haul Cable Length

You can only use the **cablelength** command on T1 ports. Use this command to compensate short haul lines (under 660 feet) for signal losses associated with line length. To specify cable length for short-haul cables use the following controller configuration command syntax:

[**no**] **cablelength short** {**110** | **220** | **330** | **440** | **550** | **660**}

Use Table 5-6 to specify the proper short-haul cable length value for cables up to 660 feet long.

***Table 5-6 T1 Short-Haul Line Buildout***

| Line Length (feet) | Cable Length Value |
|---|---|
| 0 to 110 | 110 |
| 110 to 220 | 220 |
| 220 to 330 | 330 |

*Table 5-6 T1 Short-Haul Line Buildout*

| Line Length (feet) | Cable Length Value |
|---|---|
| 330 to 440 | 440 |
| 440 to 550 | 550 |
| 550 to 660 | 660 |

## Specifying Channel Groups

A channel group is a serial interface. At a framed E1 port, you can assign bandwidth to as many as 31 channel groups numbered from 1 to 31 in 64-kbps increments called timeslots. When you specify an E1 port as unframed, you only assign a single channel group whose bandwidth is 2048 kbps. At a T1 port, you can assign bandwidth to as many as 24 channel groups in 56- or 64-kbps timeslot increments. T1 ports can not operate in unframed mode.

### Specifying E1 Channel Groups

An E1 port has 32 fixed timeslots available. In framed mode, timeslot 0 is used for framing and can not be used. Each E1 timeslot is a 64-kbps DS0 band. To create E1 interfaces that supports subscriber traffic rates from 64 kbps to 2048 kbps on any of the 24 line card ports, you combine timeslots into channel groups. You use the **channel-group** command to create one or more interfaces at any port operating in E1 mode.

The E1 channel groups have these characteristics:

- A channel group is an interface.
- You partition each line card port into channel groups. A framed port can have as many as 31 E1 channel groups consisting of from 0 to 31 timeslots. At any port:
  - A channel group that includes all 32 timeslots (0 to 31) is a single *unframed* (unchannelized) E1 interface whose bandwidth is 2048 kbps.
  - A channel group that includes 31 timeslots (1 to 31) is a single *framed* (channelized) E1 interface whose bandwidth is 2048 kbps.
  - A channel group that consists of fewer than 31 timeslots is a single framed fractional E1 interface whose bandwidth you assign from the port timeslot pool (see the "Configuring an E1 Controller" section on page 5-6).
- Each group of 31 framed timeslots can be grouped into multiple fractional E1 interfaces.
- A timeslot cannot be part of more than one E1 channel group.

After you configure a line card as an E1 controller, you can create one or more logical E1 channel groups from the timeslots available at each port using the following controller command syntax:

**[no] channel-group** *channel-group-number* {**timeslots** *list-of-timeslots* [**speed 64**] | **unframed**}

Where:

- **channel-group** *channel-group-number* is a number from 0 to 30 that you assign to identify each channel group you create.
- **timeslots** *list-of-timeslots* can be a value from 1 to 31, or a combination of timeslots from 1 to 31. You indicate a timeslot range using a hyphen, commas, or a combination of both. One timeslot defines a single E1 channel as shown by the following examples.
- **speed 64** specifies the digital data rate of a single timeslot as 64 kbps (default).

- **unframed** specifies a single digital interface whose bandwidth is 2048 kbps. If you do not specify this option, you invoke E1 framing and the channel-group bandwidth depends on the number of timeslots you assign to the interface.

Use the **no** form of the command to cancel a logical channel group previously created.

The following examples show how you use the **channel-group** command in E1 mode:

- In this example, you configure channel group 20 as nine timeslots, whose bandwidth is 576 kbps, on port 0:

```
Router(config)# controller E1 1/0/0
Router(config-controller)# channel-group 20 timeslots 1-9
Router(config-controller)#
```

  To access the interface configuration mode for this channel group, you enter:

```
Router(config)# interface serial 1/0/0:20
Router (config-if)#
```

- In this example, an E1 interface includes channel group 18 and consists of all 31 timeslots, creating a full-rate framed E1 interface at port 0:

```
Router(config)# controller E1 1/0/0
Router(config-controller)# channel-group 18 timeslots 1-31
Router(config-controller)#
```

  To access interface configuration mode for this channel group, enter:

```
Router(config)# interface serial 1/0/0:18
Router(config-if)#
```

- In this example, three channelized E1 interfaces are configured as three channel groups, which use 11 of the 31 available timeslots, at port 0:

```
Router(config)# controller E1 1/0/0
Router(config-controller)# channel-group 19 timeslots 1-6
Router(config-controller)# channel-group 20 timeslots 10
Router(config-controller)# channel-group 21 timeslots 7-9, 24
Router(config-controller)#
```

- In this example, channel groups 19, 20, and 21 are cancelled and timeslots 1 to 10, and 24 are restored to the available port 0 channel-group pool.

```
Router(config)# controller E1 1/0/0
Router(config-controller)# no channel-group 19
Router(config-controller)# no channel-group 20
Router(config-controller)# no channel-group 21
Router(config-controller)#
```

### Specifying T1 Channel Groups

A T1 port has 24 fixed timeslots available. Each T1 timeslot is a 56- or 64-kbps DS0 band depending on the configured line speed. To create T1 interfaces that support subscriber traffic rates from 56 kbps to 1344 kbps on any of the 24 line card ports operating at 56 kbps, you combine timeslots into framed channel groups.

To create T1 interfaces that support subscriber traffic rates from 64 kbps to 1536 kbps on any of the 24 line card ports operating at 64 kbps, you do the same.

You use the **channel-group** command to create one or more interfaces at any port operating in T1 mode. The T1 channel groups have these characteristics:

- A channel group is an interface.

- You partition the line card ports into channel groups. A port can have as many as 24 channel groups consisting of from 1 to 24 (DS0) timeslots. At any port:
  - A channel group that includes 24 timeslots (1 to 24) is a single *framed* (channelized) T1 interface whose bandwidth is 1536 kbps.
  - A channel group that consists of fewer than 24 timeslots is a single framed fractional T1 interface whose bandwidth you assign from the T1 port timeslot pool (see the "Configuring a T1 Controller" section on page 5-10).
- Each group of 24 framed timeslots can be grouped into multiple fractional T1 interfaces.
- A channel group cannot be part of more than one T1 interface.

After you configure a line card for T1 mode, you can create one or more logical T1 channel groups from the timeslots available at each port using the following controller command syntax:

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

Where:

- **channel-group** *channel-group-number*, **timeslots** *list-of-timeslots,* and **speed 64** (default) have the same significance as the options described for the E1 channel-group command above in the "Specifying E1 Channel Groups" section.
- **speed 56** specifies the digital data rate of a single timeslot as 56 kbps.

The following examples show how you use the **channel-group** command in T1 mode:

- In this example, you configure channel group 20 as nine timeslots, whose bandwidth is 576 kbps (64 kbps default speed) on port 0:

```
Router(config)# controller t1 1/0/0
Router(config-controller)# channel-group 20 timeslots 1-9
Router(config-controller)#
```

  To access the interface configuration mode for this channel group, you enter:

```
Router(config)# interface serial 1/0/0:20
Router (config-if)#
```

- In this example, a T1 interface includes channel group 18 and consists of all 24 timeslots, creating a full-rate framed T1 interface at port 0:

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

  To access interface configuration mode for this channel group, enter:

```
Router(config)# interface serial 1/0/0:18
Router(config-if)#
```

- In this example, three channelized T1 interfaces are configured as three channel groups, which use 11 of the 24 available timeslots, at port 0:

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

- In this example, channel groups 19, 20, and 21 are cancelled and timeslots 1 to 10, and 24 are restored to the available port 0 channel-group pool.

```
Router(config)# controller t1 1/0/0
Router(config-controller)# no channel-group 19
Router(config-controller)# no channel-group 20
Router(config-controller)# no channel-group 21
Router(config-controller)#
```

## Specifying a Clock Source

You use the **clock** command to choose whether the transmitted clock signal comes from the received clock or is generated by an internal clock source. You use the following controller command to specify the transmit timing source:

[**no**] **clock source** {**line** | **internal**}

Example:

```
Router(config-controller)# clock source line
```

The **no** form of this command sets the clock source to **line**, the default state.

**Caution** You should never configure both sides of an E1 or T1 link to **clock source line**.

## Adding a Controller Description

You can use the **description** command to identify particulars about a controller interface. You can enter up to 80 characters in your controller description string. To provide additional information about an interface you use to the following controller command:

[**no**] **description** *text*

Where: *text* is the description string.

Example:

```
Router(config-controller)# description Subscriber X
```

## Enabling T1 Facility Data Link Performance Monitoring

In T1 mode, you can specify the type of FDL (Facility Data Link) performance monitoring to use. You use FDL performance monitoring to verify the integrity of the data link and to transfer statistics between devices. FDL data is a diagnostic tool you use in remote testing of T1 network data links. You can specify either of 2 FDL performance monitoring versions for use in T1 networks:

- ANSI
- ATT

The version differences lie in the message protocols sent to or received from the remote device. To specify the FDL version, use the **fdl** command:

[**no**] **fdl** {**ansi** | **att**}

Where:

- **ansi**—Conforms with the requirements of the ANSI T1.403-1995 specification to communicate 16-bit data-link code words between the line card and the remote device.

- **att**—Conforms with the requirements of the AT&T TR 54016-1989 specification to receive 16-bit data-link code words between the line card and the remote device. Note that the 24-port 24-Port Channelized E1/T1 line card will not generate remote performance statistics requests. It only responds to received performance statistics requests.

## Specifying Framing

North American T1 facilities operate at a line speed of 1544 kbps. Framing conforms to either of 2 standards:

- D4 Superframe Format (SF)—The standard D4 SF consists of twelve 24-timeslot frames.
- Extended Superframe Format (ESF)—The ESF consists of twenty-four 24-timeslot frames.

European, Australian, and some Asian E1 facilities operate at a maximum line speed of 2048 kbps. Framing conforms to the CCITT G.704 or G.732 specification, which defines sixteen 32-timeslot multiframes. The initial timeslot (0) is used for framing.

### Specifying E1 Framing

To specify framing for an E1 interface, use the following **framing** command:

```
[no] framing {crc4 [australia] | no crc4 [australia]}
```

Example:

```
Router# configure terminal
Enter configuration commands, one per line. End with CNTRL/Z.
Router(config)# controller e1 1/0/0
Router(config-controller)# framing crc4
Router(config-controller)#
```

Where **crc4** is a checksum format used in framing timeslot 0 to provide error detection.

The default framing type is CRC4.

The **no** form of this command sets framing to the default value.

In Australia, you must also specify **australia** to obtain proper handling of the clock setting when AIS (alarm indication signal) detection occurs.

```
Router(config)# controller e1 1/0/0
Router(config-controller)# framing crc4 australia
Router(config-controller)#
```

### Specifying T1 Framing

To specify framing for a T1 interface, use the following **framing** command:

```
[no] framing {esf | sf }
```

Where:

- **esf**—Specifies the extended superframe format
- **sf**—Specifies the superframe format

Example:

```
Router# configure terminal
Enter configuration commands, one per line. End with CNTRL/Z.
Router(config)# controller t1 1/0/0
Router(config-controller)# framing sf hdlc-idle 0x7E
Router(config-controller)#
```

# Specifying Linecoding

You specify line coding to establish how physical data appears on the interface line using one of the following formats:

T1 formats:

- AMI (Alternate Mark Inversion)
- B8ZS (Binary 8 Zero Substitution)

E1 formats:

- HDB3 (High Density Bipolar 3)

These line coding formats ensure that line synchronization is maintained.

## Specifying E1 Linecoding

Linecode format HDB3 ensures that the E1 receiver remains synchronized in the presence of consecutive 0 bits in received data. In E1 mode, the line card supports HDB3 line encoding only:

```
[no] linecode hdb3
```

Where **hdb3** specifies the high-density bipolar 3 linecode in which every other 1 bit is of the opposite polarity and bipolar violations are used to automatically maintain proper 1s density.

The **no** form of this command sets the E1 default linecode to **hdb3**.

Example:

```
Router# configure terminal
Enter configuration commands, one per line. End with CNTRL/Z.
Router(config)# controller e1 1/0/0
Router(config-controller)# framing hdb3
Router(config-controller)#
```

## Specifying T1 Linecoding

You can specify either AMI or B8ZS line coding for T1 interfaces. In T1 mode, use the following linecode command:

```
[no] linecode {ami | b8zs}
```

- **ami**——specifies alternate mark inversion coding in which every other 1s bit is of the opposite polarity. If two consecutive bits have the same polarity, a bipolar violation occurs.
- **b8zs**——specifies binary 8 zero substitution in which each group of 8 consecutive zeros is replaced with a bipolar violation to maintain synchronization.

Example:

```
Router# configure terminal
Enter configuration commands, one per line. End with CNTRL/Z.
Router(config)# controller t1 1/0/0
Router(config-controller)# linecode ami
Router(config-controller)#
```

## Specifying Loopbacks

This line card supports local and network E1 and T1 loopback modes, and remote T1 loopback modes for testing, network fault isolation, and agency compliance. You can test E1 and T1 lines in local and network loopback modes. You can also test T1 lines in remote mode.

### Specifying E1 Loopbacks

If problems occur when you configure an E1 interface, you can troubleshoot the line card by using the following command in controller configuration mode:

```
[no] loopback {local | network {line | payload}}
```

Where:

- **local**—Configures the line card to loop transmitted traffic back to the line card as E1 received traffic and sends the local transmit signal to the remote receiver.

**Note** An E1 port in local loopback mode cannot send AIS due to hardware limitations. It only transmits to the line what it loops back to it's receive. As a result, the remote end will keep receiving packets from a port in local loopback.

- **network line**—Configures the line card to loop received traffic back to the remote device without passing them through the framer.
- **network payload**—Configures the line card to loop received traffic back to the remote device after passing them through the framer.

Use the **no** form of the command to terminate a loopback.

For more information on this command, see the online *Cisco 10000 Series Internet Router Troubleshooting Guide*.

Example 1:

- Configure an E1 controller port for local loopback.

```
Router(config)# controller e1 1/0/0
Router(config-controller)# loopback local
Router(config-controller)#
```

Example 2:

The following commands configure the E1 controller for local loopback at port 0, channel group 1

```
Router(config)# controller e1 1/0/0
Router(config-controller)# loopback local
Router(config-controller)#
```

### Specifying T1 Loopbacks

If problems occur when you configure an T1 interface, you can troubleshoot the line card by using the following command in controller configuration mode:

```
[no] loop {local | network {line | payload} | remote {line | payload | iboc}}
```

Where:

- **local**—Configures the line card to loop transmitted traffic back to the line card as T1 received traffic and sends the local transmit signal to the remote receiver.

**Note** A T1 port in local loopback mode cannot send AIS due to hardware limitations. It only transmits to the line what it loops back to it's receive. As a result, the remote end will keep receiving packets from a port in local loopback.

- **network line**—Configures the T1 framer for the network line submode. In this mode, the line card transmitted packets are not reframed before they are sent to the remote device. The line card also receives all incoming packets.
- **network payload**—Configures the T1 framer for the network payload submode. In this mode, the line card received packets are reframed before they are sent back to the remote device.
- **remote line**—Configures the far end T1 in line loopback using the ESF ANSI protocol.
- **remote payload**—Configures the far end T1 in payload loopback using the ESF ANSI protocol.
- **remote iboc**—Configures the far end T1 in line loopback using the SF inband protocol.

## Specifying E1 National Reserve Bits

You use national reserve bits, which are embedded in-band control timeslot TS0 to codify a framing protocol for national variations:

[**no**] **national reserve** *int sa4 sa5 sa6 sa7 sa8*

Where:

- *int*—This is the national reserved bit.
- *Sa4* to *Sa8*—Specify the country-code bit values.

Example:

```
Router(config-controller)# controller 2/0/2
Router(config-controller)# national reserve 1 1 1 1 0 1
Router(config-controller)#
```

The default [**no**] value of this command is 000000.

**Note** National reserve bits are active only when the **framing no-crc4** command is invoked (see the "Specifying E1 Framing" section on page 5-25).

## Specifying a T1 Yellow Alarm

This line card supports the ability to control generation and detection of a T1 yellow Remote Alarm Indication (RAI). You enable or inhibit the RAI yellow alarm using the **yellow** command:

[**no**] **yellow** {**generation** | **detection**}

Where:

**generation**—Enables or inhibits the near- and far-end RAI yellow alarms.

**detection**—Inhibits or enables the near- end RAI yellow alarms.

Example 1:

This command inhibits far-end yellow alarms.

```
Router# configure terminal
Enter configuration commands, one per line. End with CNTRL/Z.
Router(config)# controller t1 1/0/0
Router(config-controller)# no yellow generation
Router(config-controller)#
```

Example 2:

This command enables far-end yellow alarms.

```
Router# configure terminal
Enter configuration commands, one per line. End with CNTRL/Z.
Router(config)# controller t1 1/0/0
Router(config-controller)# yellow generation
Router(config-controller)#
```

## Specifying Shutdown

You can shut down an interface. Doing so disables all interface functions and marks the interface as unavailable when you invoke monitoring commands such as **show.** You enable or shut down operation of an interface using the **shutdown** command:

[**no**] **shutdown**

Example 1:

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

This command enables operation of the specified interface.

Example 2:

```
Router(config-controller)# shutdown
Router(config-controller)#
```

This command halts operation of the specified interface.

CHAPTER 6

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

This chapter describes the procedures for configuring the Cisco 10000 series 1-Port Channelized OC-12/STM-4 line card, hereafter known as the 1-Port Channelized OC-12/STM-4 line card.

The 1-Port Channelized OC-12/STM-4 line card uses fractional T1 (n x DS0), DS-1, E1, fractional E1 (n x DS0) and DS3 IP services to connect ISP customers to the network backbone. It provides full duplex operation at OC-12 bandwidth rates using a (single-mode) fiber optic port that provides the Cisco 10000 series router with a single interface that connects up to 768 individual channelized options.

This chapter contains the following sections:

# Software Support

Table 6-1 shows the minimum Cisco IOS release on each release train that supports the 1-Port Channelized OC-12/STM-4 line card.

*Table 6-1 1-Port Channelized OC-12/STM-4 Line Card Software Support*

| Required PRE | Minimum Cisco IOS Releases |
|---|---|
| PRE1 | Cisco IOS Release 12.0(9)SL and later releases of Cisco IOS Release 12.0SL<br>Cisco IOS Release 12.0(17)ST and later releases of Cisco IOS Release 12.0ST<br>Cisco IOS Release 12.0(22)S and later releases of Cisco IOS Release 12.0S<br>Cisco IOS Release 12.2(8)BZ and later releases of Cisco IOS Release 12.2(8)BZ for SONET mode only.<br>For both SONET and SDH mode, use Cisco IOS Release 12.0(23)S and later releases of Cisco IOS 12.0S. |
| PRE2 | Cisco IOS Release 12.3(7)XI and later releases of Cisco IOS 12.3XI<br>For both SONET and SDH mode, use Cisco IOS Release 12.2(15)BX and later releases of Cisco IOS 12.2BX.<br>Cisco IOS Release 12.2(28)SB and later releases of Cisco IOS 12.2SB |
| PRE3 | Cisco IOS Release 12.2(31)SB2 and later releases of Cisco IOS 12.2SB |

## Checking Hardware and Software Compatibility

The PRE installed in the Cisco 10000 series router chassis must support the Cisco IOS software running on the router. Use the **show version** command to check the PRE version installed.

To see if a feature is supported by a Cisco IOS release, to locate the software document for that feature, or to check the minimum software requirements of Cisco IOS software with the hardware installed on your router, Cisco maintains the Software Advisor tool on Cisco.com at
http://www.cisco.com/cgi-bin/Support/CompNav/Index.pl

This tool does not verify whether line cards within a system are compatible, but does provide the minimum Cisco IOS requirements for individual hardware line cards, modules, or options.

You must be a registered user on Cisco.com to access this tool.

# SONET Channelization

The 1-Port Channelized OC-12/STM-4 line card supports the following SONET channelization modes:

- STS-1 > DS3/T3
- STS-1 > DS3/T3 > DS1
- STS-1 > DS3/T3 > DS3 subrate
- STS-1 > VT1.5 > DS1
- STS-1 > VT2 > E1

Table 6-2 shows serial interface examples for SONET channelizations on a previously-configured SONET-framed controller on the 1-Port Channelized OC-12/STM-4 line card.

*Table 6-2 SONET Interface Examples for a SONET-Framed Controller*

| Interface Type | Syntax |
|---|---|
| STS-1,<br>over DS3/T3 | `Router(config)# **interface serial** <slot>/<subslot>/<port>.<T3#>` |
| STS-1,<br>over DS3/T3,<br>channelized to DS1 | `Router(config)# **interface serial** <slot>/<subslot>/<port>.<T3#>/<ds1#>:[chan]` |
| STS-1,<br>over subrated DS3/T3 | `Router(config)# **interface serial** <slot>/<subslot>/<port>.<T3#>` |
| STS-1,<br>over VT1.5,<br>channelized to DS1 | `Router(config)# **interface serial** <slot>/<subslot>/<port>.<VT#>/<ds1#>:[chan]` |
| STS-1,<br>over VT2,<br>channelized to E1 | `Router(config)# **interface serial** <slot>/<subslot>/<port>.<VT#>/<E1#>:[chan]` |

# SDH Channelization

The 1-Port Channelized OC-12/STM-4 line card supports the following SDH channelization modes:

- STM-1 > AU-3 > DS3/T3
- STM-1 > AU-3 > DS3/T3 > DS3 subrate
- 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

Table 6-3 shows serial interface examples for SDH channelizations on a previously-configured SDH-framed controller on the 1-Port Channelized OC-12/STM-4 line card.

*Table 6-3 SDH Interface Examples for an SDH-Framed Controller*

| Interface Type | Syntax |
|---|---|
| STM-4,<br>over STM-1,<br>over AU-3,<br>over DS3/T3 | `Router(config)# **interface serial** <slot>/<subslot>/<port>.<T3#>` |
| STM-4,<br>over STM-1,<br>over AU-3,<br>over a subrated DS3/T3 | `Router(config)# **interface serial** <slot>/<subslot>/<port>.<T3#>` |
| STM-4,<br>over STM-1<br>over AU-3,<br>over TUG-2,<br>over C-11 (DS1/T1) | `Router(config)# **interface serial** <slot>/<subslot>/<port>.<AU-3#>/<TUG-2>/<C-11>:[chan]` |

*Table 6-3 SDH Interface Examples for an SDH-Framed Controller (continued)*

| Interface Type | Syntax |
|---|---|
| STM-4,<br>over STM-1<br>over AU-3,<br>over TUG-2,<br>over C-12 (E1) | `Router(config)#` **`interface serial`**<br>`<slot>/<subslot>/<port>.<AU-3#>/<TUG-2>/<C-12>:[chan]` |
| STM-4,<br>over STM-1<br>over AU-4,<br>over TUG-3,<br>over TUG-2,<br>over C-11 (DS1/T1) | `Router(config)#` **`interface serial`**<br>`<slot>/<subslot>/<port>.<AU-4#>/<TUG-3>/<TUG-2>/<C-11>:[chan` |
| STM-4,<br>over STM-1<br>over AU-4,<br>over TUG-3,<br>over TUG-2,<br>over C-12 (E1) | `Router(config)#` **`interface serial`**<br>`<slot>/<subslot>/<port>.<AU-4#>/<TUG-3>/<TUG-2>/<C-12>:[chan]` |

# Line Card Default Values

The 1-Port Channelized OC-12/STM-4 line card has two sets of command default values. The set of default values in use is determined by the **framing** command—either SONET framing or SDH framing.

The following tables contains command default values, and indicate which configuration mode you need to be in to enter a command. Where applicable, the tables also indicate which commands apply to which line types. The tables include the commands used for modifying a default value and indicates whether a value needs to be the same (or opposite) on the remote end of the connection.

## Line Card Default Values for SONET Framing

This section contains tables of command default values that apply when the 1-Port Channelized OC-12/STM-4 line card is configured for SONET framing.

Table 6-4 lists the default values for commands that are available in SONET controller configuration mode when the 1-Port Channelized OC-12/STM-4 line card is configured for SONET framing.

Table 6-5 lists the default values for commands that are available in T3 controller configuration mode when the 1-Port Channelized OC-12/STM-4 line card is configured for SONET framing.

Table 6-7 lists the default values for commands that are available in T1 controller configuration mode, under a T3 controller, when the 1-Port Channelized OC-12/STM-4 line card is configured for SONET framing.

Table 6-6 lists the default values for commands that are available in VT controller configuration mode when the 1-Port Channelized OC-12/STM-4 line card is configured for SONET framing.

*Table 6-4 Default Values for SONET Controller Configuration Commands under SONET Framing*

| Command Name | Default Setting | Command Syntax | Remote Side Setting |
|---|---|---|---|
| **SONET Controller Configuration Mode** | | | |
| clock source | line | **clock source** {**internal** \| **line**} | One side set to **internal** |
| framing | sonet | **framing** {**sonet** \| **sdh**} | Same |
| loopback | no loopback | **[no] loopback** {**internal** \| **line**} | — |
| overhead | 1 for J0<br>0 for S1S0 | **overhead** {**j0** \| **s1s0**} | — |

*Table 6-5 Default Values for T3 Controller Configuration Commands under SONET Framing*

| Command Name | Default Setting | Command Syntax | Remote Side Setting |
|---|---|---|---|
| **T3 Controller Configuration Mode** | | | |
| cablelength | 0 | **cablelength** *length* | — |
| channelized | channelized | [**no**] **channelized** | — |
| clock source[1] | internal | **clock source** {**line** \| **internal**} | One side set to **internal** |
| framing | auto-detect | **framing** {**auto-detect** \| **c-bit** \| **m23**} | Same |
| idle pattern | 0X7e (flags) | **idle pattern** {**0x0 to 0xFF**} | Same |
| overhead | 4 for c2<br>64 for j1 | **overhead** {**c2** \| **j1**} | — |

1. The default value for SONET clock source is **line**. At least one side of the connection must be set to **internal**.

*Table 6-6 Default Values for VT Controller Configuration Commands under SONET Framing*

| Command Name | Default Setting | Command Syntax | Remote Side Setting |
|---|---|---|---|
| **VT Controller Configuration Mode** | | | |
| overhead | 2 for c2<br>64 for j1 | **overhead** {**c2** \| **j1**} | — |

*Table 6-7 Default Values for T1 and E1 Configuration Commands under SONET Framing*

| Command Name | Default Setting | Command Syntax | Remote Side Setting |
|---|---|---|---|
| t1 clock source | internal | **t1** *ch_group_number* **clock source** {**line** \| **internal**} | One side set to **internal** |
| t1 framing | esf | **t1** *t1-number* **framing** {**esf** \| **sf**} | Same |
| e1 clock source | internal | **e1** *ch_group_number* **clock source** {**line** \|**internal**} | One side set to **internal** |
| e1 framing | crc4 | **e1** *e1-number* **framing** [**crc4** \| **no-crc4**] | Same |

# Line Card Default Values for SDH Framing

This section contains tables of command default values that apply when the 1-Port Channelized OC-12/STM-4 line card is configured for SDH framing.

Table 6-8 lists the default values for commands that are available in SONET controller configuration mode when the 1-Port Channelized OC-12/STM-4 line card is configured for SDH framing.

Table 6-9 lists the default values for commands that are available in AU-3 controller configuration mode for both AU-3 and AU-4-TUG-3 when the 1-Port Channelized OC-12/STM-4 line card is configured for SDH framing.

Table 6-10 lists the default values for T3 commands when the 1-Port Channelized OC-12/STM-4 line card is configured for SDH framing.

Table 6-11 lists default values for T1 and E1 commands when the 1-Port Channelized OC-12/STM-4 line card is configured for either SONET or SDH framing.

*Table 6-8 Default Values for SONET Controller Configuration Commands under SDH Framing*

| Command Name | Default Setting | Command Syntax | Remote Side Setting |
|---|---|---|---|
| **SONET Controller Configuration Mode under SDH** | | | |
| clock source[1] | line | **[no] clock source** {**internal**\|**line**} | One side set to **internal** |
| framing | sonet | **framing** {**sonet** \| **sdh**} | Same |
| loopback | no loopback | **[no] loopback** {**internal** \| **line**} | — |
| overhead | 1 for j0<br>2 for S1S0 | **overhead** {**j0** \| **s1s0**} | — |
| aug controller | no aug controller | **[no] aug controller** <au-3\|au-4-tug-3> | — |

1. The default value for SDH clock source is **line**. At least one side of the connection must be set to **internal**.

*Table 6-9 Default Values for AU-3/AU-4-TUG-3 Controller Configuration Commands for AU-3 and AU-4-TUG-3 under SDH Framing*

| Command Name | Default Setting | Command Syntax | Remote Side Setting |
|---|---|---|---|
| **AU-3/AU-4-TUG-3 Controller Configuration Mode under SDH** | | | |
| clock source | internal | **clock source** {**line** \| **internal**} | One side set to **internal** |
| overhead | 0 for c2<br>16 for j1 | **overhead** {**c2** \| **j1**} | — |

*Table 6-10 Default Values for T3 Configuration Commands under SDH Framing*

| Command Name | Default Setting | Command Syntax | Remote Side Setting |
|---|---|---|---|
| bandwidth | | **bandwidth** *bandwidth* | — |
| crc | 16 | **crc** {**16** \| **32**} | — |
| encapsulation | HDLC | **encapsulation** {**frame-relay** \| **hdlc** \| **ppp**} | Same |
| idle-character | flags (0x7e) | **idle-character** {**flags** \| **marks**} | |
| keepalive | 10 seconds | **keepalive** *period* | — |

*Table 6-11 Default Values for T1 and E1 Interface Configuration Commands*

| Command Name | Default Setting | Command Syntax | Remote Side Setting |
|---|---|---|---|
| **T1 Interface Configuration Mode** | | | |
| crc | 16 | [**no**] **crc** [**16** \| **32**] | Same |
| DSU bandwidth | 44210 | **dsu bandwidth** *bandwidth* | Same |
| DSU mode | cisco | **dsu mode** *mode* | Same |
| encapsulation | HDLC | **encapsulation** [**hdlc** \| **ppp**] | Same |
| framing | c-bit | **framing** [**c-bit** \| **m13**] | Same |
| idle character | flags (0x7e) | **idle character** [**flags** \| **marks**] | Same |
| keepalive | 10 seconds | **keepalive** *seconds* | Same |
| mtu | 4470 | **mtu** *size* | Same |
| **E1 Interface Configuration Mode** | | | |
| crc | 16 | [**no**] **crc** [**16** \| **32**] | Same |
| encapsulation | HDLC | **encapsulation** [**hdlc** \| **ppp**] | Same |
| framing | c-bit | **framing** [**c-bit** \| **m13**] | Same |
| idle character | flags (0x7e) | **idle character** [**flags** \| **marks**] | Same |
| keepalive | 10 seconds | **keepalive** | Same |
| mtu | 4470 | **mtu** *size* | Same |

# Line Card Interface Syntax

To specify an unchannelized or 1-Port Channelized OC-12/STM-4 line card interface in a configuration command, use the syntax shown in Table 6-12.

*Table 6-12 Interface Syntax for the 1-Port Channelized OC-12/STM-4 Line Card*

| Type of Interface | Slot | Sub-slot | Port | STS-1 Path (T3) | T1 or E1 Number | Channel Group Number |
|---|---|---|---|---|---|---|
| Unchannelized | 1 to 8/ | 0/ | 0. | 1 to 12 | — | — |
| Channelized | 1 to 8/ | 0/ | 0. | 1 to 12/ | 1 to 28: for T1<br>1 to 32: for E1 | 0 to 23 for T1<br>0 to 31 for E1 |

Examples:

- Modifying T1 interface 6 in controller configuration mode:

```
Router(config)# controller t3 2/0/0.1
Router(config-controller)# t1 6 command
```

- Modifying T1 interface 6, channel group number 8 in interface configuration mode:

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

# SONET-Framed Interface Configuration Sample

You can configure up to 12 STS-1 connections on a 1-Port Channelized OC-12/STM-4 line card. Each STS-1 connection can be configured as a T3 controller or as a VT controller.

The following procedure walks you through the basic steps for creating full-rate and subrate T3 interfaces, as well as T1 and fractional T1 interfaces. Each T3 controller can be configured as a single T3 interface (full or subrate), as 28 T1 interfaces, or as an even larger number of fractional T1s.

**Step 1** Set the framing type to SONET using the framing command.

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

**Step 2** Assign an STS-1 path to a T3 interface. In this example, the path command is used to set up four T3 interfaces:

```
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
```

**Step 3** Create an interface. In the following examples, each type of interface is created in a different T3 controller (4/0/0.1 through 4/0/0.4).

#### Full-Rate T3 Interface

a. Enter controller configuration mode.

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

b. To create a full-rate T3 interface, you must eliminate the T1 interfaces by entering the **no channelized** command.

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

c. Enter interface configuration mode.

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

d. Go to Step 4.

#### Subrate T3 Interface

a. Enter controller configuration mode.

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

b. To create a subrate T3 interface, first create a full-rate one.

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

c. Enter interface configuration mode, where you can use the **dsu bandwidth** command to create a subrate T3 interface. In this example, a subrate T3 interface is created that has a bandwidth of 16000 kbps.

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

d. Go to Step 4.

#### Channelized T3 with a Full T1 Interface

a. Enter controller configuration mode.

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

b. Use the **t1 channel group** command to create a T1 interface. In the following example, T1 interface 1 (of 28) is defined as being made up of a single channel group, number 20 (any number between 0 and 23). This channel group consists of all 24 DS0 timeslots.

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

c. Go to interface configuration mode for the channel group you just created.

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

d. Go to Step 4.

**Channelized T3 with a Fractional T1 Interface**

**a.** Enter controller configuration mode.

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

**b.** Use the **t1 channel group** command to create fractional T1 interfaces. In the following example, T1 interface 3 (of 28) is defined as being made up of three channel groups, numbers 19, 20, and 21 (any numbers between 0 and 23). The channel groups consist of a total of 24 DS0 timeslots. Each channel group represents a separate interface.

```
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.** Go to interface configuration mode for one of the channel groups. For example:

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

**d.** Go to Step 4.

**Step 4** Enter the encapsulation method. This example shows the command for using Frame Relay encapsulation. You can also choose PPP or HDLC.

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

**Step 5** If IP routing is enabled on the system, you can assign an IP address and subnet mask. For example:

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

**Step 6** Add any configuration subcommands required to enable routing protocols and set the interface line characteristics.

**Step 7** Change the shutdown state to up, which enables the interface.

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

**Step 8** When you have entered all of the configuration subcommands to complete the configuration, press **Ctrl-Z** to exit configuration mode.

**Step 9** Write the new configuration to NVRAM.

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

After you create an interface configuration, you can modify it at any time by using the appropriate Cisco IOS configuration commands.

# SDH-Framed Interface Configuration Sample

If the 1-Port Channelized OC-12/STM-4 line card is configured for SDH framing, you can configure up to:

- 12 STM-0 controllers. Each STM-0 controller can be configured as one AU-3 controller, each of which can be configured a T3 interface or as 28 T1 interfaces or 21 E1 interfaces. These interfaces can be channelized into fractional T1s or E1s.
- 4 STM-1 controllers. Each STM-1 controller can be configured with three AU-4-TUG-3 controllers, each of which can be configured as 28 T1 interfaces or up to 21 E1 interfaces. These interfaces can be channelized into fractional T1s or E1s.

**Note** AU-3 and AU-4-TUG-3s are based on the higher order path, and do not operate independently, but operate in groups of three. If you shutdown any AU-3 or AU-4-TUG-3 controller, the other two controllers on the same SONET port are also shut down.

The SONET **controller** command specifies the mode, and creates three [sub]controllers of that type, which you can configure like any other controller.

The format created adds a subport extension to the controller SONET name. The form of the subport extension is different for AU-3 and AU-4-TUG-3 controllers as the following examples show:

### AU-3 Controller Syntax

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

For example, the **controller** command for AU-3 #3 on port 0 of a 4 -port channelized STM-1 line card in slot 2 is:

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

### AU-4-TUG-3 Controller Syntax

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

For example, the command for a controller created for TUG-3 #3 on AU-4 of port 0 of a 1-Port Channelized OC-12/STM-4 line card in slot 2 is:

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

## Show Controller Command Syntax

To display the settings for specific on AU-3 and AU-4-TUG-3 controllers, use the **show controller** command.

For AU-3:

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

For AU-4-TUG-3:

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

Where:

- *slot* is 1 to 8
- *subslot* is 0
- *port* is always 0
- *AU-3#* is 1 to 12
- *AU-4#* is always 1
- *TUG-3#* is 1 to 12
- *tug-2* is 1 to 7
- *t1* is 1 to 4 or *e1* is 1 to 3

**Note** The [/<tug-2>/<t1>] options are only available if a mode (C-11 or C-12) is configured on the controller.

For example, for T1 over AU-3, the command to display the settings for AU-3 #3 for a line card in slot 2 is:

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

For AU-4-TUG-3, the command to display the settings for TUG-3 #3 on AU-4 of a 1-Port Channelized OC-12/STM-4 line card in slot 2 is:

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

To display a specific TUG-2 and E1 on the same AU-4-TUG-3, use the command:

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

# AU-3 Controller Interfaces

Use the following procedure to create T3 and T1 interfaces using AU-3 controllers.

**Step 1** Enter SONET controller configuration mode from global configuration mode.

In this example, the 1-Port Channelized OC-12/STM-4 line card is in slot 1.

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

**Step 2** Use the **aug controller** command to specify the mode of operation for an STM-4 port (for an AU-3, twelve controllers are created: `2/0/0.1 to 2/0/0.12`).

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

### Full-Rate T3 Interface

**Note** T3 lines can only operate over AU-3 controllers.

**a.** Configure the controller on the port.

This example shows AU-3 controller #3 configured for a line card in slot 2.

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

**b.** Set the container mode to C-3 to establish a T3 interface.

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

C-3 mode creates a serial interface at the unchannelized T3 level.

**T1 Channel Group Interface**

**a.** Configure the controller on the single OC-12 port.

This example shows AU-3 controller #2 configured on the OC-12 port for a line card in slot 2.

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

**b.** Set the container mode to C-11 to establish a T1 interface.

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

**c.** Create a T1 interface.

In the following example, the TUG-2 interface is defined as 1 (any number between 1 and 7), the T1 interface is 2 (any number between 1 and 4), and is made up of a single channel group, 5 (any number between 0 and 23). This channel group consists of all 24 DS0 timeslots.

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

# AU-4-TUG-3 Controller Interfaces

Use the following procedure to create T1 and E1 interfaces using AU-4-TUG-3 controllers.

**Caution** The three AU-4-TUG-3 controllers created on a port are based on the same higher order path and do not operate independently from one another. In other words, when you "shutdown" any AU-4-TUG-3 controller, the other two controllers on the same SONET port are also shut down.

**Step 1** Enter SONET controller-configuration mode.

In this example, the line card is in slot 2.

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

**Step 2** Use the **aug controller** command to specify the mode of operation for the OC-12 port (for an AU-4, three controllers are created).

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

**T1 Channel Group Interface over AU-4-TUG-3/TUG-2**

**a.** Configure the controller on an STM-4 port. This example shows AU-4-TUG-3 controller #2 configured on port 0for a line card in slot 2.

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

**b.** Set the container mode to C-11 to establish a T1 interface.

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

**c.** Create a T1 interface.

In the following example, the TUG-2 interface is 1 (any number between 1 and 7), the T1 interface is 2 (any number between 1 and 4,) and is defined as a single channel group, 5 (any number between 0 and 23). This channel group consists of all 24 DS0 timeslots.

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

#### E1 Channel Group Interface over AU-4-TUG-3/TUG-2

**a.** Configure the controller on an STM-4 port.

This example shows AU-4-TUG-3 controller #2 configured on port 0 of a line card in slot 2.

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

**b.** Set the container mode to C-12 to establish an E1 interface.

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

**c.** Create an E1 interface.

In the following example, the TUG-2 interface is 1 (any number between 1 and 7), the E1 interface is 2 (any number between 1 and 3), and is defined as a single channel group, 5 (any number between 0 and 31). This channel group consists of all 31 DS0 time slots.

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

# Configuring the SONET Controller for SONET or SDH Framing

By default, the 1-Port Channelized OC-12/STM-4 line card consists of 12 STS-1 connections. To set up the 1-Port Channelized OC-12/STM-4 line card, you must configure each STS-1 as a T3 or VT interface. This section describes the commands used to create T3 and VT interfaces and test STS-1 connections. This section describes the following:

- Entering Controller-Configuration Mode, page 6-14
- Selecting SONET or SDH Framing, page 6-15
- Configuring the SONET Controller Loopback Mode, page 6-15

## Entering Controller-Configuration Mode

Use the **controller sonet** command to enter controller-configuration mode, where you can configure the controller:

**controller sonet** *slot/subslot/port*

Where *slot* is 1 to 8, and *subslot* and *port* are both 0.

The following example shows you how to enter SONET controller configuration mode for a card in slot 2:

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

## Selecting SONET or SDH Framing

To configure the 1-Port Channelized OC-12/STM-4 line card for SONET, you must first set the framing type to either SONET or SDH using the **framing** command.

If the 1-Port Channelized OC-12/STM-4 line card framing was previously configured for either SONET or SDH, and you wish to change the framing, you must enter the **no framing** command to return the line card to the default values. Then use the **framing** <*framing type*> command to set the new framing.

**Caution** If you enter the **no framing** command, the line card configuration is erased, and all default line card values are restored. Use the command with caution.

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

The default is SONET.

Use the **no** form of the command to remove framing and return the line card to the default values.

The following example shows you how to specify SONET framing for the 1-Port Channelized OC-12/STM-4 line card in slot 2:

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

## Configuring the SONET Controller Loopback Mode

You can enable SONET controller loopbacks by using the **loopback** command.

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

The default is no loopback.

Use the **no** form of the command to stop a loopback.

For more information on the **loopback** command, refer to the online *Cisco 10000 Series ESR Troubleshooting Guide*.

The commands in this example run an internal loopback on the 1-Port Channelized OC-12/STM-4 line card in slot 2:

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

# Creating a T3, VT, or AUG Controller

This section show you how to create and configure T3, VT, or AUG controllers, and contains the following sections:

- Designating an STS-1 Path as a T3 or VT under SONET Framing, page 6-16
- Designating an STM-4 Port as an AU-3 or AU-4-TUG-3 Controller under SDH Framing, page 6-16
- Configuring the SONET Controller Loopback Mode, page 6-15
- VT Commands under SONET Framing, page 6-17

- Unchannelized T3 Commands under SONET or SDH Framing, page 6-17
- Channelized T3 Commands under SONET or SDH Framing, page 6-21
- Creating T1 or E1 Channel Groups under SONET or SDH Framing, page 6-26
- Channelized T1 Commands under SONET or SDH Framing, page 6-29


# Designating an STS-1 Path as a T3 or VT under SONET Framing

When the 1-Port Channelized OC-12/STM-4 line card is configured for SONET framing, you can use the **path** command to designate an STS-1 path as a T3 or VT controller. You can designate up to 12 STS-1 channels. (Figure 6-1 on page 6-22 shows the relationship between the OC-12, T3 interfaces, VT interfaces, and T1 interfaces.

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

Where *STS_number* is the virtual T3 or VT controller. Enter a value from 1 to 12.

Use the **no** form of the command to remove a T3 controller.

In the following example, STS-1 number 1 is defined as a T3 line and controller T3 number 2/0/0.1 is generated:

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

In the next example, STS-1 number 1 is defined as a VT line and a VT controller 2/0/0.1 is generated:

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

# Designating an STM-4 Port as an AU-3 or AU-4-TUG-3 Controller under SDH Framing

When the 1-Port Channelized OC-12/STM-4 line card is configured for SDH framing, you can use the **aug controller** command to specify the mode of operation for an STM-4 port as an AU-3 controller or an AU-4-TUG-3 controller. This command creates 12 subcontrollers of the designated type, which you can then configure like any other controller.

**[no] aug controller** *<au-3|au-4-tug-3>*

Where *au-3* and *au-4-tug-3* designate the type of controller. Enter a value from 1 to 12.

Use the **no** form of the command to remove an AU-3 or AU-4-TUG-3 controller.

The format of the name for these controllers requires you to add a subport extension to the SONET controller name. The subport extension is different for AU-3 and AU-4-TUG-3 controllers.

### AU-3 Controller

For an AU-3 controller, the syntax is:

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

For example, the following command Uses AU-3 controller #3 on port 0 of a line card in slot 2.

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

#### AU-4-TUG-3 Controller

For an AU-4-TUG-3 controller, the syntax is:

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

> **Note** The twelve AU-4-TUG-3 controllers created on an SDH-framed SONET port are based on the same higher order path and do not operate independently from one another. In other words, when you "shutdown" any AU-4-TUG-3 controller, the other two controllers on the same SDH-framed SONET port are also shut down.

For example, the following command uses TUG-3 #3 on AU-4 of port 0 of an 1-Port Channelized OC-12/STM-4 line card in slot 2.

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

# VT Commands under SONET Framing

This section describes how to use the **controller vt** command to further channelize the SONET-framed STS-1 controller of the1-Port Channelized OC-12/STM-4 line card.

## Entering Controller Configuration Mode for VT

A virtual tributary (VT) controller on a 1-Port Channelized OC-12/STM-4 line card can be channelized into 28 T1 (VT1.5) interfaces or 21 E1 (VT2) interfaces. You can use the **controller vt** command to shut down a VT link or to change the settings for a T1 or E1 interface.

For information about T1 configuration settings, see "Creating T1 or E1 Channel Groups under SONET or SDH Framing" section on page 6-26, and the "Channelized T1 Commands under SONET or SDH Framing" section on page 6-29.

You can configure a VT link by entering the **controller vt** command.

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

Where *path* is a value from 1 to 12. Each number represents a VT that houses 28 T1 lines or 21 E1 lines.

For example:

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

# Unchannelized T3 Commands under SONET or SDH Framing

This section shows you how to configure an unchannelized T3 on a SONET-framed or SDH-framed controller.

If the 1-Port Channelized OC-12/STM-4 line card is configured for SONET framing, a T3 interface is channelized into 28 T1 interfaces by default. You must unchannelize the T3 interface to create a full-rate or subrate T3 interface.

If the 1-Port Channelized OC-12/STM-4 line card is configured for SDH framing, C-3 container mode automatically creates an unchannelized T3 serial interface.

This section describes the commands you use to create, customize, and test full-rate and subrate T3 interfaces. This section describes the following:


- Entering Controller Configuration Mode for T3, page 6-18
- Configuring a T3 Interface on a SONET-Framed Controller as Unchannelized, page 6-18
- Implementing Subrate T3, page 6-19
- Setting a Framing Type, page 6-19
- Specifying a DSU Mode, page 6-19
- Enabling Scrambling, page 6-20
- Specifying the Idle Character, page 6-20
- Running a T3 BER Test under SONET or SDH Framing, page 6-20


You can also use the following commands, described in the "Channelized T3 Commands under SONET or SDH Framing" section on page 6-21, when you configure unchannelized T3: **clock source, mdl, equipment**, and **loopback**.

## Entering Controller Configuration Mode for T3

To create an unchannelized T3 interface, you must first enter controller configuration mode for the T3 controller you want to configure.

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

Where *path* specifies the T3 interface number.

The following example shows how to enter controller configuration mode:

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

## Configuring a T3 Interface on a SONET-Framed Controller as Unchannelized

If the 1-Port Channelized OC-12/STM-4 line card is configured for SONET framing, you must unchannelize the T3 interface to create a full-rate or subrate T3 interface.

You can configure the T3 interface as unchannelized (clear channel) by entering the **no channelized** command.

[**no**] **channelized**

**Caution** The **no channelized** command removes all channel groups from a channelized T3 interface. If you have already configured channel groups, use this command with caution.

Use the **channelized** command to return the interface to its default. The default value for a T3 interface is channelized.

In the following example, an unchannelized T3 interface is created:

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

## Implementing Subrate T3

You can implement subrate T3 by specifying the bandwidth for an unchannelized T3 interface. To do so, use the **dsu bandwidth** *bandwidth* command from interface configuration mode.

**dsu bandwidth** *bandwidth*
[**no**] **bandwidth** *bandwidth*

Where *bandwidth* is a numeric value between 0 and 44,210 kbps.

The default bandwidth is 44,210 kbps.

To return to the default bandwidth, use the **no** form of this command.

When you specify a value, the software sets the bandwidth to the closest acceptable bandwidth based on the timeslot size for the current DSU mode.

For you to properly use this command, the remote side of the connection must have a Cisco 7200 router or Cisco 7500 router with a PA-T3 or PA-2T3 port adapter or a T3 DSU supported by the **dsu mode** command.

In the following example, a bandwidth of 16,000 kbps is specified:

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

## Setting a Framing Type

To specify a framing type for the unchannelized T3 controller, use the **framing** command.

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

The default framing type is c-bit.

Use the **no** form of this command to restore the default framing type.

In the following example, framing is set to m13:

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

## Specifying a DSU Mode

To specify a DSU mode for a selected T3 interface, use the **dsu mode** command from interface configuration mode. This command configures the line card to emulate a manufacturer's proprietary multiplexing scheme.

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

The default DSU mode is cisco.

Use the **no** form of the command to return the DSU mode to its default.

In the following example, the DSU mode is set to **cisco**:

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

## Enabling Scrambling

To enable scrambling on an unchannelized T3 interface, use the **scramble** command from interface configuration mode.

```
scramble
[no] scramble
```

The default setting for this command is no scramble (scrambling disabled).

Ensure that both sides of the link have the same scrambling setting.

In the following example, scrambling is enabled on the specified T3 interface:

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

## Specifying the Idle Character

To set a specific character on the unchannelized T3 interface to be transmitted between HDLC packets, use the **idle character** command from interface configuration mode.

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

Where:

- **flags** sets an idle character of 0x7e
- **marks** sets an idle character of all 0xff

The default idle character is 0x7e.

Use the **no** form of the command to return the idle character to its default.

In the following example, the idle character is set to flags:

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

**Note** Some systems interpret marks, 0xff, as an abort signal. Therefore, flags, 0x7e, is preferred.

## Running a T3 BER Test under SONET or SDH Framing

You can configure an unchannelized T3 interface to run a bit error rate (BER) test. The test is used to check cables and solve signal problems in the field. To send a BER test pattern on an unchannelized T3 interface, use the following interface configuration command:

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

Where:

- **errors** *number* is 1 to 255
- **pattern** *pattern* is
  - 0s—repetitive test pattern of all zeros (00000..)
  - 1s—repetitive test pattern of all ones (11111..)
  - 2^15—pseudorandom O.151 test pattern (32,768 bits in length)

  - 2^20-O153—pseudorandom O.153 test pattern (1,048,575 bits in length)
  - QRSS-2^20—pseudorandom QRSS O.151 test pattern (1,048,575 bits in length)
  - 2^23—pseudorandom O.151 test pattern (8,388,607 bits in length)
- **interval** *time* is 1 to 1440 minutes

You can terminate a BER test at any time using the **no bert** command.

For more information, refer to the online *Cisco 10000 Series Internet Router Troubleshooting Guide*.

Examples:

- Send a BER test pseudorandom pattern of 2^20 through T3 line 1 for 5 minutes.

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

- Send a repetitive pattern of all 1s through T3 line 1 for 1440 minutes.

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

# Channelized T3 Commands under SONET or SDH Framing

This section shows you how to configure a channelized T3 on a SONET-framed or SDH-framed controller.

By default, a T3 interface on a 1-Port Channelized OC-12/STM-4 line card is channelized into 28 T1 interfaces. This section describes the commands you use to customize and test a channelized T3 interface. This section describes procedures for



To use the interface for subscriber traffic, you must configure its T1 and DS0 components. For more information, see the "Creating T1 or E1 Channel Groups under SONET or SDH Framing" section on page 6-26.

Figure 6-1 illustrates the levels of configurable interface bandwidth that channelization offers.

## Entering Controller Configuration Mode for T3

You may need to enter controller configuration mode for the T3 interface. For example, enter this mode to change an unchannelized T3 interface back to a channelized one or to set MDL messages.

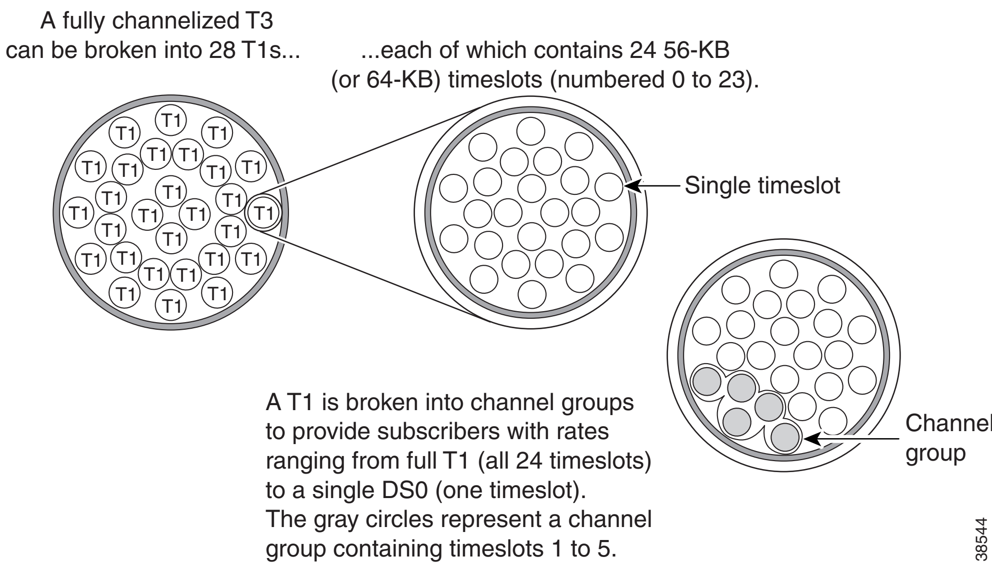
```
controller T3 slot/sub-slot/port.path
```

*Path* specifies the T3 interface number.

The following example shows how to enter controller configuration mode:

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

***Figure 6-1 Channelization of T3s***




## Configuring a T3 Interface as Channelized

A T3 interface is channelized by default. Use the **channelized** command if you had previously made the interface unchannelized.

[**no**] **channelized**
**channelized**

**Caution** The **no channelized** command removes all channel groups from a channelized T3 interface. If you have already configured channel groups, use this command with caution.

The following example shows how to create a channelized T3 interface:

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

## Setting the Framing Type

To specify a framing type for the channelized T3 controller, use the **framing** command.

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

The default is auto-detect.

Use the **no** form of the command to return the framing type to its default.

You can instruct the 1-Port Channelized OC-12/STM-4 line card to detect the framing type from the far end and transmit that same framing type:

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

## Entering MDL Messages

You can configure maintenance data link (MDL) messages (as defined in the ANSI T1.107a-1990 specification) on the channelized T3 interface.

**Note** MDL messages are supported only when the T3 framing is set for C-bit parity. (See the "Setting the Framing Type" section on page 6-22.)

To configure MDL messages, use the **mdl** command:

**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*}

Where:

- **transmit path** enables transmission of the MDL path message
- **transmit idle-signal** enables transmission of the MDL idle-signal message
- **eic** is the equipment identification code (up to 10 characters)
- **lic** is the location identification code (up to 11 characters)
- **fic** is the frame identification code (up to 10 characters)
- **unit** is the unit identification code (up to 6 characters)
- **pfi** is the facility identification code to include in the MDL path message (up to 38 characters)
- **port** is the equipment port (which initiates the idle signal) to include in the MDL idle signal message (up to 38 characters)
- **generator** is the generator number to include in the MDL test signal message (up to 38 characters)

The default is that no MDL message is configured.

Use the **no** form of the command to remove an MDL message.

Examples of configuring MDL messages follow:

- Enable the MDL path message transmission.

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

- Enable the MDL idle signal message transmission.

```
Router(config-controller)# mdl transmit idle-signal
```

- Enable the MDL test signal message transmission.

```
Router(config-controller)# mdl transmit test-signal
```

- Enter the equipment identification code.

```
Router(config-controller)# mdl string eic router A
```

- Enter the location identification code.

  `Router(config-controller)# mdl string lic test network`

- Enter the frame identification code.

  `Router(config-controller)# mdl string fic building b`

- Enter the unit identification code.

  `Router(config-controller)# mdl string unit abc`

- Enter the facility identification code.

  `Router(config-controller)# mdl string pfi string`

- Enter the port number to send in the MDL idle signal message.

  `Router(config-controller)# mdl string port string`

- Enter the generator number to send in the MDL test signal message.

  `Router(config-controller)# mdl string generator string`

## Specifying the Idle Pattern

You can set a specific pattern to be transmitted between HDLC packets on all unconfigured timeslot interfaces that belong to a channelized T3 interface. To do so, use the **idle pattern** command.

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

Where *patterns* is a number in the range of 0x0 to 0xff (hexadecimal) or 0 to 255 (decimal). You can enter this value in either hexadecimal or decimal form. Values of 0 to 254 set the idle pattern to HDLC flags (0x7e); a value of 255 sets the pattern to 0xff (all ones).

**Note** Some systems interpret marks, 0xff, as an abort signal. Therefore, flags, 0x7e, is preferred.

The default idle pattern is 0x7e.

Use the **no** form of the command to return the idle pattern to its default value.

Examples:

- Set a hexadecimal idle pattern.

```
Router(config)# controller T3 1/0/0.1
Router(config-controller)# idle pattern 0x10
```

- Set a decimal idle pattern.

```
Router(config)# controller T3 1/0/0.1
Router(config-controller)# idle pattern 23
```

## Setting the T3 Clock Source

At the prompt, set the internal or line clock source for the selected T3 controller using the **clock source** command. This command is set in controller configuration mode.

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

Where:

- **internal** specifies that the internal clock source is used
- **line** specifies that the network clock source is used

The default is clock source internal.

In this example, a T3 controller is instructed to use a line clock source:

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

## Configuring the Loopback Mode for a T3 Controller

You can configure the loopback modes for a T3 controller by using the **loopback** command:

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

The default is no loopback.

To disable loopback on the T3 controller, use the **no** form of the command.

Examples:

- Configure a T3 controller for local loopback.

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

  Local loopback simultaneously loops all channels toward the router and transmits a T3 AIS to the network. You can use local loopback to diagnose problems with the port when the port is isolated from the network cables.

- Configure a T3 port for network loopback.

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

  Network loopback loops the T3 line back towards the network and can be used to diagnose problems with cables from the central switching office to the port.

- Configure a T3 port for remote loopback.

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

  Remote loopback sends a command to loop the T3 line at the far end (central office). It can be used to diagnose problems with cables from the port adapter to the switching office.

## Configuring a T3 Controller to Respond to Remote Loopback Commands

Use the **equipment loopback** command to run loopbacks in conjunction with remote equipment.

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

Where:

- **customer** allows a port to respond to loopback commands from remote T3 equipment.
- **network** causes a controller to ignore remote T3 loopback commands.

Example:

To enable the controller's ability to respond to remote loopback requests, type:

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

To prevent a controller from responding to remote loopback commands, type:

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

**Note** Remote loopbacks are only available when you use c-bit parity framing.

For more information on the **loopback** command, refer to the online *Cisco 10000 Series Internet Router Troubleshooting Guide*.

# Creating T1 or E1 Channel Groups under SONET or SDH Framing

To create an interface that supports subscriber traffic, you must combine the timeslots into channel groups. To help you configure channel groups, consider the following T1 and E1 characteristics:

### For SONET-Framed T3 Controllers

- 1 channelized T3 is divided into 28 T1s
- 1 T1 is divided into 24 timeslots (or DS0s)

### For SONET-Framed VT Controllers

For T1:

- 1 VT controller is divided into 28 T1s (VT1.5)
- 1 T1 is divided into 24 timeslots (or DS0s)

For E1:

- 1 VT controller is divided into 21 E1s (VT2)
- 1 E1 can be divided into 31 timeslots (or DS0s)

### For SDH-Framed T3 Controllers

- 1 T3 is divided into 7 TUG-2s
- 1 TUG-2 is divided into either:
  - 4 T1s that can each be divided into 24 timeslots (or DS0s)
  - 3 E1s that can be divided into 31 timeslots (or DS0s)

## Creating Channel Groups for T1 SONET-Framed Interfaces

Use the **t1 channel-group** command to create a channel group for a SONET-framed interface.

Figure 6-1 illustrates channelization possibilities on the 1-Port Channelized OC-12/STM-4 line card.

You create a logical channel group using the following command:

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

Where:

- *t1-number* is T1 interface number 1 to 28.
- **channel-group** *channel-group-number* identifies the channel group with any number from 0 to 23.
- **timeslots** *list-of-timeslots* can be 1 to 24 or a combination of subranges within the 1 to 24 interval. You can indicate a range using a hyphen, commas, or a combination of both. One timeslot equals one DS0. (See the examples below.)
- **speed**{**56** | **64**} is an optional argument that specifies the speed of a timeslot as either 56 or 64 kbps. The default is 64. (The 56 kbps timeslots are generally used with older T1 equipment that does not support B8ZS line coding and are associated with SF framing.)

Use the **no** form of the command to remove a logical channel group.

The following examples show how to use the **t1 channel-group** command:

- In this example, T1 interface 3 includes channel group 20 and consists of 9 channelized timeslots:

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

  To enter interface configuration mode for T1 interface 3 channel group 20, enter the following:

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

- In this example, t1 interface 4 includes channel group 18 and consists of all 24 timeslots, creating a full T1 interface:

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

  To enter interface configuration mode for T1 interface 4 channel group 18, enter the following:

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

- In the following example, T1 interface number 5 is divided into three channel groups, which total 11 timeslots:

```
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
```

- In the following example, channel group 20 on T1 1 is removed:

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

## Creating Channel-Groups for SDH-Framed Interfaces

### Channel Limitations

There is a limit of 128 channels per SDH controller, and a maximum of 192 channels for the single OC-12 port on the 1-Port Channelized OC-12/STM-4 line card. This limitation applies to both T1 and E1 interfaces when you create channel groups for controllers.

For example, if you created 128 channels on T1 controller #1 on port 2, and 64 channels on T1 controller #2 on port 2, then you could not create any additional channels on T1 controller #3 or T1 controller #4 because you have reached the maximum limit of 192 channels for a single port (port 2 in this example).

To create a logical channel group, use one of the following commands:

**For T1:**

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

**For E1 (framed):**

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

**For E1 (unframed):**

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

Where:

- *tug-2 is 1 to 7*
- *t1#* is 1 to 4
- *e1#* is 1 to 3
- **channel-group** *channel#* identifies the channel group with any number from 0 to 23.
- **timeslots** *range* can be 1 to 24 for T1, or 1 to 31 for E1, or a combination of subranges within those intervals. You can indicate a range using a hyphen, commas, or a combination of both. One timeslot equals one DS0. (See the examples below.)
- **speed**{**56** | **64**} is an optional T1 argument that specifies the speed of a timeslot as either 56 or 64 kbps. The default is 64. (The 56 kbps timeslots are generally used with older T1 equipment that does not support B8ZS and are associated with SF framing.)

  **Note** The **speed** argument applies only to T1 channel groups and is not valid for E1.

- **unframed** creates an unframed E1 channel. This creates an interface with channel-group number 0.

  **Note** The unframed argument applies only to E1 channel groups and is not valid for T1.

Use the **no** form of the command to remove a logical channel group.

The following examples show how to use the **channel-group** command:

- In this example, T1 interface 3 includes channel group 20 and consists of 9 channelized timeslots:

```
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
```

  To enter interface configuration mode for T1 interface 3 channel group 20, enter the following:

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

- In this example, T1 interface 4 includes channel group 18 and consists of all 24 timeslots, creating a full T1 interface:

```
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
```

  To enter interface configuration mode for T1 interface 4 channel group 18, enter the following:

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

- In the following example, T1 interface number 5 is divided into three channel groups, which total 11 timeslots:

```
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
```

- In the following example, channel group 20 is removed:

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

# Channelized T1 Commands under SONET or SDH Framing

You can enter commands to modify aspects of a T1 interface from controller configuration mode for a T3 interface. This section describes the commands for


- Setting the Framing Format, page 6-29
- Controlling Yellow Alarms, page 6-30
- Setting the T1 Clock Source, page 6-30
- Configuring FDL, page 6-30
- Configuring a T1 BER Test, page 6-31
- Configuring Loopback Mode, page 6-31


After you configure a T1 interface, you can add encapsulation, routing, and other instructions by entering interface configuration mode. For example:

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

## Setting the Framing Format

You can specify the T1 interface framing format using the following command:

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

Where:

- *t1-number* is T1 interface 1 to 28
- **framing** is either extended super frame (ESF) or super frame (SF). You can set SF hdlc-idle to 0x7E or 0xFF
- **hdlc-idle** options allow you to set the idle pattern for the T1 interface to either 0x7e (the default) or 0xFF

The default framing format is extended super frame (ESF).

Use the **no** form of the command to return framing to its default value.

Examples:

- Set SF framing format for T1 interface 6.

```
Router(config)# controller T3 1/0/0.1
Router(config-controller)# t1 6 framing sf
```

- Set ESF framing format for T1 interface 16:

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

## Controlling Yellow Alarms

Use the **t1 yellow** command to turn the detection or generation of a yellow alarm on or off.

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

Where:

- *t1-number* is T1 interface 1 to 28
- **detection** means that the interface is told that it is failing by the remote device, causing Cisco IOS software to send a message to the console
- **generation** means that the interface notifies the remote device if it is failing, causing Cisco IOS software to send a message to the console

When you select SF framing for a full T1 interface (24 timeslots) that uses the default speed of 64, consider using the **no** *t1-number* **yellow detection** command to turn off yellow alarm detection, because the yellow alarm can be incorrectly detected with SF framing.

In the following example, T1 interface 1 is set to yellow detection:

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

## Setting the T1 Clock Source

You can set the internal or line (network) clock source for a T1 interface using the **controller** command.

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

Where:

- *t1-number* is T1 interface 1 to 28
- **internal** specifies that the internal clock source is used
- **line** specifies that the network clock source is used

The default is clock source internal.

In the following example, the interface is instructed to get its clock source from the line:

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

## Configuring FDL

You can use the **fdl ansi** command to enable 1-second transmissions of ANSI T1.403 performance reports through the facility data link (FDL), on both ends of the T1 connection.

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

Where *t1-number* is T1 interface 1 to 28.

Use the **no** form of the command to disable this feature.

**Note** You can use this command only when the T1 framing is ESF.

In the following example, FDL is enabled:

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

## Configuring a T1 BER Test

You can configure an individual T1 interface to run an independent BER test. The test is used to check cables and solve signal problems in the field. To send a BER test pattern on a T1 interface, use the following command:

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

Where:

- *t1-number* is T1 interface number 1 to 28
- *time* is 1 to 14,400 minutes
- *pattern* is
  - 2^11—pseudorandom test pattern (2048 bits in length)
  - 2^15—pseudorandom O.151 test pattern (32,768 bits in length)
  - 2^20-O153—pseudorandom O.153 test pattern (1,048,575 bits in length)
- **unframed** causes the BER test pattern to use the entire T1 bandwidth, including the T1 framing and payload bits. If **unframed** is omitted, the T1 is either SF or ESF framed as configured by the T1 n framing command, and the BER test pattern occupies only the T1 payload bits.

**Note** For each T3, you can run only one BER test at a time.

You can terminate a BER test at any time using the **no** form of the command.

For more information, refer to the online *Cisco 10000 Series Internet Router Troubleshooting Guide*.

Example:

- Send a BER test pseudorandom pattern of 2^15 through T1 interface 10 for 5 minutes.

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

## Configuring Loopback Mode

If a problem occurs when you configure a T1 interface, you can troubleshoot the line card by using the following command from controller configuration mode:

**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**]]]

Where:

- *t1-number* is T1 interface 1 to 28.
- **local** loops the router output data back toward the router at the T1 framer and sends an alarm indication signal (AIS) out toward the network.
- **network** {**line** | **payload**} loops the data back toward the network and automatically sets a local loopback at the HDLC controllers (line) or loops the payload data back toward the network and automatically sets a local loopback at the HDLC controllers (payload).
- **remote line fdl** {**ansi** | **bellcore**} sends a repeating, 16-bit ESF data link keyword to the remote end, requesting that it enter into a network line loopback. You can specify an ANSI or Bellcore keyword.
- **remote payload** [**fdl**] [**ansi**] sends a repeating, 16-bit ESF data link code word to the remote end, requesting entry into a network payload loopback. Using **fdl** and **ansi** enables the remote payload facility data link (FDL) ANSI bit loopback on the T1 channel.

Use the **no** form of the command to terminate a loopback.

For more information on this command, refer to the online *Cisco 10000 Series Internet Router Troubleshooting Guide*.

Examples:

- Configure the T3 controller for local loopback on T1 interface 1.

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

- Configure the T3 controller for remote FDL ANSI loopback on T1 interface 1.

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

CHAPTER 7

# 4-Port Channelized OC-3/STM-1 Line Card Configuration

This chapter describes the procedures for configuring the Cisco 10000 series 4-Port Channelized OC-3/STM-1 line card, hereafter known as the 4-Port Channelized OC-3/STM-1 line card.

The 4-Port Channelized OC-3/STM-1 line card uses fractional T1 (n x DS0), DS-1, E1, fractional E1 (n x DS0) and DS3 IP services to connect ISP customers to the network backbone. It provides full duplex operation at OC-3 bandwidth rates using a (single-mode) fiber optic port that provides the Cisco 10000 series router with a single interface that connects up to 768 individual channelized options.

This chapter contains the following sections:

# Software Support

Table 7-1 shows the minimum Cisco IOS release on each release train that supports the 4-Port Channelized OC-3/STM-1 line card.

***Table 7-1 4-Port Channelized OC-3/STM-1 Line Card Software Support***

| Required PRE | Minimum Cisco IOS Releases |
|---|---|
| PRE1 | Cisco IOS Release 12.0(15)SL and later releases of Cisco IOS 12.0SL<br>Cisco IOS Release 12.0(17)ST and later releases of Cisco IOS 12.0ST<br>Cisco IOS Release 12.0(22)S and later releases of Cisco IOS 12.0S<br>Cisco IOS Release 12.2(8)BZ and later releases of Cisco IOS 12.2BZ for SDH mode.<br>For both SONET and SDH mode, use Cisco IOS Release 12.0(23)S and later releases of Cisco IOS 12.0S. |
| PRE2 | Cisco IOS Release 12.3(7)XI and later releases of Cisco IOS 12.3XI<br>For both SONET and SDH mode, the use Cisco IOS Release 12.2(15)BX and later releases of Cisco IOS 12.2BX.<br>Cisco IOS Release 12.2(28)SB and later releases of Cisco IOS 12.2SB |

## Checking Hardware and Software Compatibility

The PRE installed in the Cisco 10000 series router chassis must support the Cisco IOS software running on the router. Use the **show version** command to check the PRE version installed.

To see if a feature is supported by a Cisco IOS release, to locate the software document for that feature, or to check the minimum software requirements of Cisco IOS software with the hardware installed on your router, Cisco maintains the Software Advisor tool on Cisco.com at http://www.cisco.com/cgi-bin/Support/CompNav/Index.pl

This tool does not verify whether line cards within a system are compatible, but does provide the minimum Cisco IOS requirements for individual hardware line cards, modules, or options.

You must be a registered user on Cisco.com to access this tool.

# SONET Channelization

The 4-Port Channelized OC-3/STM-1 line card supports the following SONET channelization modes:

- STS-1 > DS3/T3
- STS-1 > DS3/T3 > DS1
- STS-1 > DS3/T3 > DS3 subrate
- STS-1 > VT1.5 > DS1
- STS-1 > VT2 > E1

Table 7-2 shows serial interface examples for SONET channelizations on a previously-configured SONET-framed controller on the 4-Port Channelized OC-3/STM-1 line card.

*Table 7-2 SONET Interface Examples for a SONET-Framed Controller*

| Interface Type | Syntax |
|---|---|
| STS-1,<br>over DS3/T3 | Router(config)# **interface serial**<br><slot>/<subslot>/<port>.<T3#> |
| STS-1,<br>over DS3/T3,<br>channelized to DS1 | Router(config)# **interface serial**<br><slot>/<subslot>/<port>.<T3#>:[chan] |
| STS-1,<br>over subrated DS3/T3 | Router(config)# **interface serial**<br><slot>/<subslot>/<port>.<T3#> |
| STS-1,<br>over VT1.5,<br>channelized to DS1 | Router(config)# **interface serial**<br><slot>/<subslot>/<port>.<VT#>:[chan] |
| STS-1,<br>over VT2,<br>channelized to E1 | Router(config)# **interface serial**<br><slot>/<subslot>/<port>.<VT#>:[chan] |

# SDH Channelization

The 4-Port Channelized OC-3/STM-1 line card supports the following SDH channelization modes:

- STM-1 > AU-3 > DS3/T3
- STM-1 > AU-3 > DS3/T3 > DS3 subrate
- 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

Table 7-3 shows serial interface examples for SDH channelizations on a previously-configured SDH-framed controller on the 4-Port Channelized OC-3/STM-1 line card.

*Table 7-3 SDH Interface Examples for an SDH-Framed Controller*

| Interface Type | Syntax |
|---|---|
| STM-4,<br>over STM-1<br>over AU-3,<br>over DS3/T3 | Router(config)# **interface serial**<br><slot>/<subslot>/<port>.<AU-3#>/<T3#> |
| STM-4,<br>over STM-1<br>over AU-3,<br>over a subrated DS3/T3 | Router(config)# **interface serial**<br><slot>/<subslot>/<port>.<AU-3#>/<T3#> |

*Table 7-3 SDH Interface Examples for an SDH-Framed Controller (continued)*

| Interface Type | Syntax |
|---|---|
| STM-4,<br>over STM-1<br>over AU-3,<br>over TUG-2,<br>over C-11 (DS1/T1) | Router(config)# **interface serial**<br><slot>/<subslot>/<port>.<AU-3#>/<TUG-2>/<C-11>:[chan] |
| STM-4,<br>over STM-1<br>over AU-3,<br>over TUG-2,<br>over C-12 (E1) | Router(config)# **interface serial**<br><slot>/<subslot>/<port>.<AU-3#>/<TUG-2>/<C-12>:[chan] |
| STM-4,<br>over STM-1<br>over AU-4,<br>over TUG-3,<br>over TUG-2,<br>over C-11 (DS1/T1) | Router(config)# **interface serial**<br><slot>/<subslot>/<port>.<AU-4#>/<TUG-3>/<TUG-2>/<C-11>:[chan] |
| STM-4,<br>over STM-1<br>over AU-4,<br>over TUG-3,<br>over TUG-2,<br>over C-12 (E1) | Router(config)# **interface serial**<br><slot>/<subslot>/<port>.<AU-4#>/<TUG-3>/<TUG-2>/<C-12>:[chan] |

# Default Values

The 4-Port Channelized OC-3/STM-1 line card has two sets of command default values. The set of default values in use is determined by the **framing** command—either SONET framing or SDH framing.

The following tables contains command default values, and indicate which configuration mode you need to be in to enter a command. Where applicable, the tables also indicate which commands apply to which line types. The tables include the commands used for modifying a default value and indicates whether a value needs to be the same (or opposite) on the remote end of the connection.

## Default Values for SONET Framing

This section contains tables of command default values that apply when the 4-Port Channelized OC-3/STM-1 line card is configured for SONET framing.

Table 7-4 lists the default values for commands that are available in SONET controller configuration mode when the 4-Port Channelized OC-3/STM-1 line card is configured for SONET framing.

Table 7-5 lists the default values for commands that are available in T3 controller configuration mode when the 4-Port Channelized OC-3/STM-1 line card is configured for SONET framing.

Table 7-6 lists the default values for commands that are available in T1 controller configuration mode, under a T3 controller, when the 4-port STM-4 line card is configured for SONET framing.

Table 7-7 lists the default values for commands that are available in VT controller configuration mode when the 4-Port Channelized OC-3/STM-1 line card is configured for SONET framing.

*Table 7-4 Default Values for SONET Controller Configuration Commands under SONET Framing*

| Command Name | Default Setting | Command Syntax | Remote Side Setting |
|---|---|---|---|
| **SONET Controller Configuration Mode** | | | |
| clock source[1] | line | **clock source** {**internal** \| **line**} | One side set to **internal** |
| framing | sdh | **framing** {**sonet** \| **sdh**} | Same |
| loopback | no loopback | **[no] loopback** {**internal** \| **line**} | — |
| overhead | 1 for J0<br>0 for S1S0 | **overhead** {**j0** \| **s1s0**} | — |

1. The default value for SONET clock source is **line**. At least one side of the connection must be set to **internal**.

*Table 7-5 Default Values for T3 Controller Configuration Commands under SONET Framing*

| Command Name | Default Setting | Command Syntax | Remote Side Setting |
|---|---|---|---|
| **T3 Controller Configuration Mode** | | | |
| cablelength | 0 | **cablelength** *length* | — |
| channelized | channelized | [**no**] **channelized** | — |
| clock source | internal | **clock source** {**line** \| **internal**} | One side set to **internal** |
| framing | auto-detect | **framing** {**auto-detect** \| **c-bit** \| **m23**} | Same |
| idle pattern | 0X7e (flags) | **idle pattern** {**0x0 to 0xFF**} | Same |
| overhead | 4 for c2<br>64 for j1 | **overhead** {**c2** \| **j1**} | — |

*Table 7-6 Default Values for VT Controller Configuration Commands under SONET Framing*

| Command Name | Default Setting | Command Syntax | Remote Side Setting |
|---|---|---|---|
| **VT Controller Configuration Mode** | | | |
| overhead | 2 for c2<br>64 for j1 | **overhead** {**c2** \| **j1**} | — |

*Table 7-7 Default Values for T1 and E1 Configuration Commands under SONET Framing*

| Command Name | Default Setting | Command Syntax | Remote Side Setting |
|---|---|---|---|
| t1 clock source | internal | **t1** *ch_group_number* **clock source** {**line** \| **internal**} | One side set to **internal** |
| t1 framing | esf | **t1** *t1-number* **framing** {**esf** \| **sf**} | Same |
| e1 clock source | internal | **e1** *ch_group_number* **clock source** {**line** \|**internal**} | One side set to **internal** |
| e1 framing | crc4 | **e1** *e1-number* **framing** [**crc4** \| **no-crc4**] | Same |

# Default Values for SDH Framing

This section contains tables of command default values that apply when the 4-Port Channelized OC-3/STM-1 line card is configured for SDH framing.

Table 7-8 lists the default values for commands that are available in SONET controller configuration mode when the 4-Port Channelized OC-3/STM-1 line card is configured for SDH framing.

Table 7-9 lists the default values for commands that are available in AU-3 controller configuration mode for both AU-3 and AU-4-TUG-3 when the 4-Port Channelized OC-3/STM-1 line card is configured for SDH framing.

Table 7-10 lists the default values for T3 commands when the 4-Port Channelized OC-3/STM-1 line card is configured for SDH framing.

Table 7-11 lists default values for T1 and E1 commands when the 4-Port Channelized OC-3/STM-1 line card is configured for either SONET or SDH framing.

*Table 7-8 Default Values for SONET Controller Configuration Commands under SDH Framing*

| Command Name | Default Setting | Command Syntax | Remote Side Setting |
|---|---|---|---|
| **SONET Controller Configuration Mode under SDH** | | | |
| clock source[1] | line | **[no] clock source** {**internal**\|**line**} | One side set to **internal** |
| framing | sdh | **framing** {**sonet** \| **sdh**} | Same |
| loopback | no loopback | **[no] loopback** {**internal** \| **line**} | — |
| overhead | 1 for j0<br>2 for S1S0 | **overhead** {**j0** \| **s1s0**} | — |
| aug controller | no aug controller | **[no] aug controller** <au-3\|au-4-tug-3> | — |

1. The default value for SDH clock source is **line**. At least one side of the connection must be set to **internal**.

*Table 7-9 Default Values for AU-3/AU-4-TUG-3 Controller Configuration Commands for AU-3 and AU-4-TUG-3 under SDH Framing*

| Command Name | Default Setting | Command Syntax | Remote Side Setting |
|---|---|---|---|
| **AU-3/AU-4-TUG-3 Controller Configuration Mode under SDH** | | | |
| clock source | internal | **clock source** {**line** \| **internal**} | One side set to **internal** |
| overhead | 0 for c2<br>16 for j1 | **overhead** {**c2** \| **j1**} | — |

*Table 7-10 Default Values for T3 Configuration Commands under SDH Framing*

| Command Name | Default Setting | Command Syntax | Remote Side Setting |
|---|---|---|---|
| crc | 16 | **crc** {**16** \| **32**} | — |
| encapsulation | HDLC | **encapsulation** {**frame-relay** \| **hdlc** \| **ppp**} | Same |
| idle-character | flags (0x7e) | **idle-character** {**flags** \| **marks**} | — |
| keepalive | 10 seconds | **keepalive** *period* | — |

*Table 7-11 Default Values for T1 and E1 Interface Configuration Commands*

| Command Name | Default Setting | Command Syntax | Remote Side Setting |
|---|---|---|---|
| **T1 Interface Configuration Mode** | | | |
| crc | 16 | [**no**] **crc** [**16** \| **32**] | Same |
| DSU bandwidth | 44210 | **dsu bandwidth** *bandwidth* | Same |
| DSU mode | cisco | **dsu mode** *mode* | Same |
| encapsulation | HDLC | **encapsulation** [**hdlc** \| **ppp**] | Same |
| framing | c-bit | **framing** [**c-bit** \| **m13**] | Same |
| idle character | flags (0x7e) | **idle character** [**flags** \| **marks**] | Same |
| keepalive | 10 seconds | **keepalive** *seconds* | Same |
| mtu | 4470 | **mtu** *size* | Same |
| **E1 Interface Configuration Mode** | | | |
| crc | 16 | [**no**] **crc** [**16** \| **32**] | Same |
| encapsulation | HDLC | **encapsulation** [**hdlc** \| **ppp**] | Same |
| framing | c-bit | **framing** [**c-bit** \| **m13**] | Same |
| idle character | flags (0x7e) | **idle character** [**flags** \| **marks**] | Same |
| keepalive | 10 seconds | **keepalive** | Same |
| mtu | 4470 | **mtu** *size* | Same |

# Interface Syntax

To specify an unchannelized or channelized interface on the 4-port STM-1 line card in a configuration command, use the syntax shown in Table 7-12.

*Table 7-12 Interface Syntax for the 4-Port Channelized OC-3/STM-1 Line Card*

| Type of Interface | Slot | Sub-slot | Port | STS-1 Path (T3) | T1 or E1 Number | Channel Group Number |
|---|---|---|---|---|---|---|
| Unchannelized | 1 to 8/ | 0/ | 0 to 3. | 1 to 12 | — | — |
| Channelized | 1 to 8/ | 0/ | 0 to 3. | 1 to 12/ | 1 to 28: for T1<br>1 to 32: for E1 | 0 to 23 for T1<br>0 to 31 for E1 |

Examples:

- Modifying T1 interface 6 in controller configuration mode:

```
Router(config)# controller t3 2/0/0.1
Router(config-controller)# t1 6 command
```

- Modifying T1 interface 6, channel group number 8 in interface configuration mode:

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

# SONET-Framed Interface Configuration Sample

You can configure up to 12 STS-1 connections on a 4-Port Channelized OC-3/STM-1 line card. Each STS-1 connection can be configured as a T3 controller or as a VT controller.

The following procedure walks you through the basic steps for creating full-rate and subrate T3 interfaces, as well as T1 and fractional T1 interfaces. Each T3 controller can be configured as a single T3 interface (full or subrate), as 28 T1 interfaces, or as an even larger number of fractional T1s.

**Step 1** Set the framing type to SONET using the framing command.

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

**Step 2** Assign an STS-1 path to a T3 interface. In this example, the path command is used to set up four T3 interfaces:

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

**Step 3** Create an interface. In the following examples, each type of interface is created in a different T3 controller (4/0/0.1 through 4/0/0.4).

### Full-Rate T3 Interface

**a.** Enter controller configuration mode.

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

**b.** To create a full-rate T3 interface, you must eliminate the T1 interfaces by entering the **no channelized** command.

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

**c.** Enter interface configuration mode.

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

**d.** Go to Step 4.

#### Subrate T3 Interface

**a.** Enter controller configuration mode.

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

**b.** To create a subrate T3 interface, first create a full-rate one.

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

**c.** Enter interface configuration mode, where you can use the **dsu bandwidth** command to create a subrate T3 interface. In this example, a subrate T3 interface is created that has a bandwidth of 16000 kbps.

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

**d.** Go to Step 4.

#### Channelized T3 with a Full T1 Interface

**a.** Enter controller configuration mode.

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

**b.** Use the **t1 channel group** command to create a T1 interface. In the following example, T1 interface 1 (of 28) is defined as being made up of a single channel group, number 20 (any number between 0 and 23). This channel group consists of all 24 DS0 timeslots.

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

**c.** Go to interface configuration mode for the channel group you just created.

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

**d.** Go to Step 4.

#### Channelized T3 with a Fractional T1 Interface

**a.** Enter controller configuration mode.

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

**b.** Use the **t1 channel group** command to create fractional T1 interfaces. In the following example, T1 interface 3 (of 28) is defined as being made up of three channel groups, numbers 19, 20, and 21 (any numbers between 0 and 23). The channel groups consist of a total of 24 DS0 timeslots. Each channel group represents a separate interface.

```
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.** Go to interface configuration mode for one of the channel groups. For example:

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

**d.** Go to Step 4.

**Step 4** Enter the encapsulation method. This example shows the command for using Frame Relay encapsulation. You can also choose PPP or HDLC.

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

**Step 5** If IP routing is enabled on the system, you can assign an IP address and subnet mask. For example:

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

**Step 6** Add any configuration subcommands required to enable routing protocols and set the interface line characteristics.

**Step 7** Change the shutdown state to up, which enables the interface.

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

**Step 8** When you have entered all of the configuration subcommands to complete the configuration, press **Ctrl-Z** to exit configuration mode.

**Step 9** Write the new configuration to NVRAM.

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

---

After you create an interface configuration, you can modify it at any time by using the appropriate Cisco IOS configuration commands.

# SDH-Framed Interface Configuration Sample

If you configure the 4-port STM-1 line card for SDH framing, each OC-3/STM-1 port can be configured as:

- Three AU-3 controllers, each of which can be configured as a T3 interface or as 28 T1 interfaces or 21 E1 interfaces. These interfaces can be channelized into fractional T1s or E1s.
- Three AU-4-TUG-3 controllers, each of which can be configured as 28 T1 interfaces or up to 21 E1 interfaces. These interfaces can be channelized into fractional T1s or E1s.

**Note** AU-3 and AU-4-TUG-3s are based on the higher order path, and do not operate independently, but operate in groups of three. If you shutdown any AU-3 or AU-4-TUG-3 controller, the other two controllers on the same SONET port are also shut down.

The SONET **controller** command specifies the mode, and creates three [sub]controllers of that type, which you can configure like any other controller.

The format created adds a subport extension to the controller SONET name. The form of the subport extension is different for AU-3 and AU-4-TUG-3 controllers as the following examples show:

### AU-3 Controller Syntax

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

For example, the **controller** command for AU-3 #3 on port 0 of a 4-Port Channelized OC-3/STM-1 line card in slot 2 is:

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

### AU-4-TUG-3 Controller Syntax

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

For example, the command for a controller created for TUG-3 #3 on AU-4 of port 0 of a 4-Port Channelized OC-3/STM-1 line card in slot 2 is:

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

## Show Controller Command Syntax

To display the settings for specific on AU-3 and AU-4-TUG-3 controllers, use the **show controller** command.

For AU-3:

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

For AU-4-TUG-3:

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

Where:

- *slot* is 1 to 8
- *subslot* is 0
- *port* is always 0
- *AU-3#* is 1 to 12
- *AU-4#* is always 1
- *TUG-3#* is 1 to 12
- *tug-2* is 1 to 7
- *t1* is 1 to 4 or *e1* is 1 to 3

**Note** The [/<tug-2>/<t1>] options are only available if a mode (C-11 or C-12) is configured on the controller.

For example, for T1 over AU-3, the command to display the settings for AU-3 #3 for a line card in slot 2 is:

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

For AU-4-TUG-3, the command to display the settings for TUG-3 #3 on AU-4 of a 4-Port Channelized OC-3/STM-1 line card in slot 2 is:

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

To display a specific TUG-2 and E1 on the same AU-4-TUG-3, use the command:

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

# AU-3 Controller Interfaces

Use the following procedure to create T3 and T1 interfaces using AU-3 controllers.

**Step 1** Enter SONET controller configuration mode from global configuration mode.

In this example, the 4-port STM-1 line card is in slot 1.

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

**Step 2** Use the **aug controller** command to specify the mode of operation for an OC-3/STM-1 port (for an AU-3, twelve controllers are created: `2/0/0.1 to 2/0/0.12`).

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

### Full-Rate T3 Interface

**Note** T3 lines can only operate over AU-3 controllers.

**a.** Configure the controller on the port.

This example shows AU-3 controller #3 configured for a line card in slot 2.

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

**b.** Set the container mode to C-3 to establish a T3 interface.

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

C-3 mode creates a serial interface at the unchannelized T3 level.

### T1 Channel Group Interface

**a.** Configure the controller on the STM-1 port.

This example shows AU-3 controller #2 configured on the STM-1 port for a line card in slot 2.

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

**b.** Set the container mode to C-11 to establish a T1 interface.

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

**c.** Create a T1 interface.

In the following example, the TUG-2 interface is defined as 1 (any number between 1 and 7), the T1 interface is 2 (any number between 1 and 4), and is made up of a single channel group, 5 (any number between 0 and 23). This channel group consists of all 24 DS0 timeslots.

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

# AU-4-TUG-3 Controller Interfaces

Use the following procedure to create T1 and E1 interfaces using AU-4-TUG-3 controllers.

**Caution** The three AU-4-TUG-3 controllers created on a port are based on the same higher order path and do not operate independently from one another. In other words, when you "shutdown" any AU-4-TUG-3 controller, the other two controllers on the same SONET port are also shut down.

**Step 1** Enter SONET controller-configuration mode.

In this example, the line card is in slot 2.

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

**Step 2** Use the **aug controller** command to specify the mode of operation for the STM-1 port (for an AU-4, three controllers are created).

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

### T1 Channel Group Interface over AU-4-TUG-3/TUG-2

**a.** Configure the controller on an STM-4 port. This example shows AU-4-TUG-3 controller #2 configured on port 0for a line card in slot 2.

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

**b.** Set the container mode to C-11 to establish a T1 interface.

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

**c.** Create a T1 interface.

In the following example, the TUG-2 interface is 1 (any number between 1 and 7), the T1 interface is 2 (any number between 1 and 4,) and is defined as a single channel group, 5 (any number between 0 and 23). This channel group consists of all 24 DS0 timeslots.

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

### E1 Channel Group Interface over AU-4-TUG-3/TUG-2

**a.** Configure the controller on an STM-4 port.

This example shows AU-4-TUG-3 controller #2 configured on port 0 of a line card in slot 2.

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

**b.** Set the container mode to C-12 to establish an E1 interface.

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

**c.** Create an E1 interface.

In the following example, the TUG-2 interface is 1 (any number between 1 and 7), the E1 interface is 2 (any number between 1 and 3), and is defined as a single channel group, 5 (any number between 0 and 31). This channel group consists of all 31 DS0 time slots.

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

# Configuring the SONET Controller for SONET or SDH Framing

By default, the 4-port STM-1 line card consists of 12 STS-1 connections. To set up the 4-port STM-1 line card, you must configure each STS-1 as a T3 or VT interface. This section describes the commands used to create T3 and VT interfaces and test STS-1 connections. This section describes the following:


- Entering Controller-Configuration Mode, page 7-14
- Selecting SONET or SDH Framing, page 7-14
- Configuring the SONET Controller Loopback Mode, page 7-15


## Entering Controller-Configuration Mode

Use the **controller sonet** command to enter controller-configuration mode, where you can configure the controller:

**controller sonet** *slot/subslot/port*

Where *slot* is 1 to 8, and *subslot* and *port* are both 0.

The following example shows you how to enter SONET controller configuration mode for a card in slot 2:

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

## Selecting SONET or SDH Framing

To configure the 4-port STM-1 line card for SONET, you must first set the framing type to either SONET or SDH using the **framing** command.

If the 4-port STM-1 line card framing was previously configured for either SONET or SDH, and you wish to change the framing, you must enter the **no framing** command to return the line card to the default values. Then use the **framing** <*framing type*> command to set the new framing.

**Caution** If you enter the **no framing** command, the line card configuration is erased, and all default line card values are restored. Use the command with caution.

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

The default is SONET.

Use the **no** form of the command to remove framing and return the line card to the default values.

The following example shows you how to specify SONET framing for the 4-port STM-1 line card in slot 2:

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

## Configuring the SONET Controller Loopback Mode

You can enable SONET controller loopbacks by using the **loopback** command.

**loopback** [**internal** | **line**]
[**no**] **loopback** [**internal** | **line**]

The default is no loopback.

Use the **no** form of the command to stop a loopback.

For more information on the **loopback** command, refer to the online *Cisco 10000 Series ESR Troubleshooting Guide*.

The commands in this example run an internal loopback on the 4-Port Channelized OC-3/STM-1 line card in slot 2:

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

# Creating a T3, VT, or AUG Controller

This section shows you how to create and configure T3, VT, or AUG controllers, and contains the following sections:



## Designating an STS-1 Path as a T3 or VT under SONET Framing

When the 4-port STM-1 line card is configured for SONET framing, you can use the **path** command to designate an STS-1 path as a T3 or VT controller. You can designate up to 12 STS-1 channels. (Figure 7-1 on page 7-22 shows the relationship between the STM-1, T3 interfaces, VT interfaces, and T1 interfaces.

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

Where *STS_number* is the virtual T3 or VT controller. Enter a value from 1 to 12.

Use the **no** form of the command to remove a T3 or VT controller.

In the following example, STS-1 number 1 is defined as a T3 line and controller T3 number 2/0/0.1 is generated:

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

In the next example, STS-1 number 1 is defined as a VT line and a VT controller 2/0/0.1 is generated:

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

# Designating an STM-4 Port as an AU-3 or AU-4-TUG-3 Controller under SDH Framing

When the 4-port STM-1 line card is configured for SDH framing, you can use the **aug controller** command to specify the mode of operation for an STM-4 port as an AU-3 controller or an AU-4-TUG-3 controller. This command creates 12 subcontrollers of the designated type, which you can then configure like any other controller.

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

Where *au-3* and *au-4-tug-3* designate the type of controller. Enter a value from 1 to 12.

Use the **no** form of the command to remove an AU-3 or AU-4-TUG-3 controller.

The format of the name for these controllers requires you to add a subport extension to the SONET controller name. The subport extension is different for AU-3 and AU-4-TUG-3 controllers.

### AU-3 Controller

For an AU-3 controller, the syntax is:

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

For example, the following command Uses AU-3 controller #3 on port 0 of a line card in slot 2.

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

### AU-4-TUG-3 Controller

For an AU-4-TUG-3 controller, the syntax is:

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

**Note** The twelve AU-4-TUG-3 controllers created on an SDH-framed SONET port are based on the same higher order path and do not operate independently from one another. In other words, when you "shutdown" any AU-4-TUG-3 controller, the other two controllers on the same SDH-framed SONET port are also shut down.

For example, the following command uses TUG-3 #3 on AU-4 of port 0 of an 4-port STM-1 line card in slot 2.

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

# VT Commands under SONET Framing

This section describes how to use the **controller vt** command to further channelize the SONET-framed STS-1 controller of the 4-port STM-1 line card.

## Entering Controller Configuration Mode for VT

A virtual tributary (VT) controller on a 4-port STM-1 line card can be channelized into 28 T1 (VT1.5) interfaces or 21 E1 (VT2) interfaces. You can use the **controller vt** command to shut down a VT link or to change the settings for a T1 or E1 interface.

For information about T1 configuration settings, see "Creating T1 or E1 Channel Groups under SONET or SDH Framing" section on page 7-26, and the "Channelized T1 Commands under SONET or SDH Framing" section on page 7-29.

You can configure a VT link by entering the **controller vt** command.

*slot/sub-slot/port.path*

Where *path* is a value from 1 to 12. Each number represents a VT that houses 28 T1 lines or 21 E1 lines.

For example:

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

# Unchannelized T3 Commands under SONET or SDH Framing

This section shows you how to configure an unchannelized T3 on a SONET-framed or SDH-framed controller.

If the 4-port STM-1 line card is configured for SONET framing, a T3 interface is channelized into 28 T1 interfaces by default. You must unchannelize the T3 interface to create a full-rate or subrate T3 interface.

If the 4-Port Channelized OC-3/STM-1 line card is configured for SDH framing, C-3 container mode automatically creates an unchannelized T3 serial interface.

This section describes the commands you use to create, customize, and test full-rate and subrate T3 interfaces. This section describes the following:



You can also use the following commands, described in the "Channelized T3 Commands under SONET or SDH Framing" section on page 7-21, when you configure unchannelized T3: **clock source, mdl, equipment**, and **loopback**.

## Entering Controller Configuration Mode for T3

To create an unchannelized T3 interface, you must first enter controller configuration mode for the T3 controller you want to configure.

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

Where *path* specifies the T3 interface number.

The following example shows how to enter controller configuration mode:

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

## Configuring a T3 Interface on a SONET-Framed Controller as Unchannelized

If the 4-Port Channelized OC-3/STM-1 line card is configured for SONET framing, you must unchannelize the T3 interface to create a full-rate or subrate T3 interface.

You can configure the T3 interface as unchannelized (clear channel) by entering the **no channelized** command.

[**no**] **channelized**

**Caution** The **no channelized** command removes all channel groups from a channelized T3 interface. If you have already configured channel groups, use this command with caution.

Use the **channelized** command to return the interface to its default. The default value for a T3 interface is channelized.

In the following example, an unchannelized T3 interface is created:

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

## Implementing Subrate T3

You can implement subrate T3 by specifying the bandwidth for an unchannelized T3 interface. To do so, use the **dsu bandwidth** *bandwidth* command from interface configuration mode.

**dsu bandwidth** *bandwidth*
[**no**] **bandwidth** *bandwidth*

Where *bandwidth* is a numeric value between 0 and 44,210 kbps.

The default bandwidth is 44,210 kbps.

To return to the default bandwidth, use the **no** form of this command.

When you specify a value, the software sets the bandwidth to the closest acceptable bandwidth based on the timeslot size for the current DSU mode.

For you to properly use this command, the remote side of the connection must have a Cisco 7200 router or Cisco 7500 router with a PA-T3 or PA-2T3 port adapter or a T3 DSU supported by the **dsu mode** command.

In the following example, a bandwidth of 16,000 kbps is specified:

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

## Setting a Framing Type

To specify a framing type for the unchannelized T3 controller, use the **framing** command.

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

The default framing type is c-bit.

Use the **no** form of this command to restore the default framing type.

In the following example, framing is set to m13:

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

## Specifying a DSU Mode

To specify a DSU mode for a selected T3 interface, use the **dsu mode** command from interface configuration mode. This command configures the line card to emulate a manufacturer's proprietary multiplexing scheme.

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

The default DSU mode is cisco.

Use the **no** form of the command to return the DSU mode to its default.

In the following example, the DSU mode is set to **cisco**:

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

## Enabling Scrambling

To enable scrambling on an unchannelized T3 interface, use the **scramble** command from interface configuration mode.

**scramble**
[**no**] **scramble**

The default setting for this command is no scramble (scrambling disabled).

Ensure that both sides of the link have the same scrambling setting.

In the following example, scrambling is enabled on the specified T3 interface:

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

## Specifying the Idle Character

To set a specific character on the unchannelized T3 interface to be transmitted between HDLC packets, use the **idle character** command from interface configuration mode.

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

Where:

- **flags** sets an idle character of 0x7e
- **marks** sets an idle character of all 0xff

The default idle character is 0x7e.

Use the **no** form of the command to return the idle character to its default.

In the following example, the idle character is set to flags:

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

**Note** Some systems interpret marks, 0xff, as an abort signal. Therefore, flags, 0x7e, is preferred.

## Running a T3 BER Test under SONET or SDH Framing

You can configure an unchannelized T3 interface to run a bit error rate (BER) test. The test is used to check cables and solve signal problems in the field. To send a BER test pattern on an unchannelized T3 interface, use the following interface configuration command:

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

Where:

- **errors** *number* is 1 to 255
- **pattern** *pattern* is
  - 0s—repetitive test pattern of all zeros (00000..)
  - 1s—repetitive test pattern of all ones (11111..)
  - 2^15—pseudorandom O.151 test pattern (32,768 bits in length)
  - 2^20-O153—pseudorandom O.153 test pattern (1,048,575 bits in length)
  - QRSS-2^20—pseudorandom QRSS O.151 test pattern (1,048,575 bits in length)
  - 2^23—pseudorandom O.151 test pattern (8,388,607 bits in length)
- **interval** *time* is 1 to 1440 minutes

You can terminate a BER test at any time using the **no bert** command.

For more information, refer to the online *Cisco 10000 Series Internet Router Troubleshooting Guide*.

Examples:

- Send a BER test pseudorandom pattern of 2^20 through T3 line 1 for 5 minutes.

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

- Send a repetitive pattern of all 1s through T3 line 1 for 1440 minutes.

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

# Channelized T3 Commands under SONET or SDH Framing

This section shows you how to configure a channelized T3 on a SONET-framed or SDH-framed controller.

By default, a T3 interface on a 4-Port Channelized OC-3/STM-1 line card is channelized into 28 T1 interfaces. This section describes the commands you use to customize and test a channelized T3 interface. This section describes procedures for



To use the interface for subscriber traffic, you must configure its T1 and DS0 components. For more information, see the "Creating T1 or E1 Channel Groups under SONET or SDH Framing" section on page 7-26.

Figure 7-1 illustrates the levels of configurable interface bandwidth that channelization offers.

## Entering Controller Configuration Mode for T3

You may need to enter controller configuration mode for the T3 interface. For example, enter this mode to change an unchannelized T3 interface back to a channelized one or to set MDL messages.

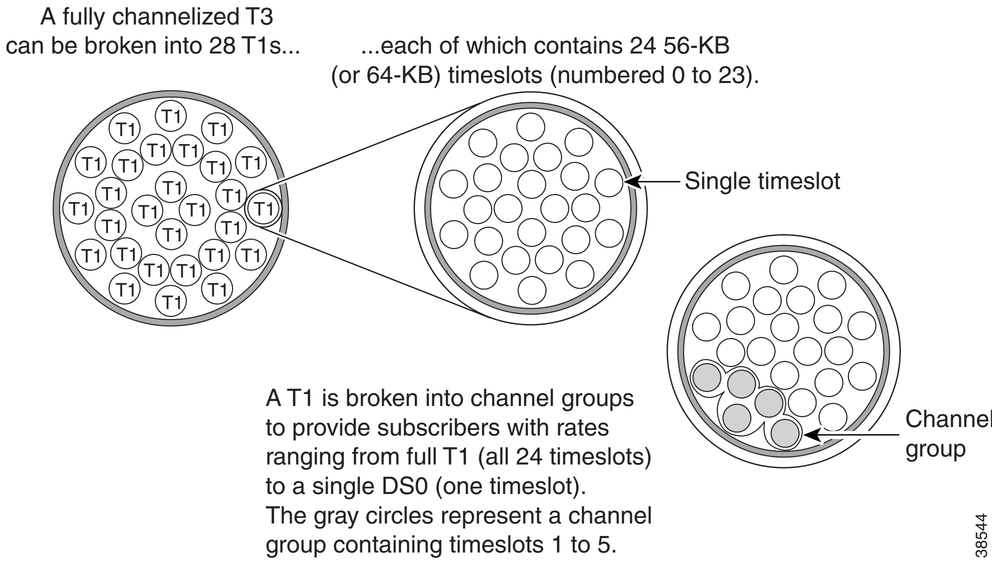
**controller T3** *slot/sub-slot/port.path*

*Path* specifies the T3 interface number.

The following example shows how to enter controller configuration mode:

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

*Figure 7-1 Channelization of T3s*




## Configuring a T3 Interface as Channelized

A T3 interface is channelized by default. Use the **channelized** command if you had previously made the interface unchannelized.

[**no**] **channelized**
**channelized**

**Caution** The **no channelized** command removes all channel groups from a 4-Port Channelized OC-3/STM-1 line card interface. If you have already configured channel groups, use this command with caution.

The following example shows how to create a 4-Port Channelized OC-3/STM-1 line card interface:

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

## Setting the Framing Type

To specify a framing type for the channelized T3 controller, use the **framing** command.

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

The default is auto-detect.

Use the **no** form of the command to return the framing type to its default.

You can instruct the 4-Port Channelized OC-3/STM-1 line card to detect the framing type from the far end and transmit that same framing type:

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

## Entering MDL Messages

You can configure maintenance data link (MDL) messages (as defined in the ANSI T1.107a-1990 specification) on the channelized T3 interface.

**Note** MDL messages are supported only when the T3 framing is set for C-bit parity. (See the "Setting the Framing Type" section on page 7-22.)

To configure MDL messages, use the **mdl** command:

**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*}

Where:

- **transmit path** enables transmission of the MDL path message
- **transmit idle-signal** enables transmission of the MDL idle-signal message
- **eic** is the equipment identification code (up to 10 characters)
- **lic** is the location identification code (up to 11 characters)
- **fic** is the frame identification code (up to 10 characters)
- **unit** is the unit identification code (up to 6 characters)
- **pfi** is the facility identification code to include in the MDL path message (up to 38 characters)
- **port** is the equipment port (which initiates the idle signal) to include in the MDL idle signal message (up to 38 characters)
- **generator** is the generator number to include in the MDL test signal message (up to 38 characters)

The default is that no MDL message is configured.

Use the **no** form of the command to remove an MDL message.

Examples of configuring MDL messages follow:

- Enable the MDL path message transmission.

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

- Enable the MDL idle signal message transmission.

```
Router(config-controller)# mdl transmit idle-signal
```

- Enable the MDL test signal message transmission.

```
Router(config-controller)# mdl transmit test-signal
```

- Enter the equipment identification code.

```
Router(config-controller)# mdl string eic router A
```

- Enter the location identification code.

```
Router(config-controller)# mdl string lic test network
```

- Enter the frame identification code.

```
Router(config-controller)# mdl string fic building b
```

- Enter the unit identification code.

```
Router(config-controller)# mdl string unit abc
```

- Enter the facility identification code.

```
Router(config-controller)# mdl string pfi string
```

- Enter the port number to send in the MDL idle signal message.

```
Router(config-controller)# mdl string port string
```

- Enter the generator number to send in the MDL test signal message.

```
Router(config-controller)# mdl string generator string
```

## Specifying the Idle Pattern

You can set a specific pattern to be transmitted between HDLC packets on all unconfigured timeslot interfaces that belong to a channelized T3 interface. To do so, use the **idle pattern** command.

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

Where *patterns* is a number in the range of 0x0 to 0xff (hexadecimal) or 0 to 255 (decimal). You can enter this value in either hexadecimal or decimal form. Values of 0 to 254 set the idle pattern to HDLC flags (0x7e); a value of 255 sets the pattern to 0xff (all ones).

**Note** Some systems interpret marks, 0xff, as an abort signal. Therefore, flags, 0x7e, is preferred.

The default idle pattern is 0x7e.

Use the **no** form of the command to return the idle pattern to its default value.

Examples:

- Set a hexadecimal idle pattern.

```
Router(config)# controller T3 1/0/0.1
Router(config-controller)# idle pattern 0x10
```

- Set a decimal idle pattern.

```
Router(config)# controller T3 1/0/0.1
Router(config-controller)# idle pattern 23
```

## Setting the T3 Clock Source

At the prompt, set the internal or line clock source for the selected T3 controller using the **clock source** command. This command is set in controller configuration mode.

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

Where:

- **internal** specifies that the internal clock source is used
- **line** specifies that the network clock source is used

The default is clock source internal.

In this example, a T3 controller is instructed to use a line clock source:

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

## Configuring the Loopback Mode for a T3 Controller

You can configure the loopback modes for a T3 controller by using the **loopback** command:

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

The default is no loopback.

To disable loopback on the T3 controller, use the **no** form of the command.

Examples:

- Configure a T3 controller for local loopback.

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

  Local loopback simultaneously loops all channels toward the router and transmits a T3 AIS to the network. You can use local loopback to diagnose problems with the port when the port is isolated from the network cables.

- Configure a T3 port for network loopback.

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

  Network loopback loops the T3 line back towards the network and can be used to diagnose problems with cables from the central switching office to the port.

- Configure a T3 port for remote loopback.

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

  Remote loopback sends a command to loop the T3 line at the far end (central office). It can be used to diagnose problems with cables from the port adapter to the switching office.

## Configuring a T3 Controller to Respond to Remote Loopback Commands

Use the **equipment loopback** command to run loopbacks in conjunction with remote equipment.

**equipment** [**customer** | **network**] l**oopback**

Where:

- **customer** allows a port to respond to loopback commands from remote T3 equipment.
- **network** causes a controller to ignore remote T3 loopback commands.

Example:

To enable the controller's ability to respond to remote loopback requests, type:

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

To prevent a controller from responding to remote loopback commands, type:

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

**Note** Remote loopbacks are only available when you use c-bit parity framing.

For more information on the **loopback** command, refer to the online *Cisco 10000 Series Internet Router Troubleshooting Guide*.

# Creating T1 or E1 Channel Groups under SONET or SDH Framing

To create an interface that supports subscriber traffic, you must combine the timeslots into channel groups. To help you configure channel groups, consider the following T1 and E1 characteristics:

### For SONET-Framed T3 Controllers

- 1 channelized T3 is divided into 28 T1s
- 1 T1 is divided into 24 timeslots (or DS0s)

### For SONET-Framed VT Controllers

For T1:

- 1 VT controller is divided into 28 T1s (VT1.5)
- 1 T1 is divided into 24 timeslots (or DS0s)

For E1:

- 1 VT controller is divided into 21 E1s (VT2)
- 1 E1 can be divided into 31 timeslots (or DS0s)

### For SDH-Framed T3 Controllers

- 1 T3 is divided into 7 TUG-2s
- 1 TUG-2 is divided into either:
  - 4 T1s that can each be divided into 24 timeslots (or DS0s)
  - 3 E1s that can be divided into 31 timeslots (or DS0s)

## Creating Channel Groups for T1 SONET-Framed Interfaces

Use the **t1 channel-group** command to create a channel group for a SONET-framed interface.

Figure 7-1 illustrates channelization possibilities on the 4-Port Channelized OC-3/STM-1 line card.

You create a logical channel group using the following command:

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

Where:

- *t1-number* is T1 interface number 1 to 28.
- **channel-group** *channel-group-number* identifies the channel group with any number from 0 to 23.

- **timeslots** *list-of-timeslots* can be 1 to 24 or a combination of subranges within the 1 to 24 interval. You can indicate a range using a hyphen, commas, or a combination of both. One timeslot equals one DS0. (See the examples below.)
- **speed**{**56** | **64**} is an optional argument that specifies the speed of a timeslot as either 56 or 64 kbps. The default is 64. (The 56 kbps timeslots are generally used with older T1 equipment that does not support B8ZS line coding and are associated with SF framing.)

Use the **no** form of the command to remove a logical channel group.

The following examples show how to use the **t1 channel-group** command:

- In this example, T1 interface 3 includes channel group 20 and consists of 9 channelized timeslots:

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

  To enter interface configuration mode for T1 interface 3 channel group 20, enter the following:

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

- In this example, t1 interface 4 includes channel group 18 and consists of all 24 timeslots, creating a full T1 interface:

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

  To enter interface configuration mode for T1 interface 4 channel group 18, enter the following:

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

- In the following example, T1 interface number 5 is divided into three channel groups, which total 11 timeslots:

```
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
```

- In the following example, channel group 20 on T1 1 is removed:

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

## Creating Channel-Groups for SDH-Framed Interfaces

### Channel Limitations

There is a limit of 128 channels per SDH controller, and a maximum of 192 channels for the 4-Port Channelized OC-3/STM-1 line card. This limitation applies to both T1 and E1 interfaces when you create channel groups for controllers.

For example, if you created 128 channels on T1 controller #1 on port 2, and 64 channels on T1 controller #2 on port 2, then you could not create any additional channels on T1 controller #3 or T1 controller #4 because you have reached the maximum limit of 192 channels for a single port (port 2 in this example).

To create a logical channel group, use one of the following commands:

### For T1:

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

**For E1 (framed):**

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

**For E1 (unframed):**

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

Where:

- *tug-2 is 1 to 7*
- *t1#* is 1 to 4
- *e1#* is 1 to 3
- **channel-group** *channel#* identifies the channel group with any number from 0 to 23.
- **timeslots** *range* can be 1 to 24 for T1, or 1 to 31 for E1, or a combination of subranges within those intervals. You can indicate a range using a hyphen, commas, or a combination of both. One timeslot equals one DS0. (See the examples below.)
- **speed**{**56** | **64**} is an optional T1 argument that specifies the speed of a timeslot as either 56 or 64 kbps. The default is 64. (The 56 kbps timeslots are generally used with older T1 equipment that does not support B8ZS and are associated with SF framing.)

**Note** The **speed** argument applies only to T1 channel groups and is not valid for E1.

- **unframed** creates an unframed E1 channel. This creates an interface with channel-group number 0.

**Note** The unframed argument applies only to E1 channel groups and is not valid for T1.

Use the **no** form of the command to remove a logical channel group.

The following examples show how to use the **channel-group** command:

- In this example, T1 interface 3 includes channel group 20 and consists of 9 channelized timeslots:

```
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
```

  To enter interface configuration mode for T1 interface 3 channel group 20, enter the following:

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

- In this example, T1 interface 4 includes channel group 18 and consists of all 24 timeslots, creating a full T1 interface:

```
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
```

  To enter interface configuration mode for T1 interface 4 channel group 18, enter the following:

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

- In the following example, T1 interface number 5 is divided into three channel groups, which total 11 timeslots:

```
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
```

- In the following example, channel group 20 is removed:

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

# Channelized T1 Commands under SONET or SDH Framing

You can enter commands to modify aspects of a T1 interface from controller configuration mode for a T3 interface. This section describes the commands for:

- Setting the Framing Format, page 7-29
- Controlling Yellow Alarms, page 7-30
- Setting the T1 Clock Source, page 7-30
- Configuring FDL, page 7-30
- Configuring a T1 BER Test, page 7-31
- Configuring Loopback Mode, page 7-31

After you configure a T1 interface, you can add encapsulation, routing, and other instructions by entering interface configuration mode. For example:

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

## Setting the Framing Format

You can specify the T1 interface framing format using the following command:

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

Where:

- *t1-number* is T1 interface 1 to 28
- **framing** is either extended super frame (ESF) or super frame (SF). You can set SF hdlc-idle to 0x7E or 0xFF
- **hdlc-idle** options allow you to set the idle pattern for the T1 interface to either 0x7e (the default) or 0xFF

The default framing format is extended super frame (ESF).

Use the **no** form of the command to return framing to its default value.

Examples:

- Set SF framing format for T1 interface 6.

```
Router(config)# controller T3 1/0/0.1
Router(config-controller)# t1 6 framing sf
```

- Set ESF framing format for T1 interface 16:

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

## Controlling Yellow Alarms

Use the **t1 yellow** command to turn the detection or generation of a yellow alarm on or off.

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

Where:

- *t1-number* is T1 interface 1 to 28
- **detection** means that the interface is told that it is failing by the remote device, causing Cisco IOS software to send a message to the console
- **generation** means that the interface notifies the remote device if it is failing, causing Cisco IOS software to send a message to the console

When you select SF framing for a full T1 interface (24 timeslots) that uses the default speed of 64, consider using the **no** *t1-number* **yellow detection** command to turn off yellow alarm detection, because the yellow alarm can be incorrectly detected with SF framing.

In the following example, T1 interface 1 is set to yellow detection:

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

## Setting the T1 Clock Source

You can set the internal or line (network) clock source for a T1 interface using the **controller** command.

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

Where:

- *t1-number* is T1 interface 1 to 28
- **internal** specifies that the internal clock source is used
- **line** specifies that the network clock source is used

The default is clock source internal.

In the following example, the interface is instructed to get its clock source from the line:

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

## Configuring FDL

You can use the **fdl ansi** command to enable 1-second transmissions of ANSI T1.403 performance reports through the facility data link (FDL), on both ends of the T1 connection.

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

Where *t1-number* is T1 interface 1 to 28.

Use the **no** form of the command to disable this feature.

**Note** You can use this command only when the T1 framing is ESF.

In the following example, FDL is enabled:

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

## Configuring a T1 BER Test

You can configure an individual T1 interface to run an independent BER test. The test is used to check cables and solve signal problems in the field. To send a BER test pattern on a T1 interface, use the following command:

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

Where:

- *t1-number* is T1 interface number 1 to 28
- *time* is 1 to 14,400 minutes
- *pattern* is
  - 2^11—pseudorandom test pattern (2048 bits in length)
  - 2^15—pseudorandom O.151 test pattern (32,768 bits in length)
  - 2^20-O153—pseudorandom O.153 test pattern (1,048,575 bits in length)
- **unframed** causes the BER test pattern to use the entire T1 bandwidth, including the T1 framing and payload bits. If **unframed** is omitted, the T1 is either SF or ESF framed as configured by the T1 n framing command, and the BER test pattern occupies only the T1 payload bits.

**Note** For each T3, you can run only one BER test at a time.

You can terminate a BER test at any time using the **no** form of the command.

For more information, refer to the online *Cisco 10000 Series Internet Router Troubleshooting Guide*.

Example:

- Send a BER test pseudorandom pattern of 2^15 through T1 interface 10 for 5 minutes.

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

## Configuring Loopback Mode

If a problem occurs when you configure a T1 interface, you can troubleshoot the line card by using the following command from controller configuration mode:

**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**]]]

Where:

- *t1-number* is T1 interface 1 to 28.
- **local** loops the router output data back toward the router at the T1 framer and sends an alarm indication signal (AIS) out toward the network.

- **network** {**line** | **payload**} loops the data back toward the network and automatically sets a local loopback at the HDLC controllers (line) or loops the payload data back toward the network and automatically sets a local loopback at the HDLC controllers (payload).
- **remote line fdl** {**ansi** | **bellcore**} sends a repeating, 16-bit ESF data link keyword to the remote end, requesting that it enter into a network line loopback. You can specify an ANSI or Bellcore keyword.
- **remote payload** [**fdl**] [**ansi**] sends a repeating, 16-bit ESF data link code word to the remote end, requesting entry into a network payload loopback. Using **fdl** and **ansi** enables the remote payload facility data link (FDL) ANSI bit loopback on the T1 channel.

Use the **no** form of the command to terminate a loopback.

For more information on this command, refer to the online *Cisco 10000 Series Internet Router Troubleshooting Guide*.

Examples:

- Configure the T3 controller for local loopback on T1 interface 1.

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

- Configure the T3 controller for remote FDL ANSI loopback on T1 interface 1.

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

CHAPTER 8

# 6-Port Channelized T3 Line Card Configuration

This chapter describes procedures for configuring the Cisco 10000 series 6-Port Channelized T3 line card, hereafter known as the 6-Port Channelized T3 line card.

The 6-Port Channelized T3 line card provides the Cisco 10000 series routers with six DS3 ports of high-density T3 service.

This chapter contains the following sections:



## Software Support

Table 8-1 shows the minimum Cisco IOS release on each release train that supports the 6-Port Channelized T3 line card.

*Table 8-1 6-Port Channelized T3 Line Card Software Support*

| Required PRE | Minimum Cisco IOS Releases |
|---|---|
| PRE1 | Cisco IOS Release 12.0(9)SL and later releases of Cisco IOS 12.0SL<br>Cisco IOS Release 12.0(17)ST and later releases of Cisco IOS 12.0ST<br>Cisco IOS Release 12.0(22)S and later releases of Cisco IOS 12.0S<br>Cisco IOS Release 12.2(8)BZ and later releases of Cisco IOS 12.2BZ |
| PRE2 | Cisco IOS Release 12.2(15)BX and later releases of Cisco IOS 12.2BX<br>Cisco IOS Release 12.3(7)XI and later releases of Cisco IOS 12. XI<br>Cisco IOS Release 12.2(28)SB and later releases of Cisco IOS 12.2SB |

## Checking Hardware and Software Compatibility

The PRE installed in the Cisco 10000 series router chassis must support the Cisco IOS software running on the router. Use the **show version** command to check the PRE version installed.

To see if a feature is supported by a Cisco IOS release, to locate the software document for that feature, or to check the minimum software requirements of Cisco IOS software with the hardware installed on your router, Cisco maintains the Software Advisor tool on Cisco.com at http://www.cisco.com/cgi-bin/Support/CompNav/Index.pl

This tool does not verify whether line cards within a system are compatible, but does provide the minimum Cisco IOS requirements for individual hardware line cards, modules, or options.

You must be a registered user on Cisco.com to access this tool.

# Default Values

Table 8-2 lists default values for the 6-Port Channelized T3 line card and indicates which values apply to which line speeds. This table also includes the command used for modifying a default value and indicates whether a value needs to be the same (or opposite) on the remote end of the connection.

*Table 8-2 6-Port Channelized T3 Line Card Default Values*

| Command Name | Default Setting | Command Syntax | Remote Side Setting | DS0 | T1 | Ch T3 | Unch T3 |
|---|---|---|---|---|---|---|---|
| **Controller Configuration Mode** | | | | | | | |
| cable length | 249 | **cablength** *feet* | — | | | x | x |
| channelized | channelized | [**no**] **channelized** | — | | | x | x |
| clock source | internal | **clock source** [**line** \| **internal**] | At least one side set to **internal** | | | x | x |
| T1 clock source | internal | **t1** *t1-number* **clock source** [**line** \| **internal**] | At least one side set to **internal** | | x | | |
| framing | auto-detect | **framing** [**c-bit** \| **m23** \| **auto-detect**] | Same | | | x | |
| T1 framing | esf | **t1** *t1-number* **framing** [**esf** \| **sf** [**hdlc-idle** {**0x7E** \| **0xFF**}]] | Same | | x | | |
| idle pattern | 0X7e (flags) | **idle pattern** [**0x0 to 0xFF**] | Same | | | x | |
| **Interface Configuration Mode** | | | | | | | |
| crc (cyclical redundancy check) | 16 | [**no**] **crc** [**16** \| **32**] | Same | x | | | x |
| DSU bandwidth | 44,210 | **dsu bandwidth** *bandwidth* | Same | | | | x |
| DSU mode | cisco | **dsu mode** *mode* | Same | | | | x |
| encapsulation | HDLC | **encapsulation** [**hdlc** \| **ppp** \| **frame-relay**] | Same | x | | | x |

*Table 8-2 6-Port Channelized T3 Line Card Default Values (continued)*

| Command Name | Default Setting | Command Syntax | Remote Side Setting | DS0 | T1 | Ch T3 | Unch T3 |
|---|---|---|---|---|---|---|---|
| framing | c-bit | **framing** [**c-bit** \| **m13**] | Same | | | | x |
| idle character | flags (0x7e) | **idle character** [**flags** \| **marks**] | Same | | | | x |
| keepalive | keepalive (10 sec) | **keepalive** | Same | x | | | x |
| mtu (maximum transmission unit) | 4470 | **mtu** *size* | Same | x | | | x |
| scramble | No scrambling | [**no**] **scramble** | Same | | | | x |

# Interface Syntax

To specify an interface number in a configuration command, use the syntax in Table 8-3 to identify interfaces on the 6-Port Channelized T3 line card.

*Table 8-3 6-Port Channelized T3 Line Card Interface Syntax*

| Type of Interface | Slot | Subslot | Port (T3 Number) | T1 Number | Channel Group Number |
|---|---|---|---|---|---|
| Unchannelized | 1 to 8/ | 0/ | 0 to 5 | — | — |
| Channelized | 1 to 8/ | 0/ | 0 to 5/ | 1 to 28 | 0 to 23 |

Examples:

- Modifying T1 interface 6 in controller configuration mode:

```
Router(config)# controller t3 2/0/0
Router(config-controller)# t1 6 command
```

- Modifying T1 interface 6, channel group number 8 in interface configuration mode:

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

# Interface Configuration Sample

Each T3 controller can be configured as a single T3 interface (full or subrate), as 28 T1 interfaces, or as an even larger number of fractional T1s. The following procedure walks you through the basic steps for creating full-rate and subrate T3 interfaces, as well as T1 and fractional T1 interfaces:

**Step 1** Create an interface. In the following examples, each type of interface is created in a different T3 controller (2/0/0 through 2/0/3).

#### Full-Rate T3 Interface

a. Enter controller configuration mode.

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

b. To create a full-rate T3 interface, you must eliminate the T1 interfaces by entering the **no channelized** command.

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

c. Go to interface configuration mode:

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

d. You can now continue to Step 2.

#### Subrate T3 Interface

a. Enter controller configuration mode.

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

b. To create a subrate T3 interface, first create a full-rate one.

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

c. Then go into interface configuration mode, where you can use the **dsu bandwidth** command to create a subrate T3 interface. In this example, a subrate T3 interface is created that has a bandwidth of 16,000 kbps.

```
Router(config)# interface serial 2/0/1
Router(config-if)# dsu bandwidth 16000
```

d. You can now continue to Step 2.

#### Full T1 Interface

a. Enter controller configuration mode.

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

b. Use the **t1 channel group** command to create a T1 interface. In the following example, T1 interface 1 (of 28) is defined as being made up of a single channel group, number 20 (any number between 0 and 23). This channel group consists of all 24 DS0 timeslots.

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

c. Go to interface configuration mode for the channel group you just created.

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

d. You can now continue to Step 2.

#### Fractional T1 Interface

a. Enter controller configuration mode.

```
Router(config)# controller t3 2/0/3
Router(config-controller)#
```

b. Use the **t1 channel group** command to create fractional T1 interfaces. In the following example, T1 interface 3 (of 28) is defined as being made up of three channel groups, numbers 19, 20, and 21. (numbers between 0 and 23 are allowed.) The channel groups consist of a total of 24 DS0 timeslots. Each channel group represents a separate interface.

```
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. Go to interface configuration mode for one of the channel groups. For example:

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

d. You can now continue to Step 2.

**Step 2** Enter the encapsulation method. This example shows the command for using Frame Relay encapsulation. You can also choose PPP or HDLC.

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

**Step 3** If IP routing is enabled on the system, assign an IP address and subnet mask. For example:

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

**Step 4** Add any configuration subcommands required to enable routing protocols and set the interface line characteristics.

**Step 5** Change the shutdown state to up, which enables the interface.

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

**Step 6** When you have entered all of the configuration subcommands to complete the configuration, press **`Ctrl-Z`** to exit configuration mode.

**Step 7** To write the new configuration to NVRAM, type

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

After you create an interface configuration, you can modify it at any time by using the appropriate Cisco IOS configuration commands.

# Unchannelized T3 Commands

By default, a T3 interface on a 6-Port Channelized T3 line card is channelized into 28 T1 interfaces. You must unchannelize the T3 interface in order to create a full-rate or subrate T3 interface. This section describes the commands you use to create, customize, and test full-rate and subrate T3 interfaces. This section describes the following:

- Entering Controller Configuration Mode for T3, page 8-6
- Configuring a T3 Interface as Unchannelized, page 8-6
- Implementing Subrate T3, page 8-6

- Setting the Framing Type, page 8-7
- Specifying the DSU Mode, page 8-7
- Enabling Scrambling, page 8-7
- Specifying an Idle Character, page 8-8
- Running a BER Test, page 8-8


You can also use the following commands, described in the "Channelized T3 Commands" section on page 8-9, when you are configuring an unchannelized T3: **cablelength**, **clock source**, **mdl, equipment**, and **loopback**.

# Entering Controller Configuration Mode for T3

To create an unchannelized T3 interface, you must first enter controller configuration mode for the T3 controller you want to configure.

**controller T3** *slot/sub-slot/port*

The following example shows how to enter controller configuration mode:

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

# Configuring a T3 Interface as Unchannelized

You can configure the T3 interface as unchannelized (clear channel) by entering the **no channelized** command.

[**no**] **channelized**
**channelized**

**Caution** The **no channelized** command removes all channel groups from a channelized T3 interface. If you have already configured channel groups, use this command with caution.

Use the **channelized** command to return the interface to its default. The default value for a T3 interface is channelized.

In the following example, an unchannelized T3 interface is created:

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

# Implementing Subrate T3

You can implement subrate T3 by specifying the bandwidth for an unchannelized T3 interface. To do so, use the command **dsu bandwidth** *bandwidth* from interface configuration mode.

**dsu bandwidth** *bandwidth*
[**no**] **bandwidth** *bandwidth*

Where *bandwidth* is a numeric value between 0 and 44210 kbps.

The default bandwidth is 44210 kbps.

To return to the default bandwidth, use the **no** form of this command.

When you specify a value, the software sets the bandwidth to the closest acceptable bandwidth, based on the timeslot size for the current DSU mode.

In order for you to use this command, the remote side of the connection must have a Cisco 7200 or Cisco 7500 with a PA-T3 or PA-2T3 port adapter or a T3 DSU supported by the **dsu mode** command.

In the following example, a bandwidth of 16000 kbps is specified:

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

# Setting the Framing Type

To specify a framing type for the unchannelized T3 controller, use the **framing** command.

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

The default framing type is C-bit.

Use the **no** form of this command to restore the default framing type.

In the following example, framing is set to m13:

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

# Specifying the DSU Mode

To specify a DSU mode for a selected T3 interface, use the **dsu mode** command from interface configuration mode. This command configures the line card to emulate a manufacturer's proprietary multiplexing scheme.

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

The default DSU mode is **cisco**.

Use the **no** form of the command to return the DSU mode to its default.

In the following example, the DSU mode is set to **cisco**:

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

# Enabling Scrambling

To enable scrambling on an unchannelized T3 interface, use the **scramble** command from interface configuration mode.

```
scramble
[no] scramble
```

The default setting for this command is no scramble (scrambling disabled).

Both sides of the link should have the same scrambling setting.

In the following example, scrambling is enabled on the specified T3 interface:

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

# Specifying an Idle Character

To set a specific character on the unchannelized T3 interface to be transmitted between HDLC packets, use the **idle character** command from interface configuration mode.

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

Where:

- **flags** sets an idle character of 0x7e.
- **marks** sets an idle character of all 0xff.

The default idle character is 0x7e.

Use the **no** form of the command to return the idle character to its default.

In the following example, the idle character is set to flags:

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

**Note** Some systems interpret marks, 0xff, as an abort signal. Therefore, flags, 0x7e, is preferred.

# Running a BER Test

You can configure an unchannelized T3 interface to run a bit error rate (BER) test. The test can be used in checking cables and solving signal problems in the field. To send a BER test pattern on an unchannelized T3 interface, use the following interface configuration command:

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

Where:

- **errors** *number* is 1 to 255.
- **pattern** *pattern* is
  - 0s—repetitive test pattern of all zeros (00000..)
  - 1s—repetitive test pattern of all ones (11111..)
  - 2^15—pseudorandom O.151 test pattern (32,768 bits long)
  - 2^20-O153—pseudorandom O.153 test pattern (1,048,575 bits long)
  - QRSS-2^20—pseudorandom QRSS O.151 test pattern (1,048,575 bits long)
  - 2^23—pseudorandom O.151 test pattern (8,388,607 bits long)
- **interval** *time* is 1 to 1440 minutes.

You can terminate a BER test at any time using the **no bert** command.

For more information, refer to the online *Cisco 10000 Series Internet Router Troubleshooting Guide*.

Examples:

- Send a BER test pseudorandom pattern of 2^20 through T3 interface 1/0/0 for 5 minutes.

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

- Send a repetitive pattern of all 1s through T3 interface 1/0/0 for 1440 minutes.

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

# Channelized T3 Commands

By default, a T3 interface on a 6-Port Channelized T3 line card is channelized into 28 T1 interfaces. This section describes the commands you use to customize and test a channelized T3 interface. This section discusses



To use the channels for subscriber traffic, you must configure the T1 and DS0 components. For more information, see the "Channel-Group Command for DS0 Timeslots and T1s" section on page 8-14.

Figure 8-1 shows the levels of configurable interface bandwidth that channelization offers.

## Entering Controller Configuration Mode for T3

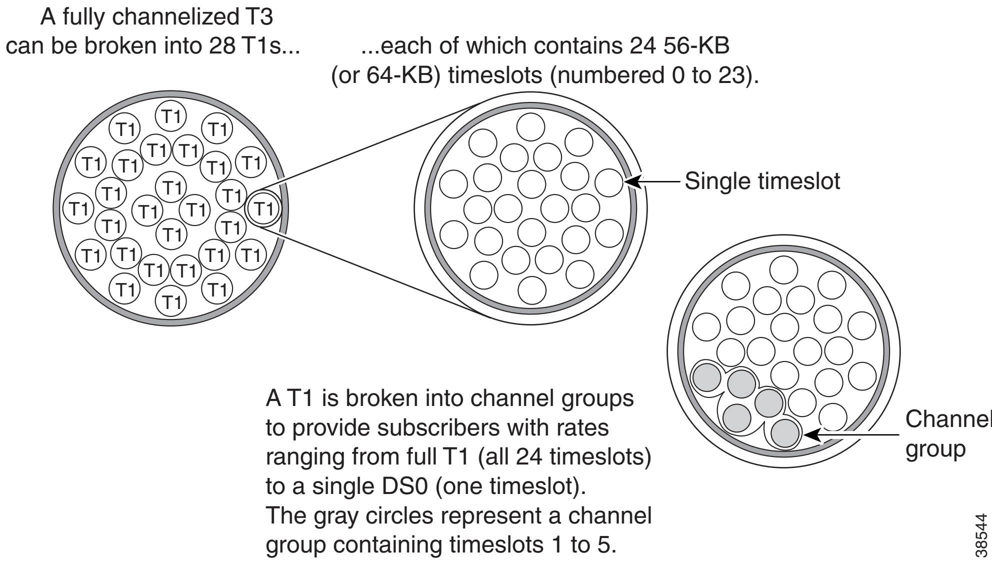
You may need to enter controller configuration mode for actions involving the T3 interface. For example, enter this mode to change an unchannelized T3 interface back to a channelized one or to set MDL messages.

**controller T3** *slot/sub-slot/port*

The following example shows how to enter controller configuration mode:

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

*Figure 8-1 Channelization of T3s*




# Configuring a T3 Interface as Channelized

A T3 interface is channelized by default. Use the **channelized** command if you had previously made the interface unchannelized and want to change the setting.

```
[no] channelized
channelized
```

**Caution** The **no channelized** command removes all channel groups from a channelized T3 interface. If you have already configured channel groups, use this command with caution.

The following example shows the creation of a channelized T3 interface:

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

# Specifying the Cable Length

To specify the cable length for the T3 controller, use the **cablelength** command.

```
cablelength feet
[no] cablelength feet
```

Where *feet* is a number from 0 to 450.

The default value is 249 feet.

Use the **no** form of this command to restore the default cable length.

In the following example, the cablelength value is set to 40 feet.

```
Router(config)# controller T3 1/0/0
Router(config-controller)# cablelength 40
```

**Note** This command causes the system to use one of two settings for impedance matching and pulse shaping, one setting for any cable length between 0 and 249 feet and another setting for any cable length greater than 250 feet. The exact value you enter is stored in the configuration file.

# Setting the Framing Type

To specify a framing type for the channelized T3 controller, use the **framing** command.

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

The default is auto-detect.

Use the **no** form of the command to return the framing type to its default.

You can instruct the 6-Port Channelized T3 line card to detect the framing type from the far end and transmit that same framing type as follows:

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

# Entering MDL Messages

You can configure maintenance data link (MDL) messages (as defined in the ANSI T1.107a-1990 specification) on the channelized T3 interface.

**Note** MDL messages are supported only when the T3 framing is set for C-bit parity. (See the "Setting the Framing Type" section on page 8-11.)

To configure MDL messages, use the **mdl** command.

**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*}

Where:

- **transmit path** enables transmission of the MDL path message.
- **transmit idle-signal** enables transmission of the MDL idle-signal message.
- **eic** is the equipment identification code (up to 10 characters).
- **lic** is the location identification code (up to 11 characters).
- **fic** is the frame identification code (up to 10 characters).
- **unit** is the unit identification code (up to 6 characters).
- **pfi** is the facility identification code to include in the MDL path message (up to 38 characters).
- **port** is the equipment port (which initiates the idle signal) to include in the MDL idle signal message (up to 38 characters).
- **generator** is the generator number to include in the MDL test signal message (up to 38 characters).

The default is that no MDL message is configured.

Use the **no** form of the command to remove an MDL message.

Examples of configuring MDL messages follow:

- Enable the MDL path message transmission.

```
Router(config)# controller T3 1/0/0
Router(config-controller)# mdl transmit path
```

- Enable the MDL idle signal message transmission.

```
Router(config-controller)# mdl transmit idle-signal
```

- Enable the MDL test signal message transmission.

```
Router(config-controller)# mdl transmit test-signal
```

- Enter the equipment identification code.

```
Router(config-controller)# mdl string eic router A
```

- Enter the location identification code.

```
Router(config-controller)# mdl string lic test network
```

- Enter the frame identification code.

```
Router(config-controller)# mdl string fic building b
```

- Enter the unit identification code.

```
Router(config-controller)# mdl string unit abc
```

- Enter the facility identification code.

```
Router(config-controller)# mdl string pfi string
```

- Enter the port number to send in the MDL idle signal message.

```
Router(config-controller)# mdl string port string
```

- Enter the generator number to send in the MDL test signal message.

```
Router(config-controller)# mdl string generator string
```

# Specifying the Idle Pattern

You can set a specific pattern to be transmitted between HDLC packets on all unconfigured timeslots that belong to a channelized T3 interface. To do so, use the **idle pattern** command.

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

Where *patterns* is a number in the range of 0x0 to 0xff (hexadecimal) or 0 to 255 (decimal). You can enter this value in either hexadecimal or decimal form. Values of 0 to 254 set the idle pattern to HDLC flags (0x7e); a value of 255 sets the pattern to 0xff (all ones).

**Note** Some systems interpret marks, 0xff, as an abort signal. Therefore, flags, 0x7e, is preferred.

The default idle pattern is 0x7e.

Use the **no** form of the command to return the idle pattern to its default value.

Examples:

- Set a hexadecimal idle pattern.

```
Router(config)# controller T3 1/0/0
Router(config-controller)# idle pattern 0x10
```

- Set a decimal idle pattern.

```
Router(config)# controller T3 1/0/0
Router(config-controller)# idle pattern 23
```

## Setting the Clock Source

At the prompt, set the internal or line clock source for the selected T3 controller using the **clock source** command. This command is set in controller configuration mode.

**clock source** {**internal** | **line**}

Where:

- **internal** specifies that the internal clock source is used
- **line** specifies that the network clock source is used

The default is clock source internal.

In this example, a T3 controller is instructed to use a line clock source.

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

## Configuring Loopback Mode

You can configure the T3 controller for a loopback mode using the **loopback** command.

**loopback** [**local** | **network** | **remote**]
[**no**] **loopback** [**local** | **network** | **remote**]

Local and network loopbacks are the same.

To cancel a loopback, use the **no** form of the command.

For more information on the **loopback** command, refer to the online *Cisco 10000 Series Internet Router Troubleshooting Guide*.

Examples:

- Configure the T3 controller for local loopback.

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

- Configure the T3 controller for remote loopback.

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

## Running Equipment Loopbacks

Use the **equipment loopback** command to run loopbacks in conjunction with remote equipment.

```
equipment [customer | network] loopback
[no] equipment [customer | network] loopback
```

Where:

- **customer** enables the line card to respond to remote T3 loopback commands from the remote T3 equipment.
- **network** causes the line card to ignore remote T3 loopback commands.

Use the **no** form of the command to terminate the loopback.

For more information on the **loopback** command, refer to the online *Cisco 10000 Series Internet Router Troubleshooting Guide*.

In the following example, an equipment network loopback is configured:

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

# Channel-Group Command for DS0 Timeslots and T1s

Each channelized T3 consists of 28 T1s, each of which is divided into 24 timeslots (or DS0s). To create an interface that supports subscriber traffic, you must combine the timeslots into channel groups. Use the **t1 channel-group** command to create a channel group.

Channel groups have these characteristics:

- A channel group is an interface.
- A channel group can be one timeslot to 24 timeslots in size.
  - A channel group that consists of 24 timeslots is a T1 interface.
  - A channel group that consists of fewer than 24 timeslots can be described as a fractional T1 interface.
- Each group of 24 timeslots can be divided into multiple fractional T1 interfaces.
- A channel group cannot be part of more than one T1.

Figure 8-1 illustrates channelization possibilities on the 6-Port Channelized T3 line card.

You create a logical channel group using the following command:

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

Where:

- *t1-number* is T1 interface number 1 to 28.
- **channel-group** *channel-group-number* identifies the channel group with any number from 0 to 23.

- **timeslots** *list-of-timeslots* can be 1 to 24 or a combination of subranges within 1 to 24. You can indicate a range using a hyphen, commas, or a combination of both. One timeslot equals one DS0. Refer to the examples below.
- **speed** {56 | 64} is an optional argument that specifies the speed of a timeslot as either 56 or 64 kbps. The default is 64. (The 56-kbps timeslots are generally used with older T1 equipment that does not support B8ZS and are associated with SF framing.)

Use the **no** form of the command to remove a logical channel group.

The following examples show how to use the **t1 channel-group** command:

- In this example, T1 interface 3 includes channel group 20 and consists of nine channelized timeslots:

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

  To enter interface configuration mode for this channel group, enter the following:

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

- In the following example, T1 interface 4 includes channel group 18 and consists of all 24 timeslots, creating a full T1 interface:

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

  To enter interface configuration mode for this channel group, you enter the following:

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

- In the following example, T1 interface number 5 is divided into three channel groups, which total 11 timeslots:

```
Router(config)# controller t3 1/0/0
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
```

- In the following example, channel group 20 is removed:

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

# Channelized T1 Commands

From controller configuration mode for a T3 interface, you can enter commands to modify aspects of a T1 interface. This section describes the commands for

- Setting the Framing Format, page 8-16
- Controlling Yellow Alarms, page 8-16
- Setting the Clock Source, page 8-17
- Configuring FDL, page 8-17
- Configuring a BER Test, page 8-17
- Configuring Loopback Mode, page 8-18

After you configure a T1 interface, you can add encapsulation, routing, and other instructions by entering interface configuration mode. For example:

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

# Setting the Framing Format

You can specify the T1 interface framing format using the following command:

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

Where:

- *t1-number* is T1 interface number 1 to 28.
- **framing** is either extended super frame (ESF) or super frame (SF). You can set SF hdlc-idle to 0x7E or 0xFF.
- **hdlc-idle** options allow you to set the idle pattern for the T1 interface to either 0x7e (the default) or 0xff.

The default framing format is extended super frame (ESF).

Use the **no** form of the command to return framing to its default value.

Examples:

- Set SF framing format for T1 interface 6.

```
Router(config)# controller T3 1/0/0
Router(config-controller)# t1 6 framing sf
```

- Set ESF framing format for T1 interface 16.

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

# Controlling Yellow Alarms

Use the **t1 yellow** command to turn the detection or generation of a yellow alarm on or off.

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

Where:

- *t1-number* is T1 interface 1 to 28
- **detection** means that the interface is told it is failing by the remote device, causing IOS to send a message to the console
- **generation** means that the interface notifies the remote device if it is failing, causing IOS to send a message to the console

When you select SF framing for a full T1 interface (24 timeslots) that uses the default speed of 64, consider using the **no** *t1-number* **yellow detection** command to turn off yellow alarm detection, because the yellow alarm can be incorrectly detected with SF framing.

In the following example, T1 interface 1 is set to yellow detection:

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

## Setting the Clock Source

You can set the internal or line (network) clock source for a T1 interface using the **controller** command.

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

Where:

- *t1-number* is T1 interface 1 to 28
- **internal** specifies that the internal clock source is used
- **line** specifies that the network clock source is used

The default is clock source internal.

In the following example, the interface is instructed to get its clock source from the line:

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

## Configuring FDL

You can enable 1-second transmissions of performance reports through the facility data link (FDL) according to the specification ANSI T1.403, on both ends of the T1 connection. To do so, use the following command:

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

Where *t1-number* is T1 interface 1 to 28.

Use the **no** form of the command to disable this feature.

**Note** You can use this command only when the T1 framing is ESF.

In the following example, FDL is enabled:

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

## Configuring a BER Test

You can configure an individual T1 interface to run an independent BER test. The test is used to check cables and solve signal problems in the field. To send a BER test pattern on a T1 interface, use the following command:

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

Where:

- *t1-number* is T1 interface number 1 to 28.
- *time* is 1 to 14,400 minutes.

- *pattern* is
  - 0s—repetitive test pattern of all zeros (00000..)
  - 1s—repetitive test pattern of all ones (11111..)
  - 2^11—pseudorandom test pattern (2048 bits long)
  - 2^15—pseudorandom O.151 test pattern (32,768 bits long)
  - 2^20-O153—pseudorandom O.153 test pattern (1,048,575 bits long)
  - 2^20-QRSS—pseudorandom QRSS O.151 test pattern (1,048,575 bits long)
  - 2^23—pseudorandom O.151 test pattern (8,388,607 bits long)
  - alt-0-1—repetitive alternating test pattern of zeros (0s) and ones (1s), for example 01010101
- **unframed** causes the BER test pattern to use the entire T1 bandwidth, including the T1 framing and payload bits. If **unframed** is omitted, the T1 is either SF or ESF framed as configured by the T1 n framing command, and the BER test pattern occupies only the T1 payload bits.

**Note** For each T3, you can run only one BER test at a time.

You can terminate a BER test at any time using the **no** form of the command.

For more information, refer to the online *Cisco 10000 Series Internet Router Troubleshooting Guide*.

Examples:

- Send a BER test pseudorandom pattern of 2^20 through T1 interface 10 for 5 minutes.

```
Router(config)# controller T3 1/0/0
Router(config-controller)# t1 10 bert pattern 2^20 interval 5 unframed
```

- Send a repetitive pattern of all 1s through T1 interface 10 for 14,400 minutes (240 hours).

```
Router(config)# controller T3 1/0/0
Router(config-controller)# t1 10 bert pattern 1s interval 14400 unframed
```

# Configuring Loopback Mode

If problems occur when you configure a T1 interface, you can troubleshoot the line card by using the following command from controller configuration mode:

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

Where:

- *t1-number* is T1 interface 1 to 28.
- **local** loops the router output data back toward the router at the T1 framer and sends an alarm indication signal (AIS) out toward the network.
- **network** {**line** | **payload**} loops the data back toward the network and automatically sets a local loopback at the HDLC controllers (line) or loops the payload data back toward the network and automatically sets a local loopback at the HDLC controller (payload).
- **remote line fdl** {**ansi** | **bellcore**} sends a repeating, 16-bit ESF data link keyword to the remote end, requesting that it enter into a network line loopback. You can specify an ANSI or Bellcore keyword.

**Note** Loopback codes are defined in ANSI T1.404-1989 Table 3 (Assigned Bit-Oriented ESF Data-Link Messages)

The 6-Port Channelized T3 line card sends the following code for remote line fdl ANSI loopback:

- Line Loopback Active Code: 0 000111 01111111
- Line Loopback Deactivate Code: 0 010010 01111111

The 6-Port Channelized T3 line card sends the following code for remote line FDL Bellcore (SmartJack) loopback:

- Network Use (loopback active) Code: 0 001001 01111111
- Network Use (loopback deactivate) Code: 0 010010 01111111

- **remote line inband** sends a repeating inband pattern to the remote end, requesting entry into a network line loopback. The inband loopback request overwrites all data in the T1 with the loop request pattern. The remote end responds to this code only after it receives the pattern continuously for at least five seconds.
  - Repeating Inband Loopback Activate Code: 00001
  - Repeating Inband Loopback Deactivate Code: 001
- **remote payload** [**fdl**] [**ansi**] sends a repeating, 16-bit ESF data link code word to the remote end, requesting entry into a network payload loopback. Using **fdl** and **ansi** enables the remote payload facility data link (FDL) ANSI bit loopback on the T1 channel.

**Note** Loopback codes are defined in ANSI T1.404-1989 Table 3 (Assigned Bit-Oriented ESF Data-Link Messages)

The 6-Port Channelized T3 line card sends the following code for remote payload fdl ANSI loopback:

- Payload Loopback Active Code: 0 001010 011111111
- Payload Loopback Deactivate Code: 0 011001 011111111

Use the **no** form of the command to terminate a loopback.

For more information on this command, refer to the online *Cisco 10000 Series Internet Router Troubleshooting Guide*.

Examples:

- Configure the T3 controller for local loopback on T1 interface 1.

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

- Configure the T3 controller for remote FDL ANSI loopback on T1 interface 1.

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

# PART 3

# Fast Ethernet and Gigabit Ethernet Line Cards

CHAPTER 9

# 8-Port Fast Ethernet Half-Height Line Card Configuration

This chapter describes procedures for configuring the Cisco 10000 series 8-Port Fast Ethernet Half-Height line card, hereafter known as the 8-Port Fast Ethernet Half-Height line card.

The 8-Port Fast Ethernet Half-Height line card contains eight 100BASE-TX ports. Each port autonegotiates between half and full duplex mode. Each port supports 100BASE-T, but does not support 10BASE-T.

This chapter contains the following sections:



**Note** For information about installing half-height line cards in subslot locations, see the *Cisco 10000 Series Router Line Card Hardware Installation Guide*.

## Software Support

Table 9-1 shows the minimum Cisco IOS release on each release train that supports the 8-Port Fast Ethernet Half-Height line card.

***Table 9-1 8-Port Fast Ethernet Half-Height Line Card Software Support***

| Required PRE | Minimum Cisco IOS Releases |
|---|---|
| PRE1 | Cisco IOS Release 12.0(23)S and later releases of Cisco IOS 12.0S |
| PRE2 | Cisco IOS Release 12.2(15)BX and later releases of Cisco IOS 12.2BX<br>Cisco IOS Release 12.3(7)XI and later releases of Cisco IOS 12.3 XI<br>Cisco IOS Release 12.2(28)SB and later releases of Cisco IOS 12.2SB |

## Checking Hardware and Software Compatibility

The PRE installed in the Cisco 10000 series router chassis must support the Cisco IOS software running on the router. Use the **show version** command to check the PRE version installed.

To see if a feature is supported by a Cisco IOS release, to locate the software document for that feature, or to check the minimum software requirements of Cisco IOS software with the hardware installed on your router, Cisco maintains the Software Advisor tool on Cisco.com at
http://www.cisco.com/cgi-bin/Support/CompNav/Index.pl

This tool does not verify whether line cards within a system are compatible, but does provide the minimum Cisco IOS requirements for individual hardware line cards, modules, or options.

You must be a registered user on Cisco.com to access this tool.

# Default Values

Table 9-2 lists default values for the 8-Port Fast Ethernet Half-Height line card.

*Table 9-2 8-Port Fast Ethernet Half-Height Line Card Defaults*

| Parameter | Default Setting | Description |
|---|---|---|
| Speed | 100 Mbps | You cannot use the **speed** command. The speed is advertised as 100 Mbps to the remote device. |
| Duplex | Autonegotiated | See the "Specifying Full or Half Duplex Mode" section on page 9-4 to manually set the duplex value.<br>**Note** The Fast Ethernet line card only responds to 802.3x pause frames from another device when it autonegatiates the duplex mode (the default). The line card does not support 802.3x flow control when you manually set half-duplex or full-duplex mode. |

# Management Port

The Fast Ethernet interface 0/0/0 is reserved for management purposes only and cannot be used for such things as customer traffic or L2TP establishment. This interface also does not allow you to configure subinterfaces. However, the interface does allow **radius**, **snmp**, or **telnet** commands.

# Configuring the Interface

To configure the 8-Port Fast Ethernet Half-Height line card interface, perform the following tasks:

- Specifying the Interface for Configuration
- Configuring the IP Address
- Specifying Full or Half Duplex Mode (optional)
- Configuring 802.1Q VLAN Encapsulation (optional)
- Configuring Routing
- Saving the Configuration
- Viewing the Configuration (optional)
- Disabling the Interface (optional)

For quick reference, refer to the configuration task commands below:

```
Router# configure terminal
Router(config)# interface FastEthernet slot/subslot/port
Router(config-if)# ip address ip-address mask
Router(config-if)# [no] {full-duplex | half-duplex} (optional)
Router(config-if)# [no] encapsulation dot1q vlan-id [native] (optional)
Router(config-if)# exit
Router(config)# router rip (For instance, use to configure RIP)
Router(config-router)# network ip-address
Router(config-router)# end
Router# copy running-config startup-config
```

## Specifying the Interface for Configuration

Follow these steps to specify the interface that you want to configure:

**Step 1** Enter global configuration mode by entering:

```
Router# configure terminal
```

**Step 2** Specify the interface you want to configure by entering:

```
Router(config)# interface FastEthernet slot/subslot/port
```

- *slot*—1 to 8 on the Cisco 10008 series router; 1 to 5 on the Cisco 10005 series router (slots 0A and 0B are used for the PREs).
- *subslot*—0 or 1. 0 is the top subslot.
- *port*—0 to 7.

**Note** Fast Ethernet 0/0/0 is reserved for management purposes only. Do not use this interface for such things as customer traffic or L2TP establishment. This interface does not allow you to configure subinterfaces.

## Configuring the IP Address

To set the IP address of the interface, enter the following command:

```
Router(config-if)# ip address ip-address mask
```

## Specifying Full or Half Duplex Mode

If the negotiation of the duplex mode fails, and a duplex mode mismatch occurs, manually set the duplex mode.

**Note** The 8-Port Fast Ethernet Half-Height line card only responds to 802.3x pause frames from another device when it autonegotiates the duplex mode (the default). The line card does not support 802.3x flow control when you manually set half-duplex or full-duplex mode.

Set the duplex mode using the following command:

```
Router(config-if)# [no] {half-duplex | full-duplex}
```

The **no** option disables the duplex setting. If no duplex mode is set, the mode reverts to autonegotiation.

## Configuring 802.1Q VLAN Encapsulation

The 8-Port Fast Ethernet Half-Height line card supports 802.1Q VLAN encapsulation. The card does not support Inter-Switch Link (ISL) encapsulation. To set 802.1Q encapsulation, enter:

```
Router(config-if)# [no] encapsulation dot1q vlan-id [native]
```

- *vlan-id*—The VLAN identifier, between 1 and 1000.
- **native**—Sets the Port VLAN ID (PVID) value of the port to the *vlan-id* value.
- **no**—Disables encapsulation.

## Configuring Routing

See the *Cisco IOS Command References* for information about configuring routing protocols, such as RIP or OSPF.

# Saving the Configuration

After completing your configuration changes, save the configuration by following these steps:

**Step 1** To exit configuration mode, either enter **Ctrl-Z** or enter the following command:

```
Router(config-if)# end
```

**Step 2** Write the new configuration to memory by entering:

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

When the configuration is stored, an OK message appears.

# Viewing the Configuration

You can check the configuration using the following command:

```
Router# show interface FastEthernet slot/subslot/port
```

# Disabling the Interface

The interface is enabled by default. To disable it, enter:

```
Router(config-if)# shutdown
```

```
To enable the interface, enter:
Router(config-if)# no shutdown
```

CHAPTER 10

# 1-Port Gigabit Ethernet Half-Height Line Card Configuration

This chapter describes procedures for configuring the Cisco 10000 series 1-Port Gigabit Ethernet Half-Height line card, hereafter known as the 1-Port Gigabit Ethernet Half-Height line card.

The 1-Port Gigabit Ethernet Half-Height line card contains a single Gigabit Ethernet port that provides a trunk uplink to switches and core routers.

This chapter contains the following sections:



**Note** For information about installing half-height line cards in subslot locations, see the *Cisco 10000 Series Router Line Card Hardware Installation Guide*.

## Software Support

Table 10-1 shows the minimum Cisco IOS release on each release train that supports the 1-Port Gigabit Ethernet Half-Height line card.

***Table 10-1 1-Port Gigabit Ethernet Half-Height Line Card Software Support***

| Required PRE | Minimum Cisco IOS Releases |
|---|---|
| PRE1 | Cisco IOS Release 12.0(23)S and later releases of Cisco IOS 12.0S |
| PRE2 | Cisco IOS Release 12.2(15)BX and later releases of Cisco IOS 12.2BX<br>Cisco IOS Release12.3(7)XI and later releases of Cisco IOS 12.3 XI<br>Cisco IOS Release 12.2(28)SB and later releases of Cisco IOS 12.2SB |
| PRE3 | Cisco IOS Release 12.2(31)SB2 and later releases of Cisco IOS 12.2SB |

## Checking Hardware and Software Compatibility

The PRE installed in the Cisco 10000 series router chassis must support the Cisco IOS software running on the router. Use the **show version** command to check the PRE version installed.

To see if a feature is supported by a Cisco IOS release, to locate the software document for that feature, or to check the minimum software requirements of Cisco IOS software with the hardware installed on your router, Cisco maintains the Software Advisor tool on Cisco.com at http://www.cisco.com/cgi-bin/Support/CompNav/Index.pl

This tool does not verify whether line cards within a system are compatible, but does provide the minimum Cisco IOS requirements for individual hardware line cards, modules, or options.

You must be a registered user on Cisco.com to access this tool.

# Default Values

Table 10-2 lists default values for the1-Port Gigabit Ethernet Half-Height line card. This table also includes the command used for modifying a default value and indicates if a value needs to be the same (or opposite) on the remote end of the connection.

***Table 10-2 1-Port Gigabit Ethernet Half-Height Line Card Defaults***

| Command Name | Default Setting | Command Syntax | Remote Side Setting |
|---|---|---|---|
| auto-negotiation | Enabled | [**no**] **negotiation auto** | Same |
| mtu (maximum transmission unit) | • 1500 to 4470 (full-height)<br>• 1500 to 9180 (half-height) | **mtu** *size* | Same |

# Interface Syntax

To specify an interface number in a configuration command, use the syntax in Table 10-3 to identify interfaces on the 1-Port Gigabit Ethernet Half-Height line card.

***Table 10-3 1-Port Gigabit Ethernet Half-Height Interface Syntax***

| Slot | Subslot | Port |
|---|---|---|
| 1 to 8/ | 0 or 1/ | 0 |

The following example shows the syntax for configuring a 1-Port Gigabit Ethernet Half-Height line card in slot 1:

```
Router(config)# interface GigabitEthernet 1/0/0
Router(config-if)#
```

# Configuring an Interface

Use the following procedure to create a basic configuration (enabling an interface and specifying IP routing) for a 1-Port Gigabit Ethernet Half-Height line card interface. Be prepared with the information you need, such as the interface IP address.

**Step 1** At the global configuration prompt, specify the new interface to configure by entering the **interface** command.

```
Router(config)# interface GigabitEthernet 1/0/0
```

**Step 2** Assign an IP address and subnet mask to the interface using the **ip address** configuration subcommand, as in the following example:

```
Router(config-if)# ip address 192.168.172.12 255.255.255.0
```

**Step 3** If necessary, modify the 1-Port Gigabit Ethernet Half-Height line card line card configuration or that of the remote device to ensure that, where appropriate, they use the same settings. For more information, refer to the Remote Side Setting column in Table 10-2.

**Step 4** Add any other configuration subcommands required for the enabling of routing protocols and adjust the interface characteristics.

**Step 5** Enter the **no shutdown** command to enable the interface.

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

**Step 6** When you have included all of the configuration subcommands to complete the configuration, enter **Ctrl-Z** to exit configuration mode.

**Step 7** Write the new configuration to memory.

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

When the configuration is stored, the system displays an OK message.

After you complete the configuration, you can check the configuration using the **show interface GigabitEthernet** *slot/subslot/port* command.

# Configuration Commands

This section lists some of the commands that you can use to further customize your 1-Port Gigabit Ethernet Half-Height line card configuration. The commands described in this section are for:

- Specifying Auto-Negotiation, page 10-3
- Assigning a MAC Address, page 10-4
- Setting and Changing Loopback Mode, page 10-4

## Specifying Auto-Negotiation

To enable auto-negotiation, enter the **negotiation auto** command.

```
negotiation auto
[no] negotiation auto
```

The default configuration is auto-negotiation enabled.

Use the **no** form of the command to disable negotiation.

**Note** Both sides must be enabled for auto-negotiation, or the link does not come up.

In the following example, negotiation is disabled:

```
Router(config)# interface GigabitEthernet 1/0/0
Router(config-if)# no negotiation auto
```

# Assigning a MAC Address

To assign a specific MAC address to a 1-Port Gigabit Ethernet Half-Height line card interface, use the **mac-address** command:

**mac-address** *address*
[**no**] **mac-address** *address*

Where *address* assigns a user-defined MAC address to a Gigabit Ethernet interface. The MAC address takes the form xxxx.xxxx.xxxx, where xxxx represents a 2-byte hexadecimal value.

**Tip** Use the **mac-address** command to preserve a MAC address when you move a card to a new slot or chassis. You may want to preserve the MAC address if you have an access list that refers to it. If you use this command, you should change the address on the original slot so that you do not have two slots with the same address.

Use **no mac-address** to restore the default MAC address.

Use the **show interfaces GigabitEthernet** *slot/subslot/port* command to see the MAC address.

In the following example, a new MAC address is assigned:

```
Router(config)# interface GigabitEthernet 1/0/0
router(config-if)# mac-address 6000.0001.0003
```

# Setting and Changing Loopback Mode

To set the loopback mode on a 1-Port Gigabit Ethernet Half-Height line card interface, use the **loopback** command in interface configuration mode.

**loopback** [**internal** | **external**]
[**no**] **loopback** [**internal** | **external**]

Where:

- **external** runs a loopback that requires a loopback connector.
- **internal** runs a loopback at the MAC controller.

Use the **no** form of the command to stop the loopback.

For more information about this command, refer to the *Cisco 10000 Series Internet Router Troubleshooting Guide*.

In the following example, an internal loopback mode is defined for a 1-Port Gigabit Ethernet Half-Height line card interface:

```
router(config)# interface GigabitEthernet 1/0/0
router(config-if)# loopback internal
```

CHAPTER 11

# 1-Port Gigabit Ethernet Line Card Configuration

This chapter describes procedures for configuring the Cisco 10000 series 1-Port Gigabit Ethernet line card, hereafter known as the 1-Port Gigabit Ethernet line card.

The 1-Port Gigabit Ethernet line card provides:

- A trunk uplink to the core for Gigabit Ethernet aggregation
- The Cisco 10000 system with an IEEE 802.3z-compliant Ethernet interface running at 1 Gbps in full-duplex mode

This chapter contains the following sections:



# Software Support

Table 11-1 shows the minimum Cisco IOS release on each release train that supports the 1-Port Gigabit Ethernet line card.

*Table 11-1 1-Port Gigabit Ethernet Line Card Software Support*

| Required PRE | Minimum Cisco IOS Releases |
|---|---|
| PRE1 | Cisco IOS Release 12.0(22)S and later releases of Cisco IOS 12.0S |
| PRE2 | Cisco IOS Release 12.0(22)S and later releases of Cisco IOS 12.0S<br>Cisco IOS Release 12.3(7)XI and later releases of Cisco IOS Release 12.3XI<br>Cisco IOS Release 12.2(28)SB and later releases of Cisco IOS 12.2SB |
| PRE3 | Cisco IOS Release 12.2(31)SB2 and later releases of Cisco IOS 12.2SB |

## Checking Hardware and Software Compatibility

The PRE installed in the Cisco 10000 series router chassis must support the Cisco IOS software running on the router. Use the **show version** command to check the PRE version installed.

To see if a feature is supported by a Cisco IOS release, to locate the software document for that feature, or to check the minimum software requirements of Cisco IOS software with the hardware installed on your router, Cisco maintains the Software Advisor tool on Cisco.com at http://www.cisco.com/cgi-bin/Support/CompNav/Index.pl

This tool does not verify whether line cards within a system are compatible, but does provide the minimum Cisco IOS requirements for individual hardware line cards, modules, or options.

You must be a registered user on Cisco.com to access this tool.

# Default Values

Table 11-2 lists default values for the 1-Port Gigabit Ethernet line card. This table also includes the command used for modifying a default value and indicates if a value needs to be the same (or opposite) on the remote end of the connection.

***Table 11-2 1-Port Gigabit Ethernet Line Card Defaults***

| Command Name | Default Setting | Command Syntax | Remote Side Setting |
|---|---|---|---|
| auto-negotiation | Enabled | [**no**] **negotiation auto** | Same |
| mtu (maximum transmission unit) | • 1500 to 4470 (full-height)<br>• 1500 to 9180 (half-height) | **mtu** *size* | Same |

# Interface Syntax

To specify an interface number in a configuration command, use the syntax in Table 11-3 to identify interfaces on the 1-Port Gigabit Ethernet line card.

***Table 11-3 1-Port Gigabit Ethernet Interface Syntax***

| Slot | Subslot | Port |
|---|---|---|
| 1 to 8/ | 0 or 1/ | 0 |

The following example shows the syntax for configuring a 1-Port Gigabit Ethernet line card in slot 1:

```
Router(config)# interface GigabitEthernet 1/0/0
Router(config-if)#
```

# Configuring an Interface

Use the following procedure to create a basic configuration (enabling an interface and specifying IP routing) for a 1-Port Gigabit Ethernet line card interface. Be prepared with the information you need, such as the interface IP address.

**Step 1** At the global configuration prompt, specify the new interface to configure by entering the **interface** command.

```
Router(config)# interface GigabitEthernet 1/0/0
```

**Step 2** Assign an IP address and subnet mask to the interface using the **ip address** configuration subcommand, as in the following example:

```
Router(config-if)# ip address 192.168.172.12 255.255.255.0
```

**Step 3** If necessary, modify the 1-Port Gigabit Ethernet line card line card configuration or that of the remote device to ensure that, where appropriate, they use the same settings. For more information, refer to the Remote Side Setting column in Table 11-2.

**Step 4** Add any other configuration subcommands required for the enabling of routing protocols and adjust the interface characteristics.

**Step 5** Enter the **no shutdown** command to enable the interface.

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

**Step 6** When you have included all of the configuration subcommands to complete the configuration, enter **Ctrl-Z** to exit configuration mode.

**Step 7** Write the new configuration to memory.

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

When the configuration is stored, the system displays an OK message.

After you complete the configuration, you can check the configuration using the **show interface GigabitEthernet** *slot/subslot/port* command.

# Configuration Commands

This section lists some of the commands that you can use to further customize your 1-Port Gigabit Ethernet line card configuration. The commands described in this section are for:


- Specifying Auto-Negotiation, page 11-4
- Assigning a MAC Address, page 11-4
- Setting and Changing Loopback Mode, page 11-5

## Specifying Auto-Negotiation

To enable auto-negotiation, enter the **negotiation auto** command.

```
negotiation auto
[no] negotiation auto
```

The default configuration is auto-negotiation enabled.

Use the **no** form of the command to disable negotiation.

**Note** Both sides must be enabled for auto-negotiation, or the link does not come up.

In the following example, negotiation is disabled:

```
Router(config)# interface GigabitEthernet 1/0/0
Router(config-if)# no negotiation auto
```

## Assigning a MAC Address

To assign a specific MAC address to a 1-Port Gigabit Ethernet line card interface, use the **mac-address** command:

```
mac-address address
[no] mac-address address
```

Where *address* assigns a user-defined MAC address to a Gigabit Ethernet interface. The MAC address takes the form xxxx.xxxx.xxxx, where xxxx represents a 2-byte hexadecimal value.

**Tip** Use the **mac-address** command to preserve a MAC address when you move a card to a new slot or chassis. You may want to preserve the MAC address if you have an access list that refers to it. If you use this command, you should change the address on the original slot so that you do not have two slots with the same address.

Use **no mac-address** to restore the default MAC address.

Use the **show interfaces GigabitEthernet** *slot/subslot/port* command to see the MAC address.

In the following example, a new MAC address is assigned:

```
Router(config)# interface GigabitEthernet 1/0/0
router(config-if)# mac-address 6000.0001.0003
```

## Setting and Changing Loopback Mode

To set the loopback mode on a 1-Port Gigabit Ethernet line card interface, use the **loopback** command in interface configuration mode.

```
loopback [internal | external]
[no] loopback [internal | external]
```

Where:

- **external** runs a loopback that requires a loopback connector.
- **internal** runs a loopback at the MAC controller.

Use the **no** form of the command to stop the loopback.

For more information about this command, refer to the *Cisco 10000 Series Internet Router Troubleshooting Guide*.

In the following example, an internal loopback mode is defined for a 1-Port Gigabit Ethernet line card interface:

```
router(config)# interface GigabitEthernet 1/0/0
router(config-if)# loopback internal
```

# Part 4

## Packet over SONET Line Cards

CHAPTER 12

# 6-Port OC-3/STM-1 Packet over SONET Line Card Configuration

This chapter describes the procedures for configuring the Cisco 10000 series 6-Port OC-3/STM-1 Packet over SONET line card, hereafter known as the 6-Port OC-3/STM-1 Packet over SONET line card.

The 6-Port OC-3/STM-1 Packet over SONET line card provides 6 STS-3/STM-1 Packet over SONET trunk uplinks that support throughput of up to 155.52 Mbps each over a SONET/ITU-T Synchronous Digital Hierarchy (SDH) interface. Each port (0 to 5) has a single-mode optical fiber intermediate-reach LC connector.

This chapter contains the following sections:



## Software Support

Table 12-1 shows the minimum Cisco IOS release on each release train that supports the 6-Port OC-3/STM-1 Packet over SONET line card.

***Table 12-1 6-Port OC-3/STM-1 Packet over SONET Line Card Software Support***

| Required PRE | Minimum Cisco IOS Releases |
|---|---|
| PRE1 | Cisco IOS Release12.0(22)S and later releases of Cisco IOS Release12.0S |
| PRE2 | Cisco IOS Release 12.0(20)ST and later releases of Cisco IOS 12.0ST<br>Cisco IOS Release12.0(22)S and later releases of Cisco IOS Release12.0S<br>Cisco IOS Release 12.2(8)BZ and later releases of Cisco IOS Release 12.2BZ<br>Cisco IOS Release 12.3(7)XI and later releases of Cisco IOS 12.3XI<br>Cisco IOS Release 12.2(28)SB and later releases of Cisco IOS 12.2SB |

## Checking Hardware and Software Compatibility

The PRE installed in the Cisco 10000 series router chassis must support the Cisco IOS software running on the router. Use the **show version** command to check the PRE version installed.

To see if a feature is supported by a Cisco IOS release, to locate the software document for that feature, or to check the minimum software requirements of Cisco IOS software with the hardware installed on your router, Cisco maintains the Software Advisor tool on Cisco.com at http://www.cisco.com/cgi-bin/Support/CompNav/Index.pl

This tool does not verify whether line cards within a system are compatible, but does provide the minimum Cisco IOS requirements for individual hardware line cards, modules, or options.

You must be a registered user on Cisco.com to access this tool.

# Default Values

Table 12-2 lists default values and command syntax for the 6-Port OC-3/STM-1 Packet over SONET line card. Table 12-2 includes the command to modify a default value, and indicates if a value needs to be the same (or opposite) on the remote end of the connection.

***Table 12-2 6-Port OC-3/STM-1 Packet over SONET Line Card Defaults***

| Command Name | Default Setting | Command Syntax | Remote Side Setting |
|---|---|---|---|
| bandwidth | 155,000 | **bandwidth** *kilobits* | Same. |
| clock source | line | **clock source** [**line** \| **internal**] | At least one side must be set to *internal.* |
| crc | 16 | **crc** [**16** \| **32**] | Same. |
| encapsulation | HDLC | **encapsulation** [**hdlc** \| **ppp**] | Same. |
| keepalive | no keepalive | [**no**] **keepalive** *period* | Same. |
| loopback | no loopback | [**no**] **loopback [line\|internal]** | Same |
| mtu (maximum transmission unit) | 4470 | **mtu** *size* | Same. |
| pos framing | SONET | **pos framing** [**sonet** \| **sdh**] | Same. |
| pos scramble-atm | No scrambling | [**no**] **pos scramble-atm** | Same. |
| pos flag (SONET overhead) | c2—0xcf<br>j0—0x01<br>s1s0—0 | **pos flag** [**c2** \| **j0** \| **s1s0**] *value* | Same. |

# Interface Syntax

To specify an interface number in a configuration command, use the syntax in Table 12-3 to identify interfaces on the 6-Port OC-3/STM-1 Packet over SONET line card.

*Table 12-3 6-Port OC-3/STM-1 Packet over SONET Line Card Interface Syntax*

| Type of Interface | Slot | Subslot | Port |
|---|---|---|---|
| Packet over SONET interface | 1 to 8 | 0 | 0 to 5 |

The following example shows the syntax for configuring a 6-Port OC-3/STM-1 Packet over SONET line card in slot 1:

```
Router(config)# interface pos 1/0/0
```

# Configuring the Interface

After you verify that the 6-Port OC-3/STM-1 Packet over SONET line card is installed correctly, use the following procedure to configure the new interface. Be prepared with the information you will need, such as the interface IP address.

Use the following procedure to create a basic configuration—enable an interface and specify IP routing.

---

**Step 1** At the global configuration prompt, specify the new interface to configure by entering the **interface pos** command and interface address. For example:

```
Router(config)# interface pos 1/0/1
```

**Step 2** Assign an IP address and a subnet mask to the interface using the **ip address** configuration subcommand, as in the following example:

```
Router(config-if)# ip address 192.168.255.255 255.255.255.0
```

**Step 3** Specify either HDLC or PPP encapsulation. For example:

```
Router(config-if)# encapsulation hdlc
```

**Step 4** If necessary, modify the 6-Port OC-3/STM-1 Packet over SONET line card configuration or that of the remote device to ensure that, where appropriate, they use the same settings. For more information, refer to the Remote Side Setting column in Table 12-2.

**Step 5** Add any other configuration subcommands required for the enabling of routing protocols and adjust the interface characteristics.

**Step 6** Enter the **no shutdown** command to enable the interface.

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

**Step 7** When you have included all of the configuration subcommands to complete the configuration, enter **Crtl-Z** to exit configuration mode.

**Step 8** Write the new configuration to memory.

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

When the configuration has been stored, the system displays an OK message.

After you have completed your configuration, you can check it using the **show interface pos** *slot/subslot/port* command.

# Configuration Commands

The following sections present some of the commands that you can use to customize your 6-Port OC-3/STM-1 Packet over SONET line card configuration.

**Note** These configuration commands are specific to each port on the 6-Port OC-3/STM-1 Packet over SONET line card.

## Setting the Clock Source

You can set the internal or line clock source by using the **clock source** command.

**clock source** {**internal** | **line**}

Where:

- **internal** specifies that the internal clock source is used
- **line** specifies that the network clock source is used

The default is clock source line.

In this example, the line card is instructed to use a line clock source.

```
Router(config)# interface pos 3/0/2
Router(config-if)# clock source line
```

## Configuring Framing

You can use the **pos framing** command to set framing to SONET STS-3c or SDH STM-1 framing.

**pos framing** [**sdh** | **sonet**]
[**no**] **pos framing**

The default is SONET.

**Note** Your system must support SDH to use this option.

Use the **no** form of the command to restore the default framing mode.

In the following example, the framing type is set to SONET:

```
Router(config)# interface pos 5/0/1
Router(config-if)# no pos framing
```

# Specifying SONET Overhead

You can use the **pos flag** command to assign values for specific elements of the frame header. This command is typically used to meet a standards requirement or to ensure interoperability with another vendor's equipment.

```
pos flag [c2 value] [j0 value] [s1s0 value]
[no] pos flag [c2 value] [j0 value] [s1s0 value]
```

Where:

- **c2** is a path signal identifier, and *value* is one of the following:
  - 0xCF for PPP or HDLC without scrambling
  - 0x16 for PPP or HDLC with scrambling
- **j0** is the section trace byte, and *value* is 0x01 for interoperability with some SDH devices in Japan
- **s1s0** is part of the payload pointer byte, and *value* is 0 for OC-3c and 2 for AU-4

Use the **no** form of the command to restore the default values.

In the following example, the c2 bit is set to 0xCF.

```
Router(config)# interface pos 4/0/02
Router(config-if)# pos flag c2 0xCF
```

# Controlling the S1 SONET Overhead Byte

In most situations, the default value for the S1 SONET overhead byte (0x0) does not need to be changed. Refer to the SONET standards for information about the possible values for the S1 SONET overhead byte and the definition of each value.

## Controlling a Transmitted S1 Overhead Byte

In Cisco IOS Release 12.2(28)SB, use the **pos flag s1-byte tx** command in interface configuration mode to control the transmission of the S1 SONET overhead byte.

**pos flag s1-byte tx** *value*

Where:

- *value* is in the range of 0x0 to 0xF
- 0x0 is the default value

In the following example the S1 SONET overhead byte is set to 0xF:

```
pos flag s1-byte tx 0xF
```

## Reacting to a Received S1 Overhead Byte

In Cisco IOS Release 12.2(28)SB, use the **pos flag s1-byte rx-communicate** command to direct the router to switch the clock source to internal when it receives an S1 SONET overhead byte with a value of 0xF. When the S1 SONET overhead byte changes from 0xF to any other value, the clock source reverts back to the clock source specified in the user configuration.

The S1 overhead byte is ignored by the receiving router unless the **pos flag s1-byte rx-communicate** command is issued.

**pos flag s1-byte rx-communicate**

**no pos flag s1-byte rx-communicate**

The following example directs the router to switch to internal clocking when it receives an S1 SONET overhead byte with a value of 0xF:

```
pos flag s1-byte rx-communicate
```

The default for the **pos flag s1-byte rx-communicate** command is disabled or off.

# Configuring Packet over SONET SPE Scrambling

You can use the **pos scramble-atm** command to scramble the Packet over SONET synchronous payload envelope (SPE). SONET payload scrambling applies a self-synchronous scrambler to the SPE of the interface to ensure sufficient bit transition density.

```
pos scramble-atm
[no] pos scramble-atm
```

The default is no Packet over SONET SPE scrambling.

Use the **no** form of the command to disable scrambling.

In the following example, scrambling is enabled:

```
Router(config)# interface pos 3/0/3
Router(config-if)# pos scramble-atm
```

# Configuring Loopback Testing

To enable loopback testing of data transmitted from the PRE line card to the 6-Port OC-3/STM-1 Packet over SONET line card and back, use the **loopback** command in interface configuration mode.

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

Where:

- **loopback line** loops any data received at the 6-Port OC-3/STM-1 Packet over SONET line card network interface back into the network
- **loopback internal** loops any data sent out the 6-Port OC-3/STM-1 Packet over SONET line card back into the 6-Port OC-3/STM-1 Packet over SONET line card

Use the **no** form of the command to stop the loopback test.

For more information on loopbacks, refer to the online *Cisco 10000 Series Internet Router Troubleshooting Guide*.

In the following example, a loopback is set for the 6-Port OC-3/STM-1 Packet over SONET line card in slot 5:

```
Router(config)# interface pos 5/0/0
Router(config-if)# loopback line
```

# Configuring APS

The 6-Port OC-3/STM-1 Packet over SONET line card supports port-level automatic protection switching (APS). For information about APS redundancy commands., see the *Cisco 10000 Series Router Software Configuration Guide*.

CHAPTER 13

# 1-Port OC-12 Packet over SONET Line Card Configuration

This chapter describes procedures for configuring the Cisco 10000 series 1-Port OC-12 Packet over SONET line card, hereafter known as the 1-Port OC-12 Packet over SONET line card.

The 1-Port OC-12 Packet over SONET line card provides a trunk uplink that supports up to 622 Mbps throughput over a standard SONET/ITU-T Synchronous Digital Hierarchy (SDH) interface using a single-mode fiber intermediate-reach SC connector.

This chapter contains the following sections:



## Software Support

Table 13-1 shows the minimum Cisco IOS release on each release train that supports the 1-Port OC-12 Packet over SONET line card.

***Table 13-1 1-Port OC-12 Packet over SONET Line Card Software Support***

| Required PRE | Minimum Cisco IOS Releases |
|---|---|
| PRE1 | Cisco IOS Release 12.0(22)S and later releases of Cisco IOS 12.0S |
| PRE2 | Cisco IOS Release 12.0(22)S and later releases of Cisco IOS 12. 0S<br>Cisco IOS Release 12.3(7)XI and later releases of Cisco IOS 12.3XI<br>Cisco IOS Release 12.2(28)SB and later releases of Cisco IOS 12.2SB |

## Checking Hardware and Software Compatibility

The PRE installed in the Cisco 10000 series router chassis must support the Cisco IOS software running on the router. Use the **show version** command to check the PRE version installed.

To see if a feature is supported by a Cisco IOS release, to locate the software document for that feature, or to check the minimum software requirements of Cisco IOS software with the hardware installed on your router, Cisco maintains the Software Advisor tool on Cisco.com at http://www.cisco.com/cgi-bin/Support/CompNav/Index.pl

This tool does not verify whether line cards within a system are compatible, but does provide the minimum Cisco IOS requirements for individual hardware line cards, modules, or options.

You must be a registered user on Cisco.com to access this tool.

# Default Values

Table 13-2 lists default values for the 1-Port OC-12 Packet over SONET line card. This table includes the command used for modifying a default value and indicates whether a value needs to be the same (or opposite) on the remote end of the connection.

***Table 13-2 1-Port OC-12 Packet over SONET Line Card Defaults***

| Command Name | Default Setting | Command Syntax | Remote Side Setting |
|---|---|---|---|
| bandwidth | 622000 | **bandwidth** *kilobits* | Same. |
| clock source | line | **clock source** [**line** \| **internal**] | At least one side must be set to *internal.* |
| crc | 32 | **crc** [**16** \| **32**] | Same. |
| encapsulation | HDLC | **encapsulation** [**hdlc** \| **ppp**] | Same. |
| keepalive | no keepalive | [**no**] **keepalive** *period* | Same. |
| mtu (maximum transmission unit) | 4470 | **mtu** *size* | Same. |
| pos framing | SONET | **pos framing** [**sonet** \| **sdh**] | Same. |
| pos scramble-atm | No scrambling | [**no**] **pos scramble-atm** | Same. |
| pos flag (SONET overhead) | c2—0xcf<br>j0—0x01<br>s1s0—0 | **pos flag** [**c2** \| **j0** \| **s1s0**] *value* | Same. |

# Interface Syntax

To specify an interface number in a configuration command, use the syntax in Table 13-3 to identify interfaces on the 1-Port OC-12 Packet over SONET line card.

*Table 13-3 1-Port OC-12 Packet over SONET Interface Syntax*

| Type of Interface | Slot | Subslot | Port |
|---|---|---|---|
| Packet over SONET interface | 1 to 8/ | 0/ | 0 |

The following example shows the syntax for configuring an 1-Port OC-12 Packet over SONET line card in slot 1:

```
Router(config)# interface pos 1/0/0
```

# Configuring the Interface

After you verify that the 1-Port OC-12 Packet over SONET line card is installed correctly, use the following procedure to configure the new interface. Be prepared with necessary information, such as the interface IP address.

Use the following procedure to create a basic configuration—enabling an interface and specifying IP routing.

**Step 1** At the global configuration prompt, specify the new interface to configure by entering the **interface pos** command and interface address. For example:

```
Router(config)# interface pos 1/0/0
```

**Step 2** Assign an IP address and a subnet mask to the interface with the **ip address** configuration subcommand, as in the following example:

```
Router(config-if)# ip address 192.168.255.255 255.255.255.0
```

**Step 3** Specify either HDLC or PPP encapsulation. For example:

```
Router(config-if)# encapsulation hdlc
```

**Step 4** If necessary, modify the 1-Port OC-12 Packet over SONET line card configuration or that of the remote device to ensure that, where appropriate, they use the same settings. For more information, refer to the Remote Side Setting column in Table 13-2.

**Step 5** Add any other configuration subcommands required for the enabling of routing protocols and adjust the interface characteristics.

**Step 6** Enter the **no shutdown** command to enable the interface.

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

**Step 7** When you have included all of the configuration subcommands to complete the configuration, enter **Crtl-Z** to exit configuration mode.

**Step 8** Write the new configuration to memory.

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

The system displays an OK message when the configuration has been stored.

After you have completed your configuration, you can check it using the **show interface pos** *slot/subslot/port*.

# Other Configuration Commands

The following sections present some of the commands that you can use to customize your Cisco 10000 1-Port OC-12 Packet over SONET line card configuration.

## Setting the Clock Source

At the prompt, set the internal or line clock source by using the **clock source** command.

**clock source** {**internal** | **line**}

Where:

- **internal** specifies that the internal clock source is used
- **line** specifies that the network clock source is used

The default is clock source internal.

In this example, the line card is instructed to use a line clock source.

```
Router(config)# interface pos 1/0/0
Router(config-if)# clock source line
```

## Configuring Framing

You can use the **pos framing** command to set framing to SONET STS-12c or SDH STM-4 framing.

**pos framing** [**sdh** | **sonet**]
[**no**] **pos framing**

The default is SONET.

Make sure your system supports SDH before using this option.

Use the **no** form of the command to restore the default framing mode.

In the following example, the framing type is set to SONET:

```
Router(config)# interface pos 5/0/0
Router(config-if)# no pos framing
```

## Specifying SONET Overhead

You can use the **pos flag** command to assign values for specific elements of the frame header. This command is typically used to meet a standards requirement or to ensure interoperability with another vendor's equipment.

**pos flag** [**c2** *value*] [**j0** *value*] [**s1s0** *value*]
[**no**] **pos flag** [**c2** *value*] [**j0** *value*] [**s1s0** *value*]

Where:

- **c2** is a path signal identifier, and *value* is one of the following:
  - 0xCF for PPP or HDLC without scrambling
  - 0x16 for PPP or HDLC with scrambling
- **j0** is the section trace byte, and *value* is 0x1 for interoperability with some SDH devices in Japan
- **s1s0** is part of the payload pointer byte, and *value* is 0 for OC-12c and 2 for AU-4

The default values are c2—0xCF, j0—0x01, and s1s0—0.

Use the **no** form of the command to restore the default values.

In the following example, the c2 bit is set to 0xCF.

```
Router(config)# interface pos 5/0/0
Router(config-if)# pos flag c2 0xCF
```

# Controlling the S1 SONET Overhead Byte

In most situations, the default value for the S1 SONET overhead byte (0x0) does not need to be changed. Refer to the SONET standards for information about the possible values for the S1 SONET overhead byte and the definition of each value.

## Controlling a Transmitted S1 Overhead Byte

In Cisco IOS Release 12.2(28)SB, use the **pos flag s1-byte tx** command in interface configuration mode to control the transmission of the S1 SONET overhead byte.

**pos flag s1-byte tx** *value*

Where:

- *value* is in the range of 0x0 to 0xF
- 0x0 is the default value

In the following example the S1 SONET overhead byte is set to 0xF:

```
pos flag s1-byte tx 0xF
```

## Reacting to a Received S1 Overhead Byte

In Cisco IOS Release 12.2(28)SB, use the **pos flag s1-byte rx-communicate** command to direct the router to switch the clock source to internal when it receives an S1 SONET overhead byte with a value of 0xF. When the S1 SONET overhead byte changes from 0xF to any other value, the clock source reverts back to the clock source specified in the user configuration.

The S1 overhead byte is ignored by the receiving router unless the **pos flag s1-byte rx-communicate** command is issued.

**pos flag s1-byte rx-communicate**

**no pos flag s1-byte rx-communicate**

The following example directs the router to switch to internal clocking when it receives an S1 SONET overhead byte with a value of 0xF:

```
pos flag s1-byte rx-communicate
```

The default for the **pos flag s1-byte rx-communicate** command is disabled or off.

# Configuring Packet over SONET SPE Scrambling

The **pos scramble-atm** command allows you to scramble the Packet over SONET synchronous payload envelope (SPE). SONET payload scrambling applies a self-synchronous scrambler to the SPE of the interface to ensure sufficient bit transition density.

```
pos scramble-atm
[no] pos scramble-atm
```

The default is no Packet over SONET SPE scrambling.

Use the **no** form of the command to disable scrambling.

In the following example, scrambling is enabled:

```
Router(config)# interface pos 5/0/0
Router(config-if)# pos scramble-atm
```

# Configuring Loopback Testing

To enable loopback testing of data transmitted from the PRE line card to the 1-Port OC-12 Packet over SONET line card and back, use the **loopback** command in interface configuration mode.

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

Where:

Both **line** and **internal** do the following:

- Loop any data received at the 1-Port OC-12 Packet over SONET line card network interface back into the network
- Loop any data received at the 1-Port OC-12 Packet over SONET line card network interface back into the PRE card

Use the **no** form of the command to stop the loopback test.

For more information on loopbacks, refer to the online *Cisco 10000 Series Internet Router Troubleshooting Guide*.

In the following example, a loopback is set for the 1-Port OC-12 Packet over SONET line card in slot 5:

```
Router(config)# interface pos 5/0/0
Router(config-if)# loopback line
```

CHAPTER 14

# 1-Port OC-48/STM-16 Packet over SONET Line Card Configuration

This chapter describes procedures for configuring the Cisco 10000 series 1-Port OC-48/STM-16 Packet over SONET line card, hereafter known as the 1-Port OC-48/STM-16 Packet over SONET line card.

The 1-Port OC-48/STM-16 Packet over SONET line card provides a trunk uplink capable of supporting up to 2.4 Gbps full duplex throughput over a standard SONET/ITU-T Synchronous Digital Hierarchy (SDH) interface, using a single-mode fiber with SC connectors.

This chapter contains the following sections:



## Software Support

Table 14-1 shows the minimum Cisco IOS release on each release train that supports the 1-Port OC-48/STM-16 Packet over SONET line card.

*Table 14-1 1-Port OC-48/STM-16 Packet over SONET Line Card Software Support*

| Required PRE | Minimum Cisco IOS Releases |
|---|---|
| PRE2 | Cisco IOS Release 12.2(15)BX and later releases of Cisco IOS 12.2BX<br>Cisco Release IOS 12.3(7)XI and later releases of Cisco IOS 12.3XI<br>Cisco IOS Release 12.2(28)SB and later releases of Cisco IOS 12.2SB |

## Checking Hardware and Software Compatibility

The PRE installed in the Cisco 10000 series router chassis must support the Cisco IOS software running on the router. Use the **show version** command to check the PRE version installed.

To see if a feature is supported by a Cisco IOS release, to locate the software document for that feature, or to check the minimum software requirements of Cisco IOS software with the hardware installed on your router, Cisco maintains the Software Advisor tool on Cisco.com at http://www.cisco.com/cgi-bin/Support/CompNav/Index.pl

This tool does not verify whether line cards within a system are compatible, but does provide the minimum Cisco IOS requirements for individual hardware line cards, modules, or options.

You must be a registered user on Cisco.com to access this tool.

# Default Values

Table 14-2 lists default values for the 1-Port OC-48/STM-16 Packet over SONET line card. This table also includes the command used for modifying a default value and indicates whether a value needs to be the same (or opposite) on the remote end of the connection.

*Table 14-2 1-Port OC-48/STM-16 Packet over SONET Line Card SONET Defaults*

| Command Name | Default Setting | Command Syntax | Remote Side Setting |
|---|---|---|---|
| bandwidth | 2488 | **bandwidth** *kilobits* | Same |
| clock source | line | **clock source** [**line** \| **internal**] | At least one side must be set to *internal.* |
| crc | 32 | **crc** [**16** \| **32**] | Same |
| encapsulation | HDLC | **encapsulation** [**hdlc** \| **ppp** \| **frame-relay**] | Same |
| keepalive | 10 | [**no**] **keepalive** *period* | Same |
| mtu (maximum transmission unit) | 4470 | **mtu** *size* | Same |
| MAC Protocol | pos | **hw-module slot** [**pos**] | Same |
| pos framing | SONET | **pos framing** [**sonet** \| **sdh**] | Same |
| pos scramble-atm | No scrambling | [**no**] **pos scramble-atm** | Same |
| pos flag (SONET overhead) | c2 to 0xcf<br>j0 to 0x01<br>s1s0 to 0 | **pos flag** [**c2** \| **j0** \| **s1s0**] *value* | Same |

# SDCC Default Values

Table 14-3 lists default values for the 1-Port OC-48/STM-16 Packet over SONET line card if you configure a SONET Section Data Communications Channel (SDCC) interface.

The table includes the command used for modifying a default value and indicates whether a value needs to be the same (or opposite) on the remote end of the connection.

*Table 14-3 1-Port OC-48/STM-16 Packet over SONET Line Card SONET SDCC Defaults*

| Command Name | Default Setting | Command Syntax | Remote Side Setting |
|---|---|---|---|
| bandwidth | 192 | **bandwidth** *kilobits* | Same |
| crc | 32 | **crc** [**16** \| **32**] | Same |
| keepalive | 10 | [**no**] **keepalive** *period* | Same |
| mtu (maximum transmission unit) | 1500 | **mtu** *size* | Same |
| loopback (internal) | off | [**no**] **loopback** | N/A |

# Interface Syntax

Use the syntax in Table 14-4 to specify an 1-Port OC-48/STM-16 Packet over SONET line card interface number in a configuration command.

*Table 14-4 OC-48 Interface Syntax*

| Type of Interface | Slot | Subslot | Port |
|---|---|---|---|
| Packet over SONET interface | 1 to 8/ | 0/ | 0 |

The following example shows the syntax for an OC-48 card in slot 1:

```
Router(config)# interface pos 1/0/0
Router(config-if)#
```

# Configuring the Interface

After verifying that the 1-Port OC-48/STM-16 Packet over SONET line card is installed correctly, use the following procedure to configure the interface and specify an IP address. Be prepared with the information you will need, such as the IP address.

**Step 1** Specify the MAC protocol using the **hw-module slot** command. In this example, the controller is configured for Packet over SONET:

```
Router(config)# hw-module slot 2 pos
```

**Step 2** At the global configuration prompt, enter interface configuration mode. In this example, the interface configuration mode is Packet Over SONET (POS):

```
Router(config)# interface pos 2/0/0
```

**Step 3** Assign an IP address and a subnet mask to the interface using the **ip address** command. For example:

```
Router(config-if)# ip address 192.168.21.1 255.255.255.0
```

**Step 4** Specify either HDLC or PPP encapsulation. The following example specifies HDLC encapsulation:

```
Router(config-if)# encapsulation hdlc
```

**Step 5** If necessary, modify the 1-Port OC-48/STM-16 Packet over SONET line card configuration or the configuration of the remote device to ensure that they use the same settings, where appropriate. For more information, refer to the Remote Side Setting column in Table 14-2.

**Step 6** Add any other configuration commands required to enable routing protocols and adjust the interface characteristics.

**Step 7** Enter the **no shutdown** command to enable the interface. For example:

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

**Step 8** When you have included all of the configuration commands to complete the configuration, enter **Ctrl-Z** to exit configuration mode.

**Step 9** Write the new configuration to memory.

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

The system displays an OK message when the configuration has been stored.

---

After you have completed the configuration, you can verify that it is correct using the **show interface pos** *slot/subslot/port* command.

# MAC Protocol Selection Command

You must specify the MAC protocol that the 1-Port OC-48/STM-16 Packet over SONET line card will use before you configure the interface, by using the following command:

**hw-module slot** *slot* [pos]

Where *slot* is 1 to 8.

The following example shows how you specify the Packet over SONET MAC protocol for a card in slot 2:

```
Router(config)# hw-module slot 2 pos
```

# Interface Configuration Mode Command

To configure an interface, you must specify an interface configuration mode by using the following command:

**interface** [**pos** | **sdcc**] *slot/subslot/port*

Where *slot* is 1 to 8, *subslot* is 0, and *port* is 0.

The following example shows how you specify Packet over SONET interface configuration mode for a card in slot 2:

```
Router(config)# interface pos 2/0/0
Router(config-if)#
```

# SDCC Interface Configuration Commands

This section lists the commands that are available when you configure a SONET Section Data Communications Channel (SDCC) interface on the router.

**Note** The SDCC interface is available only after the MAC protocol has been set to Packet over SONET. For more information about setting the MAC protocol, see the MAC Protocol Selection Command, page 14-4

## Enabling the SDCC Interface Configuration Mode

You must first enable the SDCC interface configuration mode before attempting to configure any SDCC commands. You can enable the SDCC interface mode from global configuration mode by using the **sdcc enable** command.

```
sdcc enable
[no] sdcc enable
```

The default setting is disabled.

In the following example, SDCC is enabled:

```
Router(config)# sdcc enable
```

**Caution** If you enter the **no sdcc enable** command after configuring an SDCC interface, the interface is removed from the interface list in the configuration.

## Enabling an SDCC Interface

You can administratively enable an SDCC interface using the **no shutdown** command.

```
shutdown
[no] shutdown
```

The default state of an SDCC interface is administratively up.

In the following example, an SDCC interface is enabled:

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

## Interface Selection Command

To configure an SDCC interface, you must select an interface using the **interface** command:

```
interface sdcc slot/subslot/port
```

Where *slot* is 1 to 8, *subslot* is 0, and *port* is 0.

The following example shows how you specify an SDCC interface for a card in slot 2:

```
Router(config)# interface sdcc 2/0/0
Router(config-if)#
```

# Enabling Loopback Tests

You can configure an SDCC interface for an internal loopback test by using the **loopback** command. With an internal loopback, packets that are received by the line card from the route processor are looped back to the route processor without being sent to the line.

**loopback**
[**no**] **loopback**

Use the **no** form of the command to stop the loopback test.

The default is none.

In the following example, the loopback is set to line:

```
Router(config-if)# loopback
```

# Configuring CRC Size

You can configure the CRC size for HDLC encapsulation on an SDCC interface to 16- and 32-bit CRCs using the **crc** command.

**crc** {**16** | **32**}

Where:

**16** sets the CRC to 16 bits.

**32** sets the CRC to 32 bits.

The default is 32.

In the following example, the CRC size is set to 16:

```
Router(config-if)# crc 16
```

# Configuring MTU Size

You can configure the Maximum Transmission Unit (MTU) size for up to 1500 bytes, using the **mtu** command.

**Note** The MTU size does not include the 4 bytes for the HDLC header, or the 2 or 4 bytes of the CRC.

**mtu** *mtu bytes*

Where *mtu bytes* is the MTU size from zero to 1500.

The default is 1500.

In the following example, the MTU size is set to 1000:

```
Router(config-if)# mtu 1000
```

## Configuring Hold Queue Size

You can configure a hold queue on an SDCC interface for packets received from the line using the **hold-queue** command.

**hold-queue** *number* **in**

Where *number* is the maximum number of packets that the line card will hold in the hold queue, from zero to 4096.

The default hold queue size is 75 packets.

In the following example, the hold queue size is set to 60 packets:

```
Router(config-if)# hold-queue 60 in
```

# Other Configuration Commands

The following sections present some of the commands that you can use to customize the 1-Port OC-48/STM-16 Packet over SONET line card configuration.

## Setting the Clock Source

You can set the clock source to internal or line by using the **clock source** command.

**clock source** {**internal** | **line**}

Where:

- **internal** specifies that the internal clock source is used
- **line** specifies that the network clock source is used

The default is clock source internal.

In this example, the clock source is set to line:

```
Router(config)# interface pos 1/0/0
Router(config-if)# clock source line
```

## Configuring Framing

You can use the **pos framing** command to set framing to SONET STS-3c or SDH STM-1 framing.

**pos framing** [**sdh** | **sonet**]
[**no**] **pos framing**

The default is SONET.

Make sure your system supports SDH before using this option.

Use the **no** form of the command to restore the default framing mode.

In the following example, the framing type is set to SONET:

```
Router(config)# interface pos 5/0/0
Router(config-if)# no pos framing
```

# Configuring SONET Overhead

You can use the **pos flag** command to assign values for specific elements of the frame header. This command is typically used to meet a standards requirement or to ensure interoperability with another vendor's equipment.

**pos flag** [**c2** *value*] [**j0** *value*] [**s1s0** *value*]
[**no**] **pos flag** [**c2** *value*] [**j0** *value*] [**s1s0** *value*]

Where:

- **c2** is a path signal identifier, and *value* is one of the following:
  - 0xCF for PPP or HDLC without scrambling
  - 0x16 for PPP or HDLC with scrambling
- **j0** is the section trace byte, and *value* is 0x1 for interoperability with SDH devices.
- **s1s0** is part of the payload pointer byte, *value* is 0 for OC-48c and 2 for AU-4.

The default values are c2—0xCF, j0—0x01, and s1s0—0.

Use the **no** form of the command to restore the default values.

In the following example, the c2 bit is set to 0xCF:

```
Router(config)# interface pos 5/0/0
Router(config-if)# pos flag c2 0xCF
```

# Configuring Packet over SONET SPE Scrambling

You can set the 1-Port OC-48/STM-16 Packet over SONET line card to scramble the Packet over SONET synchronous payload envelope (SPE) using the **pos scramble-atm** command. SONET payload scrambling applies a self-synchronous scrambler to the SPE of the interface to ensure sufficient bit transition density.

**pos scramble-atm**
[**no**] **pos scramble-atm**

The default is no Packet over SONET SPE scrambling.

Use the **no** form of the command to disable scrambling.

In the following example, scrambling is enabled:

```
Router(config)# interface pos 5/0/0
Router(config-if)# pos scramble-atm
```

# Configuring Loopback Testing

You can enable loopback testing of data transmitted from the PRE to the OC-48 line card and back, by using the **loopback** command in interface configuration mode.

**loopback** [**line** | **internal**]
[**no**] **loopback** [**line** | **internal**]

Where:

**internal** loops packets received from the route processor back into the line card to the route processor without being sent to the line.

**line** loops packets received from the line back onto the line at the line card level without being processed in the Route Processor.

Use the **no** form of the command to stop the loopback test.

For more information on loopbacks, refer to the online *Cisco 10000 Series Internet Router Troubleshooting Guide*.

In the following example, a loopback is set for the 1-Port OC-48/STM-16 Packet over SONET line card in slot 5:

```
Router(config)# interface pos 5/0/0
Router(config-if)# loopback line
```

# Part 5

# Unchannelized Line Card

CHAPTER 15

# 8-Port Unchannelized E3/T3 Line Card Configuration

This chapter describes procedures for configuring the Cisco 10000 series 8-Port Unchannelized E3/T3 line card, hereafter known as the 8-Port Unchannelized E3/T3 line card.

The 8-Port Unchannelized E3/T3 line card provides Cisco 10000 series routers with eight E3 or T3 high-density unchannelized interface ports.

This chapter contains the following sections:



# Software Support

Table 15-1 shows the minimum Cisco IOS release on each release train that supports the 8-Port Unchannelized E3/T3 line card.

***Table 15-1 8-Port Unchannelized E3/T3 Line Card Software Support***

| Required PRE | Minimum Cisco IOS Releases |
|---|---|
| PRE1 | Cisco IOS Release 12.0(22)S and later releases of Cisco IOS 12.0S |
| PRE2 | Cisco IOS Release 12.0(22)S and later releases of Cisco IOS 12.0S<br>Cisco IOS Release 12.0(20)ST and later releases of Cisco IOS Release 12.0ST<br>Cisco IOS Release 12.3(7)XI and later releases of Cisco IOS 12.3XI<br>Cisco IOS Release 12.2(28)SB and later releases of Cisco IOS 12.2SB |

## Checking Hardware and Software Compatibility

The PRE installed in the Cisco 10000 series router chassis must support the Cisco IOS software running on the router. Use the **show version** command to check the PRE version installed.

To see if a feature is supported by a Cisco IOS release, to locate the software document for that feature, or to check the minimum software requirements of Cisco IOS software with the hardware installed on your router, Cisco maintains the Software Advisor tool on Cisco.com at http://www.cisco.com/cgi-bin/Support/CompNav/Index.pl

This tool does not verify whether line cards within a system are compatible, but does provide the minimum Cisco IOS requirements for individual hardware line cards, modules, or options.

You must be a registered user on Cisco.com to access this tool.

# Default Values

Table 15-2 lists default configuration values for the 8-Port Unchannelized E3/T3 line card. This table also includes default values and indicates whether or not you must invoke a corresponding value at the remote side Channel Service Unit/Data Service Unit (CSU/DSU).

*Table 15-2 8-Port Unchannelized E3/T3 Line Card Parameters*

| Command Name | Default Value | Command Syntax | Remote CSU/DSU Setting |
|---|---|---|---|
| **Common Controller Configuration Mode** | | | |
| cable length | 450 | **cablelength** *feet* | — |
| clock source | internal | **clock source [line \| internal]** | At least one side set to **internal** |
| description | — | **description** *string* | — |
| dsx3mode | t3 | **[no] dxs3mode [e3 \| t3]** | Same |
| shutdown | no shutdown | **[no] shutdown** | — |
| **T3 Interface Configuration Mode** | | | |
| bert | no bert | **[no] bert pattern** *pattern* **interval** *time* **[unframed]** | No |
| description | — | **description** *string* | — |
| dsu bandwidth | 44,210 | **dsu bandwidth** *bandwidth* **[no] bandwidth** | Same |
| dsu mode | cisco | **dsu mode** [**adtran** \| **cisco** \| **digilink** \| **kentrox** \| **larscom** \| **verilink low-bit** \| **verilink high-bit**] | Same |
| encapsulation | hdlc | **encapsulation** [**hdlc** \| **ppp** \|**frame-relay**] | Same |
| equipment | customer | **equipment** [**customer** \| **network**]**loopback** | Same |
| framing | auto-detect | **framing** [**auto-detect** \| **c-bit** \| **m13**] | Same |
| idle character | flags (0x7e) | **idle character** [**flags** \| **marks**] | Same |

*Table 15-2 8-Port Unchannelized E3/T3 Line Card Parameters (continued)*

| Command Name | Default Value | Command Syntax | Remote CSU/DSU Setting |
|---|---|---|---|
| loopback | no loopback | [**no**] **loopback network** [**line** \| **payload**]<br>[**no**] **loopback** [**local** \| **remote**] | No |
| mdl | no mdl | **[no] mdl {transmit {idle-signal \| path \| test-signal} \| string {eic \| fic \| generator \| lic \| pfi \| port \| unit}** *<string>***}** | Same |
| scramble | no scramble | [**no**] **scramble** | Same |
| shutdown | no shutdown | **shutdown** | — |
| **E3 Interface Configuration Mode** | | | |
| bert | no bert | [**no**] **bert pattern** *pattern* **interval** *time* [**unframed**] | Same |
| description | — | **description** *string* | — |
| encapsulation | hdlc | **encapsulation** [**hdlc** \| **ppp** \|**frame-relay**] | Same |
| dsu bandwidth | 34,368 kbps | [**no**] **dsu bandwidth** *bandwidth* | Same |
| dsu mode | cisco | [**no**] **dsu mode** [**cisco** \| **kentrox**] | Same |
| fallback-clocking | off | **fallback-clocking on** \| **off** | Same |
| idle character | flags (0x7e) | **idle character** [**flags** \| **marks**] | Same |
| loopback | no loopback local network line | [**no**] **loopback network** [**line** \| **payload**]<br>[**no**] **loopback** [**local** \| **remote**] | No |
| scramble | no scrambling | [**no**] **scramble** | Same |
| shutdown | no shutdown | **shutdown** | — |

# Naming Convention

The Cisco 10000 series router line cards use the extended slot naming convention to identify the line card slot, subslot, and port in commands. This convention uses the following format:

```
slot/subslot/port
```

Where:

- **slot** is the slot number of the installed the line card
- **subslot** is the subslot number of the line card (the subslot number for all full-height line cards is always **0**)
- **port** is the number of the line card port

# Interface Syntax

The following tables contain the line card syntax to configure 8-Port Unchannelized E3/T3 line card interfaces for the Cisco 10008 router and Cisco 10005 router chassis.

**Note** The dsx3 controller designation is a generic term that refers to both E3 and T3 type interfaces. A dsx3 interface controls the configuration and management of DS3 channels and characteristics.

Table 15-3 specifies the 8-Port Unchannelized E3/T3 line card interface syntax for the Cisco 10008 router 8-slot chassis.

*Table 15-3 8-Port Unchannelized E3/T3 Line Card Interface Syntax for the Cisco 10008 Chassis*

| Configuration | Command | Slot | Subslot | Port Number |
|---|---|---|---|---|
| Controller | controller dsx3 *slot/0/port* | 1 to 8/ | 0/ | 0 to 7 |
| Interface T3 (DS3) | serial*slot/0/port* | 1 to 8/ | 0/ | 0 to 7 |
| Interface E3 | serial*slot/0/port* | 1 to 8/ | 0/ | 0 to 7 |

Table 15-4 specifies the 8-Port Unchannelized E3/T3 line card interface syntax for the Cisco 10005 router 5-slot chassis.

*Table 15-4 8-Port Unchannelized E3/T3 Line Card Interface Syntax for the Cisco 10005 Chassis*

| Configuration | Command | Slot | Subslot | Port Number |
|---|---|---|---|---|
| Controller | controller dsx3 *slot/0/port* | 1 to 5/ | 0/ | 0 to 7 |
| Interface T3 (DS3) | serial*slot/0/port* | 1 to 5/ | 0/ | 0 to 7 |
| Interface E3 | serial*slot/0/port* | 1 to 5/ | 0/ | 0 to 7 |

# Configuration Task Overview

You can configure each port of the 8-Port Unchannelized E3/T3 line card as a full-rate or subrate unchannelized E3 or T3 interface.

**Note** Subrate unchannelized interfaces use specific vendor DSU proprietary formats that limit interface bit rates to less than full-rate on an E3 or T3 serial interface line.

To configure an 8-Port Unchannelized E3/T3 line card interface:

1. Designate a port as a controller.
2. Configure the controller parameters and save the configuration to NVRAM.
3. Create a serial interface.
4. Specify the interface parameters for the interface and save the configuration to NVRAM.

**Note** Each time you change a port mode (from T3 to E3, or E3 to T3), the running configuration of the controller and interface of the changed port mode is erased. If you want to keep a configuration, you must save it before you change its mode.

# Configuration Examples

This section contains examples to configure both full-rate and subrate E3 or T3 interfaces on the line card. You must be in privileged EXEC mode to designate, create, and configure an E3 or T3 controller and interface.

**Tip** To view available command parameters, type **?** at the command line.

## Configuring an E3 Interface

The following sample procedure describes how to create and configure an E3 interface on port 6 of the line card in slot 4.

**Step 1** Enter global configuration mode and designate the port as a controller.

```
Router# configure terminal
Router(config)# controller dsx3 4/0/6
```

**Step 2** Configure the port as a **dsx3 e3** controller and save the configuration to NVRAM.

```
Router(config-controller)# dsx3mode e3
Router(config-controller)# end
Router# write
```

**Step 3** Enter configuration interface mode and designate port 6 as the serial interface.

```
Router# configure terminal
Router(config)# interface serial 4/0/6
```

**Step 4** Configure the E3 interface DSU parameters for the port as a full-rate or subrate interface:

**a.** Full-rate (default 34,368):

```
Router (config-if)# dsu mode cisco
Router (config-if)#
```

**b.** Subrate (any value below 34,368):

```
Router (config-if)# dsu mode cisco
Router (config-if)# dsu bandwidth 16000
```

**Step 5** Save the configuration to NVRAM.

```
Router(config-if)# end
Router# write
```

## Configuring a T3 Interface

The following sample procedure describes how to create and configure a T3 interface on port 6 of the line card in slot 4.

**Step 1** Enter global configuration mode and designate the port as a controller.

```
Router# configure terminal
Router(config)# controller dsx3 4/0/6
```

**Step 2** Configure the port as a **dsx3 t3** controller and save the configuration to NVRAM.

```
Router(config-controller)# dsx3mode t3
Router(config-controller)# end
Router# write
```

**Step 3** Enter configuration interface mode and designate port 6 as the serial interface.

```
Router# configure terminal
Router(config)# interface serial 4/0/6
```

**Step 4** Configure the T3 interface DSU parameters for the port as a full-rate or subrate interface:

**a.** Full-rate (default of 44,210):

```
Router (config-if)# dsu mode cisco
```

**b.** Subrate (any value below 44,210):

```
Router (config-if)# dsu mode cisco
Router (config-if)# dsu bandwidth 16000
```

**Step 5** Save the configuration to NVRAM.

```
Router(config-if)# end
Router# write
```

# Controller Configuration Commands

This section describes the 8-Port Unchannelized E3/T3 line card commands to modify and test an E3 or T3 controller. You must be in configuration-controller mode to modify an E3 or T3 controller.

**Note** Each time you change a port (from T3 to E3, or E3 to T3), the running configuration of the controller and interface of the changed port is erased. If you want to keep a configuration, you must save it before you change its port mode.


- Cablelength Command, page 15-6
- Clock Source Command, page 15-7
- Controller Description Command, page 15-7
- Controller dsx3mode Command, page 15-8
- Shutdown Command, page 15-8


## Cablelength Command

The **cablelength** command improves signal strength for losses associated with lengthy cables. To specify the physical length of the interface cable between the near and far-end CSU/DSU devices, use the **cablelength** command in the following format:

[**no**] **cablelength** *feet*

Where *feet* is a number from 0 to 450.

The default value is 450 feet.

Use the **no** form of this command to restore the default cable length.

In the following example, the cable length is set to 120 feet:

```
Router(config)# controller dsx3 4/0/6
Router(config-controller)# cablelength 120
```

The following cable length distances take into consideration line buildout for E3 and T3 controllers:

| Interface | Cable Length (feet) | Buildout |
|---|---|---|
| E3 | 0 to 224 | no line buildout |
| E3 | 225 to 450 | line buildout |
| T3 | 0 to 299 | no line buildout |
| T3 | 300 to 450 | line buildout |

# Clock Source Command

Use the **clock source** command to choose whether or not the transmitted clock is derived from the received clock or is generated by an internal clock source. In some cases a network configuration may require you to derive network synchronization from a remote master clock.

The **clock source** command is set in controller configuration mode using the following format:

**clock source [internal | line]**

Where:

- **internal** specifies that the internal clock source is used
- **line** specifies that the network clock source is used

The default is clock source internal.

In this example, the controller is instructed to use a line clock source.

```
Router(config)# controller dsx3 4/0/6
Router(config-controller)# clock source line
```

**Caution** Never configure both sides of an E3 or T3 link to the clock source line.

# Controller Description Command

Use the **description** command to identify particulars about the controller. You can enter up 80 characters in your controller description string using the following format:

**description** *string*

In the following example, port 6 is labeled with a description to help identify it:

```
Router(config)# controller dsx3 4/0/6
Router(config-controller)# description "Company ABC; IP = 240.21.3.8"
```

**Note** The **description** command can also be used in interface mode (see "Interface Description Command" section on page 15-10).

## Controller dsx3mode Command

Use the **dsx3mode** command to define the controller type as an E3 or T3 controller using the following format:

[**no**] **dxs3mode** [**e3** | **t3**]

The default controller type is **t3**.

Use the **no** form of this command to remove the controller.

The following example shows how to define a T3 type controller:

```
Router(config)# controller dsx3 2/0/0
Router(config-controller)# dsx3mode t3
```

## Shutdown Command

Use the **shutdown** command to shut down the controller and interface using the following format:

[**no**] **shutdown**

Use the **no** form of this command to restore the previously configured controller and interface.

- It takes 10 seconds for alarms to clear after issuing a **no shut** command on a T3 port. Shutting down a controller causes a T3 port to transmit:
  - AIS (alarm indication signal, for **m13** framing)
  - An idle signal (for **c-bit** framing)
- There is no delay for alarms to clear after issuing a **no shut** command on an E3 port.
  - Shutting down a controller causes an E3 port to transmit AIS.

**Note** The AIS (alarm indication signal), also called a blue alarm, is sent downstream to notify the downstream device that an upstream line failure has occurred.

In the following example, the controller on port 0 is shut down:

```
Router(config)# controller dsx3 1/0/0
Router(config-controller)# shutdown
```

# Interface Configuration Commands

This section describes the following command procedures to modify and test E3 and T3 interfaces on the line card:

- Running a BER Test, page 15-9
- Interface Description Command, page 15-10

- Specifying DSU Mode, page 15-10
- Specifying DSU Bandwidth, page 15-11
- Encapsulation Command, page 15-11
- Equipment Loopback Command, page 15-12
- Fallback-Clocking Command, page 15-12
- Setting the Framing Type, page 15-13
- Specifying the Idle-Character, page 15-13
- Performing Loopback Tests, page 15-14
- Entering MDL Messages, page 15-14
- Setting the National Bit, page 15-16
- Enabling Scrambling, page 15-16

You must be in configuration interface mode to modify an E3 or T3 configuration, for example:

```
Router# configure terminal
Router(config)# interface serial 1/0/0
Router(config-if)#
```

# Running a BER Test

You can configure an interface to run a BER test (bit error-rate test). The BER test checks network cables and isolates apparent signal problems in the field.

Before you begin a BER test, you need to set up a remote loopback from the far-end (remote) DSU/CSU. See "Performing Loopback Tests" section on page 15-14 for additional information about configuring the loopbacks.

The following loopback commands cause the remote CSU/DSU to perform a full-bandwidth loopback through its CSU:

- **loopback network line** causes a loopback toward the network *before* going through the CSU/DSU framer.
- **loopback network payload** causes a loopback toward the network *after* going through the CSU/DSU framer.

To run a BER test pattern on an interface, use the following interface configuration command:

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

Where:

- **errors** *number* is 1 to 255
- **pattern** *pattern* is:
  - 0s—repetitive test pattern of all zeros (00000..)
  - 1s—repetitive test pattern of all ones (11111..)
  - 2^15—pseudo-random O.151 test pattern (32,768 bits in length)
  - 2^20-O153—pseudorandom O.153 test pattern (1,048,575 bits in length)
  - 2^23—test pattern
  - alt-0-1—alternating 0s and 1s test pattern

- QRSS-2^20—pseudorandom QRSS O.151 test pattern (1,048,575 bits in length)

The default is no bert pattern test. Use the **no** form of this command to terminate a BER test.

For more information about BER testing, refer to the *Cisco IOS Interface and Hardware Component Configuration Guide* and the online *Cisco 10000 Series Internet Router Troubleshooting Guide*.

Example:

- Send a BERT pseudorandom pattern of 2^20 through T3 interface 1/0/0 for 5 minutes.

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

# Interface Description Command

Use the **description** command to identify particulars about the interface. You can enter up 80 characters in your controller description string using the following format:

**description** *string*

In the following example, port 6 is labeled with a description to help identify it:

```
Router(config)# interface 4/0/6
Router(config-if)# description "Company ABC; IP = 240.21.3.8"
```

**Note** The **description** command can also be used in controller mode (see the "Controller Description Command" section on page 15-7).

# Specifying DSU Mode

The **dsu mode** command configures the line card to emulate a manufacturer's proprietary multiplexing scheme. Be sure that the remote DSU connection at the far end is connected and set up as an identical E3 or T3 interface.

**Note** Only Cisco, Digital-Link, and Kentrox are supported on E3 interfaces.

Use the **dsu mode** command to specify a Data Service Unit (DSU) mode for a selected interface using the following format:

**dsu mode** [**Adtran** | **cisco** | **digital-link** | **kentrox** | **larscom** | **verilink-highbit** | **verilink-lowbit**]
[**no**] **dsu mode**

The default DSU mode is **cisco**. Use the **no** form of the command to return the DSU mode to the default mode.

In the following example, the DSU mode is set to **cisco**:

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

# Specifying DSU Bandwidth

Use the **dsu bandwidth** command to specify the full-rate or subrate bandwidth for an E3 or T3 interface using the following format:

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

Where:

- bandwidth = 0 to 44210 kbps for a T3 interface
  - The default value for a T3 interface is 44210 (full-rate)
- bandwidth = 0 to 34368 kbps for an E3 interface
  - The default value for an E3 interface is 34368 (full-rate)

Use the **no** form of this command to return to the full-rate default values.

In the following example, the full-rate bandwidth for an E3 interface is set to 34368:

```
Router(config)# interface serial 2/0/0
Router(config-if)# dsu bandwidth 34368
```

## Creating a Subrate E3 or T3 Interface

You can create a subrate E3 or T3 controller interface by specifying a bandwidth value that is less than full-rate for the near end DSU.

In the following example, the subrate **dsu bandwidth** for a T3 interface is set to 16000:

```
Router(config)# interface serial 4/0/6
Router(config-if)# dsu mode cisco
Router(config-if)# dsu bandwidth 16000
```

**Note** When you specify a bandwidth, the software sets that bandwidth to the closest acceptable bandwidth value based on the time-slot size for the current DSU mode. See the *Chapter 17, "Provisioning a Subrate E3 or T3 Interface"* for information about bandwidth.

# Encapsulation Command

Use the **encapsulation** command to specify the type of encapsulation for the interface using the following format:

```
encapsulation encapsulation-type
```

Where *encapsulation-type* is:

- **frame-relay**
- **hdlc** (serial HDLC synchronous)
- **ppp** (Point-to-Point Protocol)

The default encapsulation is **hldc**.

The following example sets the encapsulation type as **ppp**:

```
Router(config)# interface serial 4/0/6
Router(config-if)# encapsulation ppp
```

# Equipment Loopback Command

Use the **equipment loopback** command to specify whether or not a T3 port for the near-end device will accept remote loopback requests from the far-end device.

**Note** Because remote loopback requests are available only when **c-bit** framing is invoked, the **equipment loopback** command is useful only when framing is set to **c-bit** or **auto-frame-detect**.

```
[no] equipment [customer | network] loopback
```

The default mode is **customer**.

Use the **no** form of this command to disable the equipment selection and revert back to **auto-detect** as the framing value (see Setting the Framing Type, page 15-13).

The following examples show how to the **equipment loopback** command:

### Equipment Customer Loopback Command

When you invoke the **equipment customer loopback** command, the near-end T3 port accepts far-end loopback requests.

```
Router(config)# interface dsx3 1/0/0
Router(config-if)# equipment customer loopback
```

**Note** Specify **equipment customer loopback** as the equipment type, unless you want to block the far end from setting near-end loopbacks.

### Equipment Network Loopback Command

When you invoke the **equipment network loopback** command, the near-end T3 port ignores far-end loopback requests.

```
Router(controller)# interface dsx3 1/0/0
Router(config-if)# equipment network loopback
```

# Fallback-Clocking Command

Use the **fallback-clocking** command to force the setting on the master clock to *internal* on receipt of an AIS message. This command is used only for E3 serial interfaces.

**Note** We recommend that you use the **fallback clock** command only in countries in which regulations require it.

Use the **fallback-clocking** command using the following format:

```
[no] fallback-clocking [on | off]
```

The default is off.

Use the **no** form of this command to restore the E3 interface default, which causes the controller to revert back to programmed E3 clocking.

In the following example, the fallback-clocking is set to **on**.

```
Router(config-if)# interface dsx3 1/0/0
Router(config-if)# fallback-clocking on
```

# Setting the Framing Type

Use the **framing** command to specify the framing type for the T3 interface using the following format:

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

Where:

- **auto-detect** detects whether or not the far-end T3 line is currently using m13 or c-bit framing and automatically configures itself to match.

  **Note** If the port detects that framing rapidly toggles between m13 and c-bit framing, the port forces m13 framing.

- **c-bit** specifies c-bit framing.
- **m13** specifies m13 framing. m13 is a Cisco framing method that functions the same as m23 framing.

The default is no framing.

Use the **no** form of this command to restore the default **auto-detect** framing type.

In the following example, framing is set to **m13**:

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

# Specifying the Idle-Character

Use the **idle-character command** to set a specific character on the interface for transmission between HDLC packets using the following format:

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

Where:

- **flags** sets an idle character of **0x7e**
- **marks** sets an idle character of all **0xff**

The default idle character is 0x7e.

Use the **no** form of this command to return the idle character to its default value.

In the following example, the idle character is set to flags (**0x7e**):

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

**Note** Some systems interpret marks (0xff), as an abort signal. Therefore, the flags (0x7e) setting is preferred.

# Performing Loopback Tests

Use the **loopback** command to assign an E3 or T3 interface into loopback test using the following format:

```
[no] loopback network [line | payload]
[no] loopback [local | remote]
```

Where:

- **network line** loops the data back to the network
- **network payload** loops the payload portion back to the network
- **local** loops the line card output data back toward the line card and sends an AIS message
- **remote** sends a far end alarm control (FEAC) request to the far-end (remote) device, requesting that it enter a network line loopback

> **Note** A remote loopback test is possible only when the line is configured for c-bit framing.

The default is no loopback.

To cancel a loopback test, use the **no** form of the command.

In the following example, the loopback network is set to line:

```
Router(config)# dsx3 1/0/0
Router(config-if)# network loopback line
```

See the *Cisco IOS Interface and Hardware Component Configuration Guide,* and the *Cisco 10000 Series Internet Router Troubleshooting Guide* for details about loopback testing.

# Entering MDL Messages

Use the maintenance data link (MDL) message strings and codes to send messages in response to far-end inquiries.

> **Note** MDL messages are supported only when the T3 framing is set to c-bit (see the "Setting the Framing Type" section on page 15-13.)

You can configure MDL messages (as defined in the ANSI T1.107a-1990 specification) on a T3 interface using the **mdl** command in the following format:

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

Where:

- **transmit** enables one of the following transmissions:
  - **path** enables transmission of the MDL path message
  - **idle-signal** enables transmission of the MDL idle-signal message
  - **test-signal** enables transmission of the MDL test-signal message
- **string** enables one of the following identification codes:

- **eic** is the equipment identification code (up to 10 characters).
- **lic** is the location identification code (up to 11 characters).
- **fic** is the frame identification code (up to 10 characters).
- **unit** is the unit identification code (up to 6 characters).
- **pfi** is the facility identification code to include in the MDL path message (up to 38 characters).
- **port** is the equipment port (which initiates the idle signal) to include in the MDL idle signal message (up to 38 characters).
- **generator** is the generator number to include in the MDL test signal message (up to 38 characters).

The default is no mdl message.

Use the **no** form of the command to remove an MDL message.

Examples of configuring MDL messages are:

- Enable the MDL path message transmission.

```
Router(config-controller)# mdl transmit path
```

- Enable the MDL idle signal message transmission.

```
Router(config-controller)# mdl transmit idle-signal
```

- Enable the MDL test signal message transmission.

```
Router(config-controller)# mdl transmit test-signal
```

- Enter the equipment identification code.

```
Router(config-controller)# mdl string eic router A
```

- Enter the location identification code.

```
Router(config-controller)# mdl string lic test network
```

- Enter the frame identification code.

```
Router(config-controller)# mdl string fic building b
```

- Enter the unit identification code.

```
Router(config-controller)# mdl string unit abc
```

- Enter the facility identification code.

```
Router(config-controller)# mdl string pfi string
```

- Enter the port number to send in the MDL idle signal message.

```
Router(config-controller)# mdl string port string
```

- Enter the generator number to send in the MDL test signal message.

```
Router(config-controller)# mdl string generator string
```

# Setting the National Bit

Use the **national bit** command to set bit 12 in the E3 frame, using the following format:

**[no] national bit [0 | 1]**

The default is 0.

Use the **no** form of this command to return to the default bit.

In the following example, the national bit is set to **1** on an E3 interface:

```
Router(config)# interface serial 1/0/0
Router(config-if)# national bit 1
```

# Enabling Scrambling

Scrambling can prevent some bit patterns from being mistakenly interpreted as alarms by switches placed between the DSUs. Use the **scramble** command to assist clock recovery on the receiving end using the following format:

**[no] scramble**

The default is no scramble.

In the following example, scrambling is enabled:

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

# PART 6

## Configuration Notes and Examples

CHAPTER 16

# Preparing for Line Card Installation and Configuration

This chapter provides general information about configuring line cards on the Cisco 10000 series router.

This chapter contains the following sections:



## Treatment of Newly Inserted Line Cards

After you insert a line card into the router, the software identifies the card and executes one of three actions, depending on the contents of the running configuration file:

- If the card is not described in the running configuration file, the system enters a basic configuration into the running configuration file. For example, if you insert a Gigabit Ethernet line card into slot 4, the system adds the following lines to the running configuration:

```
card 4/0 1gigethernet-1
.
.
.
interface GigabitEthernet4/0/0
no ip address
no ip directed-broadcast
no cdp enable
```

- If the card and slot are described in the running configuration file, the system applies that configuration to the card.
- If the card is of a type different from the one listed in the running configuration file, the software replaces the configuration with a basic configuration for the newly inserted card.

To find out the type of card that a slot is configured to use, enter the **show running-config** command.

**Tip** Most **show** commands support the pipe character and **begin** argument. For example, you can use the command **show running-config | begin card** to start the display of the running configuration file at the point where it lists the cards that are configured on your Cisco 10000 series router.

# Preprovisioning Line Card Slots

The line card slot preprovisioning feature allows you to preprovision a Cisco 10000 series router line card slot to accept a particular line card. Using the **card** command, you can configure interfaces for a line card that is not yet inserted into the router chassis.

The line card slot preprovisioning feature allows you to plan for future configurations. Preprovisioning a line card slot downloads software to the line card slot for the line card you are planning to install. Preprovisioning allows you to prepare in advance a complete configuration for the line card, including items you normally need to configure after initially configuring the line card, such as PVCs under an ATM line card or VLANs under an Ethernet line card.

Personnel responsible for provisioning and setup operations can preprovision line card slots in the Cisco 10000 series router to accept a particular line card. When the functionality is needed, installers can then insert the card into the router chassis.

The following example preprovisions line card slot 2 to accept a 24-port E1/T1 line card operating in E1 mode:

```
Router(config)# card 2/0 24che1t1-1 mode e1
```

## Changing the Provisioning of a Line Card Slot

Use the **card** command to change the provisioning of a line card slot. You can remove the old line card before or after using the **card** command to change the line card provisioning. The following example shows how to change the provisioning for line card slot 5 from the 1-Port Gigabit Ethernet Half-Height line card to the 4-Port OC3/STM-1 line card.

```
Router(config)# card 5/0 4oc3atm-1
```

## Feature History for Line Card Slot Preprovisioning

| Cisco IOS Release | Description | Required PRE |
|---|---|---|
| Release 12.0(17)ST | The line card slot preprovisioning feature was introduced on the Cisco 10000 series router for the following line cards:<br>• 1-Port Gigabit Ethernet line card<br>• 1-Port OC-12 Packet Over SONET line card<br>• 6-Port OC-3/STM-1 Packet Over SONET line card<br>• 6-Port Channelized T3 line card<br>• 1-Port Channelized OC-12/STM-4 line card<br>• 4-Port Channelized OC-3/STM-1 line card<br>• 8-Port Unchannelized E3/T3 line card<br>• 1-Port OC-12 ATM line card | PRE1 |
| Release 12.0(21)SX | This feature was modified to support the 4-Port OC3/STM-1 line card with intermediate-reach optics. | PRE1 |
| Release 12.0(22)S | This feature was modified to support the 24-port E1/T1 line card and the 8-Port Unchannelized E3/T3 line card. | PRE1 |

| Cisco IOS Release | Description | Required PRE |
|---|---|---|
| Release 12.0(23)S | This feature was modified to support the 1-Port Gigabit Ethernet Half-Height line card and the 8-Port Fast Ethernet Half-Height line card. | PRE1 |
| Release 12.2(15)BX | This feature was modified to support the 1-Port OC-48/STM-16 Packet Over SONET line card. | PRE2 |
| Release 12.3XI | This feature was integrated into Cisco IOS Release 12.3XI. | PRE2 |
| Release 12.2(28)SB | This feature was modified to support the 4-Port Channelized T3 Half-Height line card and the 4-Port OC3/STM-1 line card with long-reach optics. | PRE2 |

# card Command

To preprovision a line card slot in the Cisco 10000 series router, use the **card** command in global configuration mode. To remove preprovisioning from a line card slot, use the **no** form of this command.

**card** {*slot*/*subslot*} {**lchoc12-1** | **lgigethernet-l** | **1gigethernet-hh-1** | **loc12atm-1** | **loc12pos-1** | **1oc48dpt-pos-1** | **24che1t1-1** | **4chstm-1** | **4cht3-hh-1**| **4oc3atm-1** | **4oc3atm_lr-1** | **6cht3-1** | **6oc3pos-1** | **8e3ds3-1** | **8e3ds3atm-1** | **8fastethernet-1** [**mode** {**e1** | **t1**}]}

**no card** {*slot*/*subslot*}

## Syntax Description

| | |
|---|---|
| *slot* | Number of the line card slot in the Cisco 10000 series router to be preprovisioned. |
| *subslot* | Number of the line card subslot in the Cisco 10000 series router to be preprovisioned. |
| **lchoc12-1** | Preprovisions a line card slot for a 1-Port Channelized OC-12/STM-4 line card. |
| **lgigethernet-l** | Preprovisions a line card slot for a 1-Port Gigabit Ethernet line card. |
| **1gigethernet-hh-1** | Preprovisions a line card slot for a 1-Port Gigabit Ethernet Half-Height line card. |
| **loc12atm-1** | Preprovisions a line card slot for a 1-Port OC-12 ATM line card. |
| **loc12pos-1** | Preprovisions a line card slot for a 1-Port OC-12 Packet Over SONET line card. |
| **1oc48dpt-pos-1** | Preprovisions a line card slot for a 1-Port OC-48/STM-16 Packet Over SONET line card. |
| **24che1t1-1** | Preprovisions a line card slot for a 24-Port Channelized E1/T1 line card. |
| **4chstm-1** | Preprovisions a line card slot for a 4-Port Channelized OC-3/STM-1 line card. |
| **4cht3-hh-1** | Preprovisions a line card slot for a 4-Port Channelized T3 Half-Height line card. |
| **4oc3atm-1** | Preprovisions a line card slot for a 4-Port OC3/STM-1 line card with intermediate-reach optics. |
| **4oc3atm_lr-1** | Preprovisions a line card slot for a 4-Port OC3/STM-1 line card with long-reach optics. |

| | |
|---|---|
| **6cht3-1** | Preprovisions a line card slot for a 6-Port Channelized T3 line card. |
| **6oc3pos-1** | Preprovisions a line card slot for a 6-Port OC-3/STM-1 Packet Over SONET line card. |
| **8e3ds3-1** | Preprovisions a line card slot for an 8-Port Unchannelized E3/T3 line card. |
| **8e3ds3atm-1** | Preprovisions a line card slot for an 8-Port E3/DS3 ATM line card. |
| **8fastethernet-1** | Preprovisions a line card slot for an8-Port Fast Ethernet Half-Height line card. |
| **mode** | Indicates the mode of operation of the line card. Available options are E1 and T1. |

**Defaults**

The default mode of operation for the 24-port E1/T1 line card is E1.

**Command Modes**

Global configuration

**Command History**

| Release | Modification |
|---|---|
| Release 12.0(17)ST | This command was introduced on the Cisco 10000 series router for the following line cards:<br>• 1-Port Gigabit Ethernet line card<br>• 1-Port OC-12 Packet Over SONET line card<br>• 6-Port OC-3/STM-1 Packet Over SONET line card<br>• 6-Port Channelized T3 line card<br>• 1-Port Channelized OC-12/STM-4 line card<br>• 4-Port Channelized OC-3/STM-1 line card<br>• 8-Port Unchannelized E3/T3 line card<br>• 1-Port OC-12 ATM line card |
| Release 12.0(21)SX | This command was modified to support the 4-Port OC3/STM-1 line card with intermediate-reach optics. |
| Release 12.0(22)S | This command was modified to support the 24-port E1/T1 line card and the 8-Port Unchannelized E3/T3 line card. |
| Release 12.0(23)S | This command was modified to support the 1-Port Gigabit Ethernet Half-Height line card and the 8-Port Fast Ethernet Half-Height line card. |
| Release 12.2(15)BX | This command was modified to support the 1-Port OC-48/STM-16 Packet Over SONET line card. |
| Release 12.3XI | This command was integrated into Cisco IOS Release 12.3XI. |
| Release 12.2(28)SB | This command was modified to support the 4-Port Channelized T3 Half-Height line card. |

**Usage Guidelines**

You must specify a line card slot and subslot, and the line card for which you want to preprovision the line card slot.

If you insert a line card into a line card slot that has been preprovisioned for a different line card, the line card reports a boot failure in an error message to the console. Output from the **show ip interface brief** command indicates that the line card remains in an initialized state.

You can specify a mode of operation for the 24-port E1/T1 line card. If you do not, the line card operates in the E1 mode.

**Example**

The following example preprovisions line card slot 2 to accept a 24-port E1/T1 line card operating in E1 mode:

```
Router(config)# card 2/0 24che1t1-1 mode e1
```

# Resetting Line Cards

You can use the privileged EXEC mode **hw-module** command to reset a line card. For example:

```
Router# hw-module slot 2 reset
Router#
```

Resetting a line card effectively reloads the line card software onto the line card.

CHAPTER 17

# Provisioning a Subrate E3 or T3 Interface

Subrating a T3 or E3 interface reduces the peak access rate by limiting the data transfer rate. Subrate modes configure the line card to connect with Cisco port adapters and with customer premise Data Service Units (DSUs). Bandwidth provisioning tables are provided in this guide to help you connect the Cisco 10000 series router T3 or E3 interface with several manufacturer's DSUs.

You can provision a subrate E3 or T3 interface on the following Cisco 10000 series router line cards:

- 4-Port Channelized T3 Half-Height line card
- 6-Port Channelized T3 line card
- 8-Port Unchannelized E3/T3 line card
- 1-Port Channelized OC-12/STM4 line card
- 4-Port Channelized OC-3/STM-1 line card

The chapter contains the following sections:



## Implementing a Subrate T3 or E3 Interface

DSU manufacturers supporting a subrate T3 or E3 interface use proprietary methods of multiplexing. The Cisco IOS software command line interface (CLI) supports the **dsu mode** and **dsu bandwidth** commands, which configure the line card to emulate a DSU manufacturer's proprietary multiplexing scheme.

The default DSU mode is *cisco*. If the remote end is not a Cisco DS3 subrate interface, use the **dsu mode** command to specify the DSU.

For T3 interfaces, the Cisco 10000 series router supports the following DSUs:

- Digital Link DL3100 T3 Access Multiplexer
- Larscom Access-T45 DS3 Network Service Unit
- Verilink HDM 2 182 DS3 Module
- Kentrox DataSMART T3/E3 IDSU
- Adtran T3SU 300 T3 DSU/CSU

For E3 interfaces, the Cisco 10000 series router supports the following DSUs:

- Digital Link DL3100E E3 Access Multiplexer
- Kentrox DataSMART T3/E3 IDSU

The following sample procedure shows how to use the **dsu mode** and **dsu bandwidth** commands for connecting to a Digital Link DL3100 T3 Access Multiplexer.

**Step 1** Specify a DSU mode for a selected interface using the **dsu mode** *mode* command from interface configuration mode, where *mode* is the name of the DSU manufacturer.

For example:

```
Router(config)# interface serial 1/0/0
Router(config-if)# dsu mode Digital-link
```

**Step 2** Specify the bandwidth for the T3 or E3 interface by using the **dsu bandwidth** *bandwidth* command, where *bandwidth* is a value in Kbits/sec. This value indicates the bandwidth needed to match the T3 or E3 interface to the nearest number of timeslots for the configured DSU mode.

For example:

```
Router(config)# interface serial 1/0/0
Router(config-if)# dsu bandwidth 3600
```

See the Subrate T3 Bandwidth Tables section or the Subrate E3 Bandwidth Tables section for more information.

**Step 3** After you specify the mode and bandwidth on the line card, you must specify the timeslots on the DSU. For more information, refer to the technical documentation provided with the DSU.

# Subrate T3 Bandwidth Tables

The following tables contain the **dsu bandwidth** command values, based on the dsu mode setting and the number of DSU timeslots. If you specify a dsu bandwidth value as shown in the table, the DS3 subrate interface uses the actual bandwidth and the number of DSU timeslots indicated.

The tables below provide the DSU manufacturer-specific T3 bandwidth settings for the following DSU products:

- Digital Link DL3100 T3 Access Multiplexer—Table 17-1 on page 17-2
- Larscom Access-T45 DS3 Network Service Unit—Table 17-2 on page 17-7
- Verilink HDM 2182 DS3 DSU Module—Table 17-3 on page 17-7
- Kentrox DataSMART T3/E3 IDSU—Table 17-4 on page 17-8
- Adtran T3SU 300 T3 DSU/CSU—Table 17-5 on page 17-10

***Table 17-1 Digital Link DL3100 T3 Access Multiplexer***

| DSU Timeslots | Actual Bandwidth | dsu bandwidth Value |
|---|---|---|
| 1 | 300,746 | 300 |
| 2 | 601,492 | 600 |
| 3 | 902,239 | 900 |

*Table 17-1 Digital Link DL3100 T3 Access Multiplexer (continued)*

| DSU Timeslots | Actual Bandwidth | dsu bandwidth Value |
|---|---|---|
| 4 | 1,202,985 | 1200 |
| 5 | 1,503,731 | 1500 |
| 6 | 1,804,477 | 1800 |
| 7 | 2,105,224 | 2100 |
| 8 | 2,405,970 | 2400 |
| 9 | 2,706,716 | 2700 |
| 10 | 3,007,462 | 3000 |
| 11 | 3,308,208 | 3300 |
| 12 | 3,608,955 | 3600 |
| 13 | 3,909,701 | 3900 |
| 14 | 4,210,447 | 4200 |
| 15 | 4,511,193 | 4500 |
| 16 | 4,811,939 | 4800 |
| 17 | 5,112,686 | 5100 |
| 18 | 5,413,432 | 5400 |
| 19 | 5,714,178 | 5700 |
| 20 | 6,014,924 | 6000 |
| 21 | 6,315,671 | 6300 |
| 22 | 6,616,417 | 6600 |
| 23 | 6,917,163 | 6900 |
| 24 | 7,217,909 | 7200 |
| 25 | 7,518,655 | 7500 |
| 26 | 7,819,402 | 7800 |
| 27 | 8,120,148 | 8100 |
| 28 | 8,420,894 | 8400 |
| 29 | 8,721,640 | 8700 |
| 30 | 9,022,387 | 9000 |
| 31 | 9,323,133 | 9300 |
| 32 | 9,623,879 | 9600 |
| 33 | 9,924,625 | 9900 |
| 34 | 10,225,371 | 10200 |
| 35 | 10,526,118 | 10500 |
| 36 | 10,826,864 | 10800 |
| 37 | 11,127,610 | 11100 |
| 38 | 11,428,356 | 11400 |
| 39 | 11,729,103 | 11700 |

*Table 17-1 Digital Link DL3100 T3 Access Multiplexer (continued)*

| DSU Timeslots | Actual Bandwidth | dsu bandwidth Value |
|---|---|---|
| 40 | 12,029,849 | 12000 |
| 41 | 12,330,595 | 12300 |
| 42 | 12,631,341 | 12600 |
| 43 | 12,932,087 | 12900 |
| 44 | 13,232,834 | 13200 |
| 45 | 13,533,580 | 13500 |
| 46 | 13,834,326 | 13800 |
| 47 | 14,135,072 | 14100 |
| 48 | 14,435,818 | 14400 |
| 49 | 14,736,565 | 14700 |
| 50 | 15,037,311 | 15000 |
| 51 | 15,338,057 | 15300 |
| 52 | 15,638,803 | 15600 |
| 53 | 15,939,550 | 15900 |
| 54 | 16,240,296 | 16200 |
| 55 | 16,541,042 | 16500 |
| 56 | 16,841,788 | 16800 |
| 57 | 17,142,534 | 17100 |
| 58 | 17,443,281 | 17400 |
| 59 | 17,744,027 | 17700 |
| 60 | 18,044,773 | 18000 |
| 61 | 18,345,519 | 18300 |
| 62 | 18,646,266 | 18600 |
| 63 | 18,947,012 | 18900 |
| 64 | 19,247,758 | 19200 |
| 65 | 19,548,504 | 19500 |
| 66 | 19,849,250 | 19800 |
| 67 | 20,149,997 | 20100 |
| 68 | 20,450,743 | 20500 |
| 69 | 20,751,489 | 20800 |
| 70 | 21,052,235 | 21100 |
| 71 | 21,352,982 | 21400 |
| 72 | 21,653,728 | 21700 |
| 73 | 21,954,474 | 22000 |
| 74 | 22,255,220 | 22300 |
| 75 | 22,555,966 | 22600 |

*Table 17-1 Digital Link DL3100 T3 Access Multiplexer (continued)*

| DSU Timeslots | Actual Bandwidth | dsu bandwidth Value |
|---|---|---|
| 76 | 22,856,713 | 22900 |
| 77 | 23,157,459 | 23200 |
| 78 | 23,458,205 | 23500 |
| 79 | 23,758,951 | 23800 |
| 80 | 24,059,697 | 24100 |
| 81 | 24,360,444 | 24400 |
| 82 | 24,661,190 | 24700 |
| 83 | 24,961,936 | 25000 |
| 84 | 25,262,682 | 25300 |
| 85 | 25,563,429 | 25600 |
| 86 | 25,864,175 | 25900 |
| 87 | 26,164,921 | 26200 |
| 88 | 26,465,667 | 26500 |
| 89 | 26,766,413 | 26800 |
| 90 | 27,067,160 | 27100 |
| 91 | 27,367,906 | 27400 |
| 92 | 27,668,652 | 27700 |
| 93 | 27,969,398 | 28000 |
| 94 | 28,270,145 | 28300 |
| 95 | 28,570,891 | 28600 |
| 96 | 28,871,637 | 28900 |
| 97 | 29,172,383 | 29200 |
| 98 | 29,473,129 | 29500 |
| 99 | 29,773,876 | 29800 |
| 100 | 30,074,622 | 30100 |
| 101 | 30,375,368 | 30400 |
| 102 | 30,676,114 | 30700 |
| 103 | 30,976,861 | 31000 |
| 104 | 31,277,607 | 31300 |
| 105 | 31,578,353 | 31600 |
| 106 | 31,879,099 | 31900 |
| 107 | 32,179,845 | 32200 |
| 108 | 32,480,592 | 32500 |
| 109 | 32,781,338 | 32800 |
| 110 | 33,082,084 | 33100 |
| 111 | 33,382,830 | 33400 |

*Table 17-1 Digital Link DL3100 T3 Access Multiplexer (continued)*

| DSU Timeslots | Actual Bandwidth | dsu bandwidth Value |
|---|---|---|
| 112 | 33,683,576 | 33700 |
| 113 | 33,984,323 | 34000 |
| 114 | 34,285,069 | 34300 |
| 115 | 34,585,815 | 34600 |
| 116 | 34,886,561 | 34900 |
| 117 | 35,187,308 | 35200 |
| 118 | 35,488,054 | 35500 |
| 119 | 35,788,800 | 35800 |
| 120 | 36,089,546 | 36100 |
| 121 | 36,390,292 | 36400 |
| 122 | 36,691,039 | 36700 |
| 123 | 36,991,785 | 37000 |
| 124 | 37,292,531 | 37300 |
| 125 | 37,593,277 | 37600 |
| 126 | 37,894,024 | 37900 |
| 127 | 38,194,770 | 38200 |
| 128 | 38,495,516 | 38500 |
| 129 | 38,796,262 | 38800 |
| 130 | 39,097,008 | 39100 |
| 131 | 39,397,755 | 39400 |
| 132 | 39,698,501 | 39700 |
| 133 | 39,999,247 | 40000 |
| 134 | 40,299,993 | 40300 |
| 135 | 40,600,739 | 40600 |
| 136 | 40,901,486 | 40900 |
| 137 | 41,202,232 | 41200 |
| 138 | 41,502,978 | 41500 |
| 139 | 41,803,724 | 41800 |
| 140 | 42,104,471 | 42100 |
| 141 | 42,405,217 | 42400 |
| 142 | 42,705,963 | 42700 |
| 143 | 43,006,709 | 43000 |
| 144 | 43,307,455 | 43300 |
| 145 | 43,608,202 | 43600 |
| 146 | 43,908,948 | 43900 |
| 147 | 44,209,694 | 44210 |

*Table 17-2 Larscom Access-T45 DS3 Network Service Unit*

| DSU Timeslots | Actual Bandwidth | dsu bandwidth Value |
|---|---|---|
| 1 | 3,157,835 | 3200 |
| 2 | 6,315,671 | 6300 |
| 3 | 9,473,506 | 9500 |
| 4 | 12,631,341 | 12600 |
| 5 | 15,789,176 | 15800 |
| 6 | 18,947,012 | 18900 |
| 7 | 22,104,847 | 22100 |
| 8 | 25,262,682 | 25300 |
| 9 | 28,420,518 | 28400 |
| 10 | 31,578,353 | 31600 |
| 11 | 34,736,188 | 34700 |
| 12 | 37,894,024 | 37900 |
| 13 | 41,051,859 | 41100 |
| 14 | 44,209,694 | 44210 |

*Table 17-3 Verilink HDM 2182 DS3 DSU Module*

| DSU Timeslots | Actual Bandwidth | dsu bandwidth Value |
|---|---|---|
| 1 | 1,578,918 | 1600 |
| 2 | 3,157,835 | 3200 |
| 3 | 4,736,753 | 4700 |
| 4 | 6,315,671 | 6300 |
| 5 | 7,894,588 | 7900 |
| 6 | 9,473,506 | 9500 |
| 7 | 11,052,424 | 11100 |
| 8 | 12,631,341 | 12600 |
| 9 | 14,210,259 | 14200 |
| 10 | 15,789,176 | 15800 |
| 11 | 17,368,094 | 17400 |
| 12 | 18,947,012 | 18900 |
| 13 | 20,525,929 | 20500 |
| 14 | 22,104,847 | 22100 |
| 15 | 23,683,765 | 23700 |
| 16 | 25,262,682 | 25300 |
| 17 | 26,841,600 | 26800 |
| 18 | 28,420,518 | 28400 |

*Table 17-3 Verilink HDM 2182 DS3 DSU Module (continued)*

| DSU Timeslots | Actual Bandwidth | dsu bandwidth Value |
|---|---|---|
| 19 | 29,999,435 | 30000 |
| 20 | 31,578,353 | 31600 |

*Table 17-4 Kentrox DataSMART T3/E3 IDSU*

| DSU Bandwidth | Actual Bandwidth | dsu bandwidth Value |
|---|---|---|
| 1.5 | 1,500,000 | 1500 |
| 2.0 | 2,000,000 | 2000 |
| 2.5 | 2,500,000 | 2500 |
| 3.0 | 3,000,000 | 3000 |
| 3.5 | 3,500,000 | 3500 |
| 4.0 | 4,000,000 | 4000 |
| 4.5 | 4,500,000 | 4500 |
| 5.0 | 5,000,000 | 5000 |
| 5.5 | 5,500,000 | 5500 |
| 6.0 | 6,000,000 | 6000 |
| 6.5 | 6,500,000 | 6500 |
| 7.0 | 7,000,000 | 7000 |
| 7.5 | 7,500,000 | 7500 |
| 8.0 | 8,000,000 | 8000 |
| 8.5 | 8,500,000 | 8500 |
| 9.0 | 9,000,000 | 9000 |
| 9.5 | 9,500,000 | 9500 |
| 10.0 | 10,000,000 | 10000 |
| 10.5 | 10,500,000 | 10500 |
| 11.0 | 11,000,000 | 11000 |
| 11.5 | 11,500,000 | 11500 |
| 12.0 | 12,000,000 | 12000 |
| 12.5 | 12,500,000 | 12500 |
| 13.0 | 13,000,000 | 13000 |
| 13.5 | 13,500,000 | 13500 |
| 14.0 | 14,000,000 | 14000 |
| 14.5 | 14,500,000 | 14500 |
| 15.0 | 15,000,000 | 15000 |
| 15.5 | 15,500,000 | 15500 |
| 16.0 | 16,000,000 | 16000 |

*Table 17-4 Kentrox DataSMART T3/E3 IDSU (continued)*

| DSU Bandwidth | Actual Bandwidth | dsu bandwidth Value |
|---|---|---|
| 16.5 | 16,500,000 | 16500 |
| 17.0 | 17,000,000 | 17000 |
| 17.5 | 17,500,000 | 17500 |
| 18.0 | 18,000,000 | 18000 |
| 18.5 | 18,500,000 | 18500 |
| 19.0 | 19,000,000 | 19000 |
| 19.5 | 19,500,000 | 19500 |
| 20.0 | 20,000,000 | 20000 |
| 20.5 | 20,500,000 | 20500 |
| 21.0 | 21,000,000 | 21000 |
| 21.5 | 21,500,000 | 21500 |
| 22.0 | 22,000,000 | 22000 |
| 22.5 | 22,500,000 | 22500 |
| 23.0 | 23,000,000 | 23000 |
| 23.5 | 23,500,000 | 23500 |
| 24.0 | 24,000,000 | 24000 |
| 24.5 | 24,500,000 | 24500 |
| 25.0 | 25,000,000 | 25000 |
| 25.5 | 25,500,000 | 25500 |
| 26.0 | 26,000,000 | 26000 |
| 26.5 | 26,500,000 | 26500 |
| 27.0 | 27,000,000 | 27000 |
| 27.5 | 27,500,000 | 27500 |
| 28.0 | 28,000,000 | 28000 |
| 28.5 | 28,500,000 | 28500 |
| 29.0 | 29,000,000 | 29000 |
| 29.5 | 29,500,000 | 29500 |
| 30.0 | 30,000,000 | 30000 |
| 30.5 | 30,500,000 | 30500 |
| 31.0 | 31,000,000 | 31000 |
| 31.5 | 31,500,000 | 31500 |
| 32.0 | 32,000,000 | 32000 |
| 32.5 | 32,500,000 | 32500 |
| 33.0 | 33,000,000 | 33000 |
| 33.5 | 33,500,000 | 33500 |
| 34.0 | 34,000,000 | 34000 |

*Table 17-4 Kentrox DataSMART T3/E3 IDSU (continued)*

| DSU Bandwidth | Actual Bandwidth | dsu bandwidth Value |
|---|---|---|
| 34.5 | 34,500,000 | 34500 |
| 35.0 | 35,000,000 | 35000 |
| 45.0 | 44,209,694 | 44210 |

*Table 17-5 Adtran T3SU 300 T3 DSU/CSU*

| DSU Timeslots | Actual Bandwidth | dsu bandwidth Value |
|---|---|---|
| 1 | 75,187 | 80 |
| 2 | 150,373 | 150 |
| 3 | 225,560 | 230 |
| 4 | 300,746 | 300 |
| 5 | 375,933 | 380 |
| 6 | 451,119 | 450 |
| 7 | 526,306 | 530 |
| 8 | 601,492 | 600 |
| 9 | 676,679 | 680 |
| 10 | 751,866 | 750 |
| 11 | 827,052 | 830 |
| 12 | 902,239 | 900 |
| 13 | 977,425 | 980 |
| 14 | 1,052,612 | 1050 |
| 15 | 1,127,798 | 1130 |
| 16 | 1,202,985 | 1200 |
| 17 | 1,278,171 | 1280 |
| 18 | 1,353,358 | 1350 |
| 19 | 1,428,545 | 1430 |
| 20 | 1,503,731 | 1500 |
| 21 | 1,578,918 | 1580 |
| 22 | 1,654,104 | 1650 |
| 23 | 1,729,291 | 1730 |
| 24 | 1,804,477 | 1800 |
| 25 | 1,879,664 | 1880 |
| 26 | 1,954,850 | 1950 |
| 27 | 2,030,037 | 2030 |
| 28 | 2,105,224 | 2110 |
| 29 | 2,180,410 | 2180 |

*Table 17-5 Adtran T3SU 300 T3 DSU/CSU (continued)*

| DSU Timeslots | Actual Bandwidth | dsu bandwidth Value |
|---|---|---|
| 30 | 2,255,597 | 2260 |
| 31 | 2,330,783 | 2330 |
| 32 | 2,405,970 | 2410 |
| 33 | 2,481,156 | 2480 |
| 34 | 2,556,343 | 2560 |
| 35 | 2,631,529 | 2630 |
| 36 | 2,706,716 | 2710 |
| 37 | 2,781,903 | 2780 |
| 38 | 2,857,089 | 2860 |
| 39 | 2,932,276 | 2930 |
| 40 | 3,007,462 | 3010 |
| 41 | 3,082,649 | 3080 |
| 42 | 3,157,835 | 3160 |
| 43 | 3,233,022 | 3230 |
| 44 | 3,308,208 | 3310 |
| 45 | 3,383,395 | 3380 |
| 46 | 3,458,582 | 3460 |
| 47 | 3,533,768 | 3530 |
| 48 | 3,608,955 | 3610 |
| 49 | 3,684,141 | 3680 |
| 50 | 3,759,328 | 3760 |
| 51 | 3,834,514 | 3830 |
| 52 | 3,909,701 | 3910 |
| 53 | 3,984,887 | 3980 |
| 54 | 4,060,074 | 4060 |
| 55 | 4,135,261 | 4140 |
| 56 | 4,210,447 | 4210 |
| 57 | 4,285,634 | 4290 |
| 58 | 4,360,820 | 4360 |
| 59 | 4,436,007 | 4440 |
| 60 | 4,511,193 | 4510 |
| 61 | 4,586,380 | 4590 |
| 62 | 4,661,566 | 4660 |
| 63 | 4,736,753 | 4740 |
| 64 | 4,811,939 | 4810 |
| 65 | 4,887,126 | 4890 |

*Table 17-5 Adtran T3SU 300 T3 DSU/CSU (continued)*

| DSU Timeslots | Actual Bandwidth | dsu bandwidth Value |
|---|---|---|
| 66 | 4,962,313 | 4960 |
| 67 | 5,037,499 | 5040 |
| 68 | 5,112,686 | 5110 |
| 69 | 5,187,872 | 5190 |
| 70 | 5,263,059 | 5260 |
| 71 | 5,338,245 | 5340 |
| 72 | 5,413,432 | 5410 |
| 73 | 5,488,618 | 5490 |
| 74 | 5,563,805 | 5560 |
| 75 | 5,638,992 | 5640 |
| 76 | 5,714,178 | 5710 |
| 77 | 5,789,365 | 5790 |
| 78 | 5,864,551 | 5860 |
| 79 | 5,939,738 | 5940 |
| 80 | 6,014,924 | 6010 |
| 81 | 6,090,111 | 6090 |
| 82 | 6,165,297 | 6170 |
| 83 | 6,240,484 | 6240 |
| 84 | 6,315,671 | 6320 |
| 85 | 6,390,857 | 6390 |
| 86 | 6,466,044 | 6470 |
| 87 | 6,541,230 | 6540 |
| 88 | 6,616,417 | 6620 |
| 89 | 6,691,603 | 6690 |
| 90 | 6,766,790 | 6770 |
| 91 | 6,841,976 | 6840 |
| 92 | 6,917,163 | 6920 |
| 93 | 6,992,350 | 6990 |
| 94 | 7,067,536 | 7070 |
| 95 | 7,142,723 | 7140 |
| 96 | 7,217,909 | 7220 |
| 97 | 7,293,096 | 7290 |
| 98 | 7,368,282 | 7370 |
| 99 | 7,443,469 | 7440 |
| 100 | 7,518,655 | 7520 |
| 101 | 7,593,842 | 7590 |

*Table 17-5 Adtran T3SU 300 T3 DSU/CSU (continued)*

| DSU Timeslots | Actual Bandwidth | dsu bandwidth Value |
|---|---|---|
| 102 | 7,669,029 | 7670 |
| 103 | 7,744,215 | 7740 |
| 104 | 7,819,402 | 7820 |
| 105 | 7,894,588 | 7890 |
| 106 | 7,969,775 | 7970 |
| 107 | 8,044,961 | 8040 |
| 108 | 8,120,148 | 8120 |
| 109 | 8,195,334 | 8200 |
| 110 | 8,270,521 | 8270 |
| 111 | 8,345,708 | 8350 |
| 112 | 8,420,894 | 8420 |
| 113 | 8,496,081 | 8500 |
| 114 | 8,571,267 | 8570 |
| 115 | 8,646,454 | 8650 |
| 116 | 8,721,640 | 8720 |
| 117 | 8,796,827 | 8800 |
| 118 | 8,872,013 | 8870 |
| 119 | 8,947,200 | 8950 |
| 120 | 9,022,387 | 9020 |
| 121 | 9,097,573 | 9100 |
| 122 | 9,172,760 | 9170 |
| 123 | 9,247,946 | 9250 |
| 124 | 9,323,133 | 9320 |
| 125 | 9,398,319 | 9400 |
| 126 | 9,473,506 | 9470 |
| 127 | 9,548,692 | 9550 |
| 128 | 9,623,879 | 9620 |
| 129 | 9,699,066 | 9700 |
| 130 | 9,774,252 | 9770 |
| 131 | 9,849,439 | 9850 |
| 132 | 9,924,625 | 9920 |
| 133 | 9,999,812 | 10000 |
| 134 | 10,074,998 | 10070 |
| 135 | 10,150,185 | 10150 |
| 136 | 10,225,371 | 10230 |
| 137 | 10,300,558 | 10300 |

*Table 17-5 Adtran T3SU 300 T3 DSU/CSU (continued)*

| DSU Timeslots | Actual Bandwidth | dsu bandwidth Value |
|---|---|---|
| 138 | 10,375,745 | 10380 |
| 139 | 10,450,931 | 10450 |
| 140 | 10,526,118 | 10530 |
| 141 | 10,601,304 | 10600 |
| 142 | 10,676,491 | 10680 |
| 143 | 10,751,677 | 10750 |
| 144 | 10,826,864 | 10830 |
| 145 | 10,902,050 | 10900 |
| 146 | 10,977,237 | 10980 |
| 147 | 11,052,424 | 11050 |
| 148 | 11,127,610 | 11130 |
| 149 | 11,202,797 | 11200 |
| 150 | 11,277,983 | 11280 |
| 151 | 11,353,170 | 11350 |
| 152 | 11,428,356 | 11430 |
| 153 | 11,503,543 | 11500 |
| 154 | 11,578,729 | 11580 |
| 155 | 11,653,916 | 11650 |
| 156 | 11,729,103 | 11730 |
| 157 | 11,804,289 | 11800 |
| 158 | 11,879,476 | 11880 |
| 159 | 11,954,662 | 11950 |
| 160 | 12,029,849 | 12030 |
| 161 | 12,105,035 | 12110 |
| 162 | 12,180,222 | 12180 |
| 163 | 12,255,408 | 12260 |
| 164 | 12,330,595 | 12330 |
| 165 | 12,405,782 | 12410 |
| 166 | 12,480,968 | 12480 |
| 167 | 12,556,155 | 12560 |
| 168 | 12,631,341 | 12630 |
| 169 | 12,706,528 | 12710 |
| 170 | 12,781,714 | 12780 |
| 171 | 12,856,901 | 12860 |
| 172 | 12,932,087 | 12930 |
| 173 | 13,007,274 | 13010 |

*Table 17-5 Adtran T3SU 300 T3 DSU/CSU (continued)*

| DSU Timeslots | Actual Bandwidth | dsu bandwidth Value |
|---|---|---|
| 174 | 13,082,461 | 13080 |
| 175 | 13,157,647 | 13160 |
| 176 | 13,232,834 | 13230 |
| 177 | 13,308,020 | 13310 |
| 178 | 13,383,207 | 13380 |
| 179 | 13,458,393 | 13460 |
| 180 | 13,533,580 | 13530 |
| 181 | 13,608,766 | 13610 |
| 182 | 13,683,953 | 13680 |
| 183 | 13,759,139 | 13760 |
| 184 | 13,834,326 | 13830 |
| 185 | 13,909,513 | 13910 |
| 186 | 13,984,699 | 13980 |
| 187 | 14,059,886 | 14060 |
| 188 | 14,135,072 | 14140 |
| 189 | 14,210,259 | 14210 |
| 190 | 14,285,445 | 14290 |
| 191 | 14,360,632 | 14360 |
| 192 | 14,435,818 | 14440 |
| 193 | 14,511,005 | 14510 |
| 194 | 14,586,192 | 14590 |
| 195 | 14,661,378 | 14660 |
| 196 | 14,736,565 | 14740 |
| 197 | 14,811,751 | 14810 |
| 198 | 14,886,938 | 14890 |
| 199 | 14,962,124 | 14960 |
| 200 | 15,037,311 | 15040 |
| 201 | 15,112,497 | 15110 |
| 202 | 15,187,684 | 15190 |
| 203 | 15,262,871 | 15260 |
| 204 | 15,338,057 | 15340 |
| 205 | 15,413,244 | 15410 |
| 206 | 15,488,430 | 15490 |
| 207 | 15,563,617 | 15560 |
| 208 | 15,638,803 | 15640 |
| 209 | 15,713,990 | 15710 |

*Table 17-5 Adtran T3SU 300 T3 DSU/CSU (continued)*

| DSU Timeslots | Actual Bandwidth | dsu bandwidth Value |
|---|---|---|
| 210 | 15,789,176 | 15790 |
| 211 | 15,864,363 | 15860 |
| 212 | 15,939,550 | 15940 |
| 213 | 16,014,736 | 16010 |
| 214 | 16,089,923 | 16090 |
| 215 | 16,165,109 | 16170 |
| 216 | 16,240,296 | 16240 |
| 217 | 16,315,482 | 16320 |
| 218 | 16,390,669 | 16390 |
| 219 | 16,465,855 | 16470 |
| 220 | 16,541,042 | 16540 |
| 221 | 16,616,229 | 16620 |
| 222 | 16,691,415 | 16690 |
| 223 | 16,766,602 | 16770 |
| 224 | 16,841,788 | 16840 |
| 225 | 16,916,975 | 16920 |
| 226 | 16,992,161 | 16990 |
| 227 | 17,067,348 | 17070 |
| 228 | 17,142,534 | 17140 |
| 229 | 17,217,721 | 17220 |
| 230 | 17,292,908 | 17290 |
| 231 | 17,368,094 | 17370 |
| 232 | 17,443,281 | 17440 |
| 233 | 17,518,467 | 17520 |
| 234 | 17,593,654 | 17590 |
| 235 | 17,668,840 | 17670 |
| 236 | 17,744,027 | 17740 |
| 237 | 17,819,213 | 17820 |
| 238 | 17,894,400 | 17890 |
| 239 | 17,969,587 | 17970 |
| 240 | 18,044,773 | 18040 |
| 241 | 18,119,960 | 18120 |
| 242 | 18,195,146 | 18200 |
| 243 | 18,270,333 | 18270 |
| 244 | 18,345,519 | 18350 |
| 245 | 18,420,706 | 18420 |

*Table 17-5 Adtran T3SU 300 T3 DSU/CSU (continued)*

| DSU Timeslots | Actual Bandwidth | dsu bandwidth Value |
|---|---|---|
| 246 | 18,495,892 | 18500 |
| 247 | 18,571,079 | 18570 |
| 248 | 18,646,266 | 18650 |
| 249 | 18,721,452 | 18720 |
| 250 | 18,796,639 | 18800 |
| 251 | 18,871,825 | 18870 |
| 252 | 18,947,012 | 18950 |
| 253 | 19,022,198 | 19020 |
| 254 | 19,097,385 | 19100 |
| 255 | 19,172,571 | 19170 |
| 256 | 19,247,758 | 19250 |
| 257 | 19,322,945 | 19320 |
| 258 | 19,398,131 | 19400 |
| 259 | 19,473,318 | 19470 |
| 260 | 19,548,504 | 19550 |
| 261 | 19,623,691 | 19620 |
| 262 | 19,698,877 | 19700 |
| 263 | 19,774,064 | 19770 |
| 264 | 19,849,250 | 19850 |
| 265 | 19,924,437 | 19920 |
| 266 | 19,999,624 | 20000 |
| 267 | 20,074,810 | 20070 |
| 268 | 20,149,997 | 20150 |
| 269 | 20,225,183 | 20230 |
| 270 | 20,300,370 | 20300 |
| 271 | 20,375,556 | 20380 |
| 272 | 20,450,743 | 20450 |
| 273 | 20,525,929 | 20530 |
| 274 | 20,601,116 | 20600 |
| 275 | 20,676,303 | 20680 |
| 276 | 20,751,489 | 20750 |
| 277 | 20,826,676 | 20830 |
| 278 | 20,901,862 | 20900 |
| 279 | 20,977,049 | 20980 |
| 280 | 21,052,235 | 21050 |
| 281 | 21,127,422 | 21130 |

*Table 17-5 Adtran T3SU 300 T3 DSU/CSU (continued)*

| DSU Timeslots | Actual Bandwidth | dsu bandwidth Value |
|---|---|---|
| 282 | 21,202,608 | 21200 |
| 283 | 21,277,795 | 21280 |
| 284 | 21,352,982 | 21350 |
| 285 | 21,428,168 | 21430 |
| 286 | 21,503,355 | 21500 |
| 287 | 21,578,541 | 21580 |
| 288 | 21,653,728 | 21650 |
| 289 | 21,728,914 | 21730 |
| 290 | 21,804,101 | 21800 |
| 291 | 21,879,287 | 21880 |
| 292 | 21,954,474 | 21950 |
| 293 | 22,029,661 | 22030 |
| 294 | 22,104,847 | 22100 |
| 295 | 22,180,034 | 22180 |
| 296 | 22,255,220 | 22260 |
| 297 | 22,330,407 | 22330 |
| 298 | 22,405,593 | 22410 |
| 299 | 22,480,780 | 22480 |
| 300 | 22,555,966 | 22560 |
| 301 | 22,631,153 | 22630 |
| 302 | 22,706,339 | 22710 |
| 303 | 22,781,526 | 22780 |
| 304 | 22,856,713 | 22860 |
| 305 | 22,931,899 | 22930 |
| 306 | 23,007,086 | 23010 |
| 307 | 23,082,272 | 23080 |
| 308 | 23,157,459 | 23160 |
| 309 | 23,232,645 | 23230 |
| 310 | 23,307,832 | 23310 |
| 311 | 23,383,018 | 23380 |
| 312 | 23,458,205 | 23460 |
| 313 | 23,533,392 | 23530 |
| 314 | 23,608,578 | 23610 |
| 315 | 23,683,765 | 23680 |
| 316 | 23,758,951 | 23760 |
| 317 | 23,834,138 | 23830 |

*Table 17-5 Adtran T3SU 300 T3 DSU/CSU (continued)*

| DSU Timeslots | Actual Bandwidth | dsu bandwidth Value |
|---|---|---|
| 318 | 23,909,324 | 23910 |
| 319 | 23,984,511 | 23980 |
| 320 | 24,059,697 | 24060 |
| 321 | 24,134,884 | 24130 |
| 322 | 24,210,071 | 24210 |
| 323 | 24,285,257 | 24290 |
| 324 | 24,360,444 | 24360 |
| 325 | 24,435,630 | 24440 |
| 326 | 24,510,817 | 24510 |
| 327 | 24,586,003 | 24590 |
| 328 | 24,661,190 | 24660 |
| 329 | 24,736,376 | 24740 |
| 330 | 24,811,563 | 24810 |
| 331 | 24,886,750 | 24890 |
| 332 | 24,961,936 | 24960 |
| 333 | 25,037,123 | 25040 |
| 334 | 25,112,309 | 25110 |
| 335 | 25,187,496 | 25190 |
| 336 | 25,262,682 | 25260 |
| 337 | 25,337,869 | 25340 |
| 338 | 25,413,055 | 25410 |
| 339 | 25,488,242 | 25490 |
| 340 | 25,563,429 | 25560 |
| 341 | 25,638,615 | 25640 |
| 342 | 25,713,802 | 25710 |
| 343 | 25,788,988 | 25790 |
| 344 | 25,864,175 | 25860 |
| 345 | 25,939,361 | 25940 |
| 346 | 26,014,548 | 26010 |
| 347 | 26,089,734 | 26090 |
| 348 | 26,164,921 | 26160 |
| 349 | 26,240,108 | 26240 |
| 350 | 26,315,294 | 26320 |
| 351 | 26,390,481 | 26390 |
| 352 | 26,465,667 | 26470 |
| 353 | 26,540,854 | 26540 |

*Table 17-5 Adtran T3SU 300 T3 DSU/CSU (continued)*

| DSU Timeslots | Actual Bandwidth | dsu bandwidth Value |
|---|---|---|
| 354 | 26,616,040 | 26620 |
| 355 | 26,691,227 | 26690 |
| 356 | 26,766,413 | 26770 |
| 357 | 26,841,600 | 26840 |
| 358 | 26,916,787 | 26920 |
| 359 | 26,991,973 | 26990 |
| 360 | 27,067,160 | 27070 |
| 361 | 27,142,346 | 27140 |
| 362 | 27,217,533 | 27220 |
| 363 | 27,292,719 | 27290 |
| 364 | 27,367,906 | 27370 |
| 365 | 27,443,092 | 27440 |
| 366 | 27,518,279 | 27520 |
| 367 | 27,593,466 | 27590 |
| 368 | 27,668,652 | 27670 |
| 369 | 27,743,839 | 27740 |
| 370 | 27,819,025 | 27820 |
| 371 | 27,894,212 | 27890 |
| 372 | 27,969,398 | 27970 |
| 373 | 28,044,585 | 28040 |
| 374 | 28,119,771 | 28120 |
| 375 | 28,194,958 | 28190 |
| 376 | 28,270,145 | 28270 |
| 377 | 28,345,331 | 28350 |
| 378 | 28,420,518 | 28420 |
| 379 | 28,495,704 | 28500 |
| 380 | 28,570,891 | 28570 |
| 381 | 28,646,077 | 28650 |
| 382 | 28,721,264 | 28720 |
| 383 | 28,796,450 | 28800 |
| 384 | 28,871,637 | 28870 |
| 385 | 28,946,824 | 28950 |
| 386 | 29,022,010 | 29020 |
| 387 | 29,097,197 | 29100 |
| 388 | 29,172,383 | 29170 |
| 389 | 29,247,570 | 29250 |

*Table 17-5 Adtran T3SU 300 T3 DSU/CSU (continued)*

| DSU Timeslots | Actual Bandwidth | dsu bandwidth Value |
|---|---|---|
| 390 | 29,322,756 | 29320 |
| 391 | 29,397,943 | 29400 |
| 392 | 29,473,129 | 29470 |
| 393 | 29,548,316 | 29550 |
| 394 | 29,623,503 | 29620 |
| 395 | 29,698,689 | 29700 |
| 396 | 29,773,876 | 29770 |
| 397 | 29,849,062 | 29850 |
| 398 | 29,924,249 | 29920 |
| 399 | 29,999,435 | 30000 |
| 400 | 30,074,622 | 30070 |
| 401 | 30,149,808 | 30150 |
| 402 | 30,224,995 | 30220 |
| 403 | 30,300,182 | 30300 |
| 404 | 30,375,368 | 30380 |
| 405 | 30,450,555 | 30450 |
| 406 | 30,525,741 | 30530 |
| 407 | 30,600,928 | 30600 |
| 408 | 30,676,114 | 30680 |
| 409 | 30,751,301 | 30750 |
| 410 | 30,826,487 | 30830 |
| 411 | 30,901,674 | 30900 |
| 412 | 30,976,861 | 30980 |
| 413 | 31,052,047 | 31050 |
| 414 | 31,127,234 | 31130 |
| 415 | 31,202,420 | 31200 |
| 416 | 31,277,607 | 31280 |
| 417 | 31,352,793 | 31350 |
| 418 | 31,427,980 | 31430 |
| 419 | 31,503,166 | 31500 |
| 420 | 31,578,353 | 31580 |
| 421 | 31,653,539 | 31650 |
| 422 | 31,728,726 | 31730 |
| 423 | 31,803,913 | 31800 |
| 424 | 31,879,099 | 31880 |
| 425 | 31,954,286 | 31950 |

*Table 17-5 Adtran T3SU 300 T3 DSU/CSU (continued)*

| DSU Timeslots | Actual Bandwidth | dsu bandwidth Value |
|---|---|---|
| 426 | 32,029,472 | 32030 |
| 427 | 32,104,659 | 32100 |
| 428 | 32,179,845 | 32180 |
| 429 | 32,255,032 | 32260 |
| 430 | 32,330,218 | 32330 |
| 431 | 32,405,405 | 32410 |
| 432 | 32,480,592 | 32480 |
| 433 | 32,555,778 | 32560 |
| 434 | 32,630,965 | 32630 |
| 435 | 32,706,151 | 32710 |
| 436 | 32,781,338 | 32780 |
| 437 | 32,856,524 | 32860 |
| 438 | 32,931,711 | 32930 |
| 439 | 33,006,897 | 33010 |
| 440 | 33,082,084 | 33080 |
| 441 | 33,157,271 | 33160 |
| 442 | 33,232,457 | 33230 |
| 443 | 33,307,644 | 33310 |
| 444 | 33,382,830 | 33380 |
| 445 | 33,458,017 | 33460 |
| 446 | 33,533,203 | 33530 |
| 447 | 33,608,390 | 33610 |
| 448 | 33,683,576 | 33680 |
| 449 | 33,758,763 | 33760 |
| 450 | 33,833,950 | 33830 |
| 451 | 33,909,136 | 33910 |
| 452 | 33,984,323 | 33980 |
| 453 | 34,059,509 | 34060 |
| 454 | 34,134,696 | 34130 |
| 455 | 34,209,882 | 34210 |
| 456 | 34,285,069 | 34290 |
| 457 | 34,360,255 | 34360 |
| 458 | 34,435,442 | 34440 |
| 459 | 34,510,629 | 34510 |
| 460 | 34,585,815 | 34590 |
| 461 | 34,661,002 | 34660 |

*Table 17-5 Adtran T3SU 300 T3 DSU/CSU (continued)*

| DSU Timeslots | Actual Bandwidth | dsu bandwidth Value |
|---|---|---|
| 462 | 34,736,188 | 34740 |
| 463 | 34,811,375 | 34810 |
| 464 | 34,886,561 | 34890 |
| 465 | 34,961,748 | 34960 |
| 466 | 35,036,934 | 35040 |
| 467 | 35,112,121 | 35110 |
| 468 | 35,187,308 | 35190 |
| 469 | 35,262,494 | 35260 |
| 470 | 35,337,681 | 35340 |
| 471 | 35,412,867 | 35410 |
| 472 | 35,488,054 | 35490 |
| 473 | 35,563,240 | 35560 |
| 474 | 35,638,427 | 35640 |
| 475 | 35,713,613 | 35710 |
| 476 | 35,788,800 | 35790 |
| 477 | 35,863,987 | 35860 |
| 478 | 35,939,173 | 35940 |
| 479 | 36,014,360 | 36010 |
| 480 | 36,089,546 | 36090 |
| 481 | 36,164,733 | 36160 |
| 482 | 36,239,919 | 36240 |
| 483 | 36,315,106 | 36320 |
| 484 | 36,390,292 | 36390 |
| 485 | 36,465,479 | 36470 |
| 486 | 36,540,666 | 36540 |
| 487 | 36,615,852 | 36620 |
| 488 | 36,691,039 | 36690 |
| 489 | 36,766,225 | 36770 |
| 490 | 36,841,412 | 36840 |
| 491 | 36,916,598 | 36920 |
| 492 | 36,991,785 | 36990 |
| 493 | 37,066,971 | 37070 |
| 494 | 37,142,158 | 37140 |
| 495 | 37,217,345 | 37220 |
| 496 | 37,292,531 | 37290 |
| 497 | 37,367,718 | 37370 |

*Table 17-5 Adtran T3SU 300 T3 DSU/CSU (continued)*

| DSU Timeslots | Actual Bandwidth | dsu bandwidth Value |
|---|---|---|
| 498 | 37,442,904 | 37440 |
| 499 | 37,518,091 | 37520 |
| 500 | 37,593,277 | 37590 |
| 501 | 37,668,464 | 37670 |
| 502 | 37,743,650 | 37740 |
| 503 | 37,818,837 | 37820 |
| 504 | 37,894,024 | 37890 |
| 505 | 37,969,210 | 37970 |
| 506 | 38,044,397 | 38040 |
| 507 | 38,119,583 | 38120 |
| 508 | 38,194,770 | 38190 |
| 509 | 38,269,956 | 38270 |
| 510 | 38,345,143 | 38350 |
| 511 | 38,420,329 | 38420 |
| 512 | 38,495,516 | 38500 |
| 513 | 38,570,703 | 38570 |
| 514 | 38,645,889 | 38650 |
| 515 | 38,721,076 | 38720 |
| 516 | 38,796,262 | 38800 |
| 517 | 38,871,449 | 38870 |
| 518 | 38,946,635 | 38950 |
| 519 | 39,021,822 | 39020 |
| 520 | 39,097,008 | 39100 |
| 521 | 39,172,195 | 39170 |
| 522 | 39,247,382 | 39250 |
| 523 | 39,322,568 | 39320 |
| 524 | 39,397,755 | 39400 |
| 525 | 39,472,941 | 39470 |
| 526 | 39,548,128 | 39550 |
| 527 | 39,623,314 | 39620 |
| 528 | 39,698,501 | 39700 |
| 529 | 39,773,687 | 39770 |
| 530 | 39,848,874 | 39850 |
| 531 | 39,924,061 | 39920 |
| 532 | 39,999,247 | 40000 |
| 533 | 40,074,434 | 40070 |

Table 17-5 Adtran T3SU 300 T3 DSU/CSU (continued)

| DSU Timeslots | Actual Bandwidth | dsu bandwidth Value |
|---|---|---|
| 534 | 40,149,620 | 40150 |
| 535 | 40,224,807 | 40220 |
| 536 | 40,299,993 | 40300 |
| 537 | 40,375,180 | 40380 |
| 538 | 40,450,366 | 40450 |
| 539 | 40,525,553 | 40530 |
| 540 | 40,600,739 | 40600 |
| 541 | 40,675,926 | 40680 |
| 542 | 40,751,113 | 40750 |
| 543 | 40,826,299 | 40830 |
| 544 | 40,901,486 | 40900 |
| 545 | 40,976,672 | 40980 |
| 546 | 41,051,859 | 41050 |
| 547 | 41,127,045 | 41130 |
| 548 | 41,202,232 | 41200 |
| 549 | 41,277,418 | 41280 |
| 550 | 41,352,605 | 41350 |
| 551 | 41,427,792 | 41430 |
| 552 | 41,502,978 | 41500 |
| 553 | 41,578,165 | 41580 |
| 554 | 41,653,351 | 41650 |
| 555 | 41,728,538 | 41730 |
| 556 | 41,803,724 | 41800 |
| 557 | 41,878,911 | 41880 |
| 558 | 41,954,097 | 41950 |
| 559 | 42,029,284 | 42030 |
| 560 | 42,104,471 | 42100 |
| 561 | 42,179,657 | 42180 |
| 562 | 42,254,844 | 42250 |
| 563 | 42,330,030 | 42330 |
| 564 | 42,405,217 | 42410 |
| 565 | 42,480,403 | 42480 |
| 566 | 42,555,590 | 42560 |
| 567 | 42,630,776 | 42630 |
| 568 | 42,705,963 | 42710 |
| 569 | 42,781,150 | 42780 |

*Table 17-5 Adtran T3SU 300 T3 DSU/CSU (continued)*

| DSU Timeslots | Actual Bandwidth | dsu bandwidth Value |
|---|---|---|
| 570 | 42,856,336 | 42860 |
| 571 | 42,931,523 | 42930 |
| 572 | 43,006,709 | 43010 |
| 573 | 43,081,896 | 43080 |
| 574 | 43,157,082 | 43160 |
| 575 | 43,232,269 | 43230 |
| 576 | 43,307,455 | 43310 |
| 577 | 43,382,642 | 43380 |
| 578 | 43,457,829 | 43460 |
| 579 | 43,533,015 | 43530 |
| 580 | 43,608,202 | 43610 |
| 581 | 43,683,388 | 43680 |
| 582 | 43,758,575 | 43760 |
| 583 | 43,833,761 | 43830 |
| 584 | 43,908,948 | 43910 |
| 585 | 43,984,134 | 43980 |
| 586 | 44,059,321 | 44060 |
| 587 | 44,134,508 | 44130 |
| 588 | 44,209,694 | 44210 |

# Subrate E3 Bandwidth Tables

The following tables contain the **dsu bandwidth** command values, based on the dsu mode setting and the number of DSU timeslots. If you specify a dsu bandwidth value as shown in the table, the E3 subrate interface uses the actual bandwidth and the number of DSU timeslots indicated.

The tables below provide the DSU manufacturer-specific E3 bandwidth settings for the following DSU products:

- Digital Link DL3100E E3 Access Multiplexer—Table 17-6 on page 17-26
- Kentrox DataSMART T3/E3 IDSU—Table 17-7 on page 17-29

*Table 17-6 Digital Link DL3100E E3 Access Multiplexer*

| DSU Timeslots | Actual Bandwidth | dsu bandwidth Value |
|---|---|---|
| 1 | 358,000 | 358 |
| 2 | 716,000 | 716 |
| 3 | 1,074,000 | 1074 |
| 4 | 1,432,000 | 1432 |
| 5 | 1,790,000 | 1790 |

*Table 17-6 Digital Link DL3100E E3 Access Multiplexer (continued)*

| DSU Timeslots | Actual Bandwidth | dsu bandwidth Value |
|---|---|---|
| 6 | 2,148,000 | 2148 |
| 7 | 2,506,000 | 2506 |
| 8 | 2,864,000 | 2864 |
| 9 | 3,222,000 | 3222 |
| 10 | 3,580,000 | 3580 |
| 11 | 3,938,000 | 3938 |
| 12 | 4,296,000 | 4296 |
| 13 | 4,654,000 | 4654 |
| 14 | 5,012,000 | 5012 |
| 15 | 5,370,000 | 5370 |
| 16 | 5,728,000 | 5728 |
| 17 | 6,086,000 | 6086 |
| 18 | 6,444,000 | 6444 |
| 19 | 6,802,000 | 6802 |
| 20 | 7,160,000 | 7160 |
| 21 | 7,518,000 | 7518 |
| 22 | 7,876,000 | 7876 |
| 23 | 8,234,000 | 8234 |
| 24 | 8,592,000 | 8592 |
| 25 | 8,950,000 | 8950 |
| 26 | 9,308,000 | 9308 |
| 27 | 9,666,000 | 9666 |
| 28 | 10,024,000 | 10024 |
| 29 | 10,382,000 | 10382 |
| 30 | 10,740,000 | 10740 |
| 31 | 11,098,000 | 11098 |
| 32 | 11,456,000 | 11456 |
| 33 | 11,814,000 | 11814 |
| 34 | 12,172,000 | 12172 |
| 35 | 12,530,000 | 12530 |
| 36 | 12,888,000 | 12888 |
| 37 | 13,246,000 | 13246 |
| 38 | 13,604,000 | 13604 |
| 39 | 13,962,000 | 13962 |
| 40 | 14,320,000 | 14320 |
| 41 | 14,678,000 | 14678 |

*Table 17-6 Digital Link DL3100E E3 Access Multiplexer (continued)*

| DSU Timeslots | Actual Bandwidth | dsu bandwidth Value |
|---|---|---|
| 42 | 15,036,000 | 15036 |
| 43 | 15,394,000 | 15394 |
| 44 | 15,752,000 | 15752 |
| 45 | 16,110,000 | 16110 |
| 46 | 16,468,000 | 16468 |
| 47 | 16,826,000 | 16826 |
| 48 | 17,184,000 | 17184 |
| 49 | 17,542,000 | 17542 |
| 50 | 17,900,000 | 17900 |
| 51 | 18,258,000 | 18258 |
| 52 | 18,616,000 | 18616 |
| 53 | 18,974,000 | 18974 |
| 54 | 19,332,000 | 19332 |
| 55 | 19,690,000 | 19690 |
| 56 | 20,048,000 | 20048 |
| 57 | 20,406,000 | 20406 |
| 58 | 20,764,000 | 20764 |
| 59 | 21,122,000 | 21122 |
| 60 | 21,480,000 | 21480 |
| 61 | 21,838,000 | 21838 |
| 62 | 22,196,000 | 22196 |
| 63 | 22,554,000 | 22554 |
| 64 | 22,912,000 | 22912 |
| 65 | 23,270,000 | 23270 |
| 66 | 23,628,000 | 23628 |
| 67 | 23,986,000 | 23986 |
| 68 | 24,344,000 | 24344 |
| 69 | 24,702,000 | 24702 |
| 70 | 25,060,000 | 25060 |
| 71 | 25,418,000 | 25418 |
| 72 | 25,776,000 | 25776 |
| 73 | 26,134,000 | 26134 |
| 74 | 26,492,000 | 26492 |
| 75 | 26,850,000 | 26850 |
| 76 | 27,208,000 | 27208 |
| 77 | 27,566,000 | 27566 |

*Table 17-6 Digital Link DL3100E E3 Access Multiplexer (continued)*

| DSU Timeslots | Actual Bandwidth | dsu bandwidth Value |
|---|---|---|
| 78 | 27,924,000 | 27924 |
| 79 | 28,282,000 | 28282 |
| 80 | 28,640,000 | 28640 |
| 81 | 28,998,000 | 28998 |
| 82 | 29,356,000 | 29356 |
| 83 | 29,714,000 | 29714 |
| 84 | 30,072,000 | 30072 |
| 85 | 30,430,000 | 30430 |
| 86 | 30,788,000 | 30788 |
| 87 | 31,146,000 | 31146 |
| 88 | 31,504,000 | 31504 |
| 89 | 31,862,000 | 31862 |
| 90 | 32,220,000 | 32220 |
| 91 | 32,578,000 | 32578 |
| 92 | 32,936,000 | 32936 |
| 93 | 33,294,000 | 33294 |
| 94 | 33,652,000 | 33652 |
| 95 | 34,010,000 | 34010 |

*Table 17-7 Kentrox DataSMART T3/E3 IDSU*

| DSU Timeslots | Actual Bandwidth | dsu bandwidth Value |
|---|---|---|
| 1 | 1,000,000 | 1000 |
| 1.5 | 1,500,000 | 1500 |
| 2 | 2,000,000 | 2000 |
| 2.5 | 2,500,000 | 2500 |
| 3 | 3,000,000 | 3000 |
| 3.5 | 3,500,000 | 3500 |
| 4 | 4,000,000 | 4000 |
| 4.5 | 4,500,000 | 4500 |
| 5 | 5,000,000 | 5000 |
| 5.5 | 5,500,000 | 5500 |
| 6 | 6,000,000 | 6000 |
| 6.5 | 6,500,000 | 6500 |
| 7 | 7,000,000 | 7000 |
| 7.5 | 7,500,000 | 7500 |

*Table 17-7 Kentrox DataSMART T3/E3 IDSU (continued)*

| DSU Timeslots | Actual Bandwidth | dsu bandwidth Value |
|---|---|---|
| 8 | 8,000,000 | 8000 |
| 8.5 | 8,500,000 | 8500 |
| 9 | 9,000,000 | 9000 |
| 9.5 | 9,500,000 | 9500 |
| 10 | 10,000,000 | 10000 |
| 10.5 | 10,500,000 | 10500 |
| 11 | 11,000,000 | 11000 |
| 11.5 | 11,500,000 | 11500 |
| 12 | 12,000,000 | 12000 |
| 12.5 | 12,500,000 | 12500 |
| 13 | 13,000,000 | 13000 |
| 13.5 | 13,500,000 | 13500 |
| 14 | 14,000,000 | 14000 |
| 14.5 | 14,500,000 | 14500 |
| 15 | 15,000,000 | 15000 |
| 15.5 | 15,500,000 | 15500 |
| 16 | 16,000,000 | 16000 |
| 16.5 | 16,500,000 | 16500 |
| 17 | 17,000,000 | 17000 |
| 17.5 | 17,500,000 | 17500 |
| 18 | 18,000,000 | 18000 |
| 18.5 | 18,500,000 | 18500 |
| 19 | 19,000,000 | 19000 |
| 19.5 | 19,500,000 | 19500 |
| 20 | 20,000,000 | 20000 |
| 20.5 | 20,500,000 | 20500 |
| 21 | 21,000,000 | 21000 |
| 21.5 | 21,500,000 | 21500 |
| 22 | 22,000,000 | 22000 |
| 22.5 | 22,500,000 | 22500 |
| 23 | 23,000,000 | 23000 |
| 23.5 | 23,500,000 | 23500 |
| 24 | 24,000,000 | 24000 |
| 24.5 | 24,500,000 | 24500 |
| 34 | 34,000,000 | 34000 |

CHAPTER 18

# Configuration Examples

This chapter provides Cisco IOS CLI configuration examples for the Cisco 10000 series router. Each example uses the commands you enter at the IOS command line interface (CLI).

This chapter contains the following examples:



## Example 1: Configuring an Unchannelized Subrate T3 Port, Point to Multipoint Frame Relay, and OSPF

This example provides the sequence of commands necessary to accomplish the following:

The CLI command sequence is based on the assumption that we begin at a privileged EXEC prompt. Descriptive headings inserted in the CLI text announce that the hardware or features are being enabled.

```
Router# config terminal
Enter configuration commands, one per line. End with CNTL/Z.
Router(config)# ip routing
Router(config)# !
```

### Configure an Unchannelized T3 Controller

Configure an unchannelized T3 controller and specify clock source:

```
Router(config)# controller T3 1/0/0
Router(config)# no channelized
Router(config-controller)# clock source line
Router(config-controller)# end
```

### Configure Subrate T3

Configure a subrate of 25,000 kbps on the T3 port:

```
Router(config)# interface Serial1/0/0/1:0
Router(config-if)# no ip address
Router(config-if)# no ip directed broadcast
Router(config-if)# dsu bandwidth 25000
Router(config-if)# dsu mode <dsu type>
```

### Configure Frame Relay Encapsulation

Configure Frame Relay encapsulation on interface Serial1/0/0/1:0:

```
Router(config-if)# encapsulation frame-relay
Router(config-if)# frame-relay lmi-n391dte 6
Router(config-if)# keepalive 10
Router(config-if)# frame-relay lmi-n392dte 3
Router(config-if)# frame-relay lmi-n393dte 4
Router(config-if)# no shutdown
Router(config-if)# ip address 20.0.0.1 255.255.255.0
```

### Configure Point to Multipoint Frame Relay

Configure the Cisco 10000 series router to use DLCI 101 to communicate with Router 1 and DLCI 102 to communicate with Router 2:

```
Router(config-if)# ip ospf network point-to-multipoint
Router(config-if)# frame-relay map ip 20.0.0.2 101 broadcast
Router(config-if)# frame-relay map ip 20.0.0.3 102 broadcast
Router(config-if)# no shutdown
```

### Create an OSPF Routing Process

Enable OSPF routing process 100. Define an interface on which OSPF runs and the area ID for that interface.

```
Router(config-if)# router ospf 100
Router(config-router)# network 20.0.0.0 0.255.255.255 area 0
Router(config-router)# end
```

# Example 2: OSPF, BGP, Channelized Full Rate T1

This example provides the sequence of commands necessary to accomplish the following:

- Create a Gigabit Ethernet Uplink Port, page 18-3
- Create a T3 Controller, page 18-3
- Create a Full-Rate Channelized T1 Interface, page 18-3
- Configure Frame Relay Encapsulation, page 18-3
- Enable an OSPF Routing Process, page 18-4
- Enable OSPF Route Redistribution, page 18-4
- Configure BGP to Redistribute Routes Between Autonomous Systems, page 18-4

The CLI command sequence below starts with the assumption that you begin at a privileged EXEC prompt. Descriptive headings inserted in the CLI text announce that the hardware or features are being enabled.

```
Router# config terminal
Enter configuration commands, one per line. End with CNTL/Z.
```

### Create a Gigabit Ethernet Uplink Port

Provision an operational gigabit Ethernet card:

```
Router(config)# interface GigabitEthernet8/0/0
Router(config-if)# ip address 125.1.1.2 255.255.255.0
Router(config-if)# keepalive
Router(config-if)# no shutdown
Router(config-if)# end
Router# config terminal
Enter configuration commands, one per line. End with CNTL/Z.
Router(config)# ip routing
```

### Create a T3 Controller

Enable a functional T3 controller on the CT3 line card:

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

### Create a Full-Rate Channelized T1 Interface

Configure a channelized, full-rate T1:

```
Router(config-controller)# t1 1 channel-group 0 timeslots 1-24
Router(config-controller)# t1 1 clock source Line
Router(config-controller)# interface Serial1/0/0/1:0
Router(config-if)# no ip address
```

### Configure Frame Relay Encapsulation

Enable Frame Relay encapsulation, create a Frame Relay subinterface, and specify the default LMI type:

```
Router(config-if)# encapsulation frame-relay
Router(config-if)# frame-relay lmi-n391dte 6
Router(config-if)# keepalive 10
```

```
Router(config-if)# frame-relay lmi-n392dte 3
Router(config-if)# frame-relay lmi-n393dte 4
Router(config-if)# no shutdown
Router(config-if)# interface Serial1/0/0/1:0.100 point-to-point
Router(config-subif)# ip address 128.1.1.2 255.255.255.0
Router(config-subif)# frame-relay interface-dlci 100
Router(config-fr-dlci)# no shutdown
Router(config-if)# end
Router# config terminal
Enter configuration commands, one per line. End with CNTL/Z.
```

### Enable an OSPF Routing Process

Create OSPF routing process 200, specify a range of IP addresses to be associated with the routing process, and assign an area ID to be associated with that range of IP addresses:

```
Router(config)# router ospf 200
Router(config-router)# network 125.1.1.0 0.0.0.255 area 0
```

### Enable OSPF Route Redistribution

Enable route redistribution through BGP:

```
Router(config-router)# redistribute bgp 200 subnets
```

### Configure BGP to Redistribute Routes Between Autonomous Systems

Enable BGP (starting at the first arrow), define a neighbor for autonomous system 300 (the second arrow). The **network** commands define the networks from which OSPF routes are injected into the BGP table.

```
Router(config-router)# router bgp 200
Router(config-router)# neighbor 128.1.1.1 remote-as 300
Router(config-router)# network 125.0.0.0
Router(config-router)# end
Router# config terminal
Enter configuration commands, one per line. End with CNTL/Z.
Router(config)# router bgp 200
Router(config-router)# network 130.1.0.0
Router(config-router)# end
Router# config terminal
Enter configuration commands, one per line. End with CNTL/Z.
Router(config)# router bgp 200
Router(config-router)# network 130.2.0.0
Router(config-router)# end
Router# config terminal
Enter configuration commands, one per line. End with CNTL/Z.
Router(config)# router bgp 200
Router(config-router)# network 130.3.0.0
Router(config-router)# end
Router# config terminal
Enter configuration commands, one per line. End with CNTL/Z.
```

```
Router(config)# router bgp 200
Router(config-router)# network 130.4.0.0
Router(config-router)# end
Router# config terminal
Enter configuration commands, one per line. End with CNTL/Z.
Router(config)# router bgp 200
Router(config-router)# network 130.5.0.0
Router(config-router)# end
Router#
```

# Example 3: Quality of Service Policy Propagation Using Border Gateway Protocol

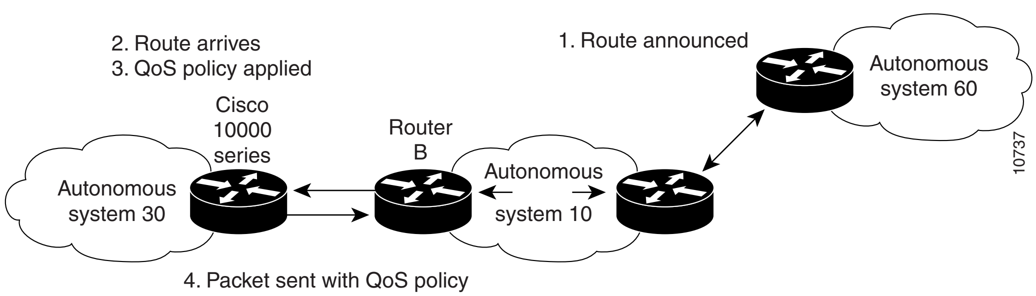
QoS Policy Propagation using Border Gateway Protocol (QPPB) allows you to classify packets by IP precedence based on BGP community lists, BGP autonomous system paths, and access lists. After a packet has been classified, you can use other QoS features such as committed access rate (CAR) and weighted random early detection (WRED) to specify and enforce policies to fit your business model.

The following example shows how to

1. Create route maps to match BGP community lists, access lists, and BGP AS paths
2. Apply IP precedence to routes learned from neighbors

In this example, the Cisco 10000 series router learns routes from autonomous system (AS) 10 and AS 60. QoS policy is applied to all packets that match the defined route maps. Any packets from the Cisco 10000 series router to AS 10 or AS 60 are sent to the appropriate QoS policy (Figure 18-1).

*Figure 18-1 Cisco 10000 Series Router Routes and QoS Policy Application*




### Cisco 10000 Series Router Configuration

```
Router(config)# router bgp 30
Router(config)# table-map precedence-map
Router(config-router)# neighbor 20.20.20.1 remote-as 10
Router(config-router)# neighbor 20.20.20.1 send-community
Router(config-router)# neigh 20.20.20.1 route-map precedence-map out
!
Router(config)# ip bgp-community new-format
```

Match community 1, set the IP precedence to priority, and set the QoS group to 1.

```
Router(config)# route-map precedence-map permit 10
Router(config-route-ma)# match community 1
```

```
Router(config-route-ma)# set ip precedence priority
Router(config-route-ma)# set ip qos-group 1
```

Match community 2 and set the IP precedence to immediate.

```
Router(config)# route-map precedence-map permit 20
Router(config-route-ma)# match community 2
Router(config-route-ma)# set ip precedence immediate
```

Match community 3 and set the IP precedence to Flash.

```
Router(config)# route-map precedence-map permit 30
Router(config-route-ma)# match community 3
Router(config-route-ma)# set ip precedence flash
```

Match community 4 and set the IP precedence to Flash-override.

```
Router(config)# route-map precedence-map permit 40
Router(config-route-ma)# match community 4
Router(config-route-ma)# set ip precedence flash-override
```

Match community 5 and set the IP precedence to critical.

```
Router(config)# route-map precedence-map permit 50
Router(config-route-ma)# match community 5
Router(config-route-ma)# set ip precedence critical
```

Match community 6 and set the IP precedence to internet.

```
Router(config)# route-map precedence-map permit 60
Router(config-route-ma)# match community 6
Router(config-route-ma)# set ip precedence internet
```

Match community 7 and set the IP precedence to network.

```
Router(config)# route-map precedence-map permit 70
Router(config-route-ma)# match community 7
Router(config-route-ma)# set ip precedence network
```

Match ip address access list 69 or match AS path 1, set the IP precedence to critical, and set the QoS group to 9.

```
Router(config)# route-map precedence-map permit 75
Router(config-route-ma)# match ip address 69
Router(config-route-ma)# match as-path 1
Router(config-route-ma)# set ip precedence critical
Router(config-route-ma)# set ip qos-group 9
```

For everything else, set the IP precedence to routine.

```
Router(config)# route-map precedence-map permit 80
Router(config-route-ma)# set ip precedence routine
```

Define the community lists.

```
Router(config)# ip community-list 1 permit 60:1
Router(config)# ip community-list 2 permit 60:2
Router(config)# ip community-list 3 permit 60:3
Router(config)# ip community-list 4 permit 60:4
Router(config)# ip community-list 5 permit 60:5
Router(config)# ip community-list 6 permit 60:6
Router(config)# ip community-list 7 permit 60:7
```

Define the AS path.

```
Router(config)# ip as-path access-list 1 permit ^10_60
```

Define the access list.

```
Router(config)# access-list 69 permit 69.0.0.0
```

#### Router B Running Configuration

```
RouterB(config)# router bgp 10
RouterB(config-router)# neighbor 30.30.30.1 remote-as 30
RouterB(config-router)# neighbor 30.30.30.1 send-community
RouterB(config-router)# neigh 30.30.30.1 route-map send_community out
!
RouterB(config)# ip bgp-community new-format
```

Match prefix 10 and set community to 60:1.

```
RouterB(config)# route-map send_community permit 10
RouterB(config-route-ma)# match ip address 10
RouterB(config-route-ma)# set community 60:1
```

Match prefix 20 and set community to 60:2.

```
RouterB(config)# route-map send_community permit 20
RouterB(config-route-ma)# match ip address 20
RouterB(config-route-ma)# set community 60:2
```

Match prefix 30 and set community to 60:3.

```
RouterB(config)# route-map send_community permit 30
RouterB(config-route-ma)# match ip address 30
RouterB(config-route-ma)# set community 60:3
```

Match prefix 40 and set community to 60:4.

```
RouterB(config)# route-map send_community permit 40
RouterB(config-route-ma)# match ip address 40
RouterB(config-route-ma)# set community 60:4
```

Match prefix 50 and set community to 60:5.

```
RouterB(config)# route-map send_community permit 50
RouterB(config-route-ma)# match ip address 50
RouterB(config-route-ma)# set community 60:5
```

Match prefix 60 and set community to 60:6.

```
RouterB(config)# route-map send_community permit 60
RouterB(config-route-ma)# match ip address 60
RouterB(config-route-ma)# set community 60:6
```

Match prefix 70 and set community to 60:7.

```
RouterB(config)# route-map send_community permit 70
RouterB(config-route-ma)# match ip address 70
RouterB(config-route-ma)# set community 60:7
```

For all others, set community to 60:8.

```
RouterB(config)# route-map send_community permit 80
RouterB(config-route-ma)# set community 60:8
```

Define the access lists.

```
RouterB(config)# access-list 10 permit 61.0.0.0
RouterB(config)# access-list 20 permit 62.0.0.0
RouterB(config)# access-list 30 permit 63.0.0.0
RouterB(config)# access-list 40 permit 64.0.0.0
```

```
RouterB(config)# access-list 50 permit 65.0.0.0
RouterB(config)# access-list 60 permit 66.0.0.0
RouterB(config)# access-list 70 permit 67.0.0.0
```

The following example shows how to configure several interfaces to classify packets based on the IP precedence and QoS group ID:

```
interface serial5/0/0/1:0
ip address 200.28.38.2 255.255.255.0
bgp-policy source ip-prec-map
no ip mroute-cache
no cdp enable
frame-relay interface-dlci 20 IETF

interface serial6/0/0/1:0
ip address 200.28.28.2 255.255.255.0
bgp-policy source qos-group
no ip mroute-cache
no cdp enable
frame-relay interface-dlci 20 IETF
```

# Example 4: MPLS Virtual Private Networks

MPLS can be used to create IP-based VPNs. MPLS VPNs offer all of the value of traditional VPNs, including reduced costs and enhanced security. In addition, because MPLS VPNs are created in Layer 3, they are more scalable, easier to configure, and easier to manage than Layer 2 VPNs.

Perform the tasks in the following sections to configure MPLS Virtual Private Networks (VPNs):

The following sections show examples of:

- Defining a VPN
- Configuring BGP Routing Sessions
- Configuring PE to PE Routing Sessions
- Configuring BGP PE to CE Routing Sessions
- Configuring RIP PE to CE Routing Sessions
- Configuring Static Route PE to CE Routing Sessions

### Defining a VPN

Enter VRF configuration mode, define the VPN routing instance by assigning a VRF name, and create routing and forwarding tables.

```
Router(config)# ip vrf go_fast_internet_company
Router(config-vrf)# rd 200.28.28.40:42
```

Create a list of import or export route target communities for the specified VRF.

```
Router(config-vrf)# route-target import 200.28.28.40:43
```

Associate the specified route map with the VRF and with an interface or subinterface.

```
Router(config-vrf)# import map go_fast
Router(config-if)# ip vrf forwarding go_fast_internet_company
```

#### Configuring BGP Routing Sessions

To configure BGP routing sessions in a provider network, use the following commands in router configuration mode on the PE router:

Configure the BGP routing process with the autonomous system number.

```
Router(config)# router bgp 42
```

Specify a neighbor's IP address or BGP peer group, identifying it to the local autonomous system.

```
Router(config-router)# neighbor 200.28.28.40
```

Activate the advertisement of the IPv4 address family.

```
Router(config-router)# neighbor 200.28.28.40 activate
```

#### Configuring PE to PE Routing Sessions

To configure PE to PE routing sessions in a provider network, use the following commands in router configuration mode on the PE router:

Define IBGP parameters for VPNv4 network-layer reachability information (NLRI) exchange.

```
Router(config-router)# address-family vpnv4 unicast
```

Define an IBGP session to exchange VPNv4 NLRIs.

```
Router(config-router-af)# neighbor 200.28.28.45 remote-as 48
Router(config-router-af)# exit
```

Activate the advertisement of the IPv4 address family.

```
Router(config-router)# neighbor 200.28.28.45 activate
```

#### Configuring BGP PE to CE Routing Sessions

To configure BGP PE to CE routing sessions, use the following commands in router configuration mode on the PE router:

Define EBGP parameters for PE to CE routing sessions.

```
Router(config-router)# address-family ipv4 unicast vrf go_fast_internet_company
```

Define an EBGP session between PE and CE routers and activate the advertisement of the IPv4 address family.

```
Router(config-router-af)# neighbor 200.28.28.46 remote-as 49
Router(config-router-af)# neighbor 200.28.28.46 activate
```

#### Configuring RIP PE to CE Routing Sessions

To configure RIP PE to CE routing sessions, use the following commands in router configuration mode on the PE router:

Enable RIP, define RIP parameters for PE to CE routing sessions, and enable RIP on the PE to CE link.

```
Router(config)# router rip
Router(config-router)# address-family ipv4 unicast vrf go_fast_internet_company
Router(config-router-af)# network 200.28.28.47
```

#### Configuring Static Route PE to CE Routing Sessions

To configure static route PE to CE routing sessions, use the following commands in router configuration mode on the PE router:

Define static route parameters for each PE to CE session and for each BGP PE to CE routing session.

```
Router(config)# ip route vrf go_fast_internet_company 200.28.28.46 255.255.255.0
200.28.28.50

Router(config-router)# address-family ipv4 unicast vrf go_fast_internet_company
```

Redistribute VRF static routes and directly connected networks into the VRF BGP table.

```
Router(config-router-af)# redistribute static
Router(config-router-af)# redistribute static connected
```

# GLOSSARY

## A

**AAA** — Authentication, authorization, and accounting.

**Access list** — List kept by a router to control access to or from the router for a number of services. For example, access lists can be used to prevent packets with a certain IP address from leaving a particular interface on the router.

**Alarm** — A status condition that shows that a module or port is experiencing an abnormal operating condition. See also Critical alarm, Major alarm, and Minor alarm.

**APS** — Automatic protection switching. A SONET switching mechanism that achieves network resiliency by automatically switching from a primary circuit to a secondary circuit. This switching process occurs if the primary circuit fails or if the error rate on the primary line exceeds a set threshold. The Cisco 10000 series ESR supports 1+1 APS, which provides permanent electrical bridging to the service and protection equipment, placed at both ends of the circuit.

## B

**Bellcore** — Bell Communications Research. An organization that performs research and development on behalf of the Regional Bell Operating Companies (RBOCs).

**BER** — Bit error rate. The ratio of received bits that contain errors to all received bits.

**BGP** — Border Gateway Protocol. An interdomain routing protocol that replaces EGP. BGP exchanges connection information with other BGP systems. It is defined by RFC 1163.

**Bit error rate** — See BER.

**Border Gateway Protocol** — See BGP.

## C

**CAR** — Committed Access Rate. A QoS feature that classifies packets based on such things as access lists, a MAC address, and IP precedence. It measures traffic rates and can be configured to take actions such as dropping packets or changing the ToS value.

**C-bit parity** — A modification of the M23 framing method for DS3 which frees the C bits for additional uses. See also M23.

**CB-WRED** Class-based weighted random early detection. Class-based WRED applies WRED to packets in different traffic classes. See also WRED.

**Channel** Communication path. Multiple channels can be multiplexed over a single cable in certain environments.

**Channel service unit** See CSU.

**Cisco IOS** Cisco system software that provides common functionality, scalability, and security for Cisco products. Cisco IOS allows centralized, integrated, and automated installation and management of internetworks, while ensuring support for a wide variety of protocols, media, services, and platforms.

**Clear channel DS3** A framed DS3 signal that is not multiplexed from 28 DS1 signals. Sometimes referred to as unchannelized DS3.

**CLI** Command line interface. Interface that allows the user to interact with the operating system by entering commands and optional arguments at the command prompt.

**Command Line Interface** See CLI.

**Committed access rate** See CAR.

**CRC** Cyclic redundancy check. Error-checking technique in which the frame recipient calculates a remainder by dividing frame contents by a prime binary divisor and then compares the calculated remainder to a value stored in the frame by the sending node.

**Critical alarm** An alarm condition that might affect most or all subscribers that connect to the reporting node. To obtain more information about a problem, use the **show facility-alarm status** command. See also Major alarm and Minor alarm.

**CSU** Channel service unit. Digital interface device that connects end-user equipment to the local digital telephone loop. Often referred to, together with DSU, as CSU/DSU. See also DSU.

**Cyclic redundancy check** See CRC.

## D

**Data service unit** See DSU.

**Differentiated service code point** See DSCP.

**DS0** Digital signal level 0. Framing specification used in transmitting digital signals over a single channel at 64 kbps on a T1 facility. Compare with DS1 and DS3.

**DS1** Digital signal level 1. Framing specification used in transmitting digital signals at 1.544 Mbps on a T1 facility (in the United States) or at 2.108 Mbps on an E1 facility (in Europe). Compare with DS0 and DS3.

**DS3** Digital signal level 3. Framing specification used for transmitting digital signals at 44.736 Mbps on a T3 facility. Compare with DS0 and DS1.

**DSCP** Differentiated service code point. Specifies a precedence value for handling packets belonging to the specified class.

**DSU** Data service unit. Device used in digital transmission that adapts the physical interface on a DTE device to a transmission facility such as T1 or E1. The DSU is also responsible for such functions as signal timing. Often used with CSU, as in CSU/DSU. See also CSU.

## E

**Edge Services Router** See ESR.

**ESF** Extended superframe. Framing type used on T1 circuits that consists of 24 frames of 192 bits each, with the 193rd bit providing timing and other functions. ESF is an enhanced version of SF. See also SF.

**ESR** Edge Services Router. A router that aggregates traffic from thousands of low- and medium-bandwidth subscriber connections and routes it on a few high-bandwidth connections to the Internet core.

**Ethernet** Baseband LAN specification. Ethernet networks use CSMA/CD and run over a variety of cable types at 10 Mbps, 100 Mbps, or 1000 Mbps. Ethernet is similar to the IEEE 802.3 series of standards. See also Fast Ethernet, Gigabit Ethernet.

**Extended Superframe Format** See ESF.

## F

**Facility data link** See FDL.

**Fast Ethernet** Any of a number of 100 Mbps Ethernet specifications. Fast Ethernet offers a speed increase 10 times that of the 10BaseT Ethernet specification, while preserving qualities such as frame format, MAC mechanisms, and MTU. Existing 10BaseT applications and network management tools can be used on Fast Ethernet networks. The Fast Ethernet specification is based on an extension to the IEEE 802.3 specification. Compare with Ethernet and Gigabit Ethernet.

**FDL** Facility data link. Embedded communications channel in ESF DS1 framing. Used to convey both bit-oriented and message-oriented signals.

**Flash memory** Nonvolatile storage that can be electrically erased and reprogrammed so that software images can be stored, booted, and rewritten as necessary. Flash memory was developed by Intel and is licensed to other semiconductor companies.

**Frame Relay** Industry-standard, switched data link layer protocol that handles multiple virtual circuits using HDLC encapsulation between connected devices. Frame Relay is more efficient than X.25, the protocol for which it is generally considered a replacement.

## G

**Gigabit Ethernet** Gigabit Ethernet. Ethernet running at a transmission speed of 1 billion bits per second.

## H

**HDLC** High Level Data Link Control. Bit-oriented synchronous data link layer protocol developed by ISO. Derived from SDLC, HDLC specifies a data encapsulation method on synchronous serial links using frame characters and checksums.

**High Level Data Link Control** See HDLC.

## I

**IOS** Internet Operating System. See Cisco IOS.

**IP** Internet Protocol. Network layer protocol in the TCP/IP stack offering a connectionless internetwork service. IP provides features for addressing, type-of-service specification, fragmentation and reassembly, and security. Defined in RFC 791.

## K

**Keepalive message** Message sent by one network device to inform another network device that the virtual circuit between the two is still active.

## L

**LAIS** Line Alarm Indication Signal. A SONET port status indicator that activates when an LAIS defect occurs and does not clear throughout the alarm integration period, which is typically 2.5 seconds. An LAIS defect occurs when bits 6, 7, and 8 of the K2 byte are 111 for three consecutive frames. This occurrence begins the alarm integration period. If this period elapses without the detection of three consecutive frames in which K2 bits 6, 7, and 8 show any pattern other than 111, the LAIS indicator activates. The LAIS indicator clears when an LAIS defect does not occur for a time interval equal to the alarm deactivation period (typically 10 seconds).

**Line card** Any I/O card that can be inserted in a modular chassis.

**LOF** Loss of Frame. A SONET port status indicator that activates when an LOF defect occurs and does not clear for an interval of time equal to the alarm integration period, which is typically 2.5 seconds. An LOF defect occurs when an out-of-frame (OOF) condition occurs and does not clear for more than 3 microseconds (ms). This occurrence begins the alarm integration period. (OOF occurs when four consecutive frames do not contain a valid frame word. OOF clears when two valid consecutive frames are detected.) The LOF indicator clears when an LOF defect is not detected for a time interval that is equal to the alarm deactivation period (typically 10 seconds).

**Loopback test** A test in which signals are sent and then directed back toward their source from some point along the communications path. Loopback tests are often used to check network interface usability.

**LOS** Loss of signal. A SONET port status indicator that activates when an LOS defect occurs and does not clear throughout the alarm integration period, which is typically 2.5 seconds. An LOS defect occurs when the OC-3 port receives all zeros for 20 microseconds (+.3 ms). This occurrence begins the alarm integration period. If this period elapses without the detection of two consecutive frames in which there are no 20-ms periods of signal loss, the LOS indicator activates. The LOS indicator clears when an LOS defect is not detected for an interval equal to the alarm deactivation period (typically 10 seconds).

## M

**M13** Generic term for equipment that multiplexes DS1s into DS3s. Sometimes used to describe a specific DS3 multiplex format. Some standards use this term to describe a synchronous multiplexing format also know as SYNTRAN. In many cases, however, M13 does not refer to the SYNTRAN format; it refers instead to the format also known as M23.

**M23** A method of multiplexing four DS1 signals into a DS2 signal, then multiplexing seven DS2 signals into a DS3 signal.

**MAC** Media Access Control. The lower of the two sublayers of the data link layer defined by the IEEE. The MAC sublayer handles access to shared media.

**MAC address** Standardized data link layer address that is required for each port or device that connects to a LAN. Other devices in the network use these addresses to locate specific ports in the network, and to create and update routing tables and data structures. MAC addresses are 6 bytes long and are controlled by the IEEE. Also known as a hardware address, MAC-layer address, or physical address.

**Maintenance data link** See MDL.

**Major alarm** One of a group of alarm conditions that are considered the second most severe of all reportable alarms. Major alarms affect several subscribers who connect to the reporting node. You can use the **show facility-alarm status** IOS command to obtain more information about the problem. See also Critical alarm and Minor alarm.

**Maximum transmission unit** See MTU.

**MDL** Maintenance data link. Embedded communications channel in C-bit parity DS3 framing. Used to convey message-oriented signals.

**Minor alarm** One of a group of alarm conditions that are considered the third most severe of all reportable alarms. Minor alarms affect a single or small number of subscribers who connect to the reporting node. You can use the **show facility-alarm status** IOS command to obtain more information about the problem. See also Critical alarm and Major alarm.

**MLP** Multilink Point-to-Point Protocol. A method of splitting, recombining, and sequencing datagrams across multiple logical data links.

**MTU** Maximum transmission unit. Maximum packet size, in bytes, that a particular interface can handle.

**Multilink Point-to-Point** See MLP.

## N

**NEBS** Network Equipment Building Systems. The Bellcore requirement for equipment deployed in a central office environment. Covers spatial and thermal requirements as well as requirements for hardware, crafts person interface, fire resistance, handling and transportation, earthquake and vibration, airborne contaminants, grounding, acoustical noise, illumination, EMC, and ESD.

**Network Equipment Building Systems** See NEBS.

**Nonvolatile RAM** See NVRAM.

**NVRAM** Nonvolatile RAM. RAM that retains its contents when a unit is powered off.

## P

**Packet** Logical grouping of information that includes a header containing control information and (usually) user data. Packets are most often used to refer to network layer units of data. The terms datagram, frame, message, and segment are also used to describe logical information groupings at various layers of the OSI reference model and in various technology circles.

**Packet over SONET** See POS.

**PCMCIA Flash disk card** A portable (credit-card size), nonvolatile storage device. PCMCIA Flash disk cards use Flash technology to store data. PCMCIA stands for Personal Computer Memory Card International Association, which sets the standard for this technology. Also called PC card.

**Performance Routing Engine** See PRE.

**Point-to-Point Protocol** See PPP.

**POS** Packet Over SONET. A high-speed means of transmitting data over a SONET fiber-optic transmission system through a direct fiber connection to a data switch or router. POS is a point-to-point dedicated leased-line approach intended purely for high-speed data applications. POS allows a user organization to pass data in its native format, without the addition of any significant level of overhead in the form of signaling and control information.

**PPP** Point-to-Point Protocol. Provides router-to-router and host-to-network connections over synchronous and asynchronous circuits.

**PRE** Performance routing engine. The central routing unit for the Cisco 10000 series ESR. The PRE performs all Layer 2 and Layer 3 packet manipulation related to routing and forwarding through the Cisco 10000 ESR. Dual PREs can be configured in a single chassis for redundancy.

## Q

**QoS** Quality of service. A measure of performance for a transmission system that reflects its transmission quality and service availability.

**QoS Policy Propagation on BGP** See QPPB.

**QPPB** QoS Policy Propagation on BGP. A feature involving the classification of packets by IP precedence based on BGP community lists, BGP autonomous system paths, and access lists. After a packet is classified, other quality of service features such as committed access rate (CAR) and weighted random early detection (WRED) can specify and enforce policies to fit a business model.

**Quality of Service** See QoS.

## R

**RAM** Random-access memory. Volatile memory that can be read and written by a microprocessor.

**Random Access Memory** See RAM.

**Redundancy** In internetworking, the duplication of devices, services, or connections so that, in the event of a failure, the redundant devices, services, or connections can perform the work of those that failed.

**RMON** Remote Monitoring. MIB agent specification described in RFC 1271 that defines functions for the remote monitoring of networked devices. The RMON specification provides numerous monitoring, problem detection, and reporting capabilities.

**ROM** Read only memory. Nonvolatile memory that can be read, but not written, by the microprocessor.

## S

**SDH** Synchronous Digital Hierarchy. European standard that defines a set of rate and format standards that are transmitted using optical signals over fiber. SDH is similar to SONET, with a basic rate of 155.52 Mbps, designated as STM-1. See also SONET and STM-1.

**SF** Super frame. Common framing type used on T1 circuits. SF consists of 12 frames of 192 bits each, with the 193rd bit providing error checking and other functions. SF has been superseded by ESF, but is still widely used. Also called D4 framing. See also ESF.

**Simple Network Management Protocol** See SNMP.

**SNMP** Simple Network Management Protocol. Network management protocol used almost exclusively in TCP/IP networks. SNMP provides a means to monitor and control network devices, and to manage configurations, statistics collection, performance, and security. See also SNMP2.

**SNMP2** SNMP Version 2. Version 2 of the network management protocol. SNMP2 supports centralized as well as distributed network management strategies, and includes improvements in the SMI, protocol operations, management architecture, and security. See also SNMP.

**SONET** Synchronous Optical Network. High-speed synchronous network specification developed by Bellcore and designed to run on optical fiber. STS-1 is the basic building block of SONET. It was approved as an international standard in 1988. See also SDH and STS-1.

**SPE** Synchronous Payload Envelope. The major portion of the SONET frame format used to carry the STS-1 signal; it is divided into an information payload section and a transport overhead system. SPE is used to address three payload structures: direct to STS-1 line rate muliplexing; asynchronous DS3 multiplexing; and synchronous DS3 multiplexing.

**STM-1** Synchronous Transport Module level 1. Basic building block signal of SDH, operating at 155.52 Mbps. Faster SDH rates are defined as STS-*n*, where *n* is a multiple of 155.52 Mbps. See also SDH.

**STS-1** Synchronous Transport Signal level 1. Basic building block signal of SONET, operating at 51.84 Mbps. Faster SONET rates are defined as STS-*n*, where *n* is a multiple of 51.84 Mbps. See also SONET.

**Subrate DS3** A generic term to describe a process in which the bandwidth of a clear channel DS3 is limited to a lower rate. Many proprietary formats exist.

**Super Frame** See SF.

**SDH** Synchronous Digital Hierarchy. European standard that defines a set of rate and format standards that are transmitted using optical signals over fiber. SDH is similar to SONET, with a basic rate of 155.52 Mbps, designated as STM-1. See also SONET and STM-1.

**SF** Super frame. Common framing type used on T1 circuits. SF consists of 12 frames of 192 bits each, with the 193rd bit providing error checking and other functions. SF has been superseded by ESF, but is still widely used. Also called D4 framing. See also ESF.

## T

**T1** Digital WAN carrier facility. T1 transmits DS1-formatted data at 1.544 Mbps through the telephone switching network.

**T3** Digital WAN carrier facility. T3 transmits DS3-formatted data at 44.736 Mbps through the telephone switching network.

**TAC** A Cisco Technical Assistance Center. There are four TACs worldwide.

**Telnet** Standard terminal emulation protocol in the TCP/IP protocol stack. Telnet is used for remote terminal connection, enabling users to log in to remote systems and use resources as if they were connected to a local system. Telnet is defined in RFC 854.

**TFTP** Trivial File Transfer Protocol. A simplified version of FTP that allows files to be transferred from one computer to another over a network.

## U

**Unchannelized DS3** See Clear channel DS3.

## V

**VT-*n*** Virtual tributary level n. The SONET format for mapping a lower-rate signal into a SONET payload. For example, VT-1.5 is used to transport a DS1 signal. See also DS1 and SONET.

## W

**Weighted Random Early Detection** See WRED.

**WRED** Weighted random early detection. RED uses an algorithm to randomly discard packets.The result of the drop is that the source detects the dropped traffic and slows its transmission. WRED combines the capabilities of the RED algorithm with IP precedence. This combination provides for preferential traffic handling for higher-priority packets. It can selectively discard lower-priority traffic when the interface starts to get congested and provide differentiated performance characteristics for different classes of service.

# INDEX

## Numerics

## A



## B

## C

## D

## H



## I



## K



## L



## M

## N



## O



## P



## Q

## R



## S



## T

## U



## V



## W



## Y