CHAPTER 18

Configuring the 1-Port Channelized OC-3/STM-1 SPA

This chapter provides information about configuring the 1-Port Channelized OC-3/STM-1 SPA on the Catalyst 6500 Series switch. It includes the following sections:

For information about managing your system images and configuration files, refer to the *Cisco IOS Configuration Fundamentals Configuration Guide, Release 12.2* and *Cisco IOS Configuration Fundamentals Command Reference, Release 12.2* publications.

For more information about the commands used in this chapter, see the *Catalyst 6500 Series Cisco IOS Command Reference, 12.2SX.* Also refer to the related Cisco IOS Release 12.2 software command reference and master index publications. For more information about accessing these publications, see the "Related Documentation" section on page xlv.

Configuration Tasks

This section describes how to configure the 1-Port Channelized OC-3/STM-1 SPA for the Catalyst 6500 Series switch and includes information about verifying the configuration. This document shows how to configure the 1-Port Channelized OC-3/STM-1 SPA in either Synchronous Optical Networking (SONET) or Synchronous Digital Hierarchy (SDH) framing modes.

It includes the following topics:

Required Configuration Tasks

This section lists the required configuration steps to configure the 1-Port Channelized OC-3/STM-1 SPA. Some of the required configuration commands implement default values that might be appropriate for your network. If the default value is correct for your network, then you do not need to configure the command.

Selecting the Physical Port and Controller

To configure or monitor the 1-Port Channelized OC-3/STM-1 SPA, you must specify the physical location of the SIP, SPA, and interface in the configuration commands. To select the physical port and controller, use the following command in configuration mode:

```
Router(config)# controller sonet slot/subslot/port
```

where:

- *slot*—Specifies the chassis slot number in the Catalyst 6500 Series switch where the SIP is installed.
- *subslot*—Specifies the secondary slot of the SIP where the SPA is installed.
- *port*—Specifies the number of the individual interface port on a SPA. Since there is only 1 port on the 1-Port Channelized OC-3/STM-1 SPA, the port number is always 0.

The following example shows how to specify the port of a 1-Port Channelized OC-3/STM-1 SPA installed in subslot 1 of a Cisco 7600 SIP-200 in slot 3:

```
Router(config)# controller sonet 3/1/0
```

For more information about identifying slots and subslots, see the "Identifying Slots and Subslots for SIPs, SSCs, and SPAs" section on page 4-2.

To configure the interface for the 1-Port Channelized OC-3/STM-1 SPA, perform this task:

	Command	Purpose
Step 1	`Router# configure terminal`	Enters configuration mode.
Step 2	`Router(config)# controller sonet slot/subslot/port`	Selects the controller to configure and enters controller configuration mode. • *slot/subslot/port*—Specifies the location of the interface. **Note** On the 1-Port Channelized OC-3/STM-1 SPA, the port number is always 0.
	To continue the configuration using SONET framing, perform the configuration described in the "Configuring for SONET Framing" section on page 18-3. To continue the configuration using SDH framing, perform the configuration described in the "Configuring for SDH Framing" section on page 18-5.	

Configuring for SONET Framing

To configure the 1-Port Channelized OC-3/STM-1 SPA for SONET framing, perform this task:

	Command	Purpose
Step 1	`Router(config-controller)# framing sonet`	Specifies SONET as the frame type. This is the default.
Step 2	`Router(config-controller)# clock source {internal | line}`	(Optional) Sets the clock source. • **internal**—Specifies that the internal clock source is used. • **line**—Specifies that the network clock source is used. This is the default. **Note** The clock source must be set to internal if the opposite end of the connection is set to line and the clock source must be set to line if the opposite end of the connection is set to internal.
Step 3	`Router(config-controller)# [no] loopback {local | network}`	(Optional) Enables or disables loopback mode on a SONET controller. • **local**—Loops data from the transmit path to the receive path. • **network**—Loops data received on the external port to the transmit path and back out the external port. By default, loopback is disabled.
Step 4	`Router(config-controller)# [no] ais-shut`	(Optional) By default, a Line Alarm Indication Signal (LAIS) is sent to the far end when a port is shut down. The **no** option disables sending LAIS when the port is administratively shut down.
Step 5	`Router(config-controller)# [no] idle pattern 0-255`	(Optional) Sets the data to be written to the idle (disabled, or unprovisioned) time-slots of a channelized path. The default idle pattern is 127 (hex 7F).
Step 6	`Router(config-controller)# [no] ber-threshold {b1-tca | b2-tca | b3-tca | sd-ber | sf-ber} exponent`	(Optional) Sets bit error rate (BER) thresholds. The BER is 10 to the negative *exponent*. These are the thresholds: • **b1-tca**—B1 BER threshold crossing alarm. • **b2-tca**—B2 BER threshold crossing alarm. • **b3-tca**—B3 BER threshold crossing alarm, applied to all channels. • **sd-ber**—Sets Signal Degrade BER threshold. • **sf-ber**—Sets Signal Fail BER threshold. The *exponent* range is 3 to 9 for all except Signal Degrade BER, which has a range of 5 to 9.
Step 7	`Router(config-controller)# sts-1 sts1#`	Selects the SONET STS-1 level to configure. For the 1-Port Channelized OC-3/STM-1 SPA, the range of *sts1#* is 1 to 3.

	Command	Purpose
Step 8	`Router(config-ctrlr-sts1)# [no] mode {vt-15 | ct3 | ct3-e1 | t3}`	Specifies the mode of operation of a STS-1 path: • **vt-15**—The STS-1 is divided into seven Virtual Tributary Groups (VTG). Each VTG is then divided into four VT1.5 channels, each carrying a DS1. • **ct3**—The STS-1 carries a DS3 signal divided into 28 DS1 channels (PDH). • **ct3-e1**—The STS-1 carries a DS3 signal divided into 21 E1 channels (PDH). • **t3**—The STS-1 carries an unchannelized (clear channel) DS3. SONET framing does not support E3 modes.
		If you select **mode vt-15**, perform the configuration described in the "Configuring DS1 (Channelized T3 mode)" section on page 18-8**.** If you select **mode ct3**, perform the configuration described in the "Configuring Channelized DS3" section on page 18-6**.** If you select **mode ct3-e1**, perform the configuration described in the "Configuring E1 (SONET Channelized T3-E1 mode)" section on page 18-11**.** If you select **mode t3**, perform the configuration described in the "Configuring an Unchannelized DS3 Serial Interface" section on page 18-13**.**
Repeat from Step 7 for each SONET STS-1 level.		

Configuring for SDH Framing

To configure the 1-Port Channelized OC-3/STM-1 SPA for SDH framing, perform this task:

	Command	Purpose
Step 1	`Router(config-controller)# framing sdh`	Specifies SDH as the frame type.
Step 2	`Router(config-controller)# aug mapping {au-3 | au-4}`	Configures administration units group (AUG) mapping. This command is available only when SDH framing is configured. • **au-3**—The following muxing/alignment/mapping will be used: VC-3 <--> AU-3 <--> AUG • **au-4**—The following muxing/alignment/mapping will be used: TUG-3 <--> VC-4 <--> AU-4 <--> AUG The default is **au-4.**
Step 3	`Router(config-controller)# au-3` *au-3#*	If you selected AUG mapping as **au-3**, you can further specify AU-3 muxing. The CLI command parser will enter into config-ctrlr-au3 parser mode, which will make only relevant commands visible. The *au-3#* range is from 1 to 3.
	`Router(config-controller)# au-4 1 tug-3` *tug-3#*	If you selected AUG mapping as **au-4**, you can further specify TUG-3 muxing. The CLI command parser will enter into config-ctrlr-tug3 parser mode, which will make only relevant commands visible. The *tug-3#* range is from 1 to 3.
Step 4	`Router(config-ctrlr-au3)# [no] mode c-11`	If you selected AUG mapping as **au-3**, you can specify c-11 mode, a container level-n channelized DS3, subdivided into 28 DS1 channels. In this mode, the AU-3 is divided into seven TUG-2 channels. Each TUG-2 is then divided into four TU-11 channels, each carrying a DS1 channel.
	`Router(config-ctrlr-tug3)# [no] mode {c-12 | t3 | e3}`	If you selected AUG mapping as **au-4**, you can specify the following modes: • **c-12**—A container level-n channelized DS3, subdivided into 21 E1 channels. The AU-4/TUG-3 is divided into seven TUG-2 channels. Each TUG-2 is then divided into three TU-12 channels, each carrying an E1 channel. • **t3**—The AU-4/TUG-3 carries an unchannelized (clear channel) DS3 channel. • **e3**—The AU-4/TUG-3 carries a unchannelized (clear channel) E3 channel.

Command	Purpose
	If you select **mode c-11**, perform the configuration described in the "Configuring DS1 (Channelized T3 mode)" section on page 18-8**.** If you select **mode c-12**, perform the configuration described in the "Configuring E1 (SDH Channelized T3/E3 mode)" section on page 18-12**.** If you select **mode t3**, perform the configuration described in the "Configuring an Unchannelized DS3 Serial Interface" section on page 18-13**.** If you select **mode e3**, perform the configuration described in the "Configuring an Unchannelized E3 Serial Interface" section on page 18-16**.**
Repeat from Step 3 for each AU-3 or TUG-3 group. **Note** If you configure an AU-3 or TUG-3 group as mode c-11 or t3, you cannot use modes c-12 or e3 for the other AU-3 or TUG-3 groups.	

Configuring Channels

Depending on the framing and channel modes you have selected, use the appropriate section for the configuration of the channels.

- Configuring Channelized DS3, page 18-6
- Configuring DS1 (Channelized T3 mode), page 18-8
- Configuring E1 (SONET Channelized T3-E1 mode), page 18-11
- Configuring E1 (SDH Channelized T3/E3 mode), page 18-12
- Configuring an Unchannelized DS3 Serial Interface, page 18-13
- Configuring an Unchannelized E3 Serial Interface, page 18-16

Configuring Channelized DS3

To configure channelized DS3, you must have selected SONET framing mode ct3.

Channelized DS3 can be configured in either config-ctrlr-sts1 or config-ctrlr-au3 parser mode. To configure channelized DS3 mode, perform this task:

	Command	Purpose
Step 1	`Router(config-ctrlr-`*xxx*[1]`)#` **[no] t3 framing** {**c-bit** | **m23** | **auto-detect**}	(Optional) Specifies the T3 framing mode. • **c-bit**—Use C-bit parity framing. • **m23**—Use M23 framing. • **auto-detect—**Detects the framing type at the device at the end of the line and switches to that framing type. If both devices are set to auto-detect, c-bit framing is used.
Step 2	`Router(config-ctrlr-`*xxx*`)#` **[no] t3 clock source** {**internal** | **line**}	(Optional) Sets the clock source. • **internal**—Specifies that the internal clock source is used. • **line**—Specifies that the network clock source is used. This is the default for T1 and E1. **Note** The clock source must be set to internal if the opposite end of the connection is set to line and the clock source must be set to line if the opposite end of the connection is set to internal.
Step 3	`Router(config-ctrlr-`*xxx*`)#` **[no] t3 loopback** {**local** | **network** [**line** | **payload**] | **remote** [**line** | **payload**]}	(Optional) Enables or disables loopback mode on a SONET controller. These are the supported loopback modes: • **local**—Loops data from the transmit path to the receive path. • **network**—Loops all data or only payload data received on the external port to the transmit path and back out the external port. • **remote**—Sends Far End Alarm and Control (FEAC) to set remote system in either **line** or **payload** loopback. Applicable only to C-bit framing. The default is no loopback.

	Command	Purpose
Step 4	`Router(config-ctrlr-xxx)# [no] t3 mdl string {eic | fic | generator | lic | pfi | port | unit} string`	(Optional) Maintenance Data Link (MDL) messages are used to communicate identification information between local and remote ports. Configures the contents of the MDL message. • **eic**—Specifies the Equipment Identification Code; can be up to 10 characters. • **fic**—Specifies the Frame Identification Code; can be up to 10 characters. • **generator**—Specifies the Generator number string sent in the MDL Test Signal message; can be up to 38 characters. • **lic**— Specifies the Location Identification Code; can be up to 11 characters. • **pfi**—Specifies the Path Facility Identification Code sent in the MDL Path message; can be up to 38 characters. • **port**—Specifies the port number string sent in the MDL Idle Signal message; can be up to 38 characters. • **unit**—Specifies the Unit Identification Code; can be up to 6 characters. The default is no MDL string.
Step 5	`Router(config-ctrlr-xxx)# [no] t3 mdl transmit {path | idle-signal | test-signal}`	(Optional) Configures the transmission of the MDL message. • **path**—Enables MDL Path message transmission. • **idle-signal**—Enables MDL Idle-Signal message transmission • **test-signal**—Enables MDL Test-Signal message transmission. The default is no MDL transmit.
Step 6	`Router(config-ctrlr-xxx)# [no] t3 equipment {customer | network} loopback`	(Optional) Determines response to remote loopback request. • **customer**—Enables the port to honor remote loopback requests. • **network**—Disables remote loopback requests. **Note** Remote loopbacks are only available in c-bit framing mode.

1. The actual command prompt is `Router(config-ctrlr-sts1)#` for SONET and `Router(config-ctrlr-au3)#` for SDH.

Configuring DS1 (Channelized T3 mode)

Two modes of operation support the DS1 channel configuration shown in this section.

With SONET framing mode VT-15, the STS-1 channel is divided into seven Virtual Tributary Groups (VTG). Each VTG contains four VT1.5 channels, each of which carries a DS1 channel. When configuring the DS1 channels, the following substitutions should be made in the commands shown:

- The command prompt is `Router(config-ctrlr-sts1)#`

- The command prefix is **vtg** *vtg-number*

  The *vtg-number* selects which VTG is being configured. The range is 1 to 7.

With SDH framing mode AU-3/C-11, the AU-3 channel is divided into seven TUG-2 channels. Each TUG-2 channel is then divided into four TU-11 channels, each carrying a DS1 channel. When configuring the DS1 channels, the following substitutions should be made in the commands shown:

- The command prompt is `Router(config-ctrlr-au3)`
- The command prefix is **tug-2** *tug-2-number*

  The *tug-2-number* selects which TUG-2 channel is being configured. The range is 1 to 7.

In configuring each DS1 channel, specify the DS1 channel number *t1#* (range 1 to 4). To configure a DS1 channel, perform this task, substituting the appropriate command prefix for *prefix* in the commands:

	Command	Purpose
Step 1	`Router(config-ctrlr-`*xxx*[1]`)#` [**no**] *prefix*[2] **t1** *t1#* **clock source** {**internal** | **line**}	(Optional) Sets the clock source. • **internal**—Specifies that the internal clock source is used. • **line**—Specifies that the network clock source is used. This is the default. **Note** The clock source must be set to internal if the opposite end of the connection is set to line and the clock source must be set to line if the opposite end of the connection is set to internal.
Step 2	`Router(config-ctrlr-`*xxx*`)#` [**no**] *prefix* **t1** *t1#* **framing** {**sf** | **esf**}	(Optional) Specifies the T1 framing type using the **framing** command. • **sf**—Specifies Super Frame as the T1 frame type. • **esf**—Specifies Extended Super Frame as the T1 frame type. This is the default. **Note** If you select sf framing, consider disabling yellow alarm detection because the yellow alarm can be incorrectly detected with sf framing.
Step 3	`Router(config-ctrlr-`*xxx*`)#` [**no**] *prefix* **t1** *t1#* **yellow** {**detection** | **generation**}	(Optional) Enables detection or generation of DS1 yellow alarms,

	Command	Purpose
Step 4	`Router(config-ctrlr-xxx)# prefix t1 t1# channel-group channel-number# timeslots range [speed {56 | 64}]`	Specifies the DS1 channel and timeslots to be mapped to each channel. • *channel-number*—Specifies a channel-group mapping (0–23) under the designated T1. • *range*—List of timeslots under the channel-group. Timeslots assigned to this T1 can be 1–24 or a combination of subranges within 1– 24. You can indicate a range using a hyphen, commas, or a combination of both. One timeslot equals one DS0. • **speed 56** or **64**— Specifies the speed of a timeslot as either 56 or 64 kbps. The default speed is 64 kbps.
	`Router(config-ctrlr-xxx)# no prefix t1 t1# channel-group channel-number#`	To alter the configuration of an existing channel group, the channel group must be removed first using the **no** form of the **channel-group** command.
Step 5	`Router(config-ctrlr-xxx)# [no] prefix t1 t1# fdl ansi`	(Optional) Enables the one-second transmission of remote performance reports via the Facility Data Link (FDL) per ANSI T1.403. This function requires that the T1 framing type is Extended Super Frame (ESF).
Step 6	`Router(config-ctrlr-xxx)# [no] prefix t1 t1# loopback {local | network line | remote {line fdl {ansi | bellcore} | payload fdl ansi}}`	(Optional) Specifies the loopback mode for testing. • **local**—Loops data from the transmit path to the receive path. • **network line**—Loops data received on the external port to the transmit path and back out the external port before going through the T1 framer. • **remote line fdl**—Sends a repeating 16-bit ESF data link code word to the remote end requesting that it enter into a network line loopback. – For ansi, the code word is (00001110 11111111). – For bellcore, the code word is (00010010 11111111). • **remote payload fdl ansi**—Sends a repeating 16-bit ESF data link code word (00010100 11111111) to the remote end requesting that it enter into a network payload loopback. **Note** Local network payload loopback is not supported due to TEMUX-84/TEMUX-84E limitations.
Step 7	`Router(config-ctrlr-xxx)# [no] prefix t1 t1# shutdown`	The **no** *args* **shutdown** command enables the interface. The *args* **shutdown** command disables the interface.
Step 8	Configure the serial interfaces. **Note** After a T1 channel is configured, it appears to the Cisco IOS software as a serial interface; therefore, all the configuration commands for a serial interface are available. However, not all commands are applicable to the T1 interface. For serial interface addressing information, refer to the "Serial Interface Naming" section on page 18-18.	

1. The actual command prompt is `Router(config-ctrlr-sts1)#` for SONET and `Router(config-ctrlr-au3)#` for SDH.
2. The actual command prefix is **vtg** *vtg-number* for SONET and **tug-2** *tug-2-number* for SDH.

Configuring E1 (SONET Channelized T3-E1 mode)

For E1 channel configuration in SONET channelized DS3 mode, you must have previously selected SONET framing mode ct3-e1. In this mode, the STS-1 is divided into 21 E1 channels.

In configuring each E1 channel, specif the E1 channel number *e1#* (range 1-21). To configure each E1, perform this task:

	Command	Purpose
Step 1	`Router(config-ctrlr-sts1)# e1 e1# channel-group channel-group# timeslots list-of-timeslots [speed {56 | 64}]`	Specifies the E1 channel and timeslots to be mapped to each channel. • *channel-number*—Specifies a channel-group mapping (0–30) under the designated E1. • *range*—List of timeslots under the channel-group. Timeslots assigned to this E1 can be 1–31 or a combination of subranges within 1–31. You can indicate a range using a hyphen, commas, or a combination of both. One timeslot equals one DS0. • **speed 56** or **64**— Specifies the speed of a timeslot as either 56 or 64 kbps. The default speed is 64 kbps.
	`Router(config-ctrlr-sts1)# no e1 e1# channel-group channel-group#`	To alter the configuration of an existing channel group, the channel group must be removed first using the **no** form of the **channel-group** command.
Step 2	`Router(config-ctrlr-sts1)# [no] e1 e1# unframed | framing {crc4 | no-crc4}`	(Optional) Sets the framing on the interface. • **unframed**—Unframed mode (G.703) uses all 32 time slots for data, none for framing signals. • **framing**—Uses a time slot for framing signals. – **crc4**—Specifies CRC4 as the E1 frame type. – **no-crc4**—Specifies no CRC4 as the E1 frame type. The default is framing with crc4.
Step 3	`Router(config-ctrlr-sts1)# [no] e1 e1# clock source {internal | line}`	(Optional) Sets the clock source. • **internal**—Specifies that the internal clock source is used. • **line**—Specifies that the network clock source is used. This is the default. **Note** The clock source must be set to internal if the opposite end of the connection is set to line and the clock source must be set to line if the opposite end of the connection is set to internal.
Step 4	`Router(config-ctrlr-sts1)# [no] e1 e1# national bits pattern`	(Optional) The national bit is reserved for national use and is set to 0 by default. Change this bit only when required for interoperability with your telephone company.

	Command	Purpose
Step 5	`Router(config-ctrlr-sts1)# [no] e1 e1# loopback [local | network]`	(Optional) Enables or disables loopback mode on a serial port. These are the supported loopback modes: • **local**—Loops data from the transmit path to the receive path. • **network**—Loops data received on the external port to the transmit path and back out the external port. The default is no loopback.
Step 6	`Router(config-ctrlr-sts1)# [no] e1 e1# shutdown`	The **no** *args* **shutdown** command enables the interface. The *args* **shutdown** command disables the interface.

Step 7 Configure the serial interfaces.

Note After an E1 channel is configured, it appears to the Cisco IOS software as a serial interface; therefore, all the configuration commands for a serial interface are available. However, not all commands are applicable to the E1 interface. For serial interface addressing information, refer to the "Serial Interface Naming" section on page 18-18.

Configuring E1 (SDH Channelized T3/E3 mode)

For E1 channel configuration in SDH channelized DS3 mode, you must have previously selected SDH framing mode with AU-4 mode C-12. In this mode, the AU-4/TUG-3 is divided into seven TUG-2 channels. Each TUG-2 channel is then divided into three TU-12 channels, each carrying an E1 channel.

In configuring each E1 channel, specify the TUG-2 channel number *tug-2#* (range 1–7) and the E1 channel number *e1#* (range 1–3). To configure each E1, perform this task:

	Command	Purpose
Step 1	`Router(config-ctrlr-tug3)# tug-2 tug-2# e1 e1# channel-group channel-group# timeslots list-of-timeslots [speed {56 | 64}]`	Specifies the E1 channel and timeslots to be mapped to each channel. • *channel-number*—Specifies a channel-group mapping (0–30) under the designated E1. • *range*—List of timeslots under the channel group. Timeslots assigned to this E1 can be 1–31 or a combination of subranges within 1–31. You can indicate a range using a hyphen, commas, or a combination of both. One timeslot equals one DS0. • **speed 56** or **64**— Specifies the speed of a timeslot as either 56 or 64 kbps. The default speed is 64 kbps.
	`Router(config-ctrlr-tug3)# no tug-2 tug-2# e1 e1# channel-group channel-group#`	To alter the configuration of an existing channel group, the channel group must be removed first using the **no** form of the **channel-group** command.

	Command	Purpose
Step 2	`Router(config-ctrlr-tug3)# [no] tug-2 tug-2# e1 e1# unframed | framing {crc4 | no-crc4}`	(Optional) Sets the framing on the interface. • **unframed**—Unframed mode (G.703) uses all 32 time slots for data, none for framing signals. • **framing**—Uses a time slot for framing signals. – **crc4**—Specifies CRC4 as the E1 frame type. – **no-crc4**—Specifies no CRC4 as the E1 frame type. The default is framing with crc4.
Step 3	`Router(config-ctrlr-tug3)# [no] tug-2 tug-2# e1 e1# clock source {internal | line}`	(Optional) Sets the clock source. • **internal**—Specifies that the internal clock source is used. • **line**—Specifies that the network clock source is used. This is the default. **Note** The clock source must be set to internal if the opposite end of the connection is set to line and the clock source must be set to line if the opposite end of the connection is set to internal.
Step 4	`Router(config-ctrlr-tug3)# [no] tug-2 tug-2# e1 e1# national bits pattern`	(Optional) The national bit is reserved for national use and is set to 0 by default. Change this bit only when required for interoperability with your telephone company.
Step 5	`Router(config-ctrlr-tug3)# [no] tug-2 tug-2# e1 e1# loopback [local | network]`	(Optional) Enables or disables loopback mode on a serial port. These are the supported loopback modes: • **local**—Loops data from the transmit path to the receive path. • **network**—Loops data received on the external port to the transmit path and back out the external port. The default is no loopback.
Step 6	`Router(config-ctrlr-tug3)# [no] tug-2 tug-2# e1 e1# shutdown`	The **no** *args* **shutdown** command enables the interface. The *args* **shutdown** command disables the interface.
Step 7	Configure the serial interfaces. **Note** After an E1 channel is configured, it appears to the Cisco IOS software as a serial interface; therefore, all the configuration commands for a serial interface are available. However, not all commands are applicable to the E1 interface. For serial interface addressing information, refer to the "Serial Interface Naming" section on page 18-18.	

Configuring an Unchannelized DS3 Serial Interface

Two modes of operation support the unchannelized DS3 configuration shown in this section:

- With SONET framing mode t3, an STS-1 channel carries a DS3 channel.
- With SDH framing mode AU-4 mode t3, an AU-4/TUG3 channel carries a DS3 channel.

To configure a DS3 serial interface, perform this task:

	Command	Purpose
Step 1	`Router(config-ctrlr-xxx[1])# end`	Exits SONET or SDH framing mode configuration.
Step 2	`Router(config)#` **`interface serial`** *`slot/subslot/port.au-4/tug-3`* `or` **`interface serial`** *`slot/subslot/port.sts1`*	Selects the T3 serial interface to configure. For serial interface addressing information, refer to the "Serial Interface Naming" section on page 18-18.
Step 3	`Router(config-if)# [no]` **`dsu mode`** `{` **`0`** `|` **`1`** `|` **`2`** `|` **`3`** `|` **`4`** `}`	Specifies the interoperability mode used by a DS3 controller. • **0**—Connects to another DS3 controller or to a Digital Link DSU (DL3100 in T3 mode). This is the default. • **1**—Connects to a Kentrox DataSMART DS3 IDSU. • **2**—Connects to a Larscom Access-T45 DS3 DSU. • **3**—Connects to an Adtran T3SU 300. • **4**—Connects to a Verilink HDM 2182. The default is **0** (Digital Link).

	Command	Purpose
Step 4	`Router(config-if)# [no] dsu bandwidth kbps`	(Optional) Specifies the allowable bandwidth in *kbps*. The default for DS3 mode is 44010 kbps. The bandwidth range and increment values are based on the specific DSU, as follows: • Digital Link DL3100 – range: 300 to 44210 kbps – increments: 300 kbps • Digital Link DL3100E – range: 358 to 34010 kbps – increments: 358 kbps • Kentrox DataSMART T3/E3 IDSU – range: 1000 to 34000 kbps (E3 mode) – range: 1500 to 44210 kbps (T3 mode) – increments: 500 kbps • Larscom Access-T45 DS3 – range: 3100 to 44210 kbps – increments: 3100 kbps • Adtran T3SU 300 – range: 80 to 44210 kbps – increments: 80 kbps • Verilink HDM 2182 – range: 1600 to 31600 kbps – increments: 1600 kbps
Step 5	`Router(config-if)# [no] scramble`	(Optional) Scrambling randomizes the pattern of ones and zeros carried in the physical layer frame in order to assist clock recovery on the receiving end. The default is no scramble.
Step 6	`Router(config-if)# [no] framing {c-bit | m13}`	(Optional) Specifies framing mode. • **c-bit**—Specifies C-bit parity framing. • **m13**—Specifies M13 framing. Unframed DS3 is not supported. Default is C-bit parity framing.
Step 7	`Router(config-if)# [no] dsu remote {fullrate | accept}`	(Optional) Specifies where the DSU bandwidth is set. • **fullrate**—Sets far end DSU to its full rate bandwidth. • **accept**—Accepts incoming remote requests to reset DSU bandwidth.
Step 8	`Router(config-if)# [no] crc {16 | 32}`	(Optional) Specifies CRC word size. Default is 16 bits (CRC-CITT).

	Command	Purpose
Step 9	`Router(config-if)# [no] loopback {local | network | dte | remote [line | payload]}`	(Optional) Specifies loopback. • **local**—Loops data from the transmit path to the receive path. • **network**—Loops data received on the external port to the transmit path and back out the external port. • **dte**—Loops back after the line interface unit (LIU) towards the terminal. • **remote**—Sends Far End Alarm and Control (FEAC) to set remote system in either **line** or **payload** loopback.
Step 10	`Router(config-if)# no shutdown`	The **no shutdown** command enables the interface. The **shutdown** command disables the interface.

1. The actual command prompt is `Router(config-ctrlr-sts1)#` for SONET and `Router(config-ctrlr-tug3)#` for SDH.

This example shows how to verify the controller configuration:

```
Router(config)# show controllers t1
T1 6/0/1 is up.
  Applique type is Channelized T1
  Cablelength is long gain36 0db
  No alarms detected.
blarm-trigger is not set
  Framing is ESF, Line Code is B8ZS, Clock Source is Line.
  Data in current interval (395 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, 0 Unavail Secs
  Total Data (last 24 hours)
     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, 0 Unavail Secs
```

Configuring an Unchannelized E3 Serial Interface

For unchannelized E3 operation, you must have selected SDH framing mode with AU-4 mode e3, which specifies an E3 channel carried over a T3 channel. The configuration must be done in serial interface configuration mode.

To configure an unchannelized E3 serial interface, perform this task:

	Command	Purpose
Step 1	`Router(config-ctrlr-tug3)# end`	Exits SDH framing mode configuration.
Step 2	`Router(config)# interface serial` *slot/subslot/port.au-4/tug-3*	Selects the E3 serial interface to configure. For serial interface addressing information, refer to the "Serial Interface Naming" section on page 18-18.

	Command	Purpose
Step 3	`Router(config-if)# dsu mode {cisco | digital-link | kentrox}`	Specifies the interoperability mode. • **cisco**—Specifies Cisco as the dsu mode. • **digital-link**—Specifies Digital Link as the dsu mode. • **kentrox**—Specifies Kentrox as the dsu mode. Default is **cisco**.
Step 4	`Router(config-if)# [no] dsu bandwidth kbps`	(Optional) Specifies the maximum allowed bandwidth in *kbps*. The available range for each DSU type is: • Cisco—Range is 300-34010 kbps. • Digital Link—Range is 300-34010 kbps. • Kentrox—Range is 1000-24500, 34010 kbps.
Step 5	`Router(config-if)# [no] scramble`	(Optional) Scrambling randomizes the pattern of ones and zeros carried in the physical layer frame in order to assist clock recovery on the receiving end. The default is no scramble.
Step 6	`Router(config-if)# [no] national bit {0 | 1}`	(Optional) The national bit is reserved for national use and is set to 0 by default. Change this bit only when required for interoperability with your telephone company.
Step 7	`Router(config-if)# [no] framing {g751 | g832}`	(Optional) Sets the framing on the interface. • **g751**—Specifies g751 framing. This is the default for E3. • **g832**—Specifies g832 framing.
Step 8	`Router(config-if)# [no] crc {16 | 32}`	(Optional) Specifies CRC word size. Default is 16 bits (CRC-CITT).
Step 9	`Router(config-if)# [no] loopback {network | local | remote}`	(Optional) Specifies loopback. • **local**—Loops data from the transmit path to the receive path. • **network**—Loops data received on the external port to the transmit path and back out the external port. • **remote**—Sends Far End Alarm and Control (FEAC) to set remote system in loopback.
Step 10	`Router(config-if)# no shutdown`	The **no shutdown** command enables the interface. The **shutdown** command disables the interface.

This example shows how to verify the controller configuration:

```
Router(config)# show controllers t1
T1 6/0/1 is up.
  Applique type is Channelized T1
  Cablelength is long gain36 0db
  No alarms detected.
blarm-trigger is not set
  Framing is ESF, Line Code is B8ZS, Clock Source is Line.
  Data in current interval (395 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, 0 Unavail Secs
  Total Data (last 24 hours)
     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, 0 Unavail Secs
```

Serial Interface Naming

After you have configured the framing type and mode, serial interface names are automatically generated, and you can configure the serial interfaces using the **interface serial** command. The interface naming format will be dependent on the framing type and mode. The name formats of the serial interfaces created are listed below.

SONET framing

- If the mode is vt-15 (VTG groups channelized to DS1):

  interface serial [*slot/subslot/port*].[*sts1/vtg/t1*]:[*channel-group*]

- If the mode is ct3 (DS3 channelized to DS1):

  interface serial [*slot/subslot/port*].[*sts1/ds1*]:[*channel-group*]

  If the mode is ct3-e1 (DS3 channelized to E1):

  interface serial [*slot/subslot/port*].[*sts1/e1*]:[*channel-group*]

  If the mode is t3 (unchannelized DS3):

  interface serial [*slot/subslot/port*]:[*sts1*]

SDH framing

If the administration units group (AUG) mapping is au-4, the au-4 value is always 1; if the AUG mapping is au-3, then the only supported mode is c-11 (carrying a T1).

- If the mode is t3 or e3 (unchannelized DS3 or E3):

  interface serial [*slot/subslot/port*].[*au-4/tug-3*]

- If the mode is ct-12 mode (DS3 container level-n channelized into E1):

  interface serial [*slot/subslot/port*].[*tug-3/tug-2/e1*]:[*channel-group*]

- If the mode is c-11 mode (DS3 container level-n channelized into DS1):

  interface serial [*slot/subslot/port*].[*au-3/tug-2/t1*]:[*channel-group*]

Optional Configurations

There are several standard, but optional, configurations that might be necessary to complete the configuration of your serial SPA.

- Configuring Encapsulation, page 18-19
- Configuring the CRC Size, page 18-19
- Configuring FDL, page 18-20
- Configuring Distributed Multilink Point-to-Point Protocol (Hardware-Based), page 18-21
- Configuring MLFR, page 18-23
- Invert Data on the T1/E1 Interface, page 18-26
- Configuring Multipoint Bridging, page 18-27
- Configuring Bridging Control Protocol Support, page 18-27
- Link Fragmentation and Interleaving (LFI) Guidelines, page 18-27

Configuring Encapsulation

When traffic crosses a WAN link, the connection needs a Layer 2 protocol to encapsulate traffic. To set the encapsulation method, perform this task:

	Command	Purpose
Step 1	`Router# configure terminal`	Enters global configuration mode.
Step 2	`Router(config)# interface serial` *interface-name*	Selects the serial interface to configure. For addressing information, refer to the "Serial Interface Naming" section on page 18-18.
Step 3	`Router(config-if)# encapsulation {hdlc | ppp | frame-relay}`	Sets the encapsulation method on the interface. • **hdlc**—High-level Data Link Control (HDLC) protocol for serial interface. This encapsulation method provides the synchronous framing and error detection functions of HDLC without windowing or retransmission. This is the default for synchronous serial interfaces. • **ppp**—PPP (for serial interface). • **frame-relay**—Frame Relay (for serial interface).

Configuring the CRC Size

The 1-Port Channelized OC-3/STM-1 SPA interface uses a 16-bit cyclic redundancy check (CRC) by default, but also supports a 32-bit CRC. CRC is an error-checking technique that uses a calculated numeric value to detect errors in transmitted data. The designators 16 and 32 indicate the length (in bits) of the frame check sequence (FCS). A CRC of 32 bits provides more powerful error detection, but adds overhead. Both the sender and receiver must use the same setting.

CRC-16, the most widely used CRC throughout the United States and Europe, is used extensively with WANs. CRC-32 is specified by IEEE 802 and as an option by some point-to-point transmission standards. It is often used on Switched Multimegabit Data Service (SMDS) networks and LANs.

To set the length of the cyclic redundancy check (CRC) on an interface, use these commands:

	Command	Purpose
Step 1	`Router# configure terminal`	Enters global configuration mode.
Step 2	`Router(config)# interface serial` *interface-name*	Selects the serial interface to configure. For addressing information, refer to the "Serial Interface Naming" section on page 18-18.
Step 3	`Router(config-if)# crc {16 | 32}`	Selects the CRC size in bits. • **16**—16-bit CRC. This is the default • **32**—32-bit CRC.

Configuring FDL

Facility Data Link (FDL) is a 4-kbps channel provided by the Extended Super Frame (ESF) T1 framing format. The FDL performs outside the payload capacity and allows you to check error statistics on terminating equipment without intrusion. To configure FDL, perform this task:

	Command	Purpose
Step 1	`Router# configure terminal`	Enters global configuration mode.
Step 2	`Router(config)# interface serial` *interface-name*	Selects the serial interface to configure. For addressing information, refer to the "Serial Interface Naming" section on page 18-18.
Step 3	`Router(config-if)# [no] t1` *t1#* `fdl {ansi | att}`	Enables the transmission of remote performance reports via the Facility Data Link (FDL). This function requires that the T1 framing type is Extended Super Frame (ESF). • **ansi**—Reports conform to the ANSI T1.403 protocol, and are sent at one-second intervals. • **att**—Reports conform to the AT&T TR54016 protocol (a subset of ANSI T1.403), and are sent only when a request has been received.

Verifying FDL

This example shows how to verify the **fdl** setting:

```
router# show controllers t1

T1 6/0/1 is up.
  Applique type is Channelized T1
  Cablelength is long gain36 0db
  No alarms detected.
  alarm-trigger is not set
  Framing is ESF, FDL is ansi, Line Code is B8ZS, Clock Source is Line.
  Data in current interval (742 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, 0 Unavail Secs
  Total Data (last 73 15 minute intervals):
```

```
     1278491 Line Code Violations, 3 Path Code Violations,
     0 Slip Secs, 1 Fr Loss Secs, 177 Line Err Secs, 0 Degraded Mins,
     3 Errored Secs, 0 Bursty Err Secs, 1 Severely Err Secs, 227 Unavail Secs
.
.
.
```

Configuring Distributed Multilink Point-to-Point Protocol (Hardware-Based)

Distributed Multilink Point-to-Point Protocol (dMLPPP) allows you to combine interfaces which correspond to an entire T1 or E1 multilink bundle. You choose the number of bundles and the number of T1 or E1 lines in each bundle.

MLPPP Configuration Restrictions and Guidelines

The following restrictions and guidelines apply to hardware-based MLPPP:

- The MLPPP command is only available on serial interfaces.
- You must enable PPP encapsulation before configuring the MLPPP commands.
- The following are link restrictions for using MLPPP:
  - Only T1 or E1 links may be in a bundle.
  - All links must be on the same SPA.
  - A maximum of 12 links may be in a bundle.
- Only three fragmentation sizes are supported: 128, 256, and 512 bytes.
- Fragmentation is enabled by default, with a default size of 512 bytes.
- Fragmentation size is configured using the **ppp multilink fragment-delay** command after using the **interface multilink** command. Among the three possible fragmentationsizes, the least size satisfying the delay criteria is configured. For example, a 192 byte packet causes a delay of 1 millisecond on a T1 link, so the nearest fragmentation size is 128 bytes.

  Use the **show ppp multilink** command to indicate the MLPPP type and the fragmentation size:

```
Router# show ppp multilink
Multilink1, bundle name is Patriot2
Bundle up for 00:00:13
Bundle is Distributed
0 lost fragments, 0 reordered, 0 unassigned
0 discarded, 0 lost received, 206/255 load
0x0 received sequence, 0x0 sent sequence
Member links: 2 active, 0 inactive (max not set, min not set)
Se4/2/0/1:0, since 00:00:13, no frags rcvd
Se4/2/0/2:0, since 00:00:10, no frags rcvd
Distributed fragmentation on. Fragment size 512.  Multilink in Hardware.
```

- Fragmentation is disabled explicitly by using the **no ppp multilink fragmentation** command after using the **interface multilink** command.

Create a Multilink Bundle

To create a multilink bundle, perform this task:

	Command	Purpose
Step 1	`Router# configure terminal`	Enters configuration mode.
Step 2	`Router(config)# [no] interface multilink` *group-number*	Creates or configures a multilink interface and enters multilink interface mode. • *group-number*—The group number for the multilink bundle.
Step 3	`Router(config-if)# [no] multilink fragment [fragment-size` *size*`]`	(Optional) Enables fragmentation and sets the fragmentation size. • *size*—Allowed sizes are 128, 256 or 512 bytes. Fragmentation is disabled by default. When fragmentation is enabled, the default size is 128 bytes.
Step 4	`Router(config-if)# [no] multilink bundle-name {authenticated | endpoint | both}`	(Optional) Specifies the criteria for naming the multilink bundle, as defined in RFC 1990. • **authenticated**—Use the peer authenticated name as the bundle name. This is the default. • **endpoint**—Use the peer endpoint discriminator as the bundle name. • **both**—Use the peer authenticated name and endpoint discriminator as the bundle name.
Step 5	`Router(config-if)# [no] multilink min-links` *number* `[mandatory]`	(Optional) Specifies the preferred minimum number of links in a multilink bundle. • *number*—Sets the minimum number of links, from 0 to 255. • **mandatory**—If the number of links in the bundle falls below the number specified, the bundle is disabled. By default, the bundle goes down only when there are no links in the bundle.
	Configure each serial interface you want to assign to the multilink bundle.	

Assign an Interface to a Multilink Bundle

To assign an interface to a multilink bundle, perform this task:

	Command	Purpose
Step 1	`Router# configure terminal`	Enters configuration mode.
Step 2	`Router(config)# interface serial` *interface-name*	Selects the interface to configure and enters interface configuration mode. For addressing information, refer to the "Serial Interface Naming" section on page 18-18.
Step 3	`Router(config-if)# [no] encapsulation ppp`	Enables PPP encapsulation.
Step 4	`Router(config-if)# [no] multilink-group` *group-number*	Assigns the interface to a multilink bundle. • *group-number*—The multilink group number for the T1 or E1 bundle.
Step 5	`Router(config-if)# ppp chap hostname` *name*	Specifies a pool of dialup routers that all appear to be the same host when authenticating with CHAP. **Note** This command is mandatory when there is more than one bundle across two routers; otherwise, it is optional.
Step 6	`Router(config-if)# [no] ppp multilink`	Enables the negotiation of multilink on an interface. This command is automatically implied for interfaces that are configured as part of a multilink bundle.
	Repeat these commands for each interface you want to assign to the multilink bundle.	

Verifying MLPPP

Use the **show ppp multilink** command to display a list of active and inactive bundles, the configured member links, and packet statistics information.

Use the **show interface multilink** *group-number* command to display multilink interface information and the LCP and multilink status.

Configuring MLFR

Multilink Frame Relay (MLFR) allows you to combine T1/E1 lines into a bundle that has the combined bandwidth of multiple T1/E1 lines. You choose the number of bundles and the number of T1/E1 lines in each bundle. This allows you to increase the bandwidth of your network links beyond that of a single T1/E1 line.

MLFR Configuration Guidelines

MLFR will function in hardware if all of the following conditions are met:

- Only T1 or E1 links may be in a bundle.
- All links must be on the same SPA.
- A maximum of 12 links may be in a bundle.

Create a Multilink Frame Relay Bundle

To create a multilink Frame Relay bundle, perform this task:

	Command	Purpose
Step 1	`Router# configure terminal`	Enters configuration mode.
Step 2	`Router(config)# [no] interface mfr number`	Creates or configures a multilink Frame Relay bundle interface. • *number*—The number for the Frame Relay bundle. The range is 0 to 2147483647.
Step 3	`Router(config-if)# [no] frame-relay multilink bid name`	(Optional) Assigns a bundle identification name to a multilink Frame Relay bundle. • *name*—The name for the Frame Relay bundle. **Note** The bundle identification (BID) will not go into effect until the interface has gone from the down state to the up state. One way to bring the interface down and back up again is by using the **shutdown** and **no shutdown** commands in interface configuration mode.
Step 4	`Router(config-if)# [no] frame-relay multilink bandwidth-class class [threshold]`	(Optional) Specifies FRF.16 *class* A, B, or C, configuring the trigger point for activating or deactivating a bundle. • **a**—Class A will bring up the bundle if at least one bundle link is active. • **b**—Class B will bring up the bundle only if all links are active, and will bring down the bundle if any link becomes inactive. • **c**—Class C wil bring the bundle up or down depending on the *threshold* number of links being active.

Assign an Interface to a Multilink Frame Relay Bundle

To assign an interface to a multilink Frame Relay bundle, perform this task:

	Command	Purpose
Step 1	`Router# configure terminal`	Enters configuration mode.
Step 2	`Router(config)# interface serial interface-name`	Selects the interface to configure and enters interface configuration mode. For addressing information, refer to the "Serial Interface Naming" section on page 18-18.

	Command	Purpose
Step 3	`Router(config-if)# [no] encapsulation frame-relay mfr number [name]`	Creates a multilink Frame Relay bundle link and associates the link with a bundle. • *number*—The number for the Frame Relay bundle. • *name*—The name for the Frame Relay bundle.
Step 4	`Router(config-if)# [no] frame-relay multilink lid name`	(Optional) Assigns a bundle link identification name with a multilink Frame Relay bundle link. • *name*—The name for the Frame Relay bundle. **Note** The bundle link identification (LID) will not go into effect until the interface has gone from the down state to the up state. One way to bring the interface down and back up again is by using the **shutdown** and **no shutdown** commands in interface configuration mode.
Step 5	`Router(config-if)# [no] frame-relay fragment size end-to-end`	(Optional) Enables and configures Frame Relay end-to-end fragmentation (FRF.12) on the serial interface. The range of fragment *size* is 16 to 1600.
Step 6	`Router(config-if)# frame-relay multilink hello seconds`	(Optional) Configures the interval at which a bundle link will send out hello messages. The default value is 10 seconds. • *seconds*—Number of seconds between hello messages sent out over the multilink bundle.
Step 7	`Router(config-if)# frame-relay multilink ack seconds`	(Optional) Configures the number of seconds that a bundle link will wait for a hello message acknowledgment before resending the hello message. The default value is 4 seconds. • *seconds*—Number of seconds a bundle link will wait for a hello message acknowledgment before resending the hello message.
Step 8	`Router(config-if)# frame-relay multilink retry number`	(Optional) Configures the maximum number of times a bundle link will resend a hello message while waiting for an acknowledgment. The default value is 2 tries. • *number*—Maximum number of times a bundle link will resend a hello message while waiting for an acknowledgment.

Verifying Multilink Frame Relay

To verify the Frame Relay multilinks, use the **show frame-relay multilink detailed** command:

```
router# show frame-relay multilink detailed

Bundle: MFR49, State = down, class = A, fragmentation disabled
```

```
BID = MFR49
No. of bundle links = 1, Peer's bundle-id =
Bundle links:

 Serial6/0/0:0, HW state = up, link state = Add_sent, LID = test
   Cause code = none, Ack timer = 4, Hello timer = 10,
   Max retry count = 2, Current count = 0,
   Peer LID = , RTT = 0 ms
   Statistics:
   Add_link sent = 21, Add_link rcv'd = 0,
   Add_link ack sent = 0, Add_link ack rcv'd = 0,
   Add_link rej sent = 0, Add_link rej rcv'd = 0,
   Remove_link sent = 0, Remove_link rcv'd = 0,
   Remove_link_ack sent = 0, Remove_link_ack rcv'd = 0,
   Hello sent = 0, Hello rcv'd = 0,
   Hello_ack sent = 0, Hello_ack rcv'd = 0,
   outgoing pak dropped = 0, incoming pak dropped = 0
```

Invert Data on the T1/E1 Interface

If the interface on the 1-Port Channelized OC-3/STM-1 SPA is used to drive a dedicated T1 line that does not have B8ZS encoding, you must invert the data stream on the connecting CSU/DSU or on the interface. Be careful not to invert data on both the CSU/DSU and the interface, as two data inversions will cancel each other out. To invert data on a T1/E1 interface, perform this task:

	Command	Purpose
Step 1	`Router# configure terminal`	Enters configuration mode.
Step 2	`Router(config)# interface serial` *interface-name*	Selects the serial interface. For addressing information, refer to the "Serial Interface Naming" section on page 18-18.
Step 3	`Router(config-if)# invert data`	Inverts the data stream.

This example shows how to verify that invert data has been set:

```
router# show running configuration
.
.
.
interface Serial6/0/0:0
 no ip address
 encapsulation ppp
 logging event link-status
 load-interval 30
 invert data
 no cdp enable
 ppp chap hostname group1
 ppp multilink
 multilink-group 1
!
.
.
.
```

Configuring Multipoint Bridging

Multipoint bridging (MPB) enables the connection of multiple ATM PVCs, Frame Relay PVCs, BCP ports, and WAN Gigabit Ethernet subinterfaces into a single broadcast domain (virtual LAN), together with the LAN ports on that VLAN. This enables service providers to add support for Ethernet-based Layer 2 services to the proven technology of their existing ATM and Frame Relay legacy networks. Customers can then use their current VLAN-based networks over the ATM or Frame Relay cloud. This also allows service providers to gradually update their core networks to the latest Gigabit Ethernet optical technologies, while still supporting their existing customer base.

For MPB configuration guidelines and restrictions and feature compatibility tables, see the "Configuring Multipoint Bridging" section on page 4-17 of Chapter 4, "Configuring the SIPs and SSC."

Configuring Bridging Control Protocol Support

The Bridging Control Protocol (BCP) enables forwarding of Ethernet frames over SONET networks and provides a high-speed extension of enterprise LAN backbone traffic through a metropolitan area. The implementation of BCP on the SPAs includes support for IEEE 802.1D, IEEE 802.1Q Virtual LAN (VLAN), and high-speed switched LANs.

For BCP configuration guidelines and restrictions and feature compatibility tables, see the "Configuring PPP Bridging Control Protocol Support" section on page 4-18 of Chapter 4, "Configuring the SIPs and SSC."

Link Fragmentation and Interleaving (LFI) Guidelines

LFI can function by using either FRF.12 or MLPPP. MLPPP LFI operates in both hardware and software while FRF.12 LFI operates only in hardware.

Hardware MLPPP LFI Guidelines

LFI using MLPPP will function only in hardware if there is just one member link in the MLPPP bundle. The link can be a fractional T1 or full T1. Note the following guidelines:

- The **ppp multilink interleave** command must be configured to enable interleaving.
- Only three fragmentation sizes are supported: 128 bytes, 256 bytes, and 512 bytes.
- Fragmentation is enabled by default, and the default size is 512 bytes.
- A policy map having a priority class must be applied to the main interface.
- When hardware-based LFI is enabled, fragmentation counters are not displayed.

FRF.12 LFI Guidelines

LFI using FRF.12 is always performed in hardware. Note the following guidelines:

- The fragmentation is configured at the main interface.
- Only three fragmentation sizes are supported: 128 bytes, 256 bytes, and 512 bytes.
- A policy map having a priority class must be applied to the main interface.
- FRF.12 can be enabled only on plain FR links.

Configuring QoS Features on Serial SPAs

For information about the QoS features supported by the serial SPAs, see the "Configuring QoS Features on a SIP" section on page 4-33 of Chapter 4, "Configuring the SIPs and SSC."

Configuring CRTP

For information about configuring cRTP, see the "Configuring Compressed Real-Time Protocol" section on page 4-4 or *Configuring Distributed Compressed Real-Time Protocol* at the following URL:

http://www.cisco.com/en/US/docs/ios/12_2/qos/configuration/guide/qcfdcrtp.html

Configuring Automatic Protection Switching (APS)

The automatic protection switching (APS) feature allows switchover of interfaces in the event of an interface failure. The protection mechanism has a 1+1 architecture in which a Protect interface is paired with each Working interface. The Working and Protect circuits are synchronized over an independent out of band (OoB) communication channel.

For detailed information about APS, see the "Configuring Automatic Protection Switching" section on page 7-50 of Chapter 7, "Configuring the ATM SPAs." For complete information on APS, including information on additional APS features, refer to the *Cisco IOS Interface Configuration Guide, Release 12.2*.

Automatic Protection Switching Configuration Guidelines

When configuring APS, consider the following guidelines:

- The Working and Protect interfaces must be compatible (for example, both OC-3c interfaces). The interfaces can be on the same SPA, different SPAs in the same switch, or different SPAs in a different switch.
- If using interfaces on different switches, the two switches must have a network connection other than the SONET connection (such as through an Ethernet LAN). Because the APS Protect Group Protocol (PGP) is UDP traffic, this network connection should be reliable with a minimum number of hops.
- The IP addresses on the Working and Protect interfaces should be in the same subnet.
- APS is not supported on SVCs.

Tip Always configure the Working interface before the Protect interface. This will prevent the Protect interface from becoming active and disabling the circuits on the Working interface.

Automatic Protection Switching Configuration Task

To configure the Working and Protect interfaces on the ATM SPAs for basic APS operation, perform the following procedure beginning in global configuration mode.

To configure an interface for APS, perform this task:

	Command	Purpose
Step 1	`Router# configure terminal`	Enters configuration mode.
Step 2	`Router(config)# interface serial` *interface-name*	Selects the Working interface to configure. For serial interface addressing information, refer to the "Serial Interface Naming" section on page 18-18.
Step 3	`Router(config-if)# ip address` *ip-address mask* [**secondary**]	Specifies the IP address and subnet mask for the Working interface. Repeat this command with the **secondary** keyword to specify additional IP addresses to be used for the interface.
Step 4	`Router(config-if)# aps group` *group-number*	Enables the use of the APS Protect Group Protocol for this Working interface. • *group-number*—Unique number identifying this pair of Working and Protect interfaces. **Note** The **aps group** command is optional if this is the only pair of Working and Protect interfaces on the switch, but is required when you configure more than one pair of Working and Protect interfaces on the same switch.
Step 5	`Router(config-if)# aps working` *circuit-number*	Identifies the interface as the Working interface. • *circuit-number*—Identification number for this particular channel in the APS pair. Because only 1+1 redundancy is supported, the only valid values are 0 or 1, and the Working interface defaults to 1.
Step 6	`Router(config-if)# aps authentication` *security-string*	(Optional) Specifies a security string that must be included in every OOB message sent between the Working and Protect interfaces. • *security-string*—Arbitrary string to be used as a password between the Working and Protect interfaces. This string must match the one configured on the Protect interface.
Step 7	`Router(config)# interface serial` *interface-name*	Selects the Protect interface to configure. For serial interface addressing information, refer to the "Serial Interface Naming" section on page 18-18.

	Command	Purpose
Step 8	`Router(config-if)# ip address ip-address mask [secondary]`	Specifies the IP address and subnet mask for the Protect interface. **Note** This should be the same address that was configured on the Working interface in Step 3. Repeat this command with the **secondary** keyword to specify additional IP addresses to be used for the interface. These should match the secondary IP addresses that are configured on the Working interface.
Step 9	`Router(config-if)# aps group group-number`	Enables the use of the APS Protect Group Protocol for this Protect interface. • *group-number*—Unique number identifying this pair of Working and Protect interfaces. **Note** The **aps group** command is optional if this is the only pair of Working and Protect interfaces on the switch, but is required when you configure more than one pair of Working and Protect interfaces on the same switch.
Step 10	`Router(config-if)# aps protect circuit-number ip-address`	Identifies this interface as the Protect interface: • *circuit-number*—Identification number for this particular channel in the APS pair. Because only 1+1 redundancy is supported, the only valid values are 0 or 1, and the Protect interface defaults to 0. • *ip-address*—The Protect interface uses this IP address to communicate with the Working interface. **Note** This IP address should be the address of the Working interface if the Protect and Working interfaces are on the same switch. If the Working and Protect interfaces are on different switches, this should be the IP address of the Ethernet interface that provides interconnectivity between the two switches.
Step 11	`Router(config-if)# aps authentication security-string`	(Optional) Specifies a security string that must be included in every OOB message sent between the Working and Protect interfaces. • *security-string*—Arbitrary string to be used as a password between the Working and Protect interfaces. This string must match the one configured on the Working interface.

	Command	Purpose
Step 12	`Router(config-if)# aps revert minutes`	(Optional) Enables the Protect interface to automatically switch back to the Working interface after the Working interface has been up for a specified number of minutes. • *minutes*—Number of minutes until the interface is switched back to the Working interface after the Working interface comes back up. **Note** If this command is not used, you must manually switch back to the Working interface using either the **aps force** *circuit-number* or the **aps manual** *circuit-number* command.
Step 13	`Router(config-if)# [no] aps unidirectional`	(Optional) Configures the Protect interface for unidirectional mode. The **no** option configures for bidirectional, the default.
Step 14	`Router(config-if)# [no] aps signalling {sonet | sdh}`	(Optional) Configures the signaling method. The default is SONET.
Step 15	`Router(config-if)# [no] aps timers hello-time hold-time`	(Optional) Sets the time in seconds between hello packets (*hello-time*) and the waiting time (*hold-time*) before the Protect interface process declares a working interface to be down. The default hello-time is 1 second. The default hold-time is 3 seconds.

Operational Commands for Automatic Protection Switching

To force or prevent APS during operation, perform this task:

	Command	Purpose
Step 1	`Router# configure terminal`	Enters configuration mode.
Step 2	`Router(config)# interface serial interface-name`	Selects the Working interface to configure. For serial interface addressing information, refer to the "Serial Interface Naming" section on page 18-18.
Step 3	`Router(config-if)# [no] aps force circuit-number`	(Optional) Manually switches the specified circuit to a Protect interface, unless a request of equal or higher priority is in effect. • *circuit-number*—Identification number for this particular channel in the APS pair. Because only 1+1 redundancy is supported, the only valid values are 0 or 1, and the Working interface defaults to 1.

	Command	Purpose
Step 4	`Router(config-if)# [no] aps manual circuit-number`	(Optional) Manually switches the circuit to a Protect interface, unless a request of equal or higher priority is in effect.
Step 5	`Router(config-if)# [no] aps lockout circuit-number`	(Optional) Prevents a working interface from switching to a protect interface.

Configuring SONET and SDH Overhead Bytes

To modify overhead bytes in SONET or SDH, perform one of these task:

Command	Purpose
`Router(config-xxx[1])# [no] overhead c2 0-255`	Changes the C2 byte in SONET or SDH/AU-3 modes. For modes vt-15, c-11, and c-12, the default is 2. For mode ct3, the default is 4.
`Router(config-controller)# [no] overhead s1s0 0-3`	Configures automatically the S1S0 bits of H1 according to the framing unless specified here.
`Router(config-controller)# [no] overhead s1s0 ignore`	Causes the s1s0 overhead bits to be ignored. This may be necessary for Australian conformance, due to the limitation of Spectra-4x155 and possibly Spectra-622.
`Router(config-xxx)# [no] overhead j0 0-255`	Specifies the J0 trace byte to be transmitted in the regenerator section overhead.
`Router(config-xxx)# [no] overhead j0 expect 0-255`	Specifies the expected value of the received J0 trace byte.
`Router(config-xxx)# [no] overhead rs-tim ignore`	Supresses the J0 trace byte mismatch alarm.
`Router(config-xxx)# [no] overhead j1 length [16 | 64]`	Sets the J1 trace bytes in the high order (STS) path overhead to either 16 or 64 bytes.
`Router(config-xxx)# [no] overhead j1 message ascii_line`	Specifies the message to transmit in the the J1 trace bytes.
`Router(config-xxx)# [no] overhead j1 expect message ascii_line`	Specifies the expected value of the received message in the J1 trace bytes.
`Router(config-xxx)# [no] overhead hp-tim ignore`	Specifies to ignore the J1 trace identifier mismatch alarm.
`Router(config-xxx)# [no] overhead j2 message ascii_line`	Specifies the message to transmit in the the J2 trace byte byte in the low order (VT) path overhead.
`Router(config-xxx)# [no] overhead j2 expect message ascii_line`	Specifies the expected value of the received message in the J2 trace byte.

Command	Purpose
`Router(config-xxx)# [no] overhead lp-tim ignore`	Specifies to ignore the J2 trace identifier mismatch alarm.
`Router(config-controller)# [no] overhead s1byte ignore`	Specifies to ignore the received synchronization byte S1 value of 0xF (do not switch to internal clock).

1. The actual command prompt is `Router(config-ctrlr-sts1)#` for SONET, `Router(config-ctrlr-au3)#` for SDH/AU-3, or `Router(config-controller)# for SDH/AU-4`.

Configuring a Bit Error Rate Test (BERT)

To start a BERT pattern on a port, perform this task in interface configuration mode:

Command	Purpose
`Router(config-xxx`[1]`)# [no] [prefix`[2]`] [t1 [t1#] | [e1 [e1#] | t3] bert pattern {0s | 1s | 2^15 | 2^20 | 2^23 | alt-0-1 | qrss} interval minutes`	Starts or stops a specific bit pattern on a DS1/E1/T3 line. For DS1 and E1, you can optionally specify a specific channel number as the **t1** or **e1** number. The interval specifies the length of the test in minutes. The range is 1 to 14400. The **no** option stops the test. **Note** Only six E1 BERTs can be performed concurrently due to TEMUX-84/TEMUX-84E limitations.

1. The actual command prompt depends on the interface selected.
2. The actual command prefix is **vtg** *vtg-number* for SONET and **tug-2** *tug-2-number* for SDH.

Following are the available BERT pattern options:

0s	Repeating pattern of zeros (...000...).
1s	Repeating pattern of ones (...111...).
2^15	Pseudorandom 0.151 test pattern that is 32,768 bits in length.
2^20	Pseudo-andom 0.153 test pattern that is 1,048,575 bits in length.
2^23	Pseudorandom 0.151 test pattern that is 8,388,607 bits in length.
alt-0-1	Repeating pattern of alternating zeros and ones (...01010...).
qrss	Pseudorandom quasi-random signal sequence (QRSS) 0.151 test pattern that is 1,048,575 bits in length.

Saving the Configuration

To save your running configuration to nonvolatile random-access memory (NVRAM), perform this task in privileged EXEC configuration mode:

Command	Purpose
`Router# copy running-config startup-config`	Writes the new configuration to NVRAM.

For more information about managing configuration files, refer to the *Cisco IOS Configuration Fundamentals Configuration Guide, Release 12.2* and *Cisco IOS Configuration Fundamentals Command Reference, Release 12.2* publications.

Verifying the Interface Configuration

Besides using the **show running-configuration** command to display your Catalyst 6500 Series switch configuration settings, you can use the **show interface serial** and the **show controllers serial** commands to get detailed information on a per-port basis for your 1-Port Channelized OC-3/STM-1 SPA.

To view the overall or per-port information, perform this task:

Command	Purpose
`Router(config)# show interface serial [interface-name]`	Displays the configuration settings of all serial interfaces or of a specific serial interface. For serial interface addressing information, refer to the "Serial Interface Naming" section on page 18-18.
`Router(config)# show controllers sonet [interface-name][brief | tabular | remote performance [brief | tabular]]`	Displays the configuration settings of all DS1 or E1 interfaces or of a specific interface. The *interface-name* is the DS1 or E1 serial interface address as shown in the "Serial Interface Naming" section on page 18-18, with the channel-group omitted. The remote performance option is only available for DS1 interfaces.
`Router(config)# show controllers t3 interface-name [brief | tabular | remote performance [brief | tabular]]`	Displays the configuration settings of all DS1 interfaces on a DS3 or of a specific DS1 interface. The *interface-name* can be *slot/subslot/port or slot/subslot/port/ds1*.

Verifying Interface Configuration and Status

To find detailed interface information for the 1-Port Channelized OC-3/STM-1 SPA, use the **show interface serial** command.

The following example provides sample output for a SPA located in the first subslot of the Cisco 7600 SIP-200 installed in slot 2 of a Catalyst 6500 Series switch:

```
Router(config)# show interface serial
Serial2/0/0.1/2 unassigned YES TFTP administratively down down
Serial2/1/0.1/1/1:0 unassigned YES unset down down
Serial2/1/0.1/2/4:0 unassigned YES unset down down
Serial2/1/0.1/2/4:1 unassigned YES unset down down
Serial2/1/0.2/1:0 unassigned YES unset down down
Serial2/1/0.2/2:0 unassigned YES unset down down
Serial2/1/0.2/3:0 unassigned YES unset down down
Serial2/1/0.3 unassigned YES unset down down
UUT#sh int Serial2/1/0.1/1/1:0
```

```
Serial2/1/0.1/1/1:0 is down, line protocol is down
Hardware is Channelized-T3
MTU 1500 bytes, BW 192 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 never
Queueing strategy: fifo
Output queue 0/40, 0 drops; input queue 0/75, 0 drops
Available Bandwidth 192 kilobits/sec
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, 2 interface resets
0 output buffer failures, 0 output buffers swapped out
0 carrier transitions alarm present
VC 2: timeslot(s): 1-3, Transmitter delay 0, non-inverted data
UUT#sh run | beg 2/1/0
controller SONET 2/1/0
ais-shut
framing sonet
clock source line
overhead j0 1
!
sts-1 1
mode vt-15
vtg 1 t1 1 channel-group 0 timeslots 1-3
vtg 2 t1 4 channel-group 0 timeslots 1-2,5-6
vtg 2 t1 4 channel-group 1 timeslots 3,7,9
!
sts-1 2
mode ct3
t1 1 channel-group 0 timeslots 1-24
t1 2 channel-group 0 timeslots 1-12
t1 3 channel-group 0 timeslots 1
!
sts-1 3
mode t3
!
controller T3 3/1/0
shutdown
cablelength 224
!
controller T3 3/1/1
shutdown
cablelength 224
!
!
interface Loopback0
ip address 172.10.11.1 255.255.255.255
.
.
```

Verifying Per-Port Interface Configuration and Status

To find detailed interface information on a per-port basis, use the **show interface serial** command and specify the port as described in the "Serial Interface Naming" section on page 18-18.

The following example provides sample output for interface port 0 on the SPA located in the first subslot of the Cisco 7600 SIP-200 installed in slot 2 of a Catalyst 6500 Series switch:

```
Router# show interface serial 2/1/0.2/1:0
Serial2/1/0.2/1:0 is down, line protocol is down
Hardware is Channelized-T3
MTU 1500 bytes, BW 1536 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 never
Queueing strategy: fifo
Output queue 0/40, 0 drops; input queue 0/75, 0 drops
Available Bandwidth 1536 kilobits/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, 2 interface resets
0 output buffer failures, 0 output buffers swapped out
0 carrier transitions alarm present
VC 5: timeslot(s): 1-24, Transmitter delay 0, non-inverted data
UUT#sh int Serial2/1/0.3
Serial2/1/0.3 is down, line protocol is down
Hardware is CHOCx SPA
MTU 4470 bytes, BW 44210 Kbit, DLY 200 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 never
Queueing strategy: fifo
Output queue 0/40, 0 drops; input queue 0/75, 0 drops
Available Bandwidth 44210 kilobits/sec
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 parity
```