Configuring Interfaces and Circuits

This chapter describes how to configure the CSS interfaces and circuits and how to bridge interfaces to Virtual LANs (VLANs). Information in this chapter applies to all CSS models, except where noted.

This chapter contains the following major sections:

- Interface and Circuit Overview
- Configuring Interfaces
- Configuring Circuits
- Configuring RIP for an IP Interface
- Configuring the Switched Port Analyzer Feature

Interface and Circuit Overview

The CSS provides Ethernet interfaces (ports) that enable you to connect servers, PCs, routers, and other devices to the CSS.

Using the **bridge** command, you assign the Ethernet interfaces to a specific VLAN. Each VLAN circuit requires an IP address. Assigning an IP address to each VLAN circuit allows the CSS to route Ethernet interfaces from VLAN to VLAN.

Using the **trunk** command, you can assign multiple VLANs to a CSS Ethernet interface port (Fast Ethernet port or Gigabit Ethernet port). A trunk is a point-to-point link carrying the traffic of several VLANs. The advantage of a trunk is to save ports by creating a link between two CSSs implementing VLANs. A trunk bundles virtual links over one physical link. The unique physical link between the two CSSs is able to carry traffic for the specified VLANs.

Note The **trunk** and **vlan** commands (and the associated software functionality) comply with the IEEE 802.1Q Standard for Local and Metropolitan Area Networks: Virtual Bridged Local Area Networks.

The CSS forwards VLAN circuit traffic to the IP interface. The IP interface passes the traffic to the IP forwarding function where the CSS compares the destination of each packet to information contained in the routing table. Once the CSS resolves the packet addresses, it forwards the packet to the appropriate VLAN and destination port.

With trunking enabled, the CSS automatically inserts a tag in every frame transmitted over the trunk link to identify the originating VLAN. When the VLAN-aware CSS receives the frame, it reviews the VLAN-tagged packet to identify the transmitting VLAN. If the VLAN is recognized, the frame is routed to the proper port and VLAN destination. If the frame is from a VLAN that is not assigned to the trunk port, the packet is ignored. By default, the CSS discards untagged packets.

For an 802.1Q trunk, you can use the **default-vlan** command to:

- Accept packets that arrive untagged on the interface
- Transmit untagged packets

By using this method, the CSS can determine which VLAN transmitted an untagged frame. This capability allows VLAN-aware CSSs and VLAN-unaware CSSs to transmit and receive information on the same cable.

Figure 1-1 illustrates the interfaces, circuits, and VLANs in a CSS, and Figure 1-2 illustrates trunking between VLANs.

Figure 1-1 CSS Interfaces and Circuits

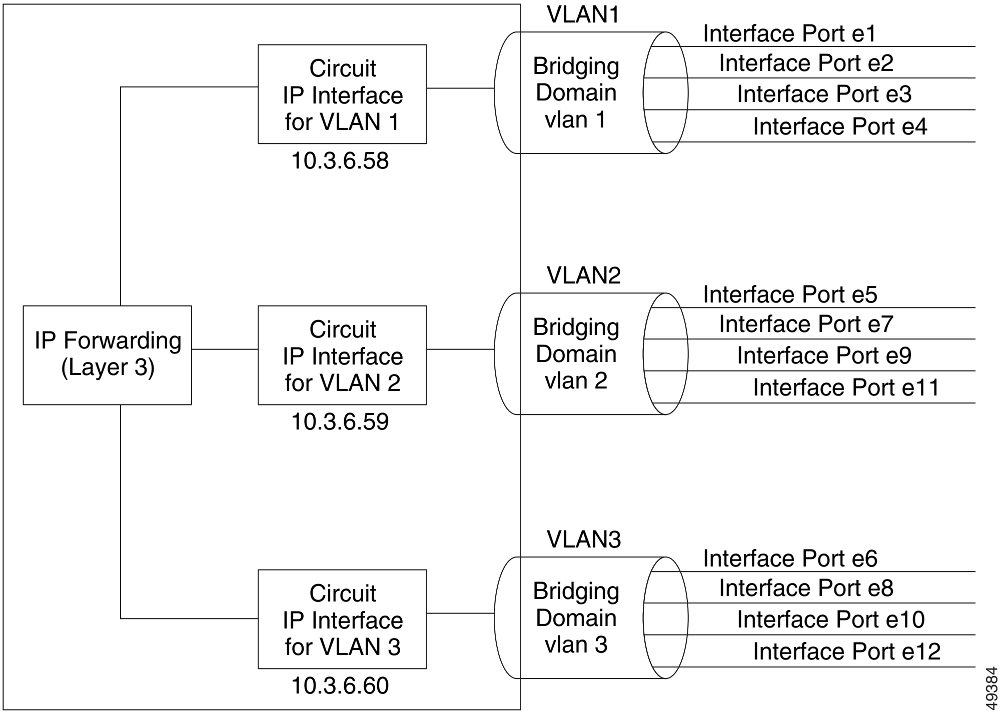
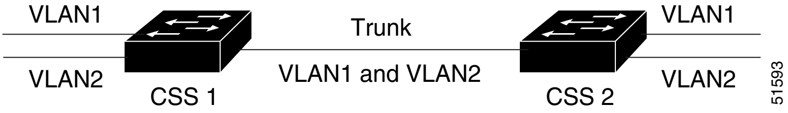


Figure 1-2 Interface Trunking Between VLANs



Interface and Circuit Configuration Quick Start

Table 1-1 provides a quick overview of the steps required to configure interfaces and circuits. Each step includes the CLI command required to complete the task. For a complete description of each feature and all the options associated with the CLI command, see the sections following Table 1-1.

Table 1-1 Interface and Circuit Configuration Quick Start

Task and Command Example
1. Log in to the CSS.

2. Enter configuration mode by typing **config**.
`# config`
`(config)#`

3. Enter the interface mode for the interface you wish to configure.
This set of interface commands applies to the CSS 11501.
`(config)# interface e1`
`(config-if[e1])#`
This set of interface commands applies to the CSS 11503 or CSS 11506.
`(config)# interface 2/1`
`(config-if[2/1])#`

4. Configure the interface duplex, speed, and flow control (default is **auto-negotiate**). `(config-if[2/1])# phy 100Mbits-FD`
5. Bridge the interface to a VLAN. All interfaces are assigned to VLAN1 by default. `(config-if[2/1])# bridge vlan 2`

6. (Optional) Enable trunking for a CSS Gigabit Ethernet or Fast Ethernet port.
`(config-if[2/1])# trunk`
`(config-if[2/1])# vlan 2`
`Create VLAN<2>, [y/n]:y`
`(config-if-vlan[2/1-2])# vlan 3`
`Create VLAN<3>, [y/n]:y`
`(config-if-vlan[2/1-3])#`

Table 1-1 Interface and Circuit Configuration Quick Start (continued)

Task and Command Example

7. (Optional) Display all circuit information for circuits that are currently active.

```
(config-if[2/1])# show circuit all
```

8. (Optional) Display the interface configuration.

```
(config-if[2/1])# show interface
(config-if[2/1])# exit
```

9. Configure circuits as required. Assign an IP address and subnet mask to each circuit.

```
(config)# circuit VLAN1
(config-circuit[VLAN1])# ip address 10.3.6.58/24
(config)# circuit VLAN3
(config-circuit[VLAN3])# ip address 10.3.6.60/24
(config-circuit-ip[VLAN3-10.3.6.60])# exit
```

10. (Optional) Display the circuit configuration.

```
(config-circuit[VLAN1])# show circuit all
```

11. (Recommended) Save your configuration changes to the startup-configuration file. If you do not save the running configuration, all configuration changes are lost upon reboot.

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

The following running-configuration example shows the results of entering the commands in Table 1-1.

```
!********************** INTERFACE *********************
interface  2/1
  phy 100Mbits-FD
  bridge vlan 2

!*********************** CIRCUIT **********************
circuit VLAN1
    ip address 10.3.6.58 255.255.255.255

circuit VLAN3
    ip address 10.3.6.60 255.255.255.255
```

Configuring Interfaces

Interfaces are ports that enable you to connect devices to the CSS and connect the CSS to the Internet. The commands to configure interfaces on the CSS 11501 differ slightly from the commands to configure interfaces on the CSS 11503 or CSS 11506 because they require a slot/port designation. The CSS 11501 does not use the slot/port designation.

This section includes the following topics:

- Configuring an Interface
- Entering a Description for the Interface
- Configuring Interface Duplex and Speed
- Setting Interface Maximum Idle Time
- Bridging an Interface to a VLAN
- Specifying VLAN Trunking for an Interface
- Configuring Spanning-Tree Bridging for a VLAN or a Trunked Interface
- Configuring Port Fast on an Interface
- Showing Interface Configurations
- Shutting Down an Interface
- Shutting Down All Interfaces
- Restarting an Interface
- Restarting All Interfaces

Configuring an Interface

To configure an Ethernet interface, use the **interface** command. Enter the interface name as follows:

- CSS 11501 - Enter the interface name in *interface port* format (for example, e1 for Ethernet interface port 1).
- CSS 11503 or CSS 11506 - Enter the interface format in *slot/port* format (for example, 3/1 for Ethernet port 1 on the I/O module in slot 3).

For example, to configure interface port 1 on a CSS 11501, access interface mode for the port by entering:

```
(config)# interface e1
(config-if[e1])#
```

For example, to configure interface 1 on a CSS 11503 or CSS 11506, access interface mode for the I/O module in slot 2 by entering:

```
(config)# interface 2/1
(config-if[2/1])#
```

Note in both examples that the CSS changes from configuration mode to the specific interface mode.

Entering a Description for the Interface

To identify the Ethernet interface, use the **description** command. Enter a quoted text string from 1 to 255 characters including spaces.

For example:

```
(config-if[2/1])# description "Connects to server17"
```

To view an interface description, use the **show running-config interface** command. For example:

```
(config-if[2/1])# show running-config interface 2/1

!*********************** INTERFACE ***********************
interface 2/1
    description "Connects to server17"
    bridge vlan 2
```

To remove an interface description, enter:

```
(config-if[2/1])# no description
```

Configuring Interface Duplex and Speed

By default, the CSS Fast Ethernet interface and Gigabit Ethernet interface are configured to auto-negotiate. The CSS automatically detects the network line speed (Fast Ethernet only) and duplex of incoming signals, and synchronizes those parameters during data transfer. Auto-negotiation enables the CSS and the other devices on the link to achieve the maximum common level of operation.

Note The CSS 1000BASE-T Gigabit Ethernet port supports 1000 Mbps full-duplex operation only and does not support auto-negotiation.

When using Fast Ethernet ports with older equipment that cannot transmit the duplex and speed with the signals, you can manually configure the speed (10 Mbps, 100 Mbps) and duplex (half or full duplex) of the CSS port to match the transmitting equipment.

When you use Gigabit Ethernet ports, if the link does not come up (perhaps due to traffic congestion), you may need to force the CSS and its link partner in to a specific mode. The CSS allows you to manually select a full duplex and flow control (pause frame) mode. Flow control allows the CSS to control traffic during congestion by notifying the other port to stop transmitting until the congestion clears. When the other device receives the pause frame, it temporarily stops transmitting data packets. When the CSS detects local congestion and becomes overwhelmed with data, the Gigabit Ethernet ports transmits a pause frame. Both the CSS Gigabit Ethernet and its link partner must be configured with the same pause method (asymmetric, symmetric, or both). By default, all Gigabit Ethernet ports are configured to full duplex mode with symmetric pause (pause frames transmitted and received by the CSS).

Note If you configure the **redundancy-phy** command on an interface of the master CSS in a box-to-box redundancy configuration and then make a change to the port settings of that interface using the **phy** command (for example, changing **auto-negotiate** to **100Mbits-FD**), the master CSS fails over to the backup CSS. To prevent the failover from occurring, first enter the **no redundancy-phy** command on the interface, change the port settings, and then reenter the **redundancy-phy** command. For information about the **redundancy-phy** command, refer to the *Cisco Content Services Switch Redundancy Guide*.

Use the **phy** command to configure the duplex, speed (Fast Ethernet ports only), and flow control (Gigabit Ethernet ports only) for the interface ports, as follows:

- **phy auto-negotiate** - Resets the Fast Ethernet and Gigabit Ethernet ports to automatically negotiate port speed and duplex of incoming signals. The CSS 1000BASE-T Gigabit Ethernet port supports 1000 Mbps full-duplex operation only and does not support auto-negotiation.

  Note Pause mode during auto-negotiation is not supported for the Fast Ethernet ports.

- **phy auto-negotiate** {**enable** | **disable**} - Disables the Gigabit Ethernet interface from automatically negotiating duplex of incoming signals. By default, auto-negotiation is enabled for all Gigabit Ethernet ports. The CSS 1000BASE-T port supports 1000 Mbps full-duplex operation only and does not support auto-negotiation.

  Gigabit Ethernet port auto-negotiation remains enabled when a pause mode command is specified so the Gigabit Ethernet interface ports can act upon the link partner’s flow control capability. If it is necessary to disable auto-negotiation for the Gigabit Ethernet port when using a pause mode, enter the **phy auto-negotiate disable** command.

- **phy 10Mbits-FD** - Sets the Fast Ethernet port to 10 Mbps and full-duplex mode.
- **phy 10Mbits-HD** - Sets the Fast Ethernet port to 10 Mbps and half-duplex mode.
- **phy 100Mbits-FD** - Sets the Fast Ethernet port to 100 Mbps and full-duplex mode.

- **phy 100Mbits-HD** - Sets the Fast Ethernet port to 100 Mbps and half-duplex mode.
- **phy 1Gbits-FD-asym** - Sets the Gigabit Ethernet port to full-duplex mode with asymmetric pause frames transmitted toward the link partner. Asymmetric pause is useful when you need the CSS to pause its link partner but not to respond to pause frames transmitted from the link partner.
- **phy 1Gbits-FD-no pause** - Sets the Gigabit Ethernet port to full-duplex mode with no pause frames transmitted or received.
- **phy 1Gbits-FD-sym** - Sets the Gigabit Ethernet port to full-duplex mode with symmetric pause (pause frames transmitted and received by the CSS). Symmetric pause is useful for point-to-point links. By default, all Gigabit Ethernet ports are configured to full-duplex mode with symmetric pause.
- **phy 1Gbits-FD-sym-asym** - Sets the Gigabit Ethernet port to full-duplex mode with symmetric and asymmetric pause frames used with the local device.

For example, to configure Fast Ethernet interface 1 on the I/O module in slot 2 of the CSS 11503 to 100 Mbps and half-duplex mode, enter:

```
(config-if[2/1])# phy 100Mbits-HD
```

For example, to configure gigabit interface 1 on the SCM in slot 1 of the CSS 11503 to full-duplex mode with asymmetric pause, enter:

```
(config-if[1/1])# phy auto-negotiate disable
(config-if[1/1])# phy 1Gbits-FD-asym
```

Setting Interface Maximum Idle Time

As a troubleshooting tool to verify an interface’s ability to receive traffic, use the **max-idle** command. If the interface does not receive traffic within the configured idle time, the CSS reinitializes the interface automatically.

Set the idle time to a value greater than the interval over which the interface is receiving traffic. For example, if the interface receives traffic every 90 seconds, set the idle time to a value greater than 90 seconds. If you set the idle time to less than 90 seconds, the CSS would continuously reinitialize the interface before the interface was able to receive traffic.

Enter an idle time from 15 to 65535 seconds. The default is 0, which disables the idle timer.

For example, to set the maximum idle time to 180 seconds for interface port 1 on a CSS 11503, the I/O module in slot 2, enter:

```
(config-if[2/1])# max-idle 180
```

To reset the idle time for an interface to its default value of 0, enter:

```
(config-if[2/1])# no max-idle
```

Bridging an Interface to a VLAN

To specify a VLAN and associate it with the specified Ethernet interface, use the **bridge vlan** command. Enter an integer from 1 to 4094 as the VLAN identifier. The default is 1. All interfaces are assigned to VLAN1 by default.

The following list defines the maximum number of VLANs supported by the specific CSS models:

- CSS 11501 and CSS 11503 - A maximum of 256 VLANs per CSS and 64 VLANs per port (FE or GE)
- CSS 11506 - A maximum of 512 VLANs per CSS and 64 VLANs per port (FE or GE)

When you specify the **bridge vlan** command, enter the word **vlan** in lowercase letters and include a space before the VLAN number (for example, **vlan 2**).

For example, to configure e1 to VLAN2 on the CSS 11501, enter:

```
(config-if[e1])# bridge vlan 2
```

The CSS Gigabit Ethernet and Fast Ethernet interface ports support trunking to multiple VLANs through the **trunk** command. In this configuration, use the **trunk** command for the Ethernet interface instead of the **bridge vlan** command (and the other associated bridge CLI commands).

To restore the default VLAN1 on the CSS 11501, enter:

```
(config-if[e7])# no bridge vlan
```

To display all interfaces and the VLANs to which they are configured, use the **show circuit** command. In the **show circuit** display, VLANs appear as VLAN (uppercase, with no space before the VLAN number). See the "Showing Circuits" section for information about the **show circuits** command.

Specifying VLAN Trunking for an Interface

To activate VLAN trunking for a CSS interface, use the **trunk** command. You specify all VLANs that include the specified port as part of the VLAN. The **trunk** command also converts the link in to a trunk link. Use the **vlan** command to specify the number of each VLAN to be associated with the Gigabit Ethernet or Fast Ethernet port. Enter an integer from 1 to 4094 as the VLAN identifier.

The following list defines the maximum number of VLANs supported by the specific CSS models:

- CSS 11501 and CSS 11503 - A maximum of 256 VLANs per CSS and 64 VLANs per port (FE or GE)
- CSS 11506 - A maximum of 512 VLANs per CSS and 64 VLANs per port (FE or GE)

The CSS software has a dependency when using the **trunk** command. For trunking to be enabled, all VLAN bridging commands for any active VLAN must first be disabled for the Gigabit Ethernet or Fast Ethernet interface by using the **no bridge vlan**, **no bridge port-priority**, **no bridge state**, and **no bridge pathcost** commands. If you do not disable VLAN bridging on an interface, the CSS software instructs you to do so.

When you specify the **trunk** command, enter the word **vlan** in lowercase letters and include a space before the VLAN number (for example, **vlan 2**). The CSS automatically prompts you to create the specified VLAN (where **y** instructs the software to create the VLAN and **n** cancels the VLAN creation).

For example, to configure Gigabit Ethernet port 1 in slot 1 for use in VLAN2, VLAN3, and VLAN9, enter:

```
(config-if[1/1])# trunk
(config-if[1/1])# vlan 2
Create VLAN<2>, [y/n]:y
(config-if-vlan[1/1-2])# vlan 3
Create VLAN<3>, [y/n]:y
(config-if-vlan[1/1-3])# vlan 9
Create VLAN<9>, [y/n]:y
(config-if-vlan[1/1-9])#
```

The **no trunk** command turns off all trunking, removes all specified **vlan** commands associated with the interface, and deletes this information from the running configuration. The interface is returned to VLAN1 by default.

To disable trunking on the specified interface and associated VLANs, enter:

```
(config-trunkif[2/3])# no trunk
```

To display all interfaces and the VLANs to which they are configured, use the **show circuit** command. In the **show circuit** output, VLANs appear as VLAN (uppercase, with no space before the VLAN number). For an interface that has trunking enabled, an "-*n*" (where *n* is the associated VLAN number) is appended to the prefix. In this example, 1/4-1 indicates slot 1, port 4, VLAN1. See the "Showing Circuits" section for information about the **show circuits** command.

Selecting a Default VLAN in a Trunk

To define a default VLAN to accept packets that arrive untagged on the interface, include the **default-vlan** command as part of the trunk/VLAN definition. The command also specifies that the packets transmitted from this VLAN will be untagged. The default VLAN must be explicitly set if you want untagged packets to be processed by the CSS. Otherwise, these packets are discarded.

The **default-vlan** command can be specified only for a single VLAN. If you attempt to use this command for another VLAN, the CSS instructs you to disable the current default VLAN using the **no default-vlan** command.

For example:

```
(config-if[1/1])# trunk
(config-if[1/1])# vlan 2
Create VLAN<2>, [y/n]:y
(config-if-vlan[1/1-2])# vlan 3
Create VLAN<3>, [y/n]:y
(config-if-vlan[1/1-3])# default-vlan
```

To remove the default VLAN selection, enter:

```
(config-if-vlan[1/1-3])# no default-vlan
```

Configuring Spanning-Tree Bridging for a VLAN or a Trunked Interface

The CSS supports configuration of Spanning-Tree Protocol (STP) bridging for an Ethernet interface in a VLAN or for a trunked Ethernet interface. Spanning-tree bridging is used to detect, and then prevent, loops in the network. You can define the bridge spanning-tree path cost, priority, and state for an Ethernet interface or for a trunked Ethernet interface. Ensure you configure the spanning-tree bridging parameters the same on all switches running STP in the network.

Note When connecting a Cisco Catalyst switch to a CSS using an 802.1Q trunk and the Spanning-Tree Protocol, the Catalyst runs a spanning-tree instance for each VLAN. When you configure an 802.1Q trunk on an Ethernet interface for the Catalyst switch, the bridge protocol data units (BPDUs) are tagged with the corresponding VLAN ID and the destination MAC address changes from the standard 01-80-C2-00-00-00 to the proprietary 01-00-0c-cc-cc-cd. This modification allows Cisco switches operating in a non-Cisco (a mix of other vendors) 802.1Q trunk environment to maintain spanning-tree states for all VLANs. Although the CSS maintains a spanning-tree instance for each VLAN as well, the CSS uses the standard 01-80-C2-00-00-00 destination MAC address for all BPDUs (tagged or untagged). When you connect a Cisco Catalyst switch to a CSS over an 802.1Q trunk, the result is that neither switch recognizes the other's BPDUs, and both assume root status. If a spanning-tree loop is detected, the Catalyst switch goes in to blocking mode on one of its looped ports.

This section includes the following topics:

- Configuring Spanning-Tree Bridge Pathcost
- Configuring Spanning-Tree Bridge Port Priority
- Configuring Spanning-Tree Bridge State

For details about globally configuring spanning-tree bridging parameters for the CSS (such as bridge aging time, forward delay time, hello time interval, and maximum age), refer to Chapter 2, Configuring Spanning-Tree Bridging for the CSS.

Configuring Spanning-Tree Bridge Pathcost

The path cost is the contribution of the interface to the vast path cost towards the spanning-tree root. Use the **bridge pathcost** command to set the spanning-tree path cost for an Ethernet interface or for a trunked Ethernet interface. Enter an integer from 1 to 65535. The default is dynamically configured based on the interface speed.

For example, to set a path cost of 9 for e7 on the CSS 11501, enter:

```
(config-if[e7])# bridge pathcost 9
```

For example, to set a path cost of 2 for the I/O module in slot 1, Ethernet port 1, in VLAN3, enter:

```
(config-if-vlan[1/1-3])# bridge pathcost 2
```

To restore the default path cost, enter:

```
(config-if-vlan[1/1-3])# no bridge pathcost
```

Configuring Spanning-Tree Bridge Port Priority

To set the spanning-tree bridge port priority for an Ethernet interface or for a trunked Ethernet interface, se the **bridge port-priority** command. If the CSS has a bridge port priority that is lower than all other switches, it will be automatically selected by the other switches as the root switch. Enter an integer from 0 to 255. The default is 128.

For example, to set a bridge port priority of 100 for e7 on the CSS 11501, enter:

```
(config-if[e7])# bridge port-priority 100
```

For example, to set a bridge port priority of 100 for the I/O module in slot 1, Ethernet port 1, in VLAN3, enter:

```
(config-if-vlan[1/1-3])# bridge port-priority 100
```

To restore the default port priority of 128, enter:

```
(config-if-vlan[1/1-3])# no bridge port-priority
```

Configuring Spanning-Tree Bridge State

By default, an Ethernet interface is set to the enabled bridge state. Use the **bridge state** command to set the spanning-tree bridge state for an Ethernet interface or for a trunked Ethernet interface.

For example, to enable the bridge state for e7 on the CSS 11501, enter:

```
(config-if[e7])# bridge state enable
```

For example, to enable the bridge state for the I/O module in slot 1, Ethernet port 1, in VLAN3, enter:

```
(config-if-vlan[1/1-3])# bridge state enable
```

To disable the bridge state, enter:

```
(config-if-vlan[1/1-3])# bridge state disable
```

Configuring Port Fast on an Interface

The Port Fast feature immediately brings a CSS Ethernet interface (port) to the Spanning Tree Protocol (STP) forwarding state from a blocking state, bypassing the listening and learning states. You can specify Port Fast for ports connected to a single workstation or server to allow those devices to immediately connect to the network, rather than waiting for the STP to converge.

Ports connected to a single workstation or server should not receive bridge protocol data units (BPDUs).

Caution The purpose of Port Fast is to minimize the time ports must wait for STP to converge. This means that the Port Fast function is effective only when used on ports connected to end stations in the network. If you enable Port Fast on a port connecting to another switch, you risk creating a spanning-tree loop. Consider using the BDPU guard feature to avoid creating a spanning-tree loop.

This section includes the following topics:

- Enabling Port Fast
- Enabling BPDU Guard
- Showing Port Fast Information

Enabling Port Fast

A port with the Port Fast feature enabled is moved directly to the spanning-tree forwarding state without waiting for the standard forward-time delay.

Caution Use Port Fast only when connecting a single end station to a CSS interface. Enabling this feature on a port connected to a switch or hub could prevent spanning tree from detecting and disabling loops in your network, which could cause broadcast storms and address-learning problems.

To enable Port Fast on a non-trunked port, use the interface mode **bridge port-fast enable** command. You cannot configure Port Fast on a trunked port. By default, Port Fast is disabled on the port.

```
(config-if[2/1])# bridge port-fast enable
```

To disable the Port Fast feature, use the interface mode **bridge port-fast disable** command.

```
(config-if[2/1])# bridge port-fast disable
```

Enabling BPDU Guard

Use the BPDU guard feature to prevent a Port Fast port on the CSS from participating in the spanning tree. When you globally enable BPDU guard on the Port Fast ports, spanning tree shuts down the ports that receive BPDUs. For information to enable Port Fast on an interface port, see the "Configuring Port Fast on an Interface" section.

In a valid configuration, the enabled Port Fast ports do not receive BPDUs. Receiving a BPDU on a Port Fast-enabled port signals an invalid configuration, such as the connection of an unauthorized device, and the BPDU guard feature puts the port in the disabled state. The BPDU guard feature provides a secure response to invalid configurations because you must manually put the port back in service.

To enable the BPDU guard on the CSS, use the global configuration **bridge bdpu-guard enabled** command:

```
(config)# bridge bpdu-guard enabled
```

To disable BPDU guard, use the global configuration **bridge bpdu-guard disabled** command:

```
(config)# bridge bpdu-guard disabled
```

Showing Port Fast Information

To display whether Port Fast is enabled or disabled on all interfaces, use the **show bridge port-fast** command. This command is available in all modes. This command also displays whether the BPDU guard feature is enabled or disabled on the CSS, and the state of the interfaces.

Table 1-2 describes the fields in the **show bridge port-fast** command output.

Table 1-2 Field Description for the show bridge port-fast Command

Field	Description
BPDU guard is *state* on this switch.	The state of the BPDU guard feature on the CSS: Enabled or Disabled.
Name	The number of the module slot and interface.
IfIndex	The interface index number.
Type	The type of interface. • **fe** indicates a Fast Ethernet interface. • **ge** indicates a Gigabit Ethernet interface.
Oper	The operational state of the interface: Up or Down.
Admin	The administration state: Enable or Down.
PortFast State	Indicates whether Port Fast is enabled or disabled on the interface.

Showing Interface Configurations

This CSS includes a series of **show** interface mode commands that enable you to view interface configuration information about the CSS. This information includes VLAN bridging, VLAN trunk status, list of valid Ethernet interfaces, interface duplex and speed values, interface statistics, and errors on an Ethernet interface.

This section includes the following topics:

- Showing Bridge Configurations
- Showing Trunking Configurations
- Showing Interface Information
- Showing Interface Duplex and Speed
- Showing Interface Statistics
- Showing Ethernet Interface Errors

Showing Bridge Configurations

The CSS enables you to show bridging information for a specific VLAN in the CSS. Use the **show bridge** command to display this bridging information.

The syntax for this command is:

show bridge [**forwarding**|**status**] {*vlan_number*}

The options and variables are as follows:

- **forwarding** - Displays the bridge forwarding table including the VLAN number, the MAC addresses, and port numbers.
- **status** - Displays the bridge spanning-tree status including the Spanning Tree Protocol (STP) state; designated root; bridge ID; root maximum age; hello time and forward delay; and port information including state, VLAN, root and port cost, and designated root and port number.
- *vlan_number* - Displays the forwarding table or spanning tree status for the specified VLAN number. To see a list of VLAN numbers, enter **show bridge** [**forwarding**|**status**] **?**

To display bridge forwarding or bridge status for a specific VLAN in the CSS, enter the **show bridge forwarding** or the **show bridge status** command with the VLAN number. Entering the **show bridge** command with a VLAN number returns a list of available VLANs.

Table 1-3 describes the fields in the **show bridge forwarding** command output.

Table 1-3 Field Descriptions for the show bridge forwarding Command

Field	Description
VLAN	The bridge interface virtual LAN number
MAC Address	The MAC address for the entries
Port Number	The port number for the bridge forwarding table

Table 1-4 describes the fields in the **show bridge status** command output.

Table 1-4 Field Descriptions for the show bridge status Command

Field	Description
STP State	The state of the Spanning-Tree Protocol: Enabled or Disabled.
Root Max Age	The timeout period, in seconds, during which the host times out root information.
Root Hello Time	The interval, in seconds, that the root bridge broadcasts its hello message to other CSSs.
Root Fwd Delay	The delay time, in seconds, that the root bridge uses for forward delay.
Designated Root	The bridge ID for the designated root.
Bridge ID	The bridge ID of this bridge.
Port	The port ID.

Table 1-4 Field Descriptions for the show bridge status Command (continued)

Field	Description
State	The state of the port. The possible states are as follows: • Block - The blocking state. A port enters the blocking state after CSS initialization. The port does not participate in frame forwarding. • Listen - The listening state. This state is the first transitional state a port enters after the blocking state. The port enters this state when STP determines that the port should participate in frame forwarding. • Learn - The learning state. The port enters the learning state from the listening state. The port in the learning state prepares to participate in frame forwarding. • Forward - The forwarding state. The port enters the forwarding state from the learning state. A port in the forwarding state forwards frames. • Disabled - The disabled state. A port in the disabled state does not participate in frame forwarding or STP. A port in the disabled state is nonoperational.
Designated Bridge	The bridge ID for the designated bridge.
Designated Root	The bridge ID for the designated root.
Root Cost	The cost of the root.
Port Cost	The cost of the port.
Desg Port	Designated port.

Showing Trunking Configurations

The CSS enables you to show VLAN trunk status information for Gigabit Ethernet and Fast Ethernet ports. To display this information, use the **show trunk** command.

Table 1-5 describes the fields in the **show trunk** command output.

Table 1-5 Field Descriptions for the show trunk Command

Field	Description
Port	The CSS port
VLAN	The VLAN on the port
Default VLAN	The configured default VLAN on the port (if there is no configured default VLAN, "None" appears in this field)

Showing Interface Information

To display a list of valid interfaces for the CSS, use the **show interface** command. For example:

```
(config)# show interface
```

To display information for a specific interface, enter the **show interface** command and the interface name. Enter the interface name as follows:

- CSS 11501 - Enter the interface name in *interface port* format (for example, e1 for Ethernet interface port 1).
- CSS 11503 or CSS 11506 - Enter the interface format in *slot/port* format (for example, 3/1 for Ethernet port 1 on the I/O module in slot 3).

For example, to show interface information for port 1 on a CSS 11503, the I/O module in slot 2, enter:

```
(config)# show interface 2/1
```

Table 1-6 describes the fields in the **show interface** command output.

Table 1-6 Field Descriptions for the show interface Command

Field	Description
Name	The name of the interface.
ifIndex	The Index for the interface.
Type	The type of interface. The possible types include: • fe - Fast Ethernet interface • ge - Gigabit Ethernet interface • console - Console interface
Oper	Operational state: Up or Down.
Admin	Administrative state: Up or Down.
Last Change	The date of the last state change.

Showing Interface Duplex and Speed

Use the **show phy** command to show duplex and speed values for all interfaces. For example:

```
(config)# show phy
```

To show duplex and speed value for a specific interface, specify the **show phy** command and the interface name. Enter the interface name as follows:

- CSS 11501 - Enter the interface name in *interface port* format (for example, e1 for Ethernet interface port 1).
- CSS 11503 or CSS 11506 - Enter the interface format in *slot/port* format (for example, 3/1 for Ethernet port 1 on the I/O module in slot 3).

For example, to show the interface and duplex speed for interface port 1 on a CSS 11506, the I/O module in slot 2, enter:

```
(config)# show phy 2/1
```

Table 1-7 describes the fields in the **show phy** command output.

Table 1-7 Field Descriptions for the show phy Command

Field	Description
Name	The name of the physical interface.
Configured Speed	The configured speed for the Ethernet interface (port) in the CSS. Auto indicates the speed is automatically negotiated.
Configured Duplex	The configured duplex for the Ethernet interface (port) in the CSS. Auto indicates the duplex is automatically negotiated.
Actual Speed	The actual speed for the Ethernet interface (port) in the CSS.
Actual Duplex	The configure duplex for the Ethernet interface (port) in the CSS.
Link	The link status: Up or Down.
Rev	Revision number of the chip.
Partner Auto	Indicates whether auto-negotiation is available on the link partner.

Showing Interface Statistics

Use the **show mibii** command to display the extended 64-bit MIB-II statistics for a specific interface, or for all interfaces in the CSS. The CSS Enterprise ap64Stats MIB defines these statistics. The Gigabit Ethernet module port statistics are an aggregation of all ports on the module.

To display the RFC 1213 32-bit statistics, include the **-32** suffix.

To display extended MIB-II statistics for a specific interface in the CSS, enter the **show mibii** command with the interface name. To see a list of interfaces in the CSS, enter **show mibii ?**.

Note Refer to the *Cisco Content Services Switch Administration Guide* for information on CSS MIBs.

Table 1-8 describes the fields in the **show mibii** command output.

Table 1-8 Field Descriptions for the show mibii Command

Field	Description
MAC	The interface address at the protocol layer immediately below the network layer in the protocol stack. For interfaces that do not have such an address (for example, a serial line), this object contains an octet string of zero length.
Administrative	The desired state of the interface (Enabled, Disabled, or Testing). The testing state indicates no operational packets can be passed.
MTU	The size of the largest datagram that can be sent or received on the interface, specified in octets. For interfaces that are used for transmitting network datagrams, this is the size of the largest network datagram that can be sent on the interface.
In Octets	The total number of octets received on the interface, including framing characters.
In Unicast	The number of subnetwork-unicast packets delivered to a higher-layer protocol.
In Multicast	The number of non-unicast (for example, subnetwork-broadcast or subnetwork-multicast) packets delivered to a higher-layer protocol.
In Errors	The number of inbound packets that contained errors preventing them from being delivered to a higher-layer protocol.
In Discards	The number of inbound packets that were chosen to be discarded even though no errors had been detected to prevent their being delivered to a higher-layer protocol. One possible reason for discarding such a packet could be to free up buffer space.
In Unknown	The number of packets received over the interface that were discarded because of an unknown or unsupported protocol.

Table 1-8 Field Descriptions for the show mibii Command (continued)

Field	Description
Last Change	The value of sysUpTime at the time the interface entered its current operational state. If the state has not changed since the time the CSS came up, the sysUptime is when the port was initialized.
Operational	The current operational state of the interface (Up, Down, or Testing). The Testing state indicates no operational packets can be passed.
Speed	An estimate of the interface's current bandwidth, in bits per second. For interfaces that do not vary in bandwidth or for those where no accurate estimation can be made, this object contains the nominal bandwidth.
Queue Len	The length of the output packet queue (in packets).
Out Octets	The total number of octets transmitted out of the interface, including framing characters.
Out Unicast	The total number of packets that higher-level protocols requested be transmitted to a subnetwork-unicast address, including those packets that were discarded or not sent.
Out Multicast	The total number of packets that higher-level protocols requested be transmitted to a non-unicast (for example, a subnetwork-broadcast or subnetwork-multicast) address, including those packets that were discarded or not sent.
Out Errors	The number of outbound packets that could not be transmitted because of errors.
Out Discards	The number of outbound packets that were chosen to be discarded even though no errors had been detected to prevent their being transmitted. One possible reason for discarding such a packet could be to free up buffer space.

To clear interface statistics, use the **clear statistics** command in SuperUser mode. For example:

```
# clear statistics
```

Showing Ethernet Interface Errors

To list the errors on an Ethernet interface, use the **show ether-errors** command and options. When required, enter the interface name as a case-sensitive unquoted text string. To see a list of interfaces, enter **show ether-errors ?**.

The command provides the following options:

- **show ether-errors** - Displays the extended 64-bit statistics for errors on all Ethernet interfaces in the CSS. The Enterprise ap64Stats MIB defines these statistics.
- **show ether-errors** *interface name* - Displays the extended 64-bit statistics for errors on a specific Ethernet interface in the CSS. The Enterprise ap64Stats MIB defines these statistics. Enter the interface name as a case-sensitive unquoted text string.
- **show ether-errors zero** - Displays the Ethernet errors for all Ethernet interfaces in the CSS and reset the statistics to zero upon retrieval.
- **show ether-errors zero** *interface name* - Displays the Ethernet errors for the specified Ethernet interface in the CSS and resets the statistics to zero upon retrieval. Enter the interface name as a case-sensitive unquoted text string.
- **show ether-errors-32** - Displays the RFC 1398 32-bit statistics, including the **-32** suffix.
- **show ether-errors-32** *interface name* - Displays the RFC 1398 32-bit statistics, including the **-32** suffix. Enter the interface name as a case-sensitive unquoted text string.

Table 1-9 describes the fields in the **show ether-errors** command output.

Table 1-9 Field Descriptions for the show ether-errors Command

Field	Description
Alignment	The number of frames with alignment errors (frames that do not end with a whole number of octets and have a bad cyclic redundancy check) received on the interface.
FCS	The number of frames received on the interface that are an integral number of octets in length but do not pass the frame check sequence (FCS) check.

Table 1-9 Field Descriptions for the show ether-errors Command (continued)

Field	Description
Single Collision	The number of successfully transmitted frames on the interface for transmissions that were inhibited by exactly one collision.
Multiple Collisions	The number of successfully transmitted frames on the interface for transmissions that were inhibited by more than one collision.
SQE Test	The number of times that the SQE TEST ERROR message is generated.
Deferred Tx	The number of frames for which the first transmission attempt on the interface is delayed because the medium is busy. The count represented by an instance of this object does not include frames involved in collisions.
Internal Rx Errors	The number of frames for which reception on the interface failed due to an internal MAC sublayer receive error.
Frame too Long	The number of frames received on the interface that exceeded the maximum permitted frame size.
Carrier Sense Errors	The number of times that the carrier sense condition was lost or never asserted when attempting to transmit a frame on the interface.
Internal Tx Errors	The number of frames for which transmission on the interface failed due to an internal MAC sublayer transmit error.
Excessive Collisions	The number of frames for which transmission on the interface failed due to excessive collisions.
Late Collisions	The number of times that a collision is detected on the interface later than 512 bit-times in to the transmission of a packet.

Shutting Down an Interface

To shut down an interface, use the **admin-shutdown** or **shut** command.

Caution Shutting down an interface terminates all connections to the interface.

For example:

- To shut down interface e3 on the CSS 11501 with the **admin-shutdown** command, enter:

```
(config-if[e3])# admin-shutdown
```

- To shut down interface e3 on the CSS 11501 with the **shut** command, enter:

```
(config-if[e3])# shut
```

When you use the **shut** command, the CSS changes the **shut** command to the **admin-shutdown** command in the running configuration.

Note If you configure the **redundancy-phy** command on an interface and then disable the interface using the **admin-shutdown** command, the master CSS fails over to the backup CSS. To prevent the CSS from failing over when you administratively disable the interface, remove the **redundancy-phy** command by entering **no redundancy-phy** before you enter the **admin-shutdown** command on that interface.

Shutting Down All Interfaces

To shut down all interfaces simultaneously, use the **admin-shutdown** command. This command is only available in the SuperUser mode. The **admin-shutdown** command provides a quick way to shut down all physical devices in the CSS.

Caution Shutting down an interface terminates all connections to the interface.

To shut down all interfaces, enter:

```
# admin-shutdown
```

Restarting an Interface

To restart an interface, use the **no admin-shutdown** or **no shut** command. For example:

- To restart interface e3 on the CSS 11501 with the **no admin-shutdown** command, enter:

  ```
  (config-if[e3])# no admin-shutdown
  ```

- To restart interface e3 on the CSS 11501 with the **no shut** command, enter:

  ```
  (config-if[e3])# no admin-shutdown
  ```

Note The CSS automatically sends a gratuitous ARP for the IP interface address when you restart the interface. The gratuitous ARP informs all network nodes about ARP mapping. The CSS transmits one ARP request packet and one ARP reply packet for every gratuitous ARP invocation.

Restarting All Interfaces

To restart all interfaces, enter:

```
# no admin-shutdown
```

Note The CSS automatically sends a gratuitous ARP for every configured IP interface address when you restart all interfaces. The gratuitous ARP informs all network nodes about ARP mapping. The CSS transmits one ARP request packet and one ARP reply packet for every gratuitous ARP invocation.

Configuring Circuits

A circuit on the CSS is a logical entity that maps IP interfaces to a logical port or group of logical ports, for example, a VLAN. Each VLAN circuit requires an IP address. Assigning an IP address to each VLAN circuit allows the CSS to route Ethernet interfaces from VLAN to VLAN. Router Discovery Protocol (RDP) settings can also be configured for each circuit VLAN to advertise the CSS to hosts.

This section includes the following topics:

- Entering Circuit Configuration Mode
- Configuring a Circuit IP Interface
- Configuring Router-Discovery Protocol Settings for a Circuit
- Showing Circuits
- Showing IP Interfaces

Entering Circuit Configuration Mode

To enter the circuit configuration mode to configure a VLAN, use the **circuit** command. Enter the specific VLAN in uppercase letters. Do not include a space between VLAN and the VLAN number. For example:

```
(config)# circuit VLAN7
(config-circuit[VLAN7])#
```

Configuring a Circuit IP Interface

This section includes the following topics:

- Configuring a Circuit IP Address
- Configuring a Circuit-IP Broadcast Address
- Configuring Circuit-IP Redirects
- Configuring Circuit-IP Unreachables
- Configuring Router-Discovery Preference for a Circuit IP Interface
- Enabling and Disabling a Circuit IP

Configuring a Circuit IP Address

To assign an IP address to a circuit, use the **ip address** command. Enter the IP address and a subnet mask in CIDR bit-count notation or a mask in dotted-decimal notation. The subnet mask range is 8 to 31.

For example, to configure an IP address and subnet mask for VLAN7, enter:

```
(config-circuit[VLAN7])# ip address 172.16.6.58/8
```

When you specify an IP address, the mode changes to the specific circuit-ip-VLAN-IP address as shown:

```
(config-circuit-ip[VLAN7-172.16.6.58])#
```

Note The CSS automatically sends a gratuitous ARP for the IP interface address when you assign an IP address to a circuit. The gratuitous ARP informs all network nodes about ARP mapping. The CSS transmits one ARP request packet and one ARP reply packet for every gratuitous ARP invocation.

To remove a local IP address from a circuit, enter the following command from circuit mode:

```
(config-circuit[VLAN7])# no ip address
```

Configuring a Circuit-IP Broadcast Address

To change the broadcast address associated with a circuit, use the **broadcast** command. If you leave the broadcast address at zero, the all-ones host is used for numbered interfaces.

The default broadcast address is an all-ones host address (for example, IP address 172.16.6.58/24 has a broadcast address of 172.16.6.58/255). This command is available in IP configuration mode.

For example, to change the broadcast address on circuit VLAN7, enter:

```
(config-circuit-ip[VLAN7-172.16.6.58])# broadcast 0.0.0.0
```

To reset the broadcast IP address to the default all-ones host address, enter:

```
(config-circuit[VLAN7-172.16.6.58])# no broadcast
```

Configuring Circuit-IP Redirects

By default, the transmission of Internet Control Message Protocol (ICMP) redirect messages is enabled. To disable the transmission of ICMP redirect messages, enter:

```
(config-circuit-ip[VLAN7-172.16.6.58])# no redirects
```

To reenable the transmission of ICMP redirect messages, use the **redirects** command. For example:

```
(config-circuit-ip[VLAN7-172.16.6.58])# redirects
```

Configuring Circuit-IP Unreachables

By default, the transmission of ICMP Destination Unreachable is enabled. To disable the transmission of ICMP Destination Unreachable messages, enter:

```
(config-circuit-ip[VLAN7-172.16.6.58])# no unreachables
```

Use the **unreachables** command to enable the transmission of ICMP Destination Unreachable messages. The default state is enabled.

For example:

```
(config-circuit-ip[VLAN7-172.16.6.58])# unreachables
```

Configuring Router-Discovery Preference for a Circuit IP Interface

To enable router discovery and configure the router discovery preference value for a circuit IP interface, use the **router-discovery** command. When enabled, router discovery transmits packets with the "all-hosts" multicast address of 244.0.0.1.

Note To enable an interface to transmit packets with the limited broadcast multicast address of 255.255.255.255, use the **router-discovery limited-broadcast** command in circuit mode (see the "Configuring Router-Discovery Limited-Broadcast" section). Router discovery is disabled by default.

Use the **router-discovery preference** command to specify the preference level for the advertised CSS circuit IP address, relative to other devices on the same network. The value is an integer from 0 (default) to 65535. If you use the default value, you do not need to use this command.

For example, to specify a router discovery preference value of 100, enter:

```
(config-circuit-ip[VLAN7-192.168.1.58])# router-discovery
(config-circuit-ip[VLAN7-192.168.1.58])# router-discovery preference
100
```

To disable router discovery, enter:

```
(config-circuit-ip[VLAN7-192.168.1.58])# no router-discovery
```

To restore the router discovery preference value to the default of 0, enter:

```
(config-circuit-ip[VLAN7-192.168.1.58])# no router-discovery
preference
```

Enabling and Disabling a Circuit IP

By default, the IP interface on a circuit is enabled. To disable the IP interfaces on a circuit, enter:

```
(config-circuit-ip[VLAN7-172.16.6.58])# no enable
```

To reenable the IP interface on a circuit, use the **enable** command. For example:

```
(config-circuit-ip[VLAN7-172.16.6.58])# enable
```

Configuring Router-Discovery Protocol Settings for a Circuit

The CSS allows you to enable Router Discovery Protocol (RDP) settings and define a router discovery preference for each circuit VLAN. RDP announces the existence of the CSS to hosts by periodically multicasting or broadcasting a router advertisement to each interface.

Use the **circuit** command to enter the circuit configuration mode before configuring RDP for a circuit VLAN.

This section includes the following topics:

- Configuring the Router-Discovery Lifetime
- Configuring Router-Discovery Limited-Broadcast
- Configuring the Router-Discovery Max-Advertisement-Interval
- Configuring the Router-Discovery Min-Advertisement-Interval

Configuring the Router-Discovery Lifetime

By default, the maximum age that hosts remember router advertisements is three times the **max-advertisement-interval**. Use the **router-discovery lifetime** command to configure the maximum age, in seconds. Enter an integer between 0 and 9000 seconds.

For example:

```
(config-circuit[VLAN7])# router-discovery lifetime 600
```

To reset the time to the default of three times the **max-advertisement-interval**, enter:

```
(config-circuit[VLAN7)# no router-discovery lifetime
```

Configuring Router-Discovery Limited-Broadcast

By default, the CSS transmits router discovery packets using the limited broadcast address is 224.0.0.1 (the “all-hosts” multicast address). Use the **router-discovery limited-broadcast** command to transmit router discovery packets using the limited broadcast address 255.255.255.255.

For example:

```
(config-circuit[VLAN7])# router-discovery limited-broadcast
```

To revert to the default of 224.0.0.1, enter:

```
(config-circuit[VLAN7)# no router-discovery limited-broadcast
```

Configuring the Router-Discovery Max-Advertisement-Interval

By default, the maximum interval timer used for router discovery advertisement from the circuit VLAN is 600 (10 minutes). Use the **router-discovery max-advertisement-interval** command to configure the maximum interval timer used for router discovery advertisement from the circuit VLAN. This command defines the maximum interval, in seconds, between sending advertisements. Enter an integer from 4 to 1800.

For example:

```
(config-circuit[VLAN7])# router-discovery max-advertisement-interval
300
```

To restore the router discovery maximum advertisement interval to the default of 600, enter:

```
(config-circuit[VLAN7])# no router-discovery
max-advertisement-interval
```

Configuring the Router-Discovery Min-Advertisement-Interval

By default, the minimum router advertisement interval is 0.75 times the maximum advertisement value. To configure the minimum interval timer used for router discovery advertisement from the circuit VLAN, use the **router-discovery min-advertisement-interval** command. This command defines the minimum interval, in seconds, between sending advertisements. Enter an integer from 0 to 1800.

The default is 0.75 times the max-advertisement-interval. If this value is greater than 0, it must be less than the value specified using the **router-discovery max-advertisement-interval** command.

For example:

```
(config-circuit[VLAN7])# router-discovery min-advertisement-interval
100
```

To reset the minimum router advertisement interval to the default of 0.75 times the maximum advertisement value, enter:

```
(config-circuit[VLAN7])# no router-discovery
min-advertisement-interval
```

Showing Circuits

Use the **show circuits** command to show circuit information. This command provides the following options:

- **show circuits** - Displays all circuit information for circuits that are currently up
- **show circuits all** - Displays all circuit information regardless of circuit state
- **show circuit name** *circuit name* - Displays circuit information for a specific circuit regardless of state

To list all circuits and their interfaces in the Up state, enter:

```
# show circuits
```

To list all circuits and their interfaces regardless of their state, enter:

```
# show circuits all
```

To list an individual circuit, enter:

```
# show circuits name VLAN5
```

Table 1-10 describes the fields in the **show circuits** command output.

Table 1-10 Field Descriptions for the show circuits Command

Field	Description
Circuit Name	The circuit name. The VLAN name appear in uppercase, with no space before the VLAN number.
Circuit State	The state of the circuit. The possible states are as follows: • active-ipEnabled • down-ipEnabled • active-ipDisabled • down-ipDisabled
IP Address	IP interface address.
Interface(s)	The interface associated with the circuit.
Operational Status	The operational status of the interface (Up or Down).

Showing IP Interfaces

Use the **show ip interfaces** command to display configured IP interfaces on the CSS. The display includes the circuit state, IP address, broadcast address, Internet Control Message Protocol (ICMP) settings, and Router Discovery Program (RDP) settings. For example:

```
# show ip interfaces
```

Table 1-11 describes the fields in the **show ip interfaces** command output.

Table 1-11 Field Descriptions for the show ip interfaces Command

Field	Description
Circuit Name	The name of the circuit associated with the IP interface.
State	The state of the IP interface. The possible states are as follows: • **Active (1)** - The interface is up • **Disabled** - The interface is disabled • **NoCircuit** - The interface is waiting for an underlying circuit
IP Address	The IP address assigned to the circuit.
Network Mask	The network mask of the circuit.
Broadcast Address	The broadcast IP address associated with the IP interface. If left at zero, the all-ones host is used for numbered interfaces. 255.255.255.255 is always used for unnumbered interfaces.
Redundancy	Indicates whether the redundancy protocol is running on the interface. The default state is Disabled.
ICMP Redirect	Indicates whether the transmission of Internet Control Message Protocol (ICMP) redirect messages is Enabled or Disabled. The default state is Enabled.
ICMP Unreachable	Indicates whether the transmission of ICMP Destination Unreachable messages is enabled or disabled. The default state is Enabled.
RIP	Indicates whether RIP is Enabled or Disabled.

Configuring RIP for an IP Interface

You can configure Routing Information Protocol (RIP) attributes on each IP interface. To configure RIP parameters and run RIP on an IP interface, use the following routing commands within the specific circuit IP mode. The default mode is to send RIP version 2 (v2) and receive either RIP or RIP2.

The timers used by RIP in the CSS include the following default values. These RIP timer values are not user-configurable in the CSS.

- Transmit (Tx) time that is a random value between 15 and 45 seconds to avoid router synchronization problems
- Route expiration time of 180 seconds (if the CSS loses the link to the next hop router, the route is immediately removed)
- Hold-down time (the amount of time the CSS transmits with an infinite metric) of 120 seconds

This section includes the following topics:

- Enabling RIP on an IP Interface
- Configuring a RIP Default Route
- Configuring a RIP Receive Version
- Configuring RIP Send Version
- Configuring RIP Packet Logging
- Showing RIP Configurations for IP Addresses

Enabling RIP on an IP Interface

To start running RIP on an IP interface, use the **rip** command. For example:

```
(config-circuit-ip[VLAN7-192.168.1.58)# rip
```

To stop running the RIP on the interface, enter:

```
(config-circuit-ip[VLAN7-192.168.1.58])# no rip
```

Configuring a RIP Default Route

By default, the CSS advertises a default route on an IP interface with a metric of 1. To advertise a default route on an IP interface with a specific metric, use the **rip default-route** command. You can also specify an optional metric in the command line. The CSS uses this metric when advertising a route. Enter a number from 1 to 15.

For example:

```
(config-circuit-ip[VLAN7-192.168.1.58])# rip default-route 9
```

Configuring a RIP Receive Version

By default, the interface receives both RIP version 1 and RIP version 2. To specify the RIP version that the interface receive, use the **rip receive** command. The options for this command are as follows:

- **rip receive both** - Receives both RIP version 1 and RIP version 2 (default)
- **rip receive none** - Receives no RIP packets
- **rip receive v1** - Receives RIP version 1 packets only
- **rip receive v2** - Receives RIP version 2 packets only

For example:

```
(config-circuit-ip[VLAN7-192.168.1.58])# rip receive both
```

Configuring RIP Send Version

By default, the interface sends RIP version 2 packets only. To specify the RIP version that the interface transmits, use the **rip send** command. The options for this command are as follows:

- **rip send none** - Sends no RIP packets
- **rip send v1** - Sends RIP version 1 packets only
- **rip send v2** - Sends RIP version 2 packets only (default)

For example:

```
(config-circuit-ip[VLAN7-192.168.1.58])# rip send v1
```

Configuring RIP Packet Logging

By default, CSS of logging received or transmitted RIP packets on the interface is disabled. Use the **rip log** command to enable the CSS to log received or transmitted RIP packets on the interface.

The options for this command are as follows:

- **rip log rx** - CSS logs RIP packets received on the interface
- **rip log tx** - CSS logs RIP packets transmitted on the interface

For example:

```
(config-circuit-ip[VLAN7-192.168.1.58])# rip log rx
```

Showing RIP Configurations for IP Addresses

Use the **show rip** command to show a RIP configuration for one IP address or all IP addresses configured in the CSS. The options for this command are as follows:

- **show rip** - Displays RIP configurations for all interfaces (including the logging of RIP packets)
- **show rip** *ip_address* - Displays a single RIP interface entry
- **show rip globals** - Displays RIP global statistics
- **show rip statistics** - Displays RIP interface statistics for all interfaces
- **show rip statistics** *ip_address* - Displays RIP interface statistics for a specific interface

Table 1-12 describes the fields in the **show rip** command output.

Table 1-12 Field Descriptions for the show rip Command

Field	Description
IP Address	The advertised RIP interface address.
State	The operational state of the RIP interface.
RIP Send	The RIP version that the interface sends. The possible values are as follows: • **none** - Do not send RIP packets • **RIPv1** - Send RIP version 1 packets only • **RIPv2** - Send RIP version 2 packets only (default)
RIP Recv	The RIP version that the interface receives. The possible values are as follows: • **both** - Receiving both version 1 and version 2 (default) • **none** - Receiving no RIP packets • **Ripv1** - Receiving RIP version 1 packets only • **Ripv2** - Receiving RIP version 2 packets only
Default Metric	The default metric used when advertising the RIP interface.
Tx Log	The setting for the logging of RIP packet transmissions (Enabled or Disabled). The default setting is disabled.
Rx Log	The setting for the logging of RIP packet received (Enabled or Disabled). The default setting is disabled.

To display global RIP statistics, enter:

```
# show rip globals
```

Table 1-13 describes the fields in the **show rip globals** command output.

Table 1-13 Field Descriptions for the show rip globals Command

Field	Description
RIP Route Changes	The global number of route changes made to the IP route database by RIP
RIP Query Responses	The global number of query responses sent to RIP query from other systems

To display the RIP interface statistics for all RIP interface entries, enter:

```
# show rip statistics
```

Table 1-14 describes the fields in the **show rip statistics** command output.

Table 1-14 Field Descriptions for the show rip statistics Command

Field	Description
System Route Changes	The global number of route changes made to the IP route database by RIP
System Global Query Responses	The global number of query responses sent to RIP query from other systems
IP Address	The RIP interface IP address
Triggered Updates Sent	The number of triggered RIP updates sent by the interface
Bad Packets Received	The number of bad RIP response packets received by the interface
Bad Routes Received	The number of bad routes in valid RIP packets received by the interface

Configuring the Switched Port Analyzer Feature

Configure the switched port analyzer (SPAN) feature on your CSS to mirror (copy) traffic passing through one CSS port (Fast Ethernet or Gigabit Ethernet) to another designated port of the same type and on the same CSS module for analysis. You can use SPAN for network troubleshooting or tuning using a network analyzer. SPAN is sometimes referred to as *port mirroring* or *port monitoring*.

A SPAN session is the association of a destination port with a source port on the same CSS module. The port that is monitored is called the source SPAN (SSPAN) port. An SSPAN port consists of two components:

- Ingress path - Network traffic entering the CSS. The CSS copies to the monitoring port packets that the SSPAN port receives (SSPAN Rx) from the network.
- Egress path - Network traffic leaving the CSS. The CSS copies to the monitoring port packets that the SSPAN port transmits (SSPAN Tx) to the network.

SPAN can monitor the ingress path, the egress path, or both. You can configure only one SSPAN port in a CSS chassis.

The port that monitors the SSPAN port is called the destination SPAN (DSPAN) port. You can configure only one DSPAN port in a CSS chassis and it must have the following characteristics:

- Same speed as the SSPAN port
- Same media type as the SSPAN port
- Local (physically resides on the same CSS module)

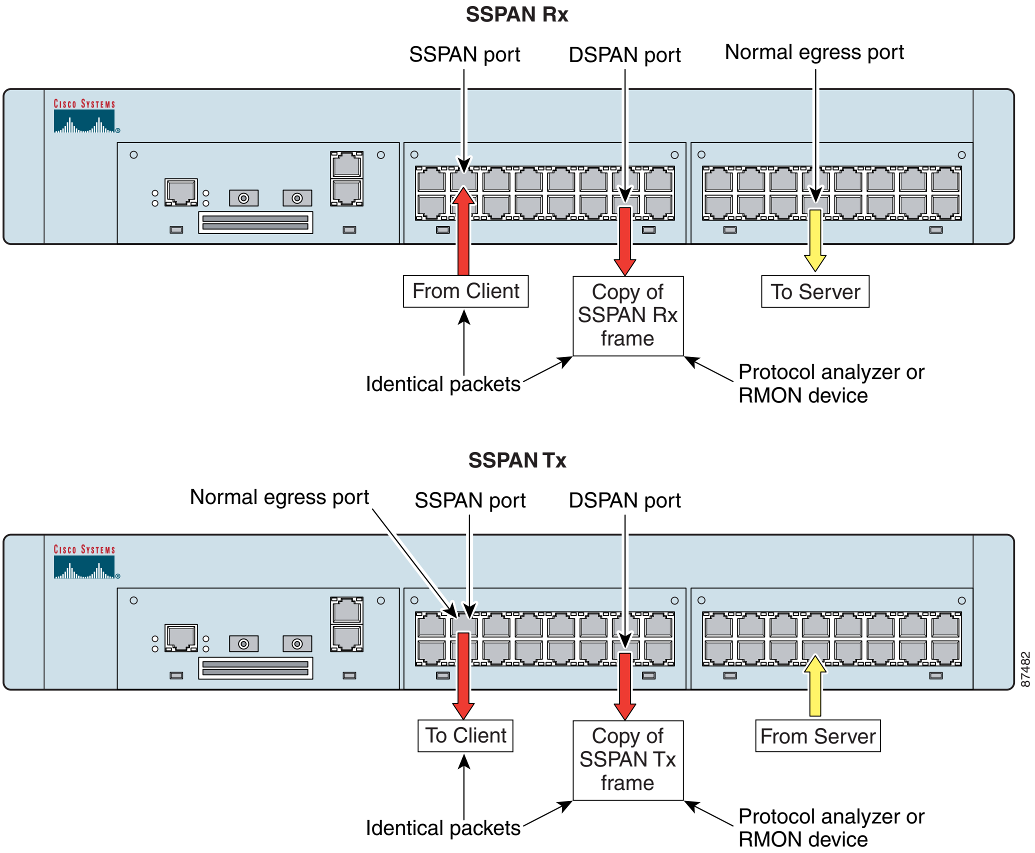
Once you configure a port as a DSPAN port, the CSS removes it from all VLANs and ignores ingress traffic on that port. In addition, the DSPAN port does not participate in STP or routing protocols such as RIP and OSPF.

Traffic copied to the DSPAN port is typically forwarded to a network analyzer, protocol analyzer, or an RMON probe. SPAN allows you to monitor CSS ports without:

- Disconnecting cables
- Requiring multiple analyzers or probes
- Needing hubs or switches

Figure 1-3 shows an example of SPAN connectivity with a protocol analyzer connected to port 2/13 on a CSS. In this example, the CSS copies all packets received or transmitted on Fast Ethernet (FE) port 2/4 (SSPAN port) to FE port 2/13 (DSPAN port). The analyzer connected to DSPAN port 2/13 receives all network traffic that the SSPAN port receives or transmits.

Figure 1-3 Example of SPAN Connectivity



This section describes how to configure SPAN on a CSS. It includes the following topics:

- Configuring SPAN on a CSS
- Verifying the SPAN Configuration on a CSS

Configuring SPAN on a CSS

To configure SPAN on a CSS, use the **setspan** command. This command instructs the CSS to monitor all incoming and/or outgoing traffic on a specified SSPAN port by copying the packets to a specified DSPAN port on the same module in the CSS. This feature is disabled by default.

The syntax of this global configuration mode command is:

setspan src_port *number* **dest_port** *number*
copyBoth|copyTxOnly|copyRxOnly

The options and variables for this command are as follows:

- **src_port** *number* - Source port keyword and number of the SSPAN port (in slot/port format) that you want to monitor. The CSS copies all packets that are received or transmitted on this port to the DSPAN port.
- **dest_port** *number* - Destination port keyword and number of the DSPAN port (in slot/port format) where you want to connect the network analyzer, protocol analyzer, or RMON probe. The CSS copies the packets that flow through the SSPAN port to the DSPAN port that you specify. The DSPAN port must reside on the same module as the SSPAN port.

  Note Once you configure a port as a DSPAN port, the CSS removes it from all VLANs and ignores ingress traffic on that port. In addition, the DSPAN port does not participate in spanning tree protocol (STP) or routing protocols such as RIP and OSPF.

- **copyBoth** - CSS copies to the DSPAN port packets that the SSPAN port transmits to the network (egress traffic) and packets that the SSPAN port receives from the network (ingress traffic).

Note If the combined traffic bandwidth of the ingress and egress traffic of the SSPAN port exceeds the bandwidth of the DSPAN port, the DSPAN port may become oversubscribed.

- **copyTxOnly** - CSS copies to the DSPAN port only those packets that the SSPAN port transmits to the network (egress traffic).
- **copyRxOnly** - CSS copies to the DSPAN port only those packets that the SSPAN port receives from the network (ingress traffic).

For example, to copy all received and transmitted packets on SSPAN port 3 of the I/O module in slot 3 to DSPAN port 12 on the same module, enter:

```
(config)# setspan src_port 3/3 dest_port 3/12 copyBoth
```

To return the SPAN feature to its default state of disabled, use the **no setspan** command. For example, to disable SPAN on the source and destination ports on CSS module 3 in the example above, enter:

```
(config)# no setspan src_port 3/3 dest_port 3/12
```

Verifying the SPAN Configuration on a CSS

To verify the SPAN configuration on a CSS, use the **show setspan** command. Table 1-15 describes the fields in the **show setspan** command output.

Table 1-15 Field Descriptions for the show setspan Command

Field	Description
SPAN Configuration	
Source	Number of the SSPAN port whose traffic you want to monitor.
Destination	Number of the DSPAN port to which the CSS copies the packets flowing through the SSPAN port. Connect the network analyzer or RMON probe to this port.

Table 1-15 Field Descriptions for the show setspan Command (continued)

Field	Description
Direction	Direction of the traffic that you want to monitor at the source port. The direction can be one of the following: • **copyBoth** - The CSS copies packets that are transmitted and received by the SSPAN port to the DSPAN port. • **copyTxOnly** - The CSS copies only packets transmitted (egress traffic) by the SSPAN port to the DSPAN port. • **copyRxOnly** - The CSS copies only packets received (ingress traffic) by the SSPAN port to the DSPAN port.