Configuring Interface Characteristics

This chapter defines the types of interfaces on the Catalyst 3750 switch and describes how to configure them. Unless otherwise noted, the term *switch* refers to a standalone switch and a switch stack.

The chapter has these sections:

Note For complete syntax and usage information for the commands used in this chapter, refer to the switch command reference for this release and the online *Cisco IOS Interface Command Reference, Release 12.2*.

Understanding Interface Types

This section describes the different types of interfaces supported by the switch with references to chapters that contain more detailed information about configuring these interface types. The rest of the chapter describes configuration procedures for physical interface characteristics.

Note The stack ports on the rear of the switch are not Ethernet ports and cannot be configured.

These sections are included:

- EtherChannel Port Groups, page 11-5
- Connecting Interfaces, page 11-5

Port-Based VLANs

A VLAN is a switched network that is logically segmented by function, team, or application, without regard to the physical location of the users. For more information about VLANs, see Chapter 13, "Configuring VLANs." Packets received on a port are forwarded only to ports that belong to the same VLAN as the receiving port. Network devices in different VLANs cannot communicate with one another without a Layer 3 device to route traffic between the VLANs.

VLAN partitions provide hard firewalls for traffic in the VLAN, and each VLAN has its own MAC address table. A VLAN comes into existence when a local port is configured to be associated with the VLAN, when the VLAN Trunking Protocol (VTP) learns of its existence from a neighbor on a trunk, or when a user creates a VLAN. VLANs can be formed with ports across the stack.

To configure normal-range VLANs (VLAN IDs 1 to 1005), use the **vlan** *vlan-id* global configuration command to enter config-vlan mode or the **vlan database** privileged EXEC command to enter VLAN database configuration mode. The VLAN configurations for VLAN IDs 1 to 1005 are saved in the VLAN database, which is downloaded to all switches in a stack. All switches in the stack build the same VLAN database. To configure extended-range VLANs (VLAN IDs 1006 to 4094), you must use config-vlan mode with VTP mode set to transparent. Extended-range VLANs are not added to the VLAN database. When VTP mode is transparent, the VTP and VLAN configuration is saved in the switch running configuration, and you can save it in the switch startup configuration file by entering the **copy running-config startup-config** privileged EXEC command. The running configuration and the saved configuration are the same for all switches in a stack.

Add ports to a VLAN by using the **switchport** interface configuration commands:

- Identify the interface.
- For a trunk port, set trunk characteristics, and if desired, define the VLANs to which it can belong.
- For an access port, set and define the VLAN to which it belongs.

Switch Ports

Switch ports are Layer 2-only interfaces associated with a physical port. Switch ports belong to one or more VLANs. A switch port can be an access port or a trunk port. You can configure a port as an access port or trunk port or let the Dynamic Trunking Protocol (DTP) operate on a per-port basis to set the switchport mode by negotiating with the port on the other end of the link. Switch ports are used for managing the physical interface and associated Layer 2 protocols and do not handle routing or bridging.

Configure switch ports by using the **switchport** interface configuration commands. Use the switchport command with no keywords to put an interface that is in Layer 3 mode into Layer 2 mode.

Note When you put an interface that is in Layer 3 mode into Layer 2 mode, the previous configuration information related to the affected interface might be lost, and the interface is returned to its default configuration.

For detailed information about configuring access port and trunk port characteristics, see Chapter 13, "Configuring VLANs."

Access Ports

An access port belongs to and carries the traffic of only one VLAN (unless it is configured as a voice VLAN port). Traffic is received and sent in native formats with no VLAN tagging. Traffic arriving on an access port is assumed to belong to the VLAN assigned to the port. If an access port receives a tagged packet (Inter-Switch Link [ISL] or 802.1Q tagged), the packet is dropped, and the source address is not learned.

Two types of access ports are supported:

- Static access ports are manually assigned to a VLAN.
- VLAN membership of dynamic access ports is learned through incoming packets. By default, a dynamic access port is a member of no VLAN, and forwarding to and from the port is enabled only when the VLAN membership of the port is discovered. Dynamic access ports on the switch are assigned to a VLAN by a VLAN Membership Policy Server (VMPS). The VMPS can be a Catalyst 6500 series switch; the Catalyst 3750 switch cannot be a VMPS server.

You can also configure an access port with an attached Cisco IP Phone to use one VLAN for voice traffic and another VLAN for data traffic from a device attached to the phone. For more information about voice VLAN ports, see Chapter 16, "Configuring Voice VLAN."

Trunk Ports

A trunk port carries the traffic of multiple VLANs and by default is a member of all VLANs in the VLAN database. Two types of trunk ports are supported:

- In an ISL trunk port, all received packets are expected to be encapsulated with an ISL header, and all transmitted packets are sent with an ISL header. Native (non-tagged) frames received from an ISL trunk port are dropped.
- An IEEE 802.1Q trunk port supports simultaneous tagged and untagged traffic. An 802.1Q trunk port is assigned a default Port VLAN ID (PVID), and all untagged traffic travels on the port default PVID. All untagged traffic and tagged traffic with a NULL VLAN ID are assumed to belong to the port default PVID. A packet with a VLAN ID equal to the outgoing port default PVID is sent untagged. All other traffic is sent with a VLAN tag.

Although by default, a trunk port is a member of every VLAN known to the VTP, you can limit VLAN membership by configuring an allowed list of VLANs for each trunk port. The list of allowed VLANs does not affect any other port but the associated trunk port. By default, all possible VLANs (VLAN ID 1 to 4094) are in the allowed list. A trunk port can only become a member of a VLAN if VTP knows of the VLAN and the VLAN is in the enabled state. If VTP learns of a new, enabled VLAN and the VLAN is in the allowed list for a trunk port, the trunk port automatically becomes a member of that VLAN and traffic is forwarded to and from the trunk port for that VLAN. If VTP learns of a new, enabled VLAN that is not in the allowed list for a trunk port, the port does not become a member of the VLAN, and no traffic for the VLAN is forwarded to or from the port.

For more information about trunk ports, see Chapter 13, "Configuring VLANs."

Routed Ports

A routed port is a physical port that acts like a port on a router; it does not have to be connected to a router. A routed port is not associated with a particular VLAN, as is an access port. A routed port behaves like a regular router interface, except that it does not support VLAN subinterfaces. Routed ports can be configured with a Layer 3 routing protocol. A routed port is a Layer 3 interface only and does not support Layer 2 protocols, such as DTP and STP.

Configure routed ports by putting the interface into Layer 3 mode with the **no switchport** interface configuration command. Then assign an IP address to the port, enable routing, and assign routing protocol characteristics by using the **ip routing** and **router** *protocol* global configuration commands.

> **Note**   Entering a **no switchport** interface configuration command shuts down the interface and then re-enables it, which might generate messages on the device to which the interface is connected. When you put an interface that is in Layer 2 mode into Layer 3 mode, the previous configuration information related to the affected interface might be lost.

The number of routed ports that you can configure is not limited by software. However, the interrelationship between this number and the number of other features being configured might impact CPU performance because of hardware limitations. See the "Configuring Layer 3 Interfaces" section on page 11-21 for information about what happens when hardware resource limitations are reached.

For more information about IP unicast and multicast routing and routing protocols, see Chapter 34, "Configuring IP Unicast Routing" and Chapter 36, "Configuring IP Multicast Routing."

> **Note**   The standard multilayer image (SMI) supports static routing and the Routing Information Protocol (RIP). For full Layer 3 routing or for fallback bridging, you must have the enhanced multilayer image (EMI) installed on the stack master.

10-Gigabit Ethernet Interfaces

The Catalyst 3750G-16TD switch has one 10-Gigabit Ethernet interface. The switch uses a 10-Gigabit Ethernet XENPAK module to establish connections to networks.

The 10-Gigabit Ethernet interface only operates in full-duplex mode. The interface can be configured as a switched port or a routed port. A stack of Catalyst 3750 switches can have up to nine 10-Gigabit Ethernet interfaces. A cross-stack EtherChannel supports up to two 10-Gigabit module ports.

For the latest information about XENPAK modules supported by the switch, refer to the release notes.

For more information about XENPAK modules, refer to your XENPAK module documentation.

> **Note**   The 10-Gigabit Ethernet module ports are referred to as 10-Gigabit Ethernet XENPAK modules in the hardware installation guide.

Switch Virtual Interfaces

A switch virtual interface (SVI) represents a VLAN of switch ports as one interface to the routing or bridging function in the system. Only one SVI can be associated with a VLAN, but you need to configure an SVI for a VLAN only when you wish to route between VLANs, to fallback-bridge nonroutable protocols between VLANs, or to provide IP host connectivity to the switch. By default, an SVI is created for the default VLAN (VLAN 1) to permit remote switch administration. Additional SVIs must be explicitly configured. SVIs provide IP host connectivity only to the system; in Layer 3 mode, you can configure routing across SVIs.

Although the switch stack supports a total or 1005 VLANs (and SVIs), the interrelationship between the number of SVIs and routed ports and the number of other features being configured might impact CPU performance because of hardware limitations. See the "Configuring Layer 3 Interfaces" section on page 11-21 for information about what happens when hardware resource limitations are reached.

SVIs are created the first time that you enter the **vlan** interface configuration command for a VLAN interface. The VLAN corresponds to the VLAN tag associated with data frames on an ISL or 802.1Q encapsulated trunk or the VLAN ID configured for an access port. Configure a VLAN interface for each VLAN for which you want to route traffic, and assign it an IP address. For more information, see the "Manually Assigning IP Information" section on page 4-10.

> **Note** When you create an SVI, it does not become active until it is associated with a physical port.

SVIs support routing protocols and bridging configurations. For more information about configuring IP routing, see Chapter 34, "Configuring IP Unicast Routing," Chapter 36, "Configuring IP Multicast Routing,"and Chapter 38, "Configuring Fallback Bridging."

> **Note** The SMI supports static routing and RIP; for more advanced routing or for fallback bridging, you must have the EMI installed on the stack master.

EtherChannel Port Groups

EtherChannel port groups provide the ability to treat multiple switch ports as one switch port. These port groups act as a single logical port for high-bandwidth connections between switches or between switches and servers. An EtherChannel balances the traffic load across the links in the channel. If a link within the EtherChannel fails, traffic previously carried over the failed link changes to the remaining links. You can group multiple trunk ports into one logical trunk port, group multiple access ports into one logical access port, or group multiple routed ports into one logical routed port. Most protocols operate over either single ports or aggregated switch ports and do not recognize the physical ports within the port group. Exceptions are the DTP, the Cisco Discovery Protocol (CDP), and the Port Aggregation Protocol (PAgP), which operate only on physical ports.

When you configure an EtherChannel, you create a port-channel logical interface and assign an interface to the EtherChannel. For Layer 3 interfaces, you manually create the logical interface by using the **interface port-channel** global configuration command. Then you manually assign an interface to the EtherChannel by using the **channel-group** interface configuration command. For Layer 2 interfaces, use the **channel-group** interface configuration command to dynamically create the port-channel logical interface. This command binds the physical and logical ports together. For more information, see Chapter 33, "Configuring EtherChannels."

Connecting Interfaces

Devices within a single VLAN can communicate directly through any switch. Ports in different VLANs cannot exchange data without going through a routing device. With a standard Layer 2 switch, ports in different VLANs have to exchange information through a router. In the configuration shown in Figure 11-1, when Host A in VLAN 20 sends data to Host B in VLAN 30, it must go from Host A to the switch, to the router, back to the switch, and then to Host B.

Figure 11-1   Connecting VLANs with Layer 2 Switches
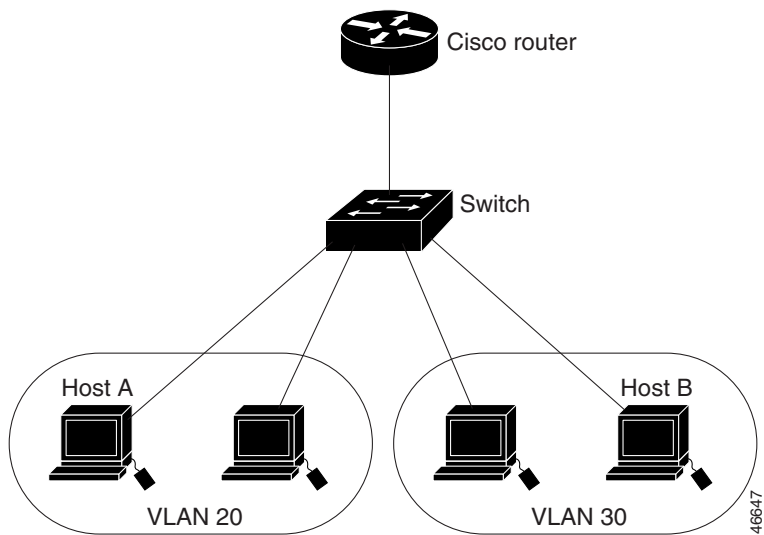
By using the switch with routing enabled, when you configure VLAN 20 and VLAN 30 each with an SVI to which an IP address is assigned, packets can be sent from Host A to Host B directly through the switch with no need for an external router (Figure 11-2).

Figure 11-2   Connecting VLANs with the Catalyst 3750 Switch
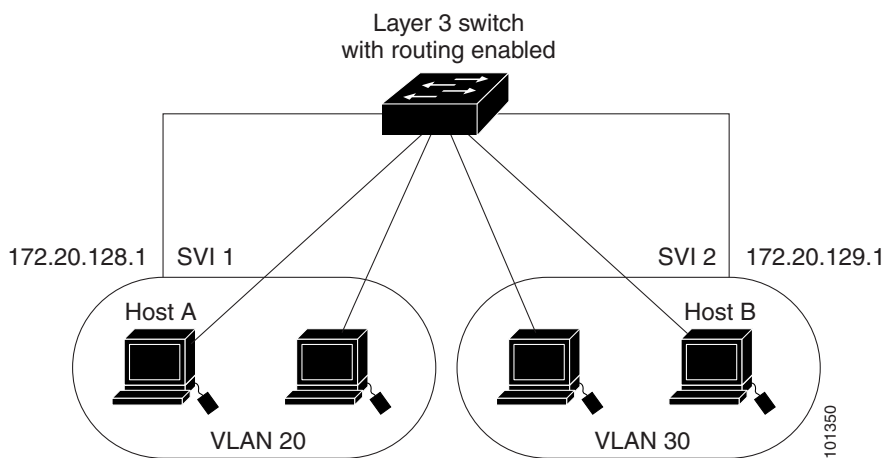
When the EMI is running on the stack master, the switch supports two methods of forwarding traffic between interfaces: routing and fallback bridging. If the SMI is on the stack master, only basic routing (static routing and RIP) is supported. Whenever possible, to maintain high performance, forwarding is done by the switch hardware. However, only IP version 4 packets with Ethernet II encapsulation can be routed in hardware. Non-IP traffic and traffic with other encapsulation methods can be fallback-bridged by hardware.

- The routing function can be enabled on all SVIs and routed ports. The switch routes only IP traffic. When IP routing protocol parameters and address configuration are added to an SVI or routed port, any IP traffic received from these ports is routed. For more information, see Chapter 34, "Configuring IP Unicast Routing," Chapter 36, "Configuring IP Multicast Routing," and Chapter 37, "Configuring MSDP."

- Fallback bridging forwards traffic that the switch does not route or traffic belonging to a nonroutable protocol, such as DECnet. Fallback bridging connects multiple VLANs into one bridge domain by bridging between two or more SVIs or routed ports. When configuring fallback bridging, you assign SVIs or routed ports to bridge groups with each SVI or routed port assigned to only one bridge group. All interfaces in the same group belong to the same bridge domain. For more information, see Chapter 38, "Configuring Fallback Bridging."

Using Interface Configuration Mode

The switch supports these interface types:

- Physical ports—including switch ports and routed ports
- VLANs—switch virtual interfaces
- Port-channels—EtherChannel of interfaces

You can also configure a range of interfaces (see the "Configuring a Range of Interfaces" section on page 11-9).

To configure a physical interface (port), enter interface configuration mode, and specify the interface type, stack member number, module number, and switch port number.

- Type—Fast Ethernet (fastethernet or fa) for 10/100 Mbps Ethernet or Gigabit Ethernet (gigabitethernet or gi) for 10/100/1000 Mbps Ethernet ports, or 10-Gigabit Ethernet (tengigabitethernet or te) for 10,000 Mbps, or small form-factor pluggable (SFP) Gigabit Ethernet interfaces.

- Stack member number—The number that identifies the switch within the stack. The switch number range is 1 to 9 and is assigned the first time the switch initializes. The default switch number, before it is integrated into a switch stack, is 1. When a switch has been assigned a stack member number, it keeps that number until another is assigned to it.

  You can use the switch port LEDs in Stack mode to identify the stack member number of a switch.

  For information about stack member numbers, see the "Stack Member Numbers" section on page 5-6.

- Module number—The module or slot number on the switch (always 0 on the Catalyst 3750 switch).

- Port number—The interface number on the switch. The port numbers always begin at 1, starting at the left when facing the front of the switch, for example, fastethernet 1/0/1, fastethernet 1/0/2. If there is more than one interface type (for example, 10/100 ports and Gigabit Ethernet ports), the port numbers start again from 1with the second interface type: gigabitethernet1/0/1, gigabitethernet 1/0/2.

You can identify physical interfaces by physically checking the interface location on the switch. You can also use the Cisco IOS **show** privileged EXEC commands to display information about a specific interface or all the interfaces on the switch. The remainder of this chapter primarily provides physical interface configuration procedures.

These are examples of identifying interfaces:

- To configure 10/100/1000 port 4 on a standalone switch, enter this command:

  ```
  Switch(config)# interface gigabitethernet1/0/4
  ```

- To configure 10/100 port 4 on stack member 3, enter this command:

  ```
  Switch(config)# interface fastethernet3/0/4
  ```

- To configure 10-Gigabit module port 1 on a standalone switch, enter this command:

  ```
  Switch(config)# interface tengigabitethernet1/0/1
  ```

- To configure 10-Gigabit module port on stack member 3, enter this command:

  ```
  Switch(config)# interface tengigabitethernet3/0/1
  ```

If the switch has SFP modules, the numbering of these ports depends on the type of other interfaces on the switch. If the port type changes from Fast Ethernet to Gigabit Ethernet (SFP), the port numbers begin again from 1; if the port type remains Gigabit Ethernet, the port numbers continue consecutively.

- To configure the first SFP port on stack member 1 with 24 10/100/1000 ports, enter this command:

  ```
  Switch(config)# interface gigabitethernet1/0/25
  ```

- To configure the first SFP port on stack member 1 with 24 10/100 ports, enter this command:

  ```
  Switch(config)# interface gigabitethernet1/0/1
  ```

Procedures for Configuring Interfaces

These general instructions apply to all interface configuration processes.

Step 1   Enter the **configure terminal** command at the privileged EXEC prompt:

```
Switch# configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
Switch(config)#
```

Step 2   Enter the **interface** global configuration command. Identify the interface type, the switch number, and the number of the connector. In this example, Gigabit Ethernet port 1 on switch 1 is selected:

```
Switch(config)# interface gigabitethernet1/0/1
Switch(config-if)#
```

> **Note**   You do not need to add a space between the interface type and interface number. For example, in the preceding line, you can specify either **gigabitethernet 1/0/1**, **gigabitethernet1/0/1**, **gi 1/0/1**, or **gi1/0/1**.

Step 3   Follow each **interface** command with the interface configuration commands that the interface requires. The commands that you enter define the protocols and applications that will run on the interface. The commands are collected and applied to the interface when you enter another interface command or enter **end** to return to privileged EXEC mode.

You can also configure a range of interfaces by using the **interface range** or **interface range macro** global configuration commands. Interfaces configured in a range must be the same type and must be configured with the same feature options.

Step 4   After you configure an interface, verify its status by using the **show** privileged EXEC commands listed in the "Monitoring and Maintaining the Interfaces" section on page 11-24.

Enter the **show interfaces** privileged EXEC command to see a list of all interfaces on or configured for the switch. A report is provided for each interface that the device supports or for the specified interface.

Configuring a Range of Interfaces

You can use the **interface range** global configuration command to configure multiple interfaces with the same configuration parameters. When you enter the interface range configuration mode, all command parameters that you enter are attributed to all interfaces within that range until you exit this mode.

Beginning in privileged EXEC mode, follow these steps to configure a range of interfaces with the same parameters:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**interface range** {*port-range* | **macro** *macro_name*}	Enter interface range configuration mode by entering the range of interfaces (VLANs or physical ports) to be configured. • You can use the **interface range** command to configure up to five port ranges or a previously defined macro. • The **macro** variable is explained in the "Configuring and Using Interface Range Macros" section on page 11-10. • In a comma-separated *port-range*, you must enter the interface type for each entry and enter spaces before and after the comma. • In a hyphen-separated *port-range*, you do not need to re-enter the interface type, but you must enter a space before the hyphen.
Step 3		You can now use the normal configuration commands to apply the configuration parameters to all interfaces in the range.
Step 4	**end**	Return to privileged EXEC mode.
Step 5	**show interfaces** [*interface-id*]	Verify the configuration of the interfaces in the range.
Step 6	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

When using the **interface range** global configuration command, note these guidelines:

- Valid entries for *port-range*:
    - **vlan** *vlan-ID - vlan-ID*, where the VLAN ID is 1 to 4094
    - **fastethernet** stack member/module/{first *port*} - {*last port*}, where the module is always 0
    - **gigabitethernet** stack member/module/{*first port*} - {*last port*}, where the module is always 0

– **port-channel** *port-channel-number - port-channel-number*, where the *port-channel-number* is 1 to 12

> ✎
>
> **Note**   When you use the **interface range** command with port channels, the first and last port channel number must be active port channels.

- You must add a space between the first interface number and the hyphen when using the **interface range** command. For example, the command **interface range ethernet 1/0/1 - 4** is a valid range; the command **interface range ethernet 1/0/1-4** is not a valid range.

- The **interface range** command only works with VLAN interfaces that have been configured with the **interface vlan** command. The **show running-config** privileged EXEC command displays the configured VLAN interfaces. VLAN interfaces not displayed by the **show running-config** command cannot be used with the **interface range** command.

- All interfaces defined as in a range must be the same type (all Fast Ethernet ports, all Gigabit Ethernet ports, all EtherChannel ports, or all VLANs), but you can enter multiple ranges in a command.

This example shows how to use the **interface range** global configuration command to set the speed on ports 1 to 4 on switch 1 to 100 Mbps:

```
Switch# configure terminal
Switch(config)# interface range tethernet1/0/1 - 4
Switch(config-if-range)# speed 100
```

This example shows how to use a comma to add different interface type strings to the range to enable Fast Ethernet ports 1 to 3 on switch 1 and Gigabit Ethernet ports 1 and 2 on switch 2 to receive flow control pause frames:

```
Switch# configure terminal
Switch(config)# interface range fastethernet1/0/1 - 3 , gigabitethernet2/0/1 - 2
Switch(config-if-range)# flowcontrol receive on
```

If you enter multiple configuration commands while you are in interface range mode, each command is executed as it is entered. The commands are not batched together and executed after you exit interface range mode. If you exit interface range configuration mode while the commands are being executed, some commands might not be executed on all interfaces in the range. Wait until the command prompt reappears before exiting interface range configuration mode.

Configuring and Using Interface Range Macros

You can create an interface range macro to automatically select a range of interfaces for configuration. Before you can use the **macro** keyword in the **interface range macro** global configuration command string, you must use the **define interface-range** global configuration command to define the macro.

Beginning in privileged EXEC mode, follow these steps to define an interface range macro:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**define interface-range** *macro_name* *interface-range*	Define the interface-range macro, and save it in NVRAM. • The *macro_name* is a 32-character maximum character string. • A macro can contain up to five comma-separated interface ranges. • Each *interface-range* must consist of the same port type.
Step 3	**interface range macro** *macro_name*	Select the interface range to be configured using the values saved in the interface-range macro called *macro_name*. You can now use the normal configuration commands to apply the configuration to all interfaces in the defined macro.
Step 4	**end**	Return to privileged EXEC mode.
Step 5	**show running-config	include define**	Show the defined interface range macro configuration.
Step 6	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

Use the **no define interface-range** *macro_name* global configuration command to delete a macro.

When using the **define interface-range** global configuration command, note these guidelines:

- Valid entries for *interface-range*:
  - **vlan** *vlan-ID - vlan-ID*, where the VLAN ID is 1 to 4094
  - **fastethernet** stack member/module/{first *port*} - {*last port*}, where the module is always 0
  - **gigabitethernet** stack member/module/{*first port*} - {*last port*}, where the module is always 0
  - **port-channel** *port-channel-number - port-channel-number*, where the *port-channel-number* is 1 to 12.

> ✎
>
> **Note**   When you use the interface ranges with port channels, the first and last port channel number must be active port channels.

- You must add a space between the first interface number and the hyphen when entering an *interface-range*. For example, **ethernet 1/0/1 - 4** is a valid range; **ethernet 1/0/1-4** is not a valid range.

- The VLAN interfaces must have been configured with the **interface vlan** command. The **show running-config** privileged EXEC command displays the configured VLAN interfaces. VLAN interfaces not displayed by the **show running-config** command cannot be used as *interface-ranges*.

- All interfaces defined as in a range must be the same type (all Fast Ethernet ports, all Gigabit Ethernet ports, all EtherChannel ports, or all VLANs), but you can combine multiple interface types in a macro.

This example shows how to define an interface-range named *enet_list* to include ports 1 and 2 on switch 1 and to verify the macro configuration:

```
Switch# configure terminal
Switch(config)# define interface-range enet_list gigabitethernet1/0/1 - 2
Switch(config)# end
Switch# show running-config | include define
define interface-range enet_list GigabitEthernet1/0/1 - 2
```

This example shows how to create a multiple-interface macro named *macro1*:

```
Switch# configure terminal
Switch(config)# define interface-range macro1 fastethernet1/0/1 - 2, gigabitethernet1/0/1
- 2
Switch(config)# end
```

This example shows how to enter interface range configuration mode for the interface-range macro *enet_list*:

```
Switch# configure terminal
Switch(config)# interface range macro enet_list
Switch(config-if-range)#
```

This example shows how to delete the interface-range macro *enet_list* and to verify that it was deleted.

```
Switch# configure terminal
Switch(config)# no define interface-range enet_list
Switch(config)# end
Switch# show run | include define
Switch#
```

Configuring Ethernet Interfaces

These sections describe the default interface configuration and the optional features that you can configure on most physical interfaces:

- Default Ethernet Interface Configuration, page 11-12
- Configuration Guidelines for 10-Gigabit Ethernet Interfaces, page 11-14
- Configuring Interface Speed and Duplex Mode, page 11-14
- Configuring IEEE 802.3z Flow Control, page 11-17
- Configuring Auto-MDIX on an Interface, page 11-18
- Configuring Power over Ethernet on an Interface, page 11-19
- Adding a Description for an Interface, page 11-20

Default Ethernet Interface Configuration

Table 11-1 shows the Ethernet interface default configuration, including some features that apply only to Layer 2 interfaces. For more details on the VLAN parameters listed in the table, see Chapter 13, "Configuring VLANs." For details on controlling traffic to the port, see Chapter 24, "Configuring Port-Based Traffic Control."

Note   To configure Layer 2 parameters, if the interface is in Layer 3 mode, you must enter the **switchport** interface configuration command without any parameters to put the interface into Layer 2 mode. This shuts down the interface and then re-enables it, which might generate messages on the device to which the interface is connected. When you put an interface that is in Layer 3 mode into Layer 2 mode, the previous configuration information related to the affected interface might be lost, and the interface is returned to its default configuration.

Table 11-1   Default Layer 2 Ethernet Interface Configuration

Feature	Default Setting
Operating mode	Layer 2 or switching mode (**switchport** command).
Allowed VLAN range	VLANs 1 – 4094.
Default VLAN (for access ports)	VLAN 1 (Layer 2 interfaces only).
Native VLAN (for 802.1Q trunks)	VLAN 1 (Layer 2 interfaces only).
VLAN trunking	Switchport mode dynamic auto (supports DTP) (Layer 2 interfaces only).
Port enable state	All ports are enabled.
Port description	None defined.
Speed	Autonegotiate. (Not supported on the 10-Gigabit interfaces.)
Duplex mode	Autonegotiate. (Not supported on the 10-Gigabit interfaces.)
Flow control	Flow control is set to **receive**: **off**. It is always off for sent packets.
EtherChannel (PAgP)	Disabled on all Ethernet ports. See Chapter 33, "Configuring EtherChannels."
Port blocking (unknown multicast and unknown unicast traffic)	Disabled (not blocked) (Layer 2 interfaces only). See the "Configuring Port Blocking" section on page 24-6.
Broadcast, multicast, and unicast storm control	Disabled. See the "Default Storm Control Configuration" section on page 24-3.
Protected port	Disabled (Layer 2 interfaces only). See the "Configuring Protected Ports" section on page 24-5.
Port security	Disabled (Layer 2 interfaces only). See the "Default Port Security Configuration" section on page 24-9.
Port Fast	Disabled.
Auto-MDIX	Enabled. **Note** The switch might not support a pre-standard powered device—such as Cisco IP phones and access points that do not fully support IEEE 802.3af—if that powered device is connected to the switch through a crossover cable. This is regardless of whether Auto-MIDX is enabled on the switch port.

Configuration Guidelines for 10-Gigabit Ethernet Interfaces

Follow these guidelines to avoid configuration problems:

- The speed and duplex features are not supported.

- The 10-Gigabit interfaces do not support these QoS features:

  – Policing

  – Auto-QoS for VoIP with Cisco IP Phones

  – Servicing the egress queues by using shaped round robin (SRR) weights

  – Limiting the bandwidth on an egress interface

- If a 10-Gigabit module port is configured as a SPAN or RSPAN destination port, its link rate decreases.

- Support for a cross-stack EtherChannel with up to two 10-Gigabit module ports.

Configuring Interface Speed and Duplex Mode

Ethernet interfaces on the switch operate at 10, 100, or 1000 Mbps, or 10,000 Mbps and in either full- or half-duplex mode. In full-duplex mode, two stations can send and receive traffic at the same time. Normally, 10-Mbps ports operate in half-duplex mode, which means that stations can either receive or send traffic.

Switch models include combinations of Fast Ethernet (10/100-Mbps) ports or Gigabit Ethernet (10/100/1000-Mbps) ports, 10-Gigabit module ports, and small form-factor pluggable (SFP) module slots supporting Gigabit SFP modules.

- You cannot configure speed or duplex mode on 10-Gigabit module ports; these ports operate only at 10,000 Mbps and in full duplex mode.

- You can configure interface speed on Fast Ethernet (10/100-Mbps) and Gigabit Ethernet (10/100/1000-Mbps) ports. You can configure duplex mode to full, half, or autonegotiate on Fast Ethernet interfaces. You can configure Gigabit Ethernet ports to full-duplex mode or to autonegotiate; you cannot configure half-duplex mode on Gigabit Ethernet ports.

- You cannot configure speed on SFP module ports, but you can configure speed to not negotiate (**nonegotiate**) if connected to a device that does not support autonegotiation. However, when a 1000BASE-T SFP module is in the SFP module port, you can configure speed as 10, 100, or 1000 Mbps, or auto.

- You cannot configure duplex mode on SFP module ports unless a 1000BASE-T SFP module or a 100BASE-FX MMF SFP module is in the port. All other SFP modules operate only in full-duplex mode.

  – When a 1000BASE-T SFP module is in the SFP module port, you can configure duplex mode to **auto** or **full**.

  – When a 100BASE-FX SFP module is in the SFP module port, you can configure duplex mode to **half** or **full**.

✎

Note   Half-duplex mode is supported on Gigabit Ethernet interfaces; however you cannot configure these interfaces to operate in half-duplex mode.

These sections describe how to configure the interface speed and duplex mode:

Configuration Guidelines

When configuring an interface speed and duplex mode, note these guidelines:

- If both ends of the line support autonegotiation, we highly recommend the default setting of **auto** negotiation.

- If one interface supports autonegotiation and the other end does not, configure duplex and speed on both interfaces; do not use the **auto** setting on the supported side.

- You cannot configure duplex mode on SFP module ports; they operate in full-duplex mode. However, when a 1000BASE-T SFP module is inserted in an SFP module port, you can configure the duplex mode to **full** or **auto** and half-duplex mode is supported with the auto configuration. When a 100BASE-FX SFP module is in the SFP module port, you can configure duplex mode to **half** or **full**. Although the **auto** keyword is available, it puts the interface in half-duplex mode (the default for this SFP module) because the 100BASE-FX SFP module does not support autonegotiation.

- You cannot configure speed on SFP module ports, except to **nonegotiate**. However, when a 1000BASE-T SFP module is in the SFP module port, the speed can be configured to **10**, **100**, **1000**, or **auto**, but not **nonegotiate**.

- When STP is enabled and a port is reconfigured, the switch can take up to 30 seconds to check for loops. The port LED is amber while STP reconfigures.

⚠

Caution   Changing the interface speed and duplex mode configuration might shut down and re-enable the interface during the reconfiguration.

Setting the Interface Speed and Duplex Parameters

Beginning in privileged EXEC mode, follow these steps to set the speed and duplex mode for a physical interface:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**interface** *interface-id*	Enter interface configuration mode and the physical interface identification.

	Command	Purpose
Step 3	**speed** {**10** | **100** | **1000** | **auto** | **nonegotiate**}	Enter the appropriate speed parameter for the interface:
		• Enter **10**, **100**, or **1000** to set a specific speed for the interface. The **1000** keyword is available only for 10/100/1000 Mbps ports or SFP module ports with a 1000BASE-T SFP module.
		• Enter **auto** to enable the interface to autonegotiate speed with the device connected to the interface.
		• The **nonegotiate** keyword is available only for SFP module ports. SFP module ports operate only at 1000 Mbps but can be configured to not negotiate if connected to a device that does not support autonegotiation.
		This command is not available on a 10-Gigabit Ethernet interface.
		Note When a 1000BASE-T SFP module is in the SFP module port, the speed can be configured to **10**, **100**, **1000**, or **auto**, but not **nonegotiate**.
Step 4	**duplex** {**auto** | **full** | **half**}	Enter the duplex parameter for the interface.
		Enable half-duplex mode (for interfaces operating only at 10 or 100 Mbps). You cannot configure half-duplex mode for interfaces operating at 1000 Mbps.
		This command is not available on a 10-Gigabit Ethernet interface.
		This command is not available on SFP module ports with these exceptions:
		• If a 1000BASE-T SFP module is inserted, you can configure duplex to **auto** or **full**.
		• If a 100BASE-FX SFP module is inserted, you can configure duplex to **full** or **half**. Although the **auto** keyword is available, it puts the interface in half-duplex mode (the default)
Step 5	**end**	Return to privileged EXEC mode.
Step 6	**show interfaces** *interface-id*	Display the interface speed and duplex mode configuration.
Step 7	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

Use the **no speed** and **no duplex** interface configuration commands to return the interface to the default speed and duplex settings (autonegotiate). To return all interface settings to the defaults, use the **default interface** *interface-id* interface configuration command.

This example shows how to set the interface speed to 10 Mbps and the duplex mode to half on a 10/100 Mbps port:

```
Switch# configure terminal
Switch(config)# interface fasttethernet1/0/3
Switch(config-if)# speed 10
Switch(config-if)# duplex half
```

This example shows how to set the interface speed to 100 Mbps on a 10/100/1000 Mbps port:

```
Switch# configure terminal
Switch(config)# interface gigabitethernet1/0/2
Switch(config-if)# speed 100
```

Configuring IEEE 802.3z Flow Control

Flow control enables connected Ethernet ports to control traffic rates during congestion by allowing congested nodes to pause link operation at the other end. If one port experiences congestion and cannot receive any more traffic, it notifies the other port to stop sending until the condition clears by sending a pause frame. Upon receipt of a pause frame, the sending device stops sending any data packets, which prevents any loss of data packets during the congestion period.

> **Note** Catalyst 3750 ports are capable of receiving, but not sending, pause frames.

You use the **flowcontrol** interface configuration command to set the interface's ability to **receive** pause frames to **on**, **off**, or **desired**. The default state is **off**.

When set to **desired**, an interface can operate with an attached device that is required to send flow-control packets or with an attached device that is not required to but can send flow-control packets.

These rules apply to flow control settings on the device:

- **receive on** (or **desired**): The port cannot send pause frames but can operate with an attached device that is required to or can send pause frames; the port can receive pause frames.

- **receive off**: Flow control does not operate in either direction. In case of congestion, no indication is given to the link partner, and no pause frames are sent or received by either device.

> **Note** For details on the command settings and the resulting flow control resolution on local and remote ports, refer to the **flowcontrol** interface configuration command in the command reference for this release.

Beginning in privileged EXEC mode, follow these steps to configure flow control on an interface:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode
Step 2	**interface** *interface-id*	Enter interface configuration mode and the physical interface to be configured.
Step 3	**flowcontrol** {**receive**} {**on**	**off**	**desired**}	Configure the flow control mode for the port.
Step 4	**end**	Return to privileged EXEC mode.
Step 5	**show interfaces** *interface-id*	Verify the interface flow control settings.
Step 6	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To disable flow control, use the **flowcontrol receive off** interface configuration command.

This example shows how to turn on flow control on a port:

```
Switch# configure terminal
Switch(config)# interface gigabitethernet1/0/1
Switch(config-if)# flowcontrol receive on
Switch(config-if)# end
```

Configuring Auto-MDIX on an Interface

When automatic medium-dependent interface crossover (Auto-MDIX) is enabled on an interface, the interface automatically detects the required cable connection type (straight through or crossover) and configures the connection appropriately. When connecting switches without the Auto-MDIX feature, you must use straight-through cables to connect to devices such as servers, workstations, or routers and crossover cables to connect to other switches or repeaters. With Auto-MDIX enabled, you can use either type of cable to connect to other devices, and the interface automatically corrects for any incorrect cabling. For more information about cabling requirements, refer to the hardware installation guide.

Auto-MDIX is enabled by default. When you enable Auto-MDIX, you must also set the speed and duplex on the interface to **auto** in order for the feature to operate correctly. Auto-MDIX is supported on all 10/100 and 10/100/1000 Mbps interfaces and on 10/100/1000 BASE-T/TX SFP interfaces. It is not supported on 1000 BASE-SX or -LX SFP interfaces.

Table 11-2 shows the link states that results from Auto-MDIX settings and correct and incorrect cabling.

Table 11-2   Link Conditions and Auto-MDIX Settings

Local Side Auto-MDIX	Remote Side Auto-MDIX	With Correct Cabling	With Incorrect Cabling
On	On	Link up	Link up
On	Off	Link up	Link up
Off	On	Link up	Link up
Off	Off	Link up	Link down

Beginning in privileged EXEC mode, follow these steps to configure Auto-MDIX on an interface:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode
Step 2	**interface** *interface-id*	Enter interface configuration mode for the physical interface to be configured.
Step 3	**speed auto**	Configure the interface to autonegotiate speed with the connected device.
Step 4	**duplex auto**	Configure the interface to autonegotiate duplex mode with the connected device.
Step 5	**mdix auto**	Enable Auto-MDIX on the interface.
Step 6	**end**	Return to privileged EXEC mode.
Step 7	**show controllers ethernet-controller** *interface-id* **phy**	Verify the operational state of the Auto-MDIX feature on the interface.
Step 8	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To disable Auto-MDIX, use the **no mdix auto** interface configuration command.

This example shows how to enable Auto-MDIX on a port:

```
Switch# configure terminal
Switch(config)# interface gigabitethernet1/0/1
Switch(config-if)# speed auto
Switch(config-if)# duplex auto
Switch(config-if)# mdix auto
Switch(config-if)# end
```

Configuring Power over Ethernet on an Interface

The switch supports both the Cisco pre-standard PoE method and the IEEE 802.3af PoE standard. The switches automatically supply power to connected pre-standard powered devices (such as Cisco IP Phones and Cisco Aironet access points) and IEEE 802.3af-compliant powered devices if the switch senses that there is no power on the circuit.

On a 24-port PoE switch, each 10/100 port provides 15.4 W of power. On a 48-port PoE switch, any 24 of the 48 10/100 ports provide 15.4 W of power, or any combination of ports provide an average of 7.7 W of power at the same time, up to a maximum switch power output of 370 W.

A powered device can receive redundant power when it is connected to a PoE switch port and to an AC power source. If a device being powered by the switch is then connected to wall power, the switch might continue to power the device. The switch continues to report that it is still powering the device whether the device is being powered by the switch or receiving power from an AC power source.

The switch detects the power required by any new device that is connected and decides whether the device requires more power than is currently available. If the switch cannot supply the required power, the new device is not powered, and the switch provides this information in the CLI **show** command messages, by sending a syslog error message, and in LED displays. Refer to the hardware installation guide for LED information.

The switch automatically maintains a power budget, monitors and tracks requests for power, and grants power only when it is available. When a PoE-capable interface is in the no-shutdown state with PoE enabled (the default), and a pre-standard or IEEE-compliant powered device is connected to the interface, the switch detects when the connected device is not being powered by an AC adaptor. When a device needing power is detected, the switch determines the device power requirements based on its type or uses an initial allocation of 15.4 W for power budgeting.

- If enough power is available, the switch grants power, updates the power budget, turns on power to the interface, and updates the LEDs.

- If granting power would exceed the system power budget, the switch denies power, ensures that power to the interface is turned off, generates a syslog message, and updates the LEDs. After power has been denied, the switch periodically rechecks the power budget and continues to attempt to grant the request for power.

- If enough power is available for all powered devices connected to a switch, power is turned on to all devices. If there is not enough available PoE, or if a device is disconnected and reconnected while other devices are waiting for power, devices to be granted or denied power cannot be predetermined.

After power is applied to an interface, the switch uses Cisco Discovery Protocol (CDP) to determine the power requirement of the connected Cisco PoE (standard and pre-standard) devices, and the switch adjusts the power budget accordingly. This does not apply to third-party PoE devices. If the switch detects a fault caused by an undervoltage, overvoltage, overtemperature, oscillator-fault, or short-circuit condition, it turns off power to the port, generates a syslog message, and updates the power budget and LEDs.

The PoE feature operates the same whether or not the switch is a stack member. The power budget is per-switch and independent of any other switch in the stack. Election of a new stack master does not affect PoE operation. The stack master keeps track of PoE status for all switches and interfaces in the stack and includes the status in output displays.

Beginning in privileged EXEC mode, follow these steps to enable or disable PoE on an interface on a PoE-capable switch.

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**interface** *interface-id*	Enter interface configuration mode for the physical interface to be configured.
Step 3	**power inline** {**auto** | **never**}	Configure PoE on the interface: • Enter **auto** (the default) to set the interface to automatically detect if a connected device requires power and to supply power to the device if enough power is available. • Enter **never** to disable power detection and supply for the interface.
Step 4	**end**	Return to privileged EXEC mode.
Step 5	**show power inline** [*interface-id* | **module** *switch-number*]	Display PoE status for a switch or switch stack, for the specified interface, or for a specified stack member.
Step 6	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

For information about the output of the **show power inline** user EXEC command, refer to the command reference for this release. For more information about PoE-related commands, see the "Troubleshooting Power over Ethernet Switch Ports" section on page 39-13.

This example shows how to enable automatic PoE on a port and the response from the **show power inline** command for the interface when a Cisco IEEE-compliant IP Phone is being supplied with power:

```
Switch# configure terminal
Switch(config)# interface fastethernet1/0/1
Switch(config-if)# power inline auto
Switch(config-if)# end
Switch# show power inline fastethernet1/0/1

Interface  Admin   Oper      Power    Device             Class
                             (Watts)
---------- ----- ---------- ------- ------------------  -----------
Fa1/0/1     auto    on        6.3     Cisco IP Phone 7960  Class 2
```

Adding a Description for an Interface

You can add a description about an interface to help you remember its function. The description appears in the output of these privileged EXEC commands: **show configuration**, **show running-config**, and **show interfaces**.

Beginning in privileged EXEC mode, follow these steps to add a description for an interface:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**interface** *interface-id*	Enter interface configuration mode, and enter the interface for which you are adding a description.
Step 3	**description** *string*	Add a description (up to 240 characters) for an interface.

	Command	Purpose
Step 4	**end**	Return to privileged EXEC mode.
Step 5	**show interfaces** *interface-id* **description** or **show running-config**	Verify your entry.
Step 6	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

Use the **no description** interface configuration command to delete the description.

This example shows how to add a description on a port and how to verify the description:

```
Switch# config terminal
Enter configuration commands, one per line.  End with CNTL/Z.
Switch(config)# interface gigabitethernet1/0/2
Switch(config-if)# description Connects to Marketing
Switch(config-if)# end
Switch# show interfaces gigabitethernet1/0/2 description
Interface Status        Protocol Description
Gi1/0/2   admin down     down     Connects to Marketing
```

Configuring Layer 3 Interfaces

The Catalyst 3750 switch supports these types of Layer 3 interfaces:

- SVIs: You should configure SVIs for any VLANs for which you want to route traffic. SVIs are created when you enter a VLAN ID following the **interface vlan** global configuration command. To delete an SVI, use the **no interface vlan** global configuration command.

> **Note** When you create an SVI, it does not become active until it is associated with a physical port. For information about assigning Layer 2 ports to VLANs, see Chapter 13, "Configuring VLANs."

- Routed ports: Routed ports are physical ports configured to be in Layer 3 mode by using the **no switchport** interface configuration command.

- Layer 3 EtherChannel ports: EtherChannel interfaces made up of routed ports.

    EtherChannel port interfaces are described in Chapter 33, "Configuring EtherChannels."

A Layer 3 switch can have an IP address assigned to each routed port and SVI.

There is no defined limit to the number of SVIs and routed ports that can be configured in a switch stack. However, the interrelationship between the number of SVIs and routed ports and the number of other features being configured might have an impact on CPU usage because of hardware limitations. If the switch is using maximum hardware resources, attempts to create a routed port or SVI have these results:

- If you try to create a new routed port, the switch generates a message that there are not enough resources to convert the interface to a routed port, and the interface remains as a switchport.

- If you try to create an extended-range VLAN, an error message is generated, and the extended-range VLAN is rejected.

- If the switch is notified by VLAN Trunking Protocol (VTP) of a new VLAN, it sends a message that there are not enough hardware resources available and shuts down the VLAN. The output of the **show vlan** user EXEC command shows the VLAN in a suspended state.

- If the switch attempts to boot up with a configuration that has more VLANs and routed ports than hardware can support, the VLANs are created, but the routed ports are shut down, and the switch sends a message that this was due to insufficient hardware resources.

All Layer 3 interfaces require an IP address to route traffic. This procedure shows how to configure an interface as a Layer 3 interface and how to assign an IP address to an interface.

Note   If the physical port is in Layer 2 mode (the default), you must enter the **no switchport** interface configuration command to put the interface into Layer 3 mode. Entering a **no switchport** command disables and then re-enables the interface, which might generate messages on the device to which the interface is connected. Furthermore, when you put an interface that is in Layer 2 mode into Layer 3 mode, the previous configuration information related to the affected interface might be lost, and the interface is returned to its default configuration

Beginning in privileged EXEC mode, follow these steps to configure a Layer 3 interface:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**interface** {{**fastethernet**	**gigabitethernet**} *interface-id*}	{**vlan** *vlan-id*}	{**port-channel** *port-channel-number*}	Enter interface configuration mode, and enter the interface to be configured as a Layer 3 interface.
Step 3	**no switchport**	For physical ports only, enter Layer 3 mode.
Step 4	**ip address** *ip_address subnet_mask*	Configure the IP address and IP subnet.
Step 5	**no shutdown**	Enable the interface.
Step 6	**end**	Return to privileged EXEC mode.
Step 7	**show interfaces** [*interface-id*]	Verify the configuration.
	show ip interface [*interface-id*]	
	show running-config interface [*interface-id*]	
Step 8	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To remove an IP address from an interface, use the **no ip address** interface configuration command.

This example shows how to configure a port as a routed port and to assign it an IP address:

```
Switch# configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
Switch(config)# interface gigabitethernet1/0/2
Switch(config-if)# no switchport
Switch(config-if)# ip address 192.20.135.21 255.255.255.0
Switch(config-if)# no shutdown
```

Configuring the System MTU

The default maximum transmission unit (MTU) size for frames received and transmitted on all interfaces on the switch stack is 1500 bytes. You can increase the MTU size for all interfaces operating at 10 or 100 Mbps by using the **system mtu** global configuration command. You can increase the MTU size to

support jumbo frames on all Gigabit Ethernet interfaces by using the **system mtu jumbo** global configuration command. Gigabit Ethernet ports are not affected by the **system mtu** command; 10/100 ports are not affected by the **system jumbo mtu** command.

You cannot set the MTU size for an individual interface; you set it for all 10/100 or all Gigabit Ethernet interfaces on the switch stack. When you change the MTU size, you must reset the switch before the new configuration takes effect.

The size of frames that can be received by the switch CPU is limited to 1992 bytes, no matter what value was entered with the **system mtu** or **system mtu jumbo** commands. Although frames that are forwarded or routed typically are not received by the CPU, in some cases packets are sent to the CPU, such as traffic sent to control traffic, SNMP, Telnet, or routing protocols.

Routed packets are subjected to MTU checks on the egress ports. The MTU value used for routed ports is derived from the **system mtu** configured value (not the **system mtu jumbo** value). That is, the routed MTU is never greater than the system MTU for any VLAN. The system MTU value is used by routing protocols when negotiating adjacencies and the MTU of the link. For example Open Shortest Path First (OSPF) protocol uses this MTU value before setting up an adjacency with a peer router. To view the MTU value for routed packets for a specific VLAN, use the **show platform port-asic mvid** privileged EXEC command.

> **Note**   If Layer 2 Gigabit Ethernet interfaces are configured to accept frames greater than the 10/100 interfaces, jumbo frames ingressing on a Layer 2 Gigabit Ethernet interface and egressing on a Layer 2 10/100 interface are dropped.

Beginning in privileged EXEC mode, follow these steps to change MTU size for all 10/100 or Gigabit Ethernet interfaces:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**system mtu** *bytes*	(Optional) Change the MTU size for all interfaces on the switch stack that are operating at 10 or 100 Mbps. The range is 1500 to 1546 bytes; the default is 1500 bytes.
Step 3	**system mtu jumbo** *bytes*	(Optional) Change the MTU size for all Gigabit Ethernet interfaces on the switch stack. The range is 1500 to 9000 bytes; the default is 1500 bytes.
Step 4	**end**	Return to privileged EXEC mode.
Step 5	**copy running-config startup-config**	Save your entries in the configuration file.
Step 6	**reload**	Reload the operating system.

If you enter a value that is outside the allowed range for the specific type of interface, the value is not accepted.

Once the switch reloads, you can verify your settings by entering the **show system mtu** privileged EXEC command.

This example shows how to set the maximum packet size for a Gigabit Ethernet port to 1800 bytes:

```
Switch(config)# system jumbo mtu 1800
Switch(config)# exit
Switch# reload
```

This example shows the response when you try to set Gigabit Ethernet interfaces to an out-of-range number:

```
Switch(config)# system mtu jumbo 25000
                                   ^
% Invalid input detected at '^' marker.
```

Monitoring and Maintaining the Interfaces

You can perform the tasks in these sections to monitor and maintain interfaces:

Monitoring Interface Status

Commands entered at the privileged EXEC prompt display information about the interface, including the versions of the software and the hardware, the configuration, and statistics about the interfaces. Table 11-3 lists some of these interface monitoring commands. (You can display the full list of **show** commands by using the **show ?** command at the privileged EXEC prompt.) These commands are fully described in the *Cisco IOS Interface Command Reference, Release 12.2*.

Table 11-3   Show Commands for Interfaces

Command	Purpose
show interfaces [*interface-id*]	Display the status and configuration of all interfaces or a specific interface.
show interfaces *interface-id* **status** [**err-disabled**]	Display interface status or a list of interfaces in an error-disabled state.
show interfaces [*interface-id*] **switchport**	Display administrative and operational status of switching (nonrouting) ports. You can use this command to find out if a port is in routing or switching mode.
show interfaces [*interface-id*] **description**	Display the description configured on an interface or all interfaces and the interface status.
show ip interface [*interface-id*]	Display the usability status of all interfaces configured for IP routing or the specified interface.
show interface [*interface-id*] **stats**	Display the input and output packets by the switching path for the interface.
show interfaces [*interface-id*] [{**transceiver calibration** | **properties** | **detail**}] *module number*]	Display physical and operational status about an SFP module.
show running-config interface [*interface-id*]	Display the running configuration in RAM for the interface.
show version	Display the hardware configuration, software version, the names and sources of configuration files, and the boot images.
show controllers ethernet-controller *interface-id* **phy**	Verify the operational state of the Auto-MDIX feature on the interface.

Clearing and Resetting Interfaces and Counters

Table 11-4 lists the privileged EXEC mode **clear** commands that you can use to clear counters and reset interfaces.

Table 11-4   Clear Commands for Interfaces

Command	Purpose
clear counters [*interface-id*]	Clear interface counters.
clear interface *interface-id*	Reset the hardware logic on an interface.
clear line [*number*	**console 0**	**vty** *number*]	Reset the hardware logic on an asynchronous serial line.

To clear the interface counters shown by the **show interfaces** privileged EXEC command, use the **clear counters** privileged EXEC command. The **clear counters** command clears all current interface counters from the interface unless optional arguments are specified to clear only a specific interface type from a specific interface number.

> **Note** The **clear counters** privileged EXEC command does not clear counters retrieved by using Simple Network Management Protocol (SNMP), but only those seen with the **show interface** privileged EXEC command.

Shutting Down and Restarting the Interface

Shutting down an interface disables all functions on the specified interface and marks the interface as unavailable on all monitoring command displays. This information is communicated to other network servers through all dynamic routing protocols. The interface is not mentioned in any routing updates.

Beginning in privileged EXEC mode, follow these steps to shut down an interface:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**interface** {**vlan** *vlan-id*}	{{**fastethernet**	**gigabitethernet**} *interface-id*}	{**port-channel** *port-channel-number*}	Select the interface to be configured.
Step 3	**shutdown**	Shut down an interface.
Step 4	**end**	Return to privileged EXEC mode.
Step 5	**show running-config**	Verify your entry.

Use the **no shutdown** interface configuration command to restart the interface.

To verify that an interface is disabled, enter the **show interfaces** privileged EXEC command. A disabled interface is shown as *administratively down* in the **show interface** command display.