CHAPTER 13

Configuring Interface Characteristics

This chapter defines the types of Catalyst interfaces and describes how to configure them.

Note For complete syntax and usage information for the commands used in this chapter, see the switch command reference for this release and the *Cisco IOS Interface Command Reference, Release 12.4* on Cisco.com.

Understanding Interface Types

This section describes the different types of supported interfaces with references to chapters that contain more detailed information about configuring these interfaces.

Connecting Interfaces, page 13-6Port-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 the Chapter 14, “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 you configure a local port 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.

To configure VLANs, use the **vlan** *vlan-id* global configuration command to enter VLAN configuration mode. The VLAN configurations for normal-range VLANs (VLAN IDs 1 to 1005) are saved in the VLAN database. If VTP is version 1 or 2, you must first set VTP mode to transparent to configure extended-range VLANs (VLAN IDs 1006 to 4094). Extended-range VLANs created in transparent mode are not added to the VLAN database but are saved in the switch running configuration. With VTP version 3, you can create extended-range VLANs in client or server mode. These VLANs are saved in the VLAN database.

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. You use switch ports for managing the physical interface and associated Layer 2 protocols.

Configure switch ports by using the **switchport** interface configuration commands.

For detailed information about configuring access port and trunk port characteristics, see Chapter 14, “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.

Supported access ports:

- Static access ports are manually assigned to a VLAN (or through a RADIUS server for use with IEEE 802.1x). For more information, see the “802.1x Authentication with VLAN Assignment” section on page 12-17.
- VLAN membership of dynamic access ports is learned through incoming packets. By default, a dynamic access port is not a member of any VLAN. Traffic forwarding to and from the port is enabled only when the port VLAN membership 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 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.

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 affects only the associated trunk port. By default, all possible VLANs (VLAN ID 1 to 4094) are in the allowed list. A trunk port can become a member of a VLAN only if VTP knows of the VLAN and if the VLAN is enabled. If VTP learns of a new, enabled VLAN and the VLAN is in the allowed list, the trunk port automatically becomes a member of that VLAN. Traffic is forwarded to and from the trunk port for that VLAN. If VTP learns of an enabled VLAN that is not in the allowed list for a trunk port, the port does not become a member of the VLAN, and traffic for the VLAN is not forwarded to or from the port.

For more information about trunk ports, see Chapter 14, "Configuring VLANs."

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. You can associate only one SVI with a VLAN. You configure an SVI for a VLAN only to route 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.

Note You cannot delete interface VLAN 1.

SVIs provide IP host connectivity only to the system.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 encapsulated trunk port 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 3-14.

Note When you create an SVI, it does not become active until it you associate it with a physical port.

EtherChannel Port Groups

EtherChannel port groups treat multiple switch ports as one switch port. An EtherChannel port group acts 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. Most protocols operate over either single ports or aggregated switch ports and do not recognize the physical ports within the port group. The DTP, the Cisco Discovery Protocol (CDP), and the Port Aggregation Protocol (PAgP) operate only on physical ports.

When you configure an EtherChannel, you create a port-channel logical interface and assign an interface to the EtherChannel. 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 39, "Configuring EtherChannels and Link-State Tracking."

Power over Ethernet Ports

PoE switch ports automatically supply power to these connected devices (if the switch senses that there is no power on the circuit):

- Cisco pre-standard powered devices (such as Cisco IP Phones and Cisco Aironet access points)
- IEEE 802.3 af-compliant powered devices

A powered device can receive redundant power when it is connected only to a PoE switch port and to an AC power source.

This section has this PoE information:

- Supported Protocols and Standards, page 13-4
- Powered-Device Detection and Initial Power Allocation, page 13-4
- Power Management Modes, page 13-5

Supported Protocols and Standards

The switch uses these protocols and standards to support PoE:

- CDP with power consumption—The powered device notifies the switch of the amount of power it is consuming. The switch does not reply to the power-consumption messages. The switch can only supply power to or remove power from the PoE port.
- Cisco intelligent power management—The powered device and the switch negotiate through power-negotiation CDP messages for an agreed power-consumption level. The negotiation allows a high-power Cisco powered device, which consumes more than 7 W, to operate at its highest power mode. The powered device first boots up in low-power mode, consumes less than 7 W, and negotiates to obtain enough power to operate in high-power mode. The device changes to high-power mode only when it receives confirmation from the switch.

  High-power devices can operate in low-power mode on switches that do not support power-negotiation CDP.

  Cisco intelligent power management is backward-compatible with CDP with power consumption; the switch responds according to the CDP message that it receives. CDP is not supported on third-party powered devices; therefore, the switch uses the IEEE classification to determine the power usage of the device.
- IEEE 802.3af—The major features of this standard are powered-device discovery, power administration, disconnect detection, and optional powered-device power classification. For more information, see the standard.

Powered-Device Detection and Initial Power Allocation

The switch detects a Cisco pre-standard or an IEEE-compliant powered device when the PoE-capable port is in the no-shutdown state, PoE is enabled (the default), and the connected device is not being powered by an AC adaptor.

After device detection, the switch determines the device power requirements based on its type:

The initial power allocation is the maximum amount of power that a powered device requires. The switch initially allocates this amount of power when it detects and powers the powered device. As the switch receives CDP messages from the powered device and as the powered device negotiates power levels with the switch through CDP power-negotiation messages, the initial power allocation might be adjusted.

- The switch classifies the detected IEEE device within a power consumption class. Based on the available power in the power budget, the switch determines if a port can be powered. Table 13-1 lists these levels.

Table 13-1 IEEE Power Classifications

Class	Maximum Power Level Required from the Switch
0 (class status unknown)	15.4 W
1	4 W
2	7 W
3	15.4 W

The switch monitors and tracks requests for power and grants power only when it is available. The switch tracks its power budget (the amount of power available on the switch for PoE). The switch performs power-accounting calculations when a port is granted or denied power to keep the power budget up to date.

After power is applied to the port, the switch uses CDP to determine the *actual* power consumption requirement of the connected Cisco powered devices, and the switch adjusts the power budget accordingly. This does not apply to third-party PoE devices. The switch processes a request and either grants or denies power. If the request is granted, the switch updates the power budget. If the request is denied, the switch ensures that power to the port is turned off, generates a syslog message, and updates the LEDs. Powered devices can also negotiate with the switch for more power.

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.

Power Management Modes

Supported PoE modes:

- **auto**—The switch automatically detects if the connected device requires power. If the switch discovers a powered device connected to the port and if the switch has enough power, it grants power, updates the power budget, turns on power to the port on a first-come, first-served basis, and updates the LEDs. For LED information, see the hardware installation guide.

  If the switch has enough power for all the powered devices, they all come up. If enough power is available for all powered devices connected to the 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, it cannot be determined which devices are granted or are denied power.

  If granting power would exceed the system power budget, the switch denies power, ensures that power to the port 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 a device being powered by the switch is then connected to wall power, the switch might continue to power the device. The switch might continue 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.

  If a powered device is removed, the switch automatically detects the disconnect and removes power from the port. You can connect a nonpowered device without damaging it.

You can specify the maximum wattage that is allowed on the port. If the IEEE class maximum wattage of the powered device is greater than the configured maximum value, the switch does not provide power to the port. If the switch powers a powered device, but the powered device later requests through CDP messages more than the configured maximum value, the switch removes power to the port. The power that was allocated to the powered device is reclaimed into the global power budget. If you do not specify a wattage, the switch delivers the maximum value. Use the **auto** setting on any PoE port. The auto mode is the default setting.

- **static**—The switch pre-allocates power to the port (even when no powered device is connected) and guarantees that power will be available for the port. The switch allocates the port configured maximum wattage, and the amount is never adjusted through the IEEE class or by CDP messages from the powered device. Because power is pre-allocated, any powered device that uses less than or equal to the maximum wattage is guaranteed to be powered when it is connected to the static port. The port no longer participates in the first-come, first-served model.

  However, if the powered-device IEEE class is greater than the maximum wattage, the switch does not supply power to it. If the switch learns through CDP messages that the powered device needs more than the maximum wattage, the powered device is shutdown.

  If you do not specify a wattage, the switch pre-allocates the maximum value. The switch powers the port only if it discovers a powered device. Use the **static** setting on a high-priority interface.

- **never**—The switch disables powered-device detection and never powers the PoE port even if an unpowered device is connected. Use this mode only when you want to make sure power is never applied to a PoE-capable port, making the port a data-only port.

For information on configuring a PoE port, see the "Configuring a Power Management Mode on a PoE Port" section on page 13-16.

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. By using the switch with routing enabled, when you configure both VLAN 20 and VLAN 30 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.

Using Interface Configuration Mode

The switch supports these interface types:

- VLANs—switch virtual interfaces
- Port channels—EtherChannel interfaces

You can also configure a range of interfaces (see the "Configuring a Range of Interfaces" section on page 13-8).

- Type—Port types depend on those supported on the switch. Possible types are: Fast Ethernet (fastethernet or fa) for 10/100 Mb/s Ethernet, Gigabit Ethernet (gigabitethernet or gi) for 10/100/1000 Mb/s Ethernet ports, 10-Gigabit Ethernet (tengigabitethernet or te) for 10,000 Mb/s, or small form-factor pluggable (SFP) module Gigabit Ethernet interfaces.
- Module number—The module or slot number on the switch (always 0).

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

Note Configuration examples and outputs in this book might not be specific to your switch, particularly regarding the presence of a stack member number.

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.

```
Switch(config-if)#
```

Note Entering a space between the interface type and interface number is optional

Step 3 Follow each **interface** command with the 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 13-20.

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*}	Specify the range of interfaces (VLANs or physical ports) to be configured, and enter interface-range configuration mode. • 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 13-9. • 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		Use the normal configuration commands to apply the configuration parameters to all interfaces in the range. Each command is executed as it is entered.
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,* depending on port types on the switch:
  - **port-channel** *port-channel-number - port-channel-number*, where the *port-channel-number* is 1 to

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.
- 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 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 2 to 100 Mb/s:

```
Switch# configure terminal
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 and Gigabit Ethernet ports 1 and 2 to receive flow-control pause frames:

```
Switch# configure terminal
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 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,* depending on port types on the switch:
  - **port-channel** *port-channel-number - port-channel-number*, where the *port-channel-number* is 1 to

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 entering an *interface-rang*.
- 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 and to verify the macro configuration:

```
Switch# configure terminal
Switch(config)# end
Switch# show running-config | include define
```

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

```
Switch# configure terminal
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

Default Ethernet Interface Configuration

Table 13-2 shows the Ethernet interface default configuration. For more details on the VLAN parameters listed in the table, see Chapter 14, "Configuring VLANs." For details on controlling traffic to the port, see Chapter 24, "Configuring Port-Based Traffic Control."

Table 13-2 Default Layer 2 Ethernet Interface Configuration

Feature	Default Setting
Allowed VLAN range	VLANs 1 to 4094.
Default VLAN (for access ports)	VLAN 1.
Native VLAN (for IEEE 802.1Q trunks)	VLAN 1.
802.1p priority-tagged traffic	Drop all packets tagged with VLAN 0.
VLAN trunking	Switchport mode dynamic auto (supports DTP).
Port enable state	All ports are enabled.
Port description	None defined.
Speed	Autonegotiate.
Duplex mode	Autonegotiate.
Flow control	Flow control is set to **receive**: **off**. It is always off for sent packets.
EtherChannel (PAgP)	Disabled on all Ethernet ports. Chapter 39, "Configuring EtherChannels and Link-State Tracking."
Port blocking (unknown multicast and unknown unicast traffic)	Disabled (not blocked). See the "Configuring Port Blocking" section on page 24-7.
Broadcast, multicast, and unicast storm control	Disabled. See the "Default Storm Control Configuration" section on page 24-3.
Protected port	Disabled. See the "Configuring Protected Ports" section on page 24-6.
Port security	Disabled. See the "Default Port Security Configuration" section on page 24-11.
Port Fast	Disabled. See the "Default Optional Spanning-Tree Configuration" section on page 19-9.
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.
Keepalive messages	Disabled on SFP module ports; enabled on all other ports.

Configuring Interface Speed and Duplex Mode

Depending on the supported port types, Ethernet interfaces on the switch operate at 10, 100, or 1000 Mb/s, or 10,000 Mb/s 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-Mb/s ports operate in half-duplex mode, which means that stations can either receive or send traffic.

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

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

- Speed and Duplex Configuration Guidelines, page 13-12
- Setting the Interface Speed and Duplex Parameters, page 13-13

Speed and Duplex Configuration Guidelines

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

- Fast Ethernet (10/100-Mb/s) ports support all speed and duplex options.
- Gigabit Ethernet (10/100/1000-Mb/s) ports support all speed options and all duplex options (auto, half, and full). However, Gigabit Ethernet ports operating at 1000 Mb/s do not support half-duplex mode.
- For SFP module ports, the speed and duplex CLI options change depending on the SFP module type:
  - The 1000BASE-*x* (where -*x* is -BX, -CWDM, -LX, -SX, and -ZX) SFP module ports support the **nonegotiate** keyword in the **speed** interface configuration command. Duplex options are not supported.
  - The 1000BASE-T SFP module ports support the same speed and duplex options as the 10/100/1000-Mb/s ports.
  - The 100BASE-*x* (where -*x* is -BX, -CWDM, -LX, -SX, and -ZX) SFP module ports support only 100 Mb/s. These modules support full- and half- duplex options but do not support autonegotiation.

  For information about which SFP modules are supported on your switch, see the product release notes.
- 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.
- 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*	Specify the physical interface to be configured, and enter interface configuration mode.
Step 3	**speed** {**10** | **100** | **1000** | **auto** [**10** | **100** | **1000**] | **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 Mb/s ports. • Enter **auto** to enable the interface to autonegotiate speed with the connected device. If you use the **10**, **100**, or the **1000** keywords with the **auto** keyword, the port autonegotiates only at the specified speeds. • The **nonegotiate** keyword is available only for SFP module ports. SFP module ports operate only at 1000 Mb/s but can be configured to not negotiate if connected to a device that does not support autonegotiation. For more information about speed settings, see the "Speed and Duplex Configuration Guidelines" section on page 13-12.
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 Mb/s). You cannot configure half-duplex mode for interfaces operating at 1000 Mb/s. For more information about duplex settings, see the "Speed and Duplex Configuration Guidelines" section on page 13-12.
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 Mb/s and the duplex mode to half on a 10/100 Mb/s port:

```
Switch# configure terminal
Switch(config-if)# speed 10
Switch(config-if)# duplex half
```

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

```
Switch# configure terminal
Switch(config-if)# speed 100
```

Configuring IEEE 802.3x 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 by sending a pause frame to stop sending until the condition clears. 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 Ports on the switch can receive, but not send, 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, see 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*	Specify the physical interface to be configured, and enter interface configuration mode.
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-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, see the hardware installation guide.

Auto-MDIX is enabled by default.

Auto-MDIX is supported on all 10/100 and 10/100/1000-Mb/s interfaces. It is not supported on 1000BASE-SX or -LX SFP module interfaces.

Table 13-3 shows the link states that result from auto-MDIX settings and correct and incorrect cabling.

Table 13-3 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*	Specify the physical interface to be configured, and enter interface configuration mode.
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	**end**	Return to privileged EXEC mode.
Step 6	**show controllers ethernet-controller** *interface-id* **phy**	Verify the operational state of the auto-MDIX feature on the interface.
Step 7	**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-if)# speed auto
Switch(config-if)# duplex auto
Switch(config-if)# mdix auto
Switch(config-if)# end
```

Configuring a Power Management Mode on a PoE Port

For most situations, the default configuration (auto mode) works well, providing plug-and-play operation. No further configuration is required. However, use the following procedure to give a PoE port higher priority, to make it data only, or to specify a maximum wattage to disallow high-power powered devices on a port.

Note When you make PoE configuration changes, the port being configured drops power. Depending on the new configuration, the state of the other PoE ports, and the state of the power budget, the port might not be powered up again. For example, port 1 is in the auto and on state, and you configure it for static mode. The switch removes power from port 1, detects the powered device, and repowers the port. If port 1 is in the auto and on state and you configure it with a maximum wattage of 10 W, the switch removes power from the port and then redetects the powered device. The switch repowers the port only if the powered device is a Class 1, Class 2, or a Cisco-only powered device.

Beginning in privileged EXEC mode, follow these steps to configure a power management mode on a PoE-capable port:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**interface** *interface-id*	Specify the physical port to be configured, and enter interface configuration mode.
Step 3	**power inline** {**auto** [**max** *max-wattage*] | **never** | **static** [**max** *max-wattage*]}	Configure the PoE mode on the port. The keywords have these meanings: • **auto**—Enable powered-device detection. If enough power is available, automatically allocate power to the PoE port after device detection. This is the default setting. • (Optional) **max** *max-wattage*—Limit the power allowed on the port. The range is 4000 to 15400 milliwatts. If no value is specified, the maximum is allowed. • **never**—Disable device detection, and disable power to the port. **Note** If a port has a Cisco powered device connected to it, do not use the **power inline never** command to configure the port. A false link-up can occur, placing the port into an error-disabled state. • **static**—Enable powered-device detection. Pre-allocate (reserve) power for a port before the switch discovers the powered device. The switch reserves power for this port even when no device is connected and guarantees that power will be provided upon device detection. The switch allocates power to a port configured in static mode before it allocates power to a port configured in auto mode.
Step 4	**end**	Return to privileged EXEC mode.
Step 5	**show power inline** [*interface-id*]	
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, see the command reference for this release. For more information about PoE-related commands, see the "Troubleshooting Power over Ethernet Switch Ports" section on page 40-12. For information about configuring voice VLAN, see Chapter 16, "Configuring Voice VLAN."

Budgeting Power for Devices Connected to a PoE Port

When Cisco powered devices are connected to PoE ports, the switch uses Cisco Discovery Protocol (CDP) to determine the *actual* power consumption of the devices, and the switch adjusts the power budget accordingly. The CDP protocol works with Cisco powered devices and does not apply to IEEE third-party powered devices. For these devices, when the switch grants a power request, the switch adjusts the power budget according to the powered-device IEEE classification. If the powered device is a Class 0 (class status unknown) or a Class 3, the switch budgets 15,400 milliwatts for the device, regardless of the actual amount of power needed. If the powered device reports a higher class than its actual consumption or does not support power classification (defaults to Class 0), the switch can power fewer devices because it uses the IEEE class information to track the global power budget.

By using the **power inline consumption** *wattage* configuration command, you can override the default power requirement specified by the IEEE classification. The difference between what is mandated by the IEEE classification and what is actually needed by the device is reclaimed into the global power budget for use by additional devices. You can then extend the switch power budget and use it more effectively.

For example, if the switch budgets 15,400 milliwatts on each PoE port, you can connect only 24 Class 0 powered devices. If your Class 0 device power requirement is actually 5000 milliwatts, you can set the consumption wattage to 5000 milliwatts and connect up to 48 devices. The total PoE output power available on a 24-port or 48-port switch is 370,000 milliwatts.

Caution You should carefully plan your switch power budget and make certain not to oversubscribe the power supply.

Note When you manually configure the power budget, you must also consider the power loss over the cable between the switch and the powered device.

When you enter the **power inline consumption default** *wattage* or the **no power inline consumption default** global configuration command, or the **power inline consumption** *wattage* or the **no power inline consumption** interface configuration command this caution message appears:

```
%CAUTION: Interface interface-id: Misconfiguring the 'power inline consumption/allocation'
command may cause damage to the switch and void your warranty. Take precaution not to
oversubscribe the power supply.
Refer to documentation.
```

If the power supply is over-subscribed to by up to 20 percent, the switch continues to operate but its reliability is reduced. If the power supply is subscribed to by more than 20 percent, the short-circuit protection circuitry triggers and shuts the switch down.

For more information about the IEEE power classifications, see the "Power over Ethernet Ports" section on page 13-4.

Beginning in privileged EXEC mode, follow these steps to configure the amount of power budgeted to a powered device connected to each PoE port on a switch:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**no cdp run**	(Optional) Disable CDP.
Step 3	**power inline consumption default** *wattage*	Configure the power consumption of powered devices connected to each the PoE port on the switch. **Note** The range for each de vice is 4000 to 15400 milliwatts. The default is 15400 milliwatts.
Step 4	**end**	Return to privileged EXEC mode.
Step 5	**show power inline consumption**	Display the power consumption status.
Step 6	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To return to the default setting, use the **no power inline consumption default** global configuration command.

Beginning in privileged EXEC mode, follow these steps to configure amount of power budgeted to a powered device connected to a specific PoE port:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**no cdp run**	(Optional) Disable CDP.
Step 3	**interface** *interface-id*	Specify the physical port to be configured, and enter interface configuration mode.
Step 4	**power inline consumption** *wattage*	Configure the power consumption of a powered device connected to a PoE port on the switch. **Note** The range for each de vice is 4000 to 15400 milliwatts. The default is 15400 milliwatts.
Step 5	**end**	Return to privileged EXEC mode.
Step 6	**show power inline consumption**	Display the power consumption status.
Step 7	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To return to the default setting, use the **no power inline consumption** interface configuration command.

For information about the output of the **show power inline consumption** privileged EXEC command, see the command reference for this release.

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*	Specify the interface for which you are adding a description, and enter interface configuration mode.
Step 3	**description** *string*	Add a description (up to 240 characters) for an interface.
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-if)# description Connects to Marketing
Switch(config-if)# end
Interface Status         .Protocol Description
```

Configuring the System MTU

The default maximum transmission unit (MTU) size for frames received and transmitted on all interfaces is 1500 bytes. You can increase the MTU size for all interfaces operating at 10 or 100 Mb/s 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 mtu jumbo** command. If you do not configure the **system mtu jumbo** command, the setting of the **system mtu** command applies to all Gigabit Ethernet interfaces.

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

Note Frames sizes that can be received by the switch CPU are limited to 1998 bytes, no matter what value was entered with the **system mtu** or **system mtu jumbo** commands. If Layer 2 Gigabit Ethernet interfaces are configured to accept frames greater than the 10/100 interfaces, jumbo frames received on a Layer 2 Gigabit Ethernet interface and sent 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*	The range is 1500 to 1998 bytes; the default is 1500 bytes.
Step 3	**system mtu jumbo** *bytes*	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 mtu jumbo 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

These sections contain interface monitoring and maintenance information:

- Monitoring Interface Status, page 13-21
- Clearing and Resetting Interfaces and Counters, page 13-21
- Shutting Down and Restarting the Interface, page 13-22

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 13-4 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 12.4* from Cisco.com.

Table 13-4 Show Commands for Interfaces

Command	Purpose
show interfaces [*interface-id*]	(Optional) Display the status and configuration of all interfaces or a specific interface.
show interfaces *interface-id* **status** [**err-disabled**]	(Optional) Display interface status or a list of interfaces in an error-disabled state.
show interfaces [*interface-id*] **switchport**	(Optional) Display administrative and operational status of switching ports.
show interfaces [*interface-id*] **description**	(Optional) Display the description configured on an interface or all interfaces and the interface status.
show ip interface [*interface-id*]	(Optional) Display the usability status of all interfaces configured for IP routing or the specified interface.
show interface [*interface-id*] **stats**	(Optional) Display the input and output packets by the switching path for the interface.
show interfaces transceiver properties	(Optional)
show interfaces [*interface-id*] [{**transceiver 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**	Display the operational state of the auto-MDIX feature on the interface.

Clearing and Resetting Interfaces and Counters

Table 13-5 lists the privileged EXEC mode **clear** commands that you can use to clear counters and reset interfaces.

Table 13-5 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 you specify optional arguments that 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.