



Configuring Interface Characteristics

- [Information About Interface Characteristics, on page 1](#)
- [How to Configure Interface Characteristics, on page 11](#)
- [Setting Interface Speed and Duplex Mode: Example, on page 28](#)
- [Configuring Layer 3 Interfaces: Example, on page 28](#)
- [Configuring Breakout Interfaces : Example, on page 28](#)
- [Example: Configuring the USB Inactivity Timeout, on page 30](#)
- [Feature History for Configuring Interface Characteristics, on page 30](#)

Information About Interface Characteristics

The following sections provide information about interface characteristics.

Interface Types

This section describes the different types of interfaces supported by the device. The rest of the chapter describes configuration procedures for physical interface characteristics.

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. 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.

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, to configure extended-range VLANs (VLAN IDs 1006 to 4094), you must first set VTP mode to transparent. Extended-range VLANs created in transparent mode are not added to the VLAN database but are saved in the device running configuration. With VTP version 3, you can create extended-range VLANs in client or server mode in addition to transparent mode. These VLANs are saved in the VLAN database.

Add ports to a VLAN by using the **switchport** command in interface configuration mode.

- 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.

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 IEEE 802.1Q tagged), the packet is dropped, and the source address is not learned.

The types of access ports supported are:

- Static access ports are manually assigned to a VLAN (or through a RADIUS server for use with IEEE 802.1x).

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.

Trunk Ports

A trunk port carries the traffic of multiple VLANs and by default is a member of all VLANs in the VLAN database. The IEEE 802.1Q trunk port type is supported. An IEEE 802.1Q trunk port supports simultaneous tagged and untagged traffic. An IEEE 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 become a member of a VLAN only if VTP knows of the VLAN and if 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.

Tunnel Ports

Tunnel ports are used in IEEE 802.1Q tunneling to segregate the traffic of customers in a service-provider network from other customers who are using the same VLAN number. You configure an asymmetric link

from a tunnel port on a service-provider edge switch to an IEEE 802.1Q trunk port on the customer switch. Packets entering the tunnel port on the edge switch, already IEEE 802.1Q-tagged with the customer VLANs, are encapsulated with another layer of an IEEE 802.1Q tag (called the metro tag), containing a VLAN ID unique in the service-provider network, for each customer. The double-tagged packets go through the service-provider network keeping the original customer VLANs separate from those of other customers. At the outbound interface, also a tunnel port, the metro tag is removed, and the original VLAN numbers from the customer network are retrieved.

Tunnel ports cannot be trunk ports or access ports and must belong to a VLAN unique to each customer.

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.



Note The Network Essentials license supports static routing and the Routing Information Protocol (RIP). For full Layer 3 routing, you must enable the Network Advantage license on the standalone device, or the active device, or the standby device.

Switch Virtual Interfaces

A switch virtual interface (SVI) represents a VLAN of switch ports as one interface to the routing 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 device. By default, an SVI is created for the default VLAN (VLAN 1) to permit remote device 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 ISL or IEEE 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.

You can also use the interface range command to configure existing VLAN SVIs within the range. The commands entered under the interface range command are applied to all existing VLAN SVIs within the range. You can enter the command **interface range create vlan** *x - y* to create all VLANs in the specified range that do not already exist. When the VLAN interface is created, **interface range vlan** *id* can be used to configure the VLAN interface.

Although the device supports a total of 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.

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

EtherChannel Port Groups

EtherChannel port groups treat multiple switch ports as one switch port. These port groups act as a single logical port for high-bandwidth connections between devices or between devices 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, group multiple tunnel ports into one logical tunnel 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.

Uplink Ports

The supervisor module has 10 uplink ports, named 1 to 10. The first eight uplink ports, 1 to 8, use Small Form-Factor Pluggable (SFP) transceivers or SFP+ transceivers and uplink ports 9 and 10 use Quad Small Form-Factor Pluggable (QSFP) transceivers. Ports 1 to 8 are 10-Gigabit Ethernet ports that support both 10G and 1G transceivers. Ports 9 and 10 are the QSFP ports that support 40 Gigabit Ethernet uplinks.

By default the 10-Gigabit Ethernet ports 1 to 8 are enabled.

40-Gigabit Ethernet Port 9 can be enabled only when ports 1 to 4 are enabled. 40-Gigabit Ethernet Port 10 can be enabled only when ports 5 to 9 are enabled.

Power over Ethernet

The Power over Ethernet (PoE) technology allows PoE (802.3af standard), PoE+ (802.3at) ports to supply power for the operation of a device.

Cisco Universal Power Over Ethernet (Cisco UPoE) extends the IEEE PoE+ standard to double the power per port to 60 watts.

For more information, see the *Configuring PoE* section of this guide

Using the Switch USB Ports

The device has two USB ports on the front panel — a USB mini-Type B console port and a USB Type A port.

USB Mini-Type B Console Port

The device has the following console ports:

- USB mini-Type B console connection
- RJ-45 console port

Console output appears on devices connected to both ports, but console input is active on only one port at a time. By default, the USB connector takes precedence over the RJ-45 connector.



Note Windows PCs require a driver for the USB port. See the hardware installation guide for driver installation instructions.

Use the supplied USB Type A-to-USB mini-Type B cable to connect a PC or other device to the device. The connected device must include a terminal emulation application. When the device detects a valid USB connection to a powered-on device that supports host functionality (such as a PC), input from the RJ-45 console is immediately disabled, and input from the USB console is enabled. Removing the USB connection immediately reenables input from the RJ-45 console connection. An LED on the device shows which console connection is in use.

Console Port Change Logs

At software startup, a log shows whether the USB or the RJ-45 console is active. Every device always first displays the RJ-45 media type.

In the sample output, Device 1 has a connected USB console cable. Because the bootloader did not change to the USB console, the first log from Device 1 shows the RJ-45 console. A short time later, the console changes and the USB console log appears. Device 2 and Device 3 have connected RJ-45 console cables.

When the USB cable is removed or the PC de-activates the USB connection, the hardware automatically changes to the RJ-45 console interface:

You can configure the console type to always be RJ-45, and you can configure an inactivity timeout for the USB connector.

USB Type A Port

The USB Type A port provides access to external USB flash devices, also known as thumb drives or USB keys. The port supports Cisco USB flash drives, USB 2.0 and USB 3.0, with capacities from 128 MB to 16 GB (USB devices with port densities of 128 MB, 256 MB, 1 GB, 4 GB, 8 GB, and 16 GB are supported). USB 3.0 is also called SuperSpeed USB, used for higher file transfer rates. You can use standard Cisco IOS command-line interface (CLI) commands to read, write, erase, and copy to or from the flash device. You can also configure the devices to boot from the USB flash drive.

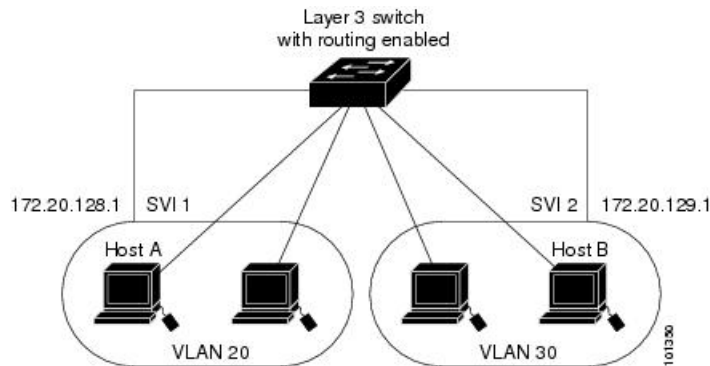
USB 2.0 Host Port

The USB 2.0 host port provides access to external USB flash devices, also known as thumb drives or USB keys. The port supports Cisco USB flash drives with capacities from 128 MB to 16 GB (USB devices with port densities of 128 MB, 256 MB, 1 GB, 4 GB, 8 GB, and 16 GB are supported). You can use standard Cisco IOS command-line interface (CLI) commands to read, write, erase, and copy to or from the flash device. You can also configure the device to boot from the USB flash drive.

Interface Connections

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 device, ports in different VLANs have to exchange information through a router. By using the device 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 device with no need for an external router.

Figure 1: Connecting VLANs with the Switch



When the Network Advantage license is used on the device or the active device, the device uses the routing method to forward traffic between interfaces. If the Network Essentials license is used on the device or the active device, only basic routing (static routing and RIP) is supported. Whenever possible, to maintain high performance, forwarding is done by the device hardware. However, only IPv4 packets with Ethernet II encapsulation are routed in hardware.

The routing function can be enabled on all SVIs and routed ports. The device 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.

Interface Configuration Mode

The device supports these interface types:

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

You can also configure a range of interfaces.

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

- **Type**—Gigabit Ethernet (GigabitEthernet or gi) for 10/100/1000 Mb/s Ethernet ports, 2.5-Gigabit Ethernet (TwoGigabitEthernet or tw) for 2.5 Gb/s, 5-Gigabit Ethernet (FiveGigabitEthernet or fi) for 5 Gb/s, 10-Gigabit Ethernet (TenGigabitEthernet or te) for 10 Gb/s, 25-Gigabit Ethernet (TwentyFiveGigE or twe) for 25 Gb/s, small form-factor pluggable (SFP) module Gigabit Ethernet and 10-Gigabit Ethernet interfaces and quad small-form-factor pluggable (QSFP) module 40-Gigabit Ethernet (FortyGigabitEthernet or fo) for 40 Gb/s.
- **Switch number**—The number that identifies the given device. The number range is assigned the first time the device initializes.
- **Module number**—The module or slot number on the device: switch (downlink) ports are 0, and uplink ports are 1.
- On a device with SFP uplink ports, the module number is 1 and the port numbers restart. For example, if the device has 24 10/100/1000 ports, the SFP module ports are gigabitethernet1/1/1 through gigabitethernet1/1/4 or tengigabitethernet1/1/1 through tengigabitethernet1/1/4.

You can identify physical interfaces by physically checking the interface location on the device. You can also use the **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 how to identify interfaces on standalone devices:

Breakout Interfaces

support dual mode breakout cables. Breakout cables enable a single 40G QSFP+ interface to be split into four 10G SFP+ interfaces. Dual mode breakout cables support both 4x10G conversion and straight 40G support. Breakout cable support is available on the following switch models and network modules with few [Limitations for Breakout Interfaces](#):

Switch Models

Network Modules

Limitations for Breakout Interfaces

- Only the first four ports support dual mode QSFP breakout cables. See [Configuring Breakout Interfaces, on page 18](#) for the list of configurable interfaces.
- To enable breakout for dual mode QSFP breakout cables, the **hw-module breakout module slot port port-range switch switch-num** command must be configured on the first four ports of the switch. The range for the variables in the **hw-module breakout module slot port port-range switch switch-num** command are given below:
 - *slot* — Slot number of port depending on the chassis model.
 - *port-range* — Single port or range of ports on which breakout is configured. The range varies from 1 to 4.
 - *switch-num* — Switch number in the stack. The range varies from 1 to 8.

See [Configuring Breakout Interfaces, on page 18](#) for the list of configurable interfaces.

Default Ethernet Interface Configuration

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.

This table shows the Ethernet interface default configuration, including some features that apply only to Layer 2 interfaces.

Table 1: Default Layer 2 Ethernet Interface Configuration

Feature	Default Setting
Operating mode	Layer 2 or switching mode (switchport command).
Allowed VLAN range	VLANs 1 to 4094.
Default VLAN (for access ports)	VLAN 1 (Layer 2 interfaces only).
Native VLAN (for IEEE 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.
Duplex mode	Autonegotiate.
Flow control	Flow control is set to receive: on . It is always off for sent packets.
EtherChannel (PAgP)	Disabled on all Ethernet ports.
Port blocking (unknown multicast and unknown unicast traffic)	Disabled (not blocked) (Layer 2 interfaces only).
Broadcast, multicast, and unicast storm control	Disabled.
Protected port	Disabled (Layer 2 interfaces only).
Port security	Disabled (Layer 2 interfaces only).
Port Fast	Disabled.

Feature	Default Setting
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-MDIX is enabled on the switch port.
Power over Ethernet (PoE)	Enabled (auto).

Interface Speed and Duplex Mode

Ethernet interfaces on the switch operate at 10, 100, 1000 Mb/s, 2.5 Gb/s, 5 Gb/s, 10 Gb/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 modules include Gigabit Ethernet (10/100/1000-Mb/s) ports. The switch also includes multigigabit ethernet ports which support speeds up to 2.5 Gb/s (100/1000/2500-Mb/s), 5 Gb/s (100/1000/2500/5000-Mb/s), 10 Gb/s (100/1000/2500/5000/10000-Mb/s); SFP modules that support speeds up to 1 Gb/s, SFP+ modules that support speeds up to 10 Gb/s, SFP28 modules that support speeds up to 25 Gb/s, QSFP modules that support speeds up to 40 Gb/s.

Speed and Duplex Configuration Guidelines

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

- Gigabit Ethernet (10/100/1000-Mb/s) ports and multigigabit ethernet ports (2.5 Gb/s, 5Gb/s, 10 Gb/s) support all speed options and all duplex options (auto, half, and full). However, Gigabit Ethernet ports operating at 1000 Mb/s and above do not support half-duplex mode.
- 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 device can take up to 30 seconds to check for loops. The port LED is amber while STP reconfigures. As best practice, we suggest configuring the speed and duplex options on a link to auto or to fixed on both the ends. If one side of the link is configured to auto and the other side is configured to fixed, the link will not be up and this is expected.
- As best practice, we suggest configuring the speed and duplex options on a link to auto or to fixed on both the ends. If one side of the link is configured to auto and the other side is configured to fixed, the link will not be up and this is expected.



Caution

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

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 The switch ports 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 **on**.

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.

Layer 3 Interfaces

The device 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. You cannot delete interface VLAN 1.



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

When configuring SVIs, you can use the **switchport autostate exclude** command on a port to exclude that port from being included in determining SVI line-state. To disable autostate on the SVI, use the **no autostate** command on the SVI.

- **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.

A Layer 3 device 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 device or in a device 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 device is using its maximum hardware resources, attempts to create a routed port or SVI have these results:

- If you try to create a new routed port, the device 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 device 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 device 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 device sends a message that this was due to insufficient hardware resources.



Note 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:

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

How to Configure Interface Characteristics

Configuring Interfaces

These general instructions apply to all interface configuration processes.

Procedure

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. Enter your password, if prompted.
Step 2	configure terminal Example:	Enters global configuration mode.

	Command or Action	Purpose
	Device# <code>configure terminal</code>	
Step 3	interface Example: Device(config)# <code>interface gigabitethernet1/0/1</code> Device(config-if)#	Identifies the interface type, and the number of the connector. Note You do not need to add a space between the interface type and the 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 4	Follow each interface command with the interface configuration commands that the interface requires.	Defines 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.
Step 5	interface range or interface range macro	(Optional) Configures a range of interfaces. Note Interfaces configured in a range must be the same type and must be configured with the same feature options.
Step 6	show interfaces	Displays 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.

Adding a Description for an Interface

SUMMARY STEPS

1. `enable`
2. `configure terminal`
3. `interface interface-id`
4. `description string`
5. `end`
6. `show interfaces interface-id description`
7. `copy running-config startup-config`

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> <code>enable</code>	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.

	Command or Action	Purpose
Step 2	configure terminal Example: Device# <code>configure terminal</code>	Enters global configuration mode.
Step 3	interface <i>interface-id</i> Example: Device(config)# <code>interface gigabitethernet1/0/2</code>	Specifies the interface for which you are adding a description, and enter interface configuration mode.
Step 4	description <i>string</i> Example: Device(config-if)# <code>description Connects to Marketing</code>	Adds a description for an interface.
Step 5	end Example: Device(config-if)# <code>end</code>	Returns to privileged EXEC mode.
Step 6	show interfaces <i>interface-id</i> <i>description</i>	Verifies your entry.
Step 7	copy running-config startup-config Example: Device# <code>copy running-config startup-config</code>	(Optional) Saves your entries in the configuration file.

Configuring a Range of Interfaces

To configure multiple interfaces with the same configuration parameters, use the **interface range** global configuration command. 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.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface range** {*port-range* | **macro** *macro_name*}
4. **end**
5. **show interfaces** [*interface-id*]
6. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. Enter your password, if prompted.
Step 2	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 3	interface range { <i>port-range</i> macro <i>macro_name</i> } Example: Device(config)# interface range macro	Specifies the range of interfaces (VLANs or physical ports) to be configured, and enter interface-range configuration mode. <ul style="list-style-type: none"> • 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 Configuring and Using Interface Range Macros. • In a comma-separated <i>port-range</i>, you must enter the interface type for each entry and enter spaces before and after the comma. • In a hyphen-separated <i>port-range</i>, you do not need to re-enter the interface type, but you must enter a space before the hyphen. <p>Note Use the normal configuration commands to apply the configuration parameters to all interfaces in the range. Each command is executed as it is entered.</p>
Step 4	end Example: Device(config)# end	Returns to privileged EXEC mode.
Step 5	show interfaces [<i>interface-id</i>] Example: Device# show interfaces	Verifies the configuration of the interfaces in the range.

	Command or Action	Purpose
Step 6	copy running-config startup-config Example: Device# <code>copy running-config startup-config</code>	(Optional) Saves your entries in the configuration file.

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.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **define interface-range** *macro_name* *interface-range*
4. **interface range macro** *macro_name*
5. **end**
6. **show running-config | include define**
7. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> <code>enable</code>	Enables privileged EXEC mode. Enter your password, if prompted.
Step 2	configure terminal Example: Device# <code>configure terminal</code>	Enters global configuration mode.
Step 3	define interface-range <i>macro_name</i> <i>interface-range</i> Example:	Defines the interface-range macro, and saves it in NVRAM. <ul style="list-style-type: none"> • The <i>macro_name</i> is a 32-character maximum character string. • A macro can contain up to five comma-separated interface ranges. • Each <i>interface-range</i> must consist of the same port type.

	Command or Action	Purpose
		Note 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.
Step 4	interface range macro <i>macro_name</i> Example: Device(config)# interface range macro enet_list	Selects the interface range to be configured using the values saved in the interface-range macro called <i>macro_name</i> . You can now use the normal configuration commands to apply the configuration to all interfaces in the defined macro.
Step 5	end Example: Device(config)# end	Returns to privileged EXEC mode.
Step 6	show running-config include define Example: Device# show running-config include define	Shows the defined interface range macro configuration.
Step 7	copy running-config startup-config Example: Device# copy running-config startup-config	(Optional) Saves your entries in the configuration file.

Configuring Ethernet Interfaces

Setting the Interface Speed and Duplex Parameters

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface** *interface-id*
4. **speed** {10 | 100 | 1000 | auto [10 | 100 | 1000 | 10000] | nonegotiate}
5. **duplex** {auto | full | half}
6. **end**
7. **show interfaces** *interface-id*
8. **copy running-config startup-config**
9. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 3	interface <i>interface-id</i> Example: Device(config)# interface gigabitethernet1/0/3	Specifies the physical interface to be configured, and enter interface configuration mode.
Step 4	speed { 10 100 1000 auto [10 100 1000 10000] nonegotiate } Example: Device(config-if)# speed 10	Enter the appropriate speed parameter for the interface: <ul style="list-style-type: none"> • Enter 10, 100, 1000, or 10000 to set a specific speed for the interface. • Enter auto to enable the interface to autonegotiate speed with the connected device. If you specify a speed and also set 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.
Step 5	duplex { auto full half } Example: Device(config-if)# duplex half	Enter the duplex parameter for the interface. Enable half-duplex mode (for interfaces operating only at 10 or 100 Mb/s). Half duplex is not supported on multigigabit ethernet ports configured for speed of 1000 Mb/s. You can configure the duplex setting when the speed is set to auto .
Step 6	end Example: Device(config-if)# end	Returns to privileged EXEC mode.

	Command or Action	Purpose
Step 7	show interfaces <i>interface-id</i> Example: Device# <code>show interfaces gigabitethernet1/0/3</code>	Displays the interface speed and duplex mode configuration.
Step 8	copy running-config startup-config Example: Device# <code>copy running-config startup-config</code>	(Optional) Saves your entries in the configuration file.
Step 9	copy running-config startup-config Example: Device# <code>copy running-config startup-config</code>	(Optional) Saves your entries in the configuration file.

Configuring Breakout Interfaces

For information about device compatibility, see the [Transceiver Module Group \(TMG\) Compatibility Matrix](#).

Configuring IEEE 802.3x Flow Control

SUMMARY STEPS

1. **configure terminal**
2. **interface** *interface-id*
3. **flowcontrol** {receive} {on | off | desired}
4. **end**
5. **show interfaces** *interface-id*
6. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	configure terminal Example: Device# <code>configure terminal</code>	Enters global configuration mode
Step 2	interface <i>interface-id</i> Example:	Specifies the physical interface to be configured, and enters interface configuration mode.

	Command or Action	Purpose
	<code>Device(config)# interface gigabitethernet1/0/1</code>	
Step 3	flowcontrol {receive} {on off desired} Example: <code>Device(config-if)# flowcontrol receive on</code>	Configures the flow control mode for the port.
Step 4	end Example: <code>Device(config-if)# end</code>	Returns to privileged EXEC mode.
Step 5	show interfaces interface-id Example: <code>Device# show interfaces gigabitethernet1/0/1</code>	Verifies the interface flow control settings.
Step 6	copy running-config startup-config Example: <code>Device# copy running-config startup-config</code>	(Optional) Saves your entries in the configuration file.

Configuring Layer 3 Interfaces

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface {gigabitethernet interface-id} | {vlan vlan-id} | {port-channel port-channel-number}**
4. **no switchport**
5. **ip address ip_address subnet_mask**
6. **no shutdown**
7. **end**
8. **show interfaces [interface-id]**
9. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable	Enables privileged EXEC mode.

	Command or Action	Purpose
	Example: Device> enable	Enter your password, if prompted.
Step 2	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 3	interface { gigabitethernet <i>interface-id</i> } { vlan <i>vlan-id</i> } { port-channel <i>port-channel-number</i> } Example: Device(config)# interface gigabitethernet1/0/2	Specifies the interface to be configured as a Layer 3 interface, and enters interface configuration mode.
Step 4	no switchport Example: Device(config-if)# no switchport	(For physical ports only) Enters Layer 3 mode.
Step 5	ip address <i>ip_address subnet_mask</i> Example: Device(config-if)# ip address 192.20.135.21 255.255.255.0	Configures the IP address and IP subnet.
Step 6	no shutdown Example: Device(config-if)# no shutdown	Enables the interface.
Step 7	end Example: Device(config-if)# end	Returns to privileged EXEC mode.
Step 8	show interfaces [<i>interface-id</i>]	Verifies the configuration.
Step 9	copy running-config startup-config Example: Device# copy running-config startup-config	(Optional) Saves your entries in the configuration file.

Configuring a Logical Layer 3 GRE Tunnel Interface

Before you begin

Generic Routing Encapsulation (GRE) is a tunneling protocol used to encapsulate network layer protocols inside virtual point-to-point links. A GRE tunnel only provides encapsulation and not encryption.



Note

- GRE tunnels are supported on the hardware on Cisco Catalyst 9000 switches. When GRE is configured without tunnel options, packets are hardware-switched. When GRE is configured with tunnel options (such as key, checksum, and so on), packets are switched in the software. A maximum of 1000 GRE tunnels are supported.
- Other features such as Access Control Lists (ACL) and Quality of Service (QoS) are not supported for the GRE tunnels.
- The **tunnel path-mtu-discovery** command is not supported for GRE tunnels. To avoid fragmentation, you can set the maximum transmission unit (MTU) of both ends of the GRE tunnel to the lowest value by using the **ip mtu 256** command.

To configure a GRE tunnel, perform this task:

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface tunnel** *number*
4. **ip address** *ip_address*/*subnet_mask*
5. **tunnel source** {*ip_address* | *type_number*}
6. **tunnel destination** {*host_name* | *ip_address*}
7. **tunnel mode gre ip**
8. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. Enter your password, if prompted.
Step 2	configure terminal Example: Device# configure terminal	Enters global configuration mode.

	Command or Action	Purpose
Step 3	interface tunnel <i>number</i> Example: Device(config)# interface tunnel 2	Enables tunneling on the interface.
Step 4	ip address <i>ip_address</i> <i>subnet_mask</i> Example: Device(config)# ip address 100.1.1.1 255.255.255.0	Configures the IP address and IP subnet.
Step 5	tunnel source { <i>ip_address</i> <i>type_number</i> } Example: Device(config)# tunnel source 10.10.10.1	Configures the tunnel source.
Step 6	tunnel destination { <i>host_name</i> <i>ip_address</i> } Example: Device(config)# tunnel destination 10.10.10.2	Configures the tunnel destination.
Step 7	tunnel mode gre ip Example: Device(config)# tunnel mode gre ip	Configures the tunnel mode.
Step 8	end Example: Device(config)# end	Exits configuration mode.

Configuring SVI Autostate Exclude

SUMMARY STEPS

1. enable
2. configure terminal
3. interface *interface-id*
4. switchport autostate exclude
5. end
6. show running config interface *interface-id*
7. copy running-config startup-config

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example:	Enables privileged EXEC mode. Enter your password, if prompted.

	Command or Action	Purpose
	Device> enable	
Step 2	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 3	interface <i>interface-id</i> Example: Device(config)# interface gigabitethernet1/0/2	Specifies a Layer 2 interface (physical port or port channel), and enters interface configuration mode.
Step 4	switchport autostate exclude Example: Device(config-if)# switchport autostate exclude	Excludes the access or trunk port when defining the status of an SVI line state (up or down)
Step 5	end Example: Device(config-if)# end	Returns to privileged EXEC mode.
Step 6	show running config interface <i>interface-id</i>	(Optional) Shows the running configuration. Verifies the configuration.
Step 7	copy running-config startup-config Example: Device# copy running-config startup-config	(Optional) Saves your entries in the configuration file.

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.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface {vlan *vlan-id*} | { gigabitethernet *interface-id*} | {port-channel *port-channel-number*}**

4. **shutdown**
5. **no shutdown**
6. **end**
7. **show running-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. Enter your password, if prompted.
Step 2	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 3	interface { <i>vlan vlan-id</i> } { gigabitethernet <i>interface-id</i> } { port-channel <i>port-channel-number</i> } Example: Device(config)# interface gigabitethernet1/0/2	Selects the interface to be configured.
Step 4	shutdown Example: Device(config-if)# shutdown	Shuts down an interface.
Step 5	no shutdown Example: Device(config-if)# no shutdown	Restarts an interface.
Step 6	end Example: Device(config-if)# end	Returns to privileged EXEC mode.
Step 7	show running-config Example: Device# show running-config	Verifies your entries.

Configuring USB Inactivity Timeout

The configurable inactivity timeout reactivates the RJ-45 console port if the USB console port is activated but no input activity occurs on it for a specified time period. When the USB console port is deactivated due to a timeout, you can restore its operation by disconnecting and reconnecting the USB cable.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **line console 0**
4. **usb-inactivity-timeout switch** *switch_number timeout-minutes*
5. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. Enter your password, if prompted.
Step 2	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 3	line console 0 Example: Device(config)# line console 0	Configures the console and enters line configuration mode.
Step 4	usb-inactivity-timeout switch <i>switch_number timeout-minutes</i> Example: Device(config-line)# usb-inactivity-timeout switch 1 30	Specifies an inactivity timeout for the console port. The range is 1 to 240 minutes. The default is to have no timeout configured.
Step 5	copy running-config startup-config Example: Device# copy running-config startup-config	(Optional) Saves your entries in the configuration file.

Monitoring Interface Characteristics

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 2: Show Commands for Interfaces

Command	Purpose
show interfaces <i>interface-id</i> status [err-disabled]	Displays interface status or a list of interfaces in the error-disabled state.
show interfaces [<i>interface-id</i>] switchport	Displays administrative and operational status of switching (nonrouting) ports. You can use this command to find out if a port is in routing or in switching mode.
show interfaces [<i>interface-id</i>] description	Displays the description configured on an interface or all interfaces and the interface status.
show ip interface [<i>interface-id</i>]	Displays the usability status of all interfaces configured for IP routing or the specified interface.
show interface [<i>interface-id</i>] stats	Displays the input and output packets by the switching path for the interface.
show interfaces <i>interface-id</i>	(Optional) Displays speed and duplex on the interface.
show interfaces transceiver dom-supported-list	(Optional) Displays Digital Optical Monitoring (DOM) status on the connect SFP modules.
show interfaces transceiver properties	(Optional) Displays temperature, voltage, or amount of current on the interface.
show interfaces [<i>interface-id</i>] [{ transceiver properties detail }] <i>module number</i>	Displays physical and operational status about an SFP module.
show running-config interface [<i>interface-id</i>]	Displays the running configuration in RAM for the interface.
show version	Displays the hardware configuration, software version, the names and sources of configuration files, and the boot images.
show controllers ethernet-controller <i>interface-id</i> phy	Displays the operational state of the auto-MDIX feature on the interface.

Clearing and Resetting Interfaces and Counters

Table 3: Clear Commands for Interfaces

Command	Purpose
clear counters [<i>interface-id</i>]	Clears interface counters.
clear interface <i>interface-id</i>	Resets the hardware logic on an interface.
clear line [<i>number</i> console 0 <i>vtty number</i>]	Resets the hardware logic on an asynchronous serial line.



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.

Configuration Examples for Interface Characteristics

Adding a Description to an Interface: Example

Configuring a Range of Interfaces: Examples

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: Examples

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:

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

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

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

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

Setting Interface Speed and Duplex Mode: Example

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

```
Device# configure terminal
Device(config)# interface gigabitethernet1/0/3
Device(config-if)# speed 10
Device(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:

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

Configuring Layer 3 Interfaces: Example

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

Configuring Breakout Interfaces : Example

The following is a sample output of **show interface status** command with 40G QSFP module inserted into port number 1, 3, 11 and 12.

```
Device# configure terminal
```

Port	Name	Status	Vlan	Duplex	Speed	Type
Fo2/0/1		notconnect	4094	full	40G	QSFP
	H40G AOCxM SFP					
Fo2/0/2		notconnect	1	auto	auto	unknown
Fo2/0/3		notconnect	1	full	40G	QSFP
	40G SR4 SFP					
Fo2/0/4		notconnect	1	auto	auto	unknown
Fo2/0/5		notconnect	1	auto	auto	unknown
Fo2/0/6		notconnect	1	auto	auto	unknown
Fo2/0/7		notconnect	1	auto	auto	unknown
Fo2/0/8		notconnect	1	auto	auto	unknown
Fo2/0/9		notconnect	1	auto	auto	unknown
Fo2/0/10		notconnect	1	auto	auto	unknown
Fo2/0/11		connected	1	full	40G	QSFP
	H40G AOCxM SFP					

```

Fo2/0/12                connected    1                full    40G QSFP
H40G AOCxM SFP
Fo2/0/13                notconnect    1                full    10G unknown
Fo2/0/14                notconnect    1                full    10G unknown
.....
.....
.....
..... (Output truncated).....
    
```

The following is a sample output of **show interface status** command when 40G QSFP module inserted in port number 3 is removed and 4x10G breakout cable is inserted into port number 3 after using the command **hw-mod breakout module 1 port 3 switch 2**. Port number 3 — Fo2/0/3 — is split into four 10G ports — Te2/0/9, Te2/0/10, Te2/0/11 and Te2/0/12.

```

Device# configure terminal
Device (config)# hw-mod breakout module 1 port 3 switch 2
Device (config)#
*May  5 22:33:20.939 UTC: %PLATFORM_PM-6-MODULE_REMOVED: SFP module with
interface name Fo2/0/3 removed
*May  5 22:33:23.721 UTC: %PLATFORM_PM-6-FRULINK_REMOVED: 1x40G Port3
uplink module removed from switch 2 slot 1
*May  5 22:33:25.608 UTC: %PLATFORM_PM-6-FRULINK_INSERTED: BC:4x10G Port3
uplink module inserted in the switch 2 slot 1
*May  5 22:33:25.722 UTC: %LINK-3-UPDOWN: Interface
FortyGigabitEthernet2/0/3, changed state to down
*May  5 22:33:30.275 UTC: %PLATFORM_PM-6-MODULE_INSERTED: SFP module
inserted with interface name Te2/0/9
*May  5 22:33:32.509 UTC: %PLATFORM_PM-6-MODULE_INSERTED: SFP module
inserted with interface name Te2/0/10
*May  5 22:33:34.744 UTC: %PLATFORM_PM-6-MODULE_INSERTED: SFP module
inserted with interface name Te2/0/11
*May  5 22:33:36.979 UTC: %PLATFORM_PM-6-MODULE_INSERTED: SFP module
inserted with interface name Te2/0/12
Device (config)# end
Device # show interface status

*May  5 22:33:47.297 UTC: %SYS-5-CONFIG_I: Configured from console by
consoleface status
    
```

```

Port      Name                Status      Vlan      Duplex  Speed Type
Fo2/0/1   H40G AOCxM SFP      notconnect  4094      full    40G QSFP
Fo2/0/2   notconnect          1          auto     auto   unknown
Fo2/0/4   notconnect          1          auto     auto   unknown
Fo2/0/5   notconnect          1          auto     auto   unknown
Fo2/0/6   notconnect          1          auto     auto   unknown
Fo2/0/7   notconnect          1          auto     auto   unknown
Fo2/0/8   notconnect          1          auto     auto   unknown
Fo2/0/9   notconnect          1          auto     auto   unknown
Fo2/0/10  notconnect          1          auto     auto   unknown
Fo2/0/11  connected           1          full     40G QSFP
H40G AOCxM SFP
Fo2/0/12  connected           1          full     40G QSFP
    
```

```

H40G AOCxM SFP
Fo2/0/13                notconnect 1          full  10G unknown
Fo2/0/14                notconnect 1          full  10G unknown
Fo2/0/15                notconnect 1          auto  auto unknown
.....
.....
..... (Output truncated) .....
Te2/0/9                 notconnect 1          full  40G
Te2/0/10                notconnect 1          full  40G
Te2/0/11                notconnect 1          full  40G QSFP
40G SR4 SFP
Te2/0/12                notconnect 1          full  40G
.....
.....
..... (Output truncated) .....

```

Example: Configuring the USB Inactivity Timeout

The following example shows how to configure the inactivity timeout to 30 minutes:

```

Device# configure terminal
Device(config)# line console 0
Device(config-line)# usb-inactivity-timeout switch 1 30

```

The following example shows how to disable the configuration:

```

Device# configure terminal
Device(config)# line console 0
Device(config-line)# no usb-inactivity-timeout switch 1

```

If there is no (input) activity on a USB console port for the configured number of minutes, the inactivity timeout setting applies to the RJ-45 port, and a log shows this occurrence:

```
*Mar  1 00:47:25.625: %USB_CONSOLE-6-INACTIVITY_DISABLE: Console media-type USB disabled
due to inactivity, media-type reverted to RJ45.
```

At this point, the only way to reactivate the USB console port is to disconnect and reconnect the cable.

When the USB cable on the switch has been disconnected and reconnected, a log similar to this appears:

```
*Mar  1 00:48:28.640: %USB_CONSOLE-6-MEDIA_USB: Console media-type is USB.
```

Feature History for Configuring Interface Characteristics

This table provides release and related information for the features explained in this module.

These features are available in all the releases subsequent to the one they were introduced in, unless noted otherwise.

Release	Feature	Feature Information
Cisco IOS XE Everest 16.6.1	Interface Characteristics	Interface Characteristics includes interface types, connections, configuration modes, speed, and other aspects of configuring a physical interface on a device.
Cisco IOS XE Everest 16.6.4	IEEE 802.3x Flow Control	The default value for flowcontrol interface configuration command was modified to on on all the models of the series.

Use the Cisco Feature Navigator to find information about platform and software image support. To access Cisco Feature Navigator, go to <http://www.cisco.com/go/cfn>.

