Configuring Interfaces

This chapter contains the following topics:

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest feature information and caveats, see the release notes for your platform and software release. Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to http://www.cisco.com/go/cfn. An account on Cisco.com is not required.

Pre-requisites for Configuring Interfaces

You can define the wireless management, AP-manager, virtual, and management interface parameters using the Startup Wizard. However, you can display and configure interface parameters through either the GUI or CLI after the controller is running.

For Cisco 5700 Series Controllers in a non-link-aggregation (non-LAG) configuration, the management interface must be on a different VLAN than any dynamic AP-manager interface. Otherwise, the management interface cannot fail over to the port that the AP-manager is on.

To configure interfaces, you must configure the default gateway, router, and the IP route using the following commands:

- **ip default-gateway** 154.4.0.1
- **default-router** 154.51.0.1
- **ip route** 0.0.0.0 0.0.0.0 154.4.0.1

Restrictions for Configuring Interfaces

Information About Interfaces

An interface is a logical entity on the controller. An interface has multiple parameters associated with it, including an IP address, default gateway, VLAN identifier, and DHCP server. The following interfaces available on the controller:

- Wireless Management Interface
- AP Manager Interface
- Dynamic Interface

The wireless management interface is used for access point join functions, mobility, RRM , and also used for peer connections (MC - MC connections) and MC to MA connections.

Typically, you define the management, AP-manager, virtual, and service-port interface parameters using the Startup Wizard. However, you can display and configure interface parameters through either the GUI or CLI after the controller is running.

Interface Types

This section describes the different types of interfaces supported by the controller. 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. VLANs can be formed with ports across the stack.

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 controller 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.

In a switch stack, the VLAN database is downloaded to all switches in a stack, and all switches in the stack build the same VLAN database. The running configuration and the saved configuration are the same for all switches in a stack.

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

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

Ports

ports are Layer 2-only interfaces associated with a physical port. ports belong to one or more VLANs. A controller port can be an access port, a trunk port, or a tunnel 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. You must manually configure tunnel ports as part of an asymmetric link connected to an IEEE 802.1Q trunk port. ports are used for managing the physical interface and associated Layer 2 protocols and do not handle routing or bridging.

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

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.

Two types of access ports are supported:

- Static access ports are manually assigned to a VLAN (or through a RADIUS server for use with IEEE 802.1x.
- VLAN membership of dynamic access ports is learned through incoming packets. By default, a dynamic access port is not a member of any VLAN, and forwarding to and from the port is enabled only when the VLAN membership of the port is discovered. Dynamic access ports on the controller are assigned to a VLAN by a VLAN Membership Policy Server (VMPS). The VMPS can be a Catalyst 6500 series switch; the controller 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.

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 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 IP Base image supports static routing and the Routing Information Protocol (RIP). For full Layer 3 routing or for fallback bridging, you must enable the IP Services image on the standalone controller, or the active controller.

Switch Virtual Interfaces

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

Note You cannot delete interface VLAN 1.

SVIs provide IP host connectivity only to the system; in Layer 3 mode, you can configure routing across SVIs.

Although the switch stack or controller 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.

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.

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

SVIs support routing protocols and bridging configurations.

Note The IP base feature set supports static routing and RIP. For more advanced routing or for fallback bridging, enable the IP services feature set on the standalone switch or the active switch. For information about using the software activation feature to install a software license for a specific feature set, see the *Cisco IOS Software Activation* document.

SVI Autostate Exclude

The line state of an SVI with multiple ports on a VLAN is in the *up* state when it meets these conditions:

- The VLAN exists and is active in the VLAN database on the controller
- The VLAN interface exists and is not administratively down.
- At least one Layer 2 (access or trunk) port exists, has a link in the *up* state on this VLAN, and is in the spanning-tree forwarding state on the VLAN.

Note The protocol link state for VLAN interfaces come up when the first switchport belonging to the corresponding VLAN link comes up and is in STP forwarding state.

The default action, when a VLAN has multiple ports, is that the SVI goes down when all ports in the VLAN go down. You can use the SVI autostate exclude feature to configure a port so that it is not included in the SVI line-state up-or-down calculation. For example, if the only active port on the VLAN is a monitoring port, you might configure autostate exclude on that port so that the VLAN goes down when all other ports go down. When enabled on a port, **autostate exclude** applies to all VLANs that are enabled on that port.

The VLAN interface is brought up when one Layer 2 port in the VLAN has had time to converge (transition from STP listening-learning state to forwarding state). This prevents features such as routing protocols from using the VLAN interface as if it were fully operational and minimizes other problems, such as routing black holes.

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 controllers or between controllers 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.

10-Gigabit Ethernet Interfaces

A 10-Gigabit Ethernet interface operates only in full-duplex mode. The interface can be configured as a switched or routed port.

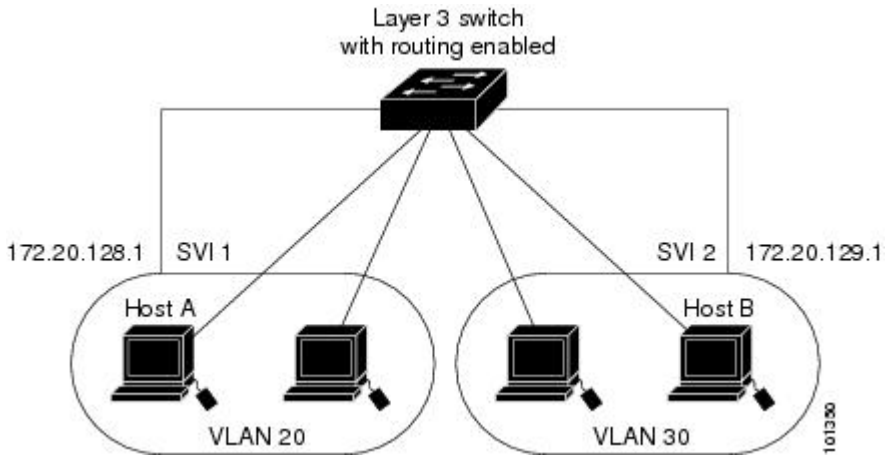
For more information about the Cisco TwinGig Converter Module, see the controller hardware installation guide and your transceiver module documentation.

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

Figure 1: Connecting VLANs with the Switch



Interface Configuration Mode

The controller supports these interface types:

- Physical ports—controller 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, stack member number (only stacking-capable switches), module number, and controller port number, and enter interface configuration mode.

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

  You can use the switch port LEDs in Stack mode to identify the stack member number of a controller.
- Module number—The module or slot number on the controller: switch (downlink) ports are 0, and uplink ports are 1.
- Port number—The interface number on the controller. The 10/100/1000 port numbers always begin at 1, starting with the far left port when facing the front of the controller, for example, gigabitethernet1/0/1 or gigabitethernet1/0/8.

You can identify physical interfaces by physically checking the interface location on the controller. 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 a stacking-capable controller:

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

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

- To configure 10-Gigabit Ethernet port 1 on a standalone controller, enter this command:

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

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

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

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– 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.

Feature	Default Setting
Speed	Autonegotiate. (Not supported on the 10-Gigabit interfaces.)
Duplex mode	Autonegotiate. (Not supported on the 10-Gigabit interfaces.)
Flow control	Flow control is set to **receive: off**. It is always off for sent packets.
EtherChannel (PAgP)	Disabled on all Ethernet ports.
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.
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.
Power over Ethernet (PoE)	Enabled (auto).

Layer 3 Interfaces

The controller 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 also configure SVI autostate exclude on a port in the SVI to exclude that port from being included in determining SVI line-state status.

- 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 controller 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 controller or in a controller 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 controller 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 controller 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 controller 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 controller 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 controller sends a message that this was due to insufficient hardware resources.

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

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

Configuring Interfaces

This module lists the generic steps used to configure any interface on the controller. You must use the following steps to configure interfaces on the controller:

Before You Begin

-

SUMMARY STEPS

1. **configure terminal**
2. **global configuration**
3. **interface**
4. **show interface summary**
5. **show interface detail management**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:**	Enables you to enter configure terminal configured mode at the privileged prompt.
Step 2	**global configuration** **Example:** global configuration	Identify interface details, for example the interface type, connector, and so on and enter global configuration mode. Enables you to identify the interface and enter global configuration mode.
Step 3	**interface** **Example:**	Follow each **interface** command with the interface configuration commands that the interface requires. The commands that you enter define the protocols and applications that will run on the configuration commands. Interfaces configured in a range must be the same type and must be configured with the same feature options. The commands are collected and applied to the interface when you enter another **interface** command or enter end to return to privileged EXEC mode. Enables you to configure the supported interfaces on the controller.
Step 4	**show interface summary** **Example:**	Verify the status of the configured interface using the **show interface summary**. Enables you to view the status of the configured interface.
Step 5	**show interface detail management** **Example:**	Verify the status of the configured interface using the **show interface detail management**. Enables you to view the status of the configured 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:** `Controller> enable`	Enables privileged EXEC mode. Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Controller# configure terminal`	Enters global configuration mode.
Step 3	**interface** *interface-id* **Example:** `Controller(config)# interface gigabitethernet1/0/2`	Specifies the interface for which you are adding a description, and enter interface configuration mode.
Step 4	**description** *string* **Example:** `Controller(config-if)# description Connects to Marketing`	Adds a description (up to 240 characters) for an interface.
Step 5	**end** **Example:** `Controller(config-if)# end`	Returns to privileged EXEC mode.

	Command or Action	Purpose
Step 6	**show interfaces** *interface-id* **description**	Verifies your entry.
Step 7	**copy running-config startup-config** **Example:** `Controller# copy running-config startup-config`	(Optional) Saves your entries in the configuration file.

Configuring a Range of Interfaces: Examples

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

```
Controller# configure terminal
Controller(config)# interface range gigabitethernet1/0/1 - 4
Controller(config-if-range)# speed 100
```

This example shows how to use a comma to add different interface type strings to the range to enable Gigabit Ethernet ports 1 to 3 and 10-Gigabit Ethernet ports 1 and 2 to receive flow-control pause frames:

```
Controller# configure terminal
Controller(config)# interface range gigabitethernet1/0/1 - 3 , tengigabitethernet1/0/1 - 2
Controller(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: 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:

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

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

```
Controller# configure terminal
Controller(config)# define interface-range macro1 gigabitethernet1/0/1 - 2,
gigabitethernet1/0/5 - 7, tengigabitethernet1/0/1 -2
Controller(config)# end
```

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

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

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

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

Configuring Interfaces

These general instructions apply to all interface configuration processes.

DETAILED STEPS

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

Step 3	**interface**
	
	Example:
	
	`Controller(config)# interface gigabitethernet1/0/1`
	`Controller(config-if)#`
		Identifies the interface type, the controller number (only on stacking-capable switches), 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.

	Command or Action	Purpose
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.

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** **Example:** `Controller> enable`	Enables privileged EXEC mode. Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Controller# configure terminal`	Enters global configuration mode.
Step 3	**interface** {**gigabitethernet** *interface-id*} | {**vlan** *vlan-id*} | {**port-channel** *port-channel-number*} **Example:** `Controller(config)# interface gigabitethernet1/0/2`	Specifies the interface to be configured as a Layer 3 interface, and enter interface configuration mode.

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

	Command or Action	Purpose
Step 6	**end** **Example:** `Controller(config-if)# end`	Returns to privileged EXEC mode.
Step 7	**show running-config** **Example:** `Controller# show running-config`	Verifies your entries.

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 *interface-id* **status** [**err-disabled**]	Displays interface status or a list of interfaces in the error-disabled state.
show interfaces [*interface-id*] **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 [*interface-id*] **description**	Displays the description configured on an interface or all interfaces and the interface status.
show ip interface [*interface-id*]	Displays the usability status of all interfaces configured for IP routing or the specified interface.
show interface [*interface-id*] **stats**	Displays the input and output packets by the switching path for the interface.
show interfaces *interface-id*	(Optional) Displays speed and duplex on the interface.

Command	Purpose
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 [*interface-id*] [{**transceiver properties** | **detail**}] *module number*]	Displays physical and operational status about an SFP module.
show running-config interface [*interface-id*]	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 *interface-id* **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 [*interface-id*]	Clears interface counters.
clear interface *interface-id*	Resets the hardware logic on an interface.
clear line [*number* | **console 0** | **vty** *number*]	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.

Viewing Wireless Interfaces (GUI)

You can view the wireless interfaces available in the controller by choosing **Monitor > Controller > System > Wireless Interface.** in the controller web UI. The following details of the wireless interface page are displayed.

Parameter	Description
Interface Type	Displays the operator-defined interface type. Values are as follows: • Static—Wireless Management. • AP-Manager. • Service-Port—The Ten Gigabit Ethernet port located on the back of the controller • Virtual interfaces.
Interface Name	Displays the name of the interface. Values are as follows: • Management—802.11 distribution system wired network. • Service-port—System service interface. • Virtual—Loopback interface for the web interface to work. This is available in the controller by default. You need not explicitly configure this interface. • AP-manager—Can be on the same subnet as the management IP address, but must have a different IP address than the management interface. • name—Operator-defined interface assignment, without any spaces.
IP Address	Displays the IP address of the Controller and its distribution port.
IP Netmask	Displays the destination subnet mask.
MAC Address	Displays the MAC address of the interface.
VLAN ID	Displays the virtual LAN assignment of the interface.

Configuring Ports (GUI)

You can configure ports in controller using the web UI. To do this, you must follow the steps defined in this module in the web UI.

You can create the following types of port using the controller web UI.

- Loopback Interfaces
- EtherChannel Port
- Ten Gigabit Ethernet Interfaces
- Gigabit Ethernet Interfaces

SUMMARY STEPS

1. Choose **Configuration > Controller > System > Interfaces** > **Port Summary**.
2. Click on the port in the port summary table to view the details of the selected port.
3. Click **Apply**.

DETAILED STEPS

Step 1 Choose **Configuration > Controller > System > Interfaces** > **Port Summary**.
Displays all the ports and details of the ports in the controller.

Step 2 Click on the port in the port summary table to view the details of the selected port.
The Edit Port details page appears. To edit the values listed in the page, enter values for the parameters listed in the Edit page.

Note You must configure the selected port as a Layer2 or Layer3 interface.

Step 3 Click **Apply**.

Configuring Wireless Interface (GUI)

You can configure wireless interface the in controller using the web user interface (GUI). To do this, you must follow the steps defined in this module in the GUI.

SUMMARY STEPS

1. Choose **Configuration > Controller > System > Interfaces** > **Wireless Summary.**
2. Click **New**.
3. Select the interface to configure the AP management interface(s) and management interface.
4. Click **Apply**.

DETAILED STEPS

Step 1 Choose **Configuration > Controller > System > Interfaces** > **Wireless Summary.**
Displays all the wireless interfaces and details of the interfaces in the controller.

Step 2 Click **New**.
The New page appears.

Step 3 Select the interface to configure the AP management interface(s) and management interface.
You can configure one management and one or multiple AP management interfaces in the controller using the web UI.

Step 4 Click **Apply**.

Feature History and Information For Configuring Interfaces

Command History

Release	Modification
Cisco IOS XE 3.2SE	This command was introduced.