<p style="text-align: right;">C H A P T E R **6**</p>

# Configuring Interfaces

This chapter describes how to configure interfaces for the Catalyst 4500 series switches. It also provides guidelines, procedures, and configuration examples.

This chapter includes the following major sections:

> **Note** For complete syntax and usage information for the switch commands used in this chapter, first look at the *Cisco Catalyst 4500 Series Switch Command Reference* and related publications at this location:
>
> http://www.cisco.com/en/US/products/hw/switches/ps4324/index.html
>
> If the command is not found in the Catalyst 4500 Series Switch Command Reference, it will be found in the larger Cisco IOS library. Refer to the *Catalyst 4500 Series Switch Cisco IOS Command Reference* and related publications at this location:
>
> http://www.cisco.com/en/US/products/ps6350/index.html

# Overview of Interface Configuration

By default, all interfaces are enabled. The 10/100-Mbps Ethernet interfaces autonegotiate connection speed and duplex. The 10/100/1000-Mbps Ethernet interfaces negotiate speed, duplex, and flow control. The 1000-Mbps Ethernet interfaces negotiate flow control only. Autonegotiation automatically selects the fastest speed possible on that port for the given pair. If a speed is explicitly stated for an interface, that interface will default to half duplex unless it is explicitly set for full duplex.

Many features are enabled on a per-interface basis. When you enter the **interface** command, you must specify the following:

- Interface type:
  - Fast Ethernet (use the **fastethernet** keyword)
  - Gigabit Ethernet (use the **gigabitethernet** keyword)
  - 10-Gigabit Ethernet (use the **tengigabitethernet** keyword)
- Slot number—The slot in which the interface module is installed. Slots are numbered starting with 1, from top to bottom.
- Interface number—The interface number on the module. The interface numbers always begin with 1. When you are facing the front of the switch, the interfaces are numbered from left to right.

You can identify interfaces by physically checking the slot/interface location on the switch. You can also use the Cisco IOS **show** commands to display information about a specific interface or all the interfaces.

# Using the interface Command

These general instructions apply to all interface configuration processes:

**Step 1**   At the privileged EXEC prompt, enter the **configure terminal** command to enter global configuration mode:

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

**Step 2**   In global configuration mode, enter the **interface** command. Identify the interface type and the number of the connector on the interface card. The following example shows how to select Fast Ethernet, slot 5, interface 1:

```
Switch(config)# interface fastethernet 5/1
Switch(config-if)#
```

**Step 3**   Interface numbers are assigned at the factory at the time of installation or when modules are added to a system. Enter the **show interfaces** EXEC command to see a list of all interfaces installed on your switch. A report is provided for each interface that your switch supports, as shown in this display:

```
Switch(config-if)#Ctrl-Z
Switch#show interfaces
Vlan1 is up, line protocol is down
  Hardware is Ethernet SVI, address is 0004.dd46.7aff (bia 0004.dd46.7aff)
  MTU 1500 bytes, BW 1000000 Kbit, DLY 10 usec,
     reliability 255/255, txload 1/255, rxload 1/255
  Encapsulation ARPA, loopback not set
  ARP type: ARPA, ARP Timeout 04:00:00
  Last input never, output never, output hang never
```

```
       Last clearing of "show interface" counters never
       Input queue: 0/75/0/0 (size/max/drops/flushes); Total output drops: 0
       Queueing strategy: fifo
       Output queue: 0/40 (size/max)
       5 minute input rate 0 bits/sec, 0 packets/sec
       5 minute output rate 0 bits/sec, 0 packets/sec
          0 packets input, 0 bytes, 0 no buffer
          Received 0 broadcasts, 0 runts, 0 giants, 0 throttles
          0 input errors, 0 CRC, 0 frame, 0 overrun, 0 ignored
          0 packets output, 0 bytes, 0 underruns
          0 output errors, 0 interface resets
          0 output buffer failures, 0 output buffers swapped out
GigabitEthernet1/1 is up, line protocol is down
       Hardware is Gigabit Ethernet Port, address is 0004.dd46.7700 (bia 0004.dd46.7700)
       MTU 1500 bytes, BW 1000000 Kbit, DLY 10 usec,
          reliability 255/255, txload 1/255, rxload 1/255
       Encapsulation ARPA, loopback not set
       Keepalive set (10 sec)
       Auto-duplex, Auto-speed
       ARP type: ARPA, ARP Timeout 04:00:00
       Last input never, output never, output hang never
       Last clearing of "show interface" counters never
       Input queue: 0/2000/0/0 (size/max/drops/flushes); Total output drops: 0
       Queueing strategy: fifo
       Output queue: 0/40 (size/max)
       5 minute input rate 0 bits/sec, 0 packets/sec
       5 minute output rate 0 bits/sec, 0 packets/sec
          0 packets input, 0 bytes, 0 no buffer
          Received 0 broadcasts, 0 runts, 0 giants, 0 throttles
          0 input errors, 0 CRC, 0 frame, 0 overrun, 0 ignored
          0 input packets with dribble condition detected
          0 packets output, 0 bytes, 0 underruns
          0 output errors, 0 collisions, 0 interface resets
          0 babbles, 0 late collision, 0 deferred
          0 lost carrier, 0 no carrier
          0 output buffer failures, 0 output buffers swapped out
GigabitEthernet1/2 is up, line protocol is down
       Hardware is Gigabit Ethernet Port, address is 0004.dd46.7701 (bia 0004.dd46.7701)
       MTU 1500 bytes, BW 1000000 Kbit, DLY 10 usec,
          reliability 255/255, txload 1/255, rxload 1/255
       Encapsulation ARPA, loopback not set
       Keepalive set (10 sec)
       Auto-duplex, Auto-speed
       ARP type: ARPA, ARP Timeout 04:00:00
       Last input never, output never, output hang never
       Last clearing of "show interface" counters never
       Input queue: 0/2000/0/0 (size/max/drops/flushes); Total output drops: 0
       Queueing strategy: fifo
       Output queue: 0/40 (size/max)
       5 minute input rate 0 bits/sec, 0 packets/sec
       5 minute output rate 0 bits/sec, 0 packets/sec
          0 packets input, 0 bytes, 0 no buffer
          Received 0 broadcasts, 0 runts, 0 giants, 0 throttles
          0 input errors, 0 CRC, 0 frame, 0 overrun, 0 ignored
          0 input packets with dribble condition detected
          0 packets output, 0 bytes, 0 underruns
          0 output errors, 0 collisions, 0 interface resets
          0 babbles, 0 late collision, 0 deferred
          0 lost carrier, 0 no carrier
          0 output buffer failures, 0 output buffers swapped out
--More--
<...output truncated...>
```

**Step 4**  To begin configuring Fast Ethernet interface 5/5, as shown in the following example, enter the **interface** keyword, interface type, slot number, and interface number in global configuration mode:

```
Switch# configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
Switch(config)# interface fastethernet 5/5
Switch(config-if)#
```

**Note**  You do not need to add a space between the interface type and interface number. For example, in the preceding line you can specify either *fastethernet 5/5* or *fastethernet5/5*.

**Step 5**  Follow each **interface** command with the interface configuration commands your particular interface requires. The commands you enter define the protocols and applications that will run on the interface. The commands are collected and applied to the **interface** command until you enter another **interface** command or press **Ctrl-Z** to exit interface configuration mode and return to privileged EXEC mode.

**Step 6**  After you configure an interface, check its status by using the EXEC **show** commands listed in the "Monitoring and Maintaining the Interface" section on page 6-30.

# Configuring a Range of Interfaces

The interface-range configuration mode allows you to configure multiple interfaces with the same configuration parameters. When you enter the interface-range configuration mode, all command parameters you enter are attributed to all interfaces within that range until you exit interface-range configuration mode.

To configure a range of interfaces with the same configuration, perform this task:

| Command | Purpose |
|---|---|
| Switch(config)# **interface range** {**vlan** *vlan_ID - vlan_ID*} \| {{**fastethernet** \| **gigabitethernet** \| **tengigabitethernet** \| **macro** *macro_name*} *slot/interface - interface*} [**,** {**vlan** *vlan_ID - vlan_ID*} {{**fastethernet** \| **gigabitethernet** \| **tengigabitethernet** \| **macro** *macro_name*} *slot/interface - interface*}] | Selects the range of interfaces to be configured. Note the following: <br><br> • You are required to enter a space before the dash. <br><br> • You can enter up to five comma-separated ranges. <br><br> • You are not required to enter spaces before or after the comma. |

**Note**  When you use the **interface range** command, you must add a space between the **vlan**, **fastethernet**, **gigabitethernet**, **tengigabitethernet**, or **macro** keyword and the dash. For example, the command **interface range fastethernet 5/1 - 5** specifies a valid range; the command **interface range fastethernet 5/1-5** does not contain a valid range command.

**Note** The **interface range** command works only with VLAN interfaces that have been configured with the **interface vlan** command (the **show running-configuration** command displays the configured VLAN interfaces). VLAN interfaces that are not displayed by the **show running-configuration** command cannot be used with the **interface range** command.

This example shows how to reenable all Fast Ethernet interfaces 5/1 to 5/5:

```
Switch(config)# interface range fastethernet 5/1 - 5
Switch(config-if-range)# no shutdown
Switch(config-if-range)#
*Oct  6 08:24:35: %LINK-3-UPDOWN: Interface FastEthernet5/1, changed state to up
*Oct  6 08:24:35: %LINK-3-UPDOWN: Interface FastEthernet5/2, changed state to up
*Oct  6 08:24:35: %LINK-3-UPDOWN: Interface FastEthernet5/3, changed state to up
*Oct  6 08:24:35: %LINK-3-UPDOWN: Interface FastEthernet5/4, changed state to up
*Oct  6 08:24:35: %LINK-3-UPDOWN: Interface FastEthernet5/5, changed state to up
*Oct  6 08:24:36: %LINEPROTO-5-UPDOWN: Line protocol on Interface FastEthernet5/
5, changed state to up
*Oct  6 08:24:36: %LINEPROTO-5-UPDOWN: Line protocol on Interface FastEthernet5/
3, changed state to up
*Oct  6 08:24:36: %LINEPROTO-5-UPDOWN: Line protocol on Interface FastEthernet5/
4, changed state to up
Switch(config-if)#
```

This example shows how to use a comma to add different interface type strings to the range to re-enable all Fast Ethernet interfaces ranging from 5/1 to 5/5 and both Gigabit Ethernet interfaces 1/1 and 1/2:

```
Switch(config-if)# interface range fastethernet 5/1 - 5, gigabitethernet 1/1 - 2
Switch(config-if)# no shutdown
Switch(config-if)#
*Oct  6 08:29:28: %LINK-3-UPDOWN: Interface FastEthernet5/1, changed state to up
*Oct  6 08:29:28: %LINK-3-UPDOWN: Interface FastEthernet5/2, changed state to up
*Oct  6 08:29:28: %LINK-3-UPDOWN: Interface FastEthernet5/3, changed state to up
*Oct  6 08:29:28: %LINK-3-UPDOWN: Interface FastEthernet5/4, changed state to up
*Oct  6 08:29:28: %LINK-3-UPDOWN: Interface FastEthernet5/5, changed state to up
*Oct  6 08:29:28: %LINK-3-UPDOWN: Interface GigabitEthernet1/1, changed state to
 up
*Oct  6 08:29:28: %LINK-3-UPDOWN: Interface GigabitEthernet1/2, changed state to
 up
*Oct  6 08:29:29: %LINEPROTO-5-UPDOWN: Line protocol on Interface FastEthernet5/
5, changed state to up
*Oct  6 08:29:29: %LINEPROTO-5-UPDOWN: Line protocol on Interface FastEthernet5/
3, changed state to up
*Oct  6 08:29:29: %LINEPROTO-5-UPDOWN: Line protocol on Interface FastEthernet5/
4, changed state to up
Switch(config-if)#
```

**Note**   If you enter multiple configuration commands while you are in interface-range configuration mode, each command is run as it is entered (they are not batched together and run after you exit interface-range configuration mode). If you exit interface-range configuration mode while the commands are being run, some commands might not be run on all interfaces in the range. Wait until the command prompt is displayed before exiting interface-range configuration mode.

# Using the Ethernet Management Port

This section has this information:
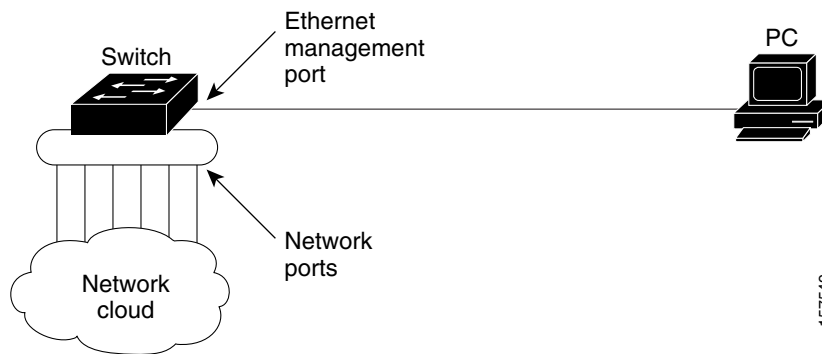
# Understanding the Ethernet Management Port

The Ethernet management port, also referred to as the *Fa1 or fastethernet1 port*, is a Layer 3 host port to which you can connect a PC. You can use the Ethernet management port instead of the switch console port for network management. When managing a switch, connect the PC to the Ethernet management port on a Catalyst 4500 series switch. (Figure 6-1).

**Note** When connecting a PC to the Ethernet management port, you must assign an IP address.

*Figure 6-1*    *Connecting a Switch to a PC*



By default, the Ethernet management port is enabled. The switch cannot route packets from the Ethernet management port to a network port, and from the network port to the Ethernet port. To obtain these, the Fa1 interface is automatically placed in a separate routing domain (or VRF domain), called *mgmtVrf*. (You observe the *ip Vrf forwarding mgmtVrf* line in the running configuration when you boot up.) For details, read the

Even though the Ethernet management port does not support routing, you might need to enable routing protocols on the port. As illustrated in Figure 6-2, you must enable routing protocols on the Ethernet management port when the PC is multiple hops away from the switch and the packets must pass through multiple Layer 3 devices to reach the PC.

*Figure 6-2*    *Network Example with Routing Protocols Enabled*
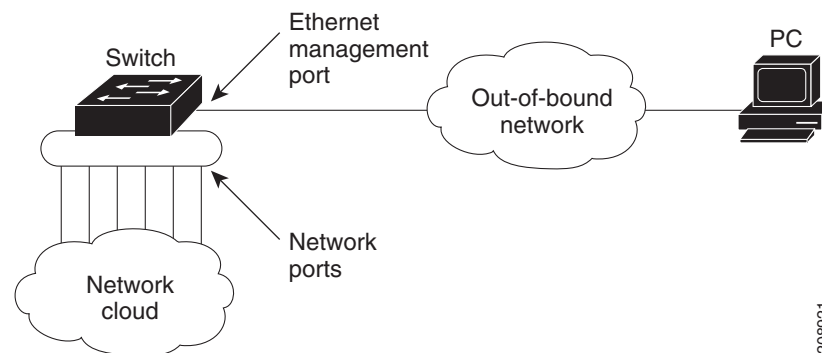


The specific implementation of Ethernet management port depends on the redundancy model you are applying.

For details on configuring SSO and ISSU, refer to Chapter 8, "Configuring Supervisor Engine Redundancy Using RPR and SSO" and Chapter 5, "Configuring the Cisco IOS In Service Software Upgrade Process"

## Fa1 Interface and mgmtVrf

⚠

**Caution**    The Ethernet management port is intended for out-of-band access only.  Like the console port, the Ethernet management port has direct access to critical resources on the switch.  Connecting this port to an in-band network might cause performance degradation and vulnerability to a denial of service attack.

All features that use fa1 now need to be VRF-aware.

✎

**Note**    You cannot configure any other interface in the same routing domain and you cannot configure a different routing domain for the Fa1 interface.

On bootup the fa1 port assumes the following default configuration:

```
ip unicast-routing

ip vrf mgmtVrf
!
interface FastEthernet1
 ip vrf forwarding mgmtVrf
speed auto
duplex auto

Switch# show ip vrf
  Name                             Default RD          Interfaces
  mgmtVrf                                                Fa1
```

Because the management port is placed in mgmtVrf, you should be aware of the VRF aware commands required for the following tasks:

✎

**Note**    Command usage specific to the mgmtVrf are mentioned below. The additional configuration needed to make the feature work needs to be configured.

### Ping

If you want to ping an IP address that is reachable through an fa1 port, enter the following command:

```
Switch# ping vrf  mgmtVrf ip address
```

For example,

```
Switch# ping vrf mgmtVrf 20.20.20.1
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 20.20.20.1, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/2/4 ms
```

### TraceRoute

```
Switch# traceroute vrf mgmtVrf ip address

   Eg: Switch# traceroute vrf mgmtVrf 20.20.20.1
   Type escape sequence to abort.
   Tracing the route to 20.20.20.1
    1 20.20.20.1 0 msec 0 msec *
```

### Telnet

If you want to Telnet to a remote switch through the Fa1 port, enter the following command:

```
Switch# telnet <word> /vrf mgmtVrf
word  IP address or hostname of a remote system
```

<u>An example</u>

```
Switch# telnet 20.20.20.1 /vrf mgmtVrf
Trying 20.20.20.1 ... Open
User Access Verification
Password:
switch> en
Password:
switch#
```

### TFTP

If you want to use Fa1 port for TFTP operation, configure the Fa1 port as the source interface for TFTP as follows:

```
Switch(config# ip tftp source-interface fastEthernet1
```

### FTP

If you want to use an Fa1 port for an FTP operation, configure the Fa1 port as the source interface for FTP as follows:

```
Switch(config)# ip ftp source-interface fastEthernet1
```

### SSH

If you want initiate SSH from your switch through the Fa1 port, enter the following command:

```
Switch# ssh -l <login name> -vrf mgmtVrf <ip address>
```

For example,

```
Switch# ssh -l xyz -vrf mgmtVrf 20.20.20.1
```

## SSO Model

On a redundant chassis, management port behavior differs from that of a standard Ethernet port in that each supervisor engine possesses a management port, and only the port on the active supervisor engine is enabled. The management port on the standby supervisor engine is always disabled; it cannot switch any kind of traffic.

When a switchover occurs, the management port of the standby supervisor engine (now, active) is enabled and can be used to switch traffic, while the management port on the "old" active supervisor engine i disabled.

> **Note**    The IOS configuration for the management port is synchronized between the two supervisor engines. So, under IOS, they will possess the same IP address. To avoid address overlapping during a switchover on a redundant chassis, you should assign a different IP address on the management port from the one you assigned to the same port in the ROMMON configuration.

## ISSU Model

In SSO mode, the running configurations on the active and standby supervisor engines must match. So, you cannot enable the management port on a redundant chassis if one of the two supervisor engines is running an IOS image older than 12.2(50)SG (where the Management port is not supported).

When you perform an ISSU upgrade or downgrade between an image prior to Cisco IOS Release 12.2(50)SF and Cisco IOS Release 12.2(50)SG, IOS automatically disables the management port. The port configuration is restored when both images running on the supervisor engines are at least 12.2(50)SG. A warning message is also displayed to flag the event.

# Supported Features on the Ethernet Management Port

The Ethernet management port supports these features:

- Express Setup
- Telnet with passwords
- TFTP
- Secure Shell (SSH)
- DHCP-based autoconfiguration
- SNMP (only the ENTITY-MIB and the IF-MIB)
- IP ping
- Interface features
  - Speed—10 Mb/s, 100 Mb/s, 1000Mb/s, and autonegotiation
  - Duplex mode—Full, half, and autonegotiation
  - Loopback detection
- Cisco Discovery Protocol (CDP) (only on WS-C4900M and WS-C4948)
- IPv4 access control lists (ACLs)
- Routing protocols (only on WS-C4900M and WS-C4948)
- AAA

⚠

**Caution**   Before enabling a feature on the Ethernet management port, ensure that the feature is supported. If you try to configure an unsupported feature on an Ethernet management port, the feature might not work properly, and the switch might fail.

# Configuring the Ethernet Management Port

To specify the Ethernet management port, enter **fastethernet1**.

To disable the port, use the **shutdown** interface configuration command. To enable the port, use the **no shutdown** interface configuration command.

To determine the link status to the PC, you can monitor the LED for the Ethernet management port:

> The LED is green (on) when the link is active.

> The LED is off when the link is down.

> The LED is amber when there is a POST failure.

To display the link status, use the **show interfaces fastethernet 1** privileged EXEC command.

# Defining and Using Interface-Range Macros

You can define 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 command string, you must define the macro.

To define an interface-range macro, perform this task:

**Table 6-1**

| Command | Purpose |
|---|---|
| Switch(config)# **define interface-range** *macro_name* {**vlan** *vlan_ID – vlan_ID*} \| {{**fastethernet** \| **gigabitethernet**} *slot/interface* **-** *interface*} [**,** {**vlan** *vlan_ID – vlan_ID*} {{**fastethernet** \| **gigabitethernet**} *slot/interface* **-** *interface*}] | Defines the interface-range macro and saves it in the running configuration file. |

This example shows how to define an interface-range macro named **enet_list** to select Fast Ethernet interfaces 5/1 through 5/4:

```
Switch(config)# define interface-range enet_list fastethernet 5/1 - 4
```

To show the defined interface-range macro configuration, perform this task:

**Table 6-2**

| Command | Purpose |
|---|---|
| Switch# **show running-config** | Shows the defined interface-range macro configuration. |

This example shows how to display the defined interface-range macro named **enet_list**:

```
Switch# show running-config | include define
define interface-range enet_list FastEthernet5/1 - 4
Switch#
```

To use an interface-range macro in the **interface range** command, perform this task:

**Table 6-3**

| Command | Purpose |
|---------|---------|
| Switch(config)# **interface range macro** *name* | Selects the interface range to be configured using the values saved in a named interface-range macro. |

This example shows how to change to the interface-range configuration mode using the interface-range macro **enet_list**:

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

# Deploying SFP+ in X2 Ports

> **Note**    This feature is supported on Supervisor Engine 6-E X2 ports as well as the WS-X4606-10GE (or WS-X4606-X2-E), WS-X4908-10GE, WS-X4904-10GE, and WS-C4900M chassis.

To use an SFP+ in an X2 port to obtain 10-Gigabit Ethernet bandwidth, the Catalyst 4500 series switch supports OneX Convertor modules. When you plug a OneX Converter module into an X2 port, it converts the X2 port into an SFP+ port into which you can plug in an SFP+. An SFP+ in a OneX Convertor module provides the same functionality as an X2 and maintains the same port numbering.

The output for the **show idprom tengigabitethernet slot/interface** command displays the contents of both the SFP+ and the OneX Convertor module seeproms when an SFP+ in a OneX Convertor module is plugged into an X2 port.

# Deploying 10-Gigabit Ethernet and Gigabit Ethernet SFP Ports on Supervisor Engine V-10GE

> **Note**    LAN Base image does not support 10-Gigabit Ethernet uplinks.

> **Note**    On a Catalyst 4510R series switch, if you enable both the 10-Gigabit Ethernet and Gigabit Ethernet SFP uplink ports, you must re-boot the switch. On the Catalyst 4503, 4506, and 4507R series switches, this capability is automatically enabled.

Prior to Cisco IOS Release 12.2(25)SG, the Cisco Catalyst 4500 Supervisor Engine V-10GE allowed you to enable either the dual wire-speed 10-Gigabit Ethernet ports, or four alternatively wired Gigabit Ethernet SFP uplink ports. With Cisco IOS Release 12.2(25)SG, you can simultaneously deploy the dual 10 Gigabit Ethernet ports and the four Gigabit Ethernet SFP ports on the Catalyst 4503, Catalyst 4506, and Catalyst 4507R chassis.

When you deploy a Catalyst 4510R chassis, one of the following configurations is supported:

- Dual 10-Gigabit Ethernet ports (X2 optics) only.

- Four Gigabit Ethernet ports (SFP optics) only.

- Both the dual 10-Gigabit Ethernet and the four Gigabit Ethernet ports, with the understanding that the tenth slot (Flex-Slot) will only support a 2-port gigabit interface converter (GBIC) line card (WS-X4302-GB) when in this mode.

- You cannot place a linecard with a backplane traffic capacity exceeding 6Gbps in slots 8, 9 and 10 of a Catalyst 4510R-E chassis when used with a Supervisor Engine 6-E.

To select the 10-Gigabit Ethernet or the Gigabit Ethernet SFP uplink port, perform this task:

| | Command | Purpose |
|---|---|---|
| Step 1 | Switch# **configure terminal** | Establishes global configuration mode. |
| Step 2 | Switch(config)# **hw-module uplink select** [**all** \| **gigabitethernet** \| **tengigabitethernet**] | Selects the port type to enable. |

**Note** On a Supervisor Engine V-10GE (WS-X4516-10GE) in a 10 slot chassis (Catalyst 4510R and 4510RE), if a startup configuration with a new uplink mode is copied into flash memory and the system is power cycled, the system will not come up with the new uplink mode. After copying the startup configuration with the new uplink mode into flash memory, the uplink mode must be changed to the new uplink mode through the command interface before the system is power cycled. This ensures that the system comes up in the new uplink mode.

The following example shows how to enable both 10-Gigabit Ethernet and Gigabit Ethernet SFP uplink ports on a Catalyst 4510R series switch:

```
Switch# configure terminal
Switch(config)# hw-module uplink select all
Warning: This configuration mode will place slot 10 in flex slot mode
```

**Note** When you modify the uplink mode, you must re-boot the switch.

# Deploying 10-Gigabit Ethernet or Gigabit Ethernet Ports on WS-X4606-X2-E and Supervisor Engine 6-E

To increase the flexibility of X2 ports on both Supervisor Engine 6-E and WS-X4606-X2-E, the Catalyst 4500 switch as well as the 4900M chassis supports TwinGig Convertor modules. When you plug a TwinGig Convertor module into an X2 hole, it converts a single X2 hole (capable of holding one pluggable X2 optic) into two SFP holes (capable of holding two pluggable SFP optics). This enables you to have 10 Gigabit ports and 1-Gigabit ports on the same linecard. It also allows you to use Gigabit ports, and then switch to a 10-Gigabit port, when needed.

Topics include:

- Port Numbering TwinGig Convertors, page 6-14
- Limitations on Using a TwinGig Convertor, page 6-14
- Selecting X2/TwinGig Convertor Mode, page 6-15

# Port Numbering TwinGig Convertors

When a TwinGig Convertor is enabled or disabled, the number and type of ports on the linecard change dynamically. The terminology must reflect this behavior. In Cisco IOS, 10-Gigabit ports are named *TenGigabit* and 1-Gigabit ports are named *Gigabit*. Starting with Cisco IOS Release 12.2(40)SG, to avoid having two ports named TenGigabit1/1 and Gigabit1/1, the 10-Gigabit and 1-Gigabit port numbers are independent. For example, for a WS-X4606-X2-E module with six X2 holes, the X2 ports are named *TenGigabit slot-num/<1-6>*, and the SFP ports are named *Gigabit slot-num/<7-18>*.

*Figure 6-3        Faceplate for WS-X4606-10GE*
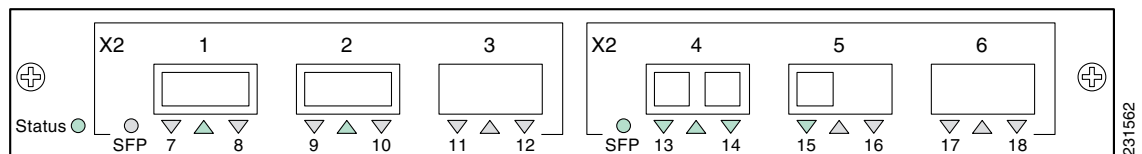


In Cisco IOS, ports 1 through 18 always exist. This means that you can apply configurations on them and they display in the CLI output. However, only the X2 or the SFP ports can be active at any particular time. For example, if an X2 is plugged into the second hole, the X2 port 2 is active and SFP ports 9 and 10 are inactive. If a TwinGig Convertor is plugged into the second hole, the X2 port 2 is inactive, and the SFP ports 9 and 10 are active. The inactive ports are treated analogously to the inactive ports on Supervisor Engines IV and V-10GE, where at no time are all of the uplinks connected to the switching ASICs.

**Note**    When using both TwinGig and X2 transceivers on the WS-X4606-X2-E module, place ports 1-3 in one group and ports 4-6 in another. (The mode selected with the **show hw-module module port-group** command determines the behaviour. See "Selecting X2/TwinGig Convertor Mode".) Mixing within a port group will not work. For example, you cannot have an X2 in port 1 and a TwinGig in port 2 and expect both of them to function.

# Limitations on Using a TwinGig Convertor

In a Supervisor Engine 6-E system, the ports are connected to the switching engine through a stub ASIC. This stub ASIC imposes some limitations on the ports: Gig and 10 Gig ports cannot be mixed on a single stub ASIC; they must either be all 10 Gig (X2), or all Gig (TwinGig Converter and SFP). The faceplates of X2 modules show this stub port grouping, either with actual physical grouping with a box drawn around a grouping.

# Selecting X2/TwinGig Convertor Mode

The default configuration mode is X2. So, if you plan to deploy 10-Gigabit interfaces, you don't need to configure anything. However, if you want to deploy Gigabit interfaces (that is, use TwinGig Convertors), you must configure the associated port-group:

- To determine how the X2 holes on a module are grouped, enter the **show hw-module module** *m* **port-group** *p* command.

> **Note**    For a 10-Gigabit port that accepts a HAMM, you must place it into 1-Gigabit mode instead of 10-Gigabit mode.

If you configure a 10-Gigabit port as 1-Gigabit port, an output similar to the following displays:

```
Switch# show hw-module module 5 port-group
Module Port-group Active                          Inactive
-------------------------------------------------------------
   5        1    Gi5/3-6                          Te5/1-2
```

Instead, if the port is set to the default, 10-Gigabit mode, an output similar to the following displays:

```
Switch# show hw-module module 6 port-group
Module Port-group Active                          Inactive
-------------------------------------------------------------
   6        1    Te6/1-2                          Gi6/3-6
```

```
Switch# show int status mod 1

Port      Name            Status       Vlan    Duplex  Speed Type
Te1/1                     notconnect   1         full    10G 10GBase-LR
Te1/2                     connected    1         full    10G 10GBase-LR
Te1/3                     notconnect   1         full    10G No X2
Te1/4                     notconnect   1         full    10G No X2
Te1/5                     notconnect   1         full    10G No X2
Te1/6                     notconnect   1         full    10G No X2
Gi1/7                     inactive     1         full   1000 No Gbic
Gi1/8                     inactive     1         full   1000 No Gbic
Gi1/9                     inactive     1         full   1000 No Gbic
Gi1/10                    inactive     1         full   1000 No Gbic
Gi1/11                    inactive     1         full   1000 No Gbic
Gi1/12                    inactive     1         full   1000 No Gbic
Gi1/13                    inactive     1         full   1000 No Gbic
Gi1/14                    inactive     1         full   1000 No Gbic
Gi1/15                    inactive     1         full   1000 No Gbic
Gi1/16                    inactive     1         full   1000 No Gbic
Gi1/17                    inactive     1         full   1000 No Gbic
Gi1/18                    inactive     1         full   1000 No Gbic
Switch#
```

- To configure the modes of operation for each X2 port group in which you want to deploy Gigabit, enter the **hw-module module** *m* **port-group** *p* **select gigabitethernet** command. This configuration is preserved across power cycles and reloads.

To deploy Gigabit Ethernet interfaces using the TwinGig Convertor, perform this task:

| | Command | Purpose |
|---|---|---|
| Step 1 | Switch# **configure terminal** | Establishes global configuration mode. |
| Step 2 | Switch(config)# **hw-module module *m* port-group *p* select [gigabitethernet | tengigabitethernet]** | Selects the mode of operation for each X2 port-group. Default is TenGigabit Ethernet (x2). |
| Step 3 | Switch(config)# **exit** | Exits configuration mode. |
| Step 4 | Switch# **show int status mod *n*** | Verifies the setting. |

This example shows how to select Gigabit Ethernet interfaces on a WS-X4606-X2-E using the TwinGig Convertor:

```
Switch# config terminal
Enter configuration commands, one per line.  End with CNTL/Z.
Switch(config)# hw-module module 1 port-group 1 select gigabitethernet
Switch(config)# exit
Switch# show int status mod 1
Port        Name                Status     Vlan       Duplex  Speed Type
Te1/1                           inactive   1            full   10G No X2
Te1/2                           inactive   1            full   10G No X2
Te1/3                           inactive   1            full   10G No X2
Te1/4                           notconnect 1            full   10G No X2
Te1/5                           notconnect 1            full   10G No X2
Te1/6                           notconnect 1            full   10G No X2
Gi1/7                           notconnect 1            full   1000 No Gbic
Gi1/8                           notconnect 1            full   1000 No Gbic
Gi1/9                           notconnect 1            full   1000 No Gbic
Gi1/10                          notconnect 1            full   1000 No Gbic
Gi1/11                          notconnect 1            full   1000 No Gbic
Gi1/12                          notconnect 1            full   1000 No Gbic
Gi1/13                          inactive   1            full   1000 No Gbic
Gi1/14                          inactive   1            full   1000 No Gbic
Gi1/15                          inactive   1            full   1000 No Gbic
Gi1/16                          inactive   1            full   1000 No Gbic
Gi1/17                          inactive   1            full   1000 No Gbic
Gi1/18                          inactive   1            full   1000 No GbicI
```

# Invoking Shared-backplane Uplink Mode on Supervisor Engine 6-E

This feature enables you to use all 4 10-Gigabit Ethernet ports on the supervisor engines as blocking ports when in redundant mode.

Prior to Cisco IOS Release 12.2(40)SG, the Catalyst 4500 Supervisor Engine V-10GE allowed you to enable either the dual wire-speed 10-Gigabit Ethernet ports, or four Twin-Gigabit converter based Gigabit Ethernet SFP uplink ports when operating in redundant mode. With Cisco IOS Release 12.2(40)SG, you can deploy (1) all 4 10-Gigabit Ethernet ports, (2) 2 blocking ports on an active supervisor engine and 2 blocking ports on the standby supervisor engine, or (3) all 8 Gigabit Ethernet SFP ports, 4 on the active supervisor and 4 on the standby supervisor engine. This capability is supported on all Catalyst 4500 and 4500E series chassis.

To enable shared-backplane mode, enter this command:

| Command | Purpose |
|---|---|
| Switch(config)# **hw-mod uplink mode shared-backplane** | A reload of the active supervisor is required to apply the new configuration. |

To disable shared-backplane mode, enter this command:

| Command | Purpose |
|---|---|
| Switch(config)# **no hw-mod uplink mode shared-backplane** | A reload of the active supervisor is required to apply the new configuration. |

# Digital Optical Monitoring Transceiver Support

Command line interface (CLI) commands (show inventory, show idprom interface) are used on transceivers to obtain serial number, model name, inventory information.

The following commands are specific to the transceivers that support the DOM capability:

- Displays current values and thresholds for all sensor on a particular interface transceiver:

  **show interfaces <int-name> transceiver** [**detail**] [**threshold**]

- Enables or disables the *entSensorThresholdNotification* for all sensors in all the transceivers:

  **snmp-server enable trap transceiver**

- Enables or disables transceiver monitoring:

  **transceiver type all**
  **monitoring**

> **Note**   This feature is only available when a DOM capable transceiver is present and configured for monitoring. The frequency at which the sensor information is refreshed depends on default values configured in the transceiver SEEPROM (Serial Electrically Erasable Programmable Read Only Memory).

> **Note**   For details on transceiver module compatibility, refer to this URL:
>
> http://www.cisco.com/en/US/products/hw/modules/ps5455/products_device_support_tables_list.html

# Configuring Optional Interface Features

The following subsections describe optional procedures:

- Configuring Ethernet Interface Speed and Duplex Mode, page 6-18
- Configuring Flow Control, page 6-21
- Configuring Jumbo Frame Support, page 6-23

# Configuring Ethernet Interface Speed and Duplex Mode

## Speed and Duplex Mode Configuration Guidelines

**Note** You do not configure the client device for autonegotiation. Rather, you configure the switch with the speed, or range of speeds, that you want to autonegotiate.

You can configure the interface speed and duplex mode parameters to **auto** and allow the Catalyst 4500 series switch to negotiate the interface speed and duplex mode between interfaces. If you decide to configure the interface **speed** and **duplex** commands manually, consider the following:

- If you enter the **no speed** command, the switch automatically configures both interface **speed** and **duplex** to **auto**.
- When you set the interface speed to **1000** (Mbps) or **auto 1000**, the duplex mode is full duplex. You cannot change the duplex mode.
- If the interface speed is set to **10** or **100**, the duplex mode is set to half duplex by default unless you explicitly configure it.

**Caution** Changing the interface speed and duplex mode configuration might shut down and restart the interface during the reconfiguration.

## Setting the Interface Speed

If you set the interface speed to **auto** on a 10/100-Mbps Ethernet interface, speed and duplex are autonegotiated. The forced 10/100 autonegotiation feature allows you to limit interface speed auto negotiation up to 100 Mbps on a 10/100/1000BASE-T port.

To set the port speed for a 10/100-Mbps Ethernet interface, perform this task:

| | Command | Purpose |
|---|---|---|
| **Step 1** | `Switch(config)# ` **`interface fastethernet`** `slot/interface` | Specifies the interface to be configured. |
| **Step 2** | `Switch(config-if)# ` **`speed`** [**`10`** \| **`100`** \| **`auto`** [**`10`** \| **`100`**]] | Sets the interface speed of the interface. |

This example shows how to set the interface speed to 100 Mbps on the Fast Ethernet interface 5/4:

```
Switch(config)# interface fastethernet 5/4
Switch(config-if)# speed 100
```

This example shows how to allow Fast Ethernet interface 5/4 to autonegotiate the speed and duplex mode:

```
Switch(config)# interface fastethernet 5/4
Switch(config-if)# speed auto
```

**Note**    This is analogous to specifying **speed auto 10 100**.

This example shows how to limit the interface speed to 10 and 100 Mbps on the Gigabit Ethernet interface 1/1 in auto-negotiation mode:

```
Switch(config)# interface gigabitethernet 1/1
Switch(config-if)# speed auto 10 100
```

This example shows how to limit speed negotiation to 100 Mbps on the Gigabit Ethernet interface 1/1:

```
Switch(config)# interface gigabitethernet 1/1
Switch(config-if)# speed auto 100
```

**Note**    Turning off autonegotiation on a Gigabit Ethernet interface will result in the port being forced into 1000 Mbps and full-duplex mode.

To turn off the port speed autonegotiation for Gigabit Ethernet interface 1/1, perform this task:

| | Command | Purpose |
|---|---|---|
| Step 1 | Switch(config)# **interface gigabitethernet1/1** | Specifies the interface to be configured. |
| Step 2 | Switch(config-if)# **speed nonegotiate** | Disables autonegotiation on the interface. |

To restore autonegotiation, enter the **no speed nonegotiate** command in the interface configuration mode.

**Note**    For the blocking ports on the WS-X4416 module, do not set the speed to autonegotiate.

## Setting the Interface Duplex Mode

**Note**    When the interface is set to 1000 Mbps, you cannot change the duplex mode from full duplex to half duplex.

To set the duplex mode of a Fast Ethernet interface, perform this task:

| | Command | Purpose |
|---|---|---|
| Step 1 | Switch(config)# **interface fastethernet** *slot/interface* | Specifies the interface to be configured. |
| Step 2 | Switch(config-if)# **duplex** [**auto** | **full** | **half**] | Sets the duplex mode of the interface. |

This example shows how to set the interface duplex mode to full on Fast Ethernet interface 5/4:

```
Switch(config)# interface fastethernet 5/4
Switch(config-if)# duplex full
```

## Displaying the Interface Speed and Duplex Mode Configuration

To display the interface speed and duplex mode configuration for an interface, perform this task:

| Command | Purpose |
|---------|---------|
| Switch# **show interfaces** [**fastethernet** \| **gigabitethernet** \| **tengigabitethernet**] *slot/interface* | Displays the interface speed and duplex mode configuration. |

This example shows how to display the interface speed and duplex mode of Fast Ethernet interface 6/1:

```
Switch# show interface fastethernet 6/1
FastEthernet6/1 is up, line protocol is up
  Hardware is Fast Ethernet Port, address is 0050.547a.dee0 (bia 0050.547a.dee0)
  MTU 1500 bytes, BW 100000 Kbit, DLY 100 usec,
     reliability 255/255, txload 1/255, rxload 1/255
  Encapsulation ARPA, loopback not set
  Keepalive set (10 sec)
  Full-duplex, 100Mb/s
  ARP type: ARPA, ARP Timeout 04:00:00
  Last input 00:00:54, output never, output hang never
  Last clearing of "show interface" counters never
  Input queue: 50/2000/0/0 (size/max/drops/flushes); Total output drops: 0
  Queueing strategy: fifo
  Output queue: 0/40 (size/max)
  5 minute input rate 0 bits/sec, 0 packets/sec
  5 minute output rate 0 bits/sec, 0 packets/sec
     50 packets input, 11300 bytes, 0 no buffer
     Received 50 broadcasts, 0 runts, 0 giants, 0 throttles
     0 input errors, 0 CRC, 0 frame, 0 overrun, 0 ignored
     0 input packets with dribble condition detected
     1456 packets output, 111609 bytes, 0 underruns
     0 output errors, 0 collisions, 0 interface resets
     0 babbles, 0 late collision, 0 deferred
     1 lost carrier, 0 no carrier
     0 output buffer failures, 0 output buffers swapped out
Switch#
```

## Adding a Description for an Interface

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

To add a description for an interface, enter the following command:

| Command | Purpose |
|---------|---------|
| Switch(config-if)# **description** *string* | Adds a description for an interface. |

This example shows how to add a description on Fast Ethernet interface 5/5:

```
Switch(config)# interface fastethernet 5/5
```

```
Switch(config-if)# description Channel-group to "Marketing"
```

# Configuring Flow Control

Gigabit Ethernet ports use Flow Control to slow-down the transmission of incoming packets. If a buffer on a Gigabit Ethernet port runs out of space, the port transmits a special packet that requests remote ports to delay sending packets for a period of time. The port can also receive this special packet from its link-partner for the same purpose. This special packet is called a *pause frame*.

The default settings for Gigabit Ethernet interfaces are as follows:

- Sending pause frames is off—non-oversubscribed Gigabit Ethernet interfaces.
- Receiving pause frames is desired—non-oversubscribed Gigabit Ethernet interfaces.
- Sending pause frames is on—Oversubscribed Gigabit Ethernet interfaces.
- Receiving pause frames is desired—Oversubscribed Gigabit Ethernet interfaces

The default settings for Tengigabit Ethernet interfaces are as follows:

- Sending pause frames is off.
- Receiving pause frames is on.

> **Note**    **desired** is not a flow control option on the Tengigabit Ethernet interfaces.

To configure flow control, perform this task:

|        | Command | Purpose |
|--------|---------|---------|
| Step 1 | Switch# **configure terminal** | Enters global configuration mode. |
| Step 2 | Switch(config)# **interface** *interface-id* | Enters interface configuration mode and specifies the interface to be enabled for flowcontrol. |
| Step 3 | Switch(config-if)# **flowcontrol** {**receive** \| **send**} {**off** \| **on** \| **desired**} | Configures a Gigabit Ethernet port to send or receive pause frames. |
| Step 4 | Switch(config-if)# **end** | Returns to configuration mode. |
| Step 5 | Switch(config)# **end** | Returns to privileged EXEC mode. |

This example shows how to configure flow control on an oversubscribed Gigabit Ethernet port 7/5:

```
Switch# configure terminal
Switch(config)# interface g7/5
Switch(config-if)# flowcontrol send on
Switch(config-if)# end
Switch)# show interfaces gigabitEthernet 7/5 capabilities
GigabitEthernet7/5
  Model:                WS-X4548-GB-RJ45-RJ-45
  Type:                 10/100/1000-TX
  Speed:                10,100,1000,auto
  Duplex:               half,full,auto
  Trunk encap. type:    802.1Q,ISL
  Trunk mode:           on,off,desirable,nonegotiate
  Channel:              yes
  Broadcast suppression: percentage(0-100), hw
  Flowcontrol:          rx-(off,on,desired),tx-(off,on,desired)
```

```
VLAN Membership:       static, dynamic
Fast Start:            yes
Queuing:               rx-(N/A), tx-(1p3q1t, Sharing/Shaping)
CoS rewrite:           yes
ToS rewrite:           yes
Inline power:          no
SPAN:                  source/destination
UDLD:                  yes
Link Debounce:         no
Link Debounce Time:    no
Port Security:         yes
Dot1x:                 yes
Maximum MTU:           1552 bytes (Baby Giants)
Multiple Media Types:  no
Diagnostic Monitoring: N/A

Switch)# show flowcontrol interface GigabitEthernet 7/5
Port      Send FlowControl  Receive FlowControl  RxPause TxPause
          admin    oper     admin    oper
--------- -------- -------- -------- --------     ------- -------
Gi7/5     on       off      desired  off          0       0
```

This example shows the output of the **show interfaces** and **show flowcontrol** commands on an non-oversubscribed Gigabit Ethernet port 5/5:

```
Switch# show interfaces gigabitEthernet 5/5 capabilities
GigabitEthernet5/5
  Model:                 WS-X4306-GB-Gbic
  Type:                  No Gbic
  Speed:                 1000
  Duplex:                full
  Trunk encap. type:     802.1Q,ISL
  Trunk mode:            on,off,desirable,nonegotiate
  Channel:               yes
  Broadcast suppression: percentage(0-100), hw
  Flowcontrol:           rx-(off,on,desired),tx-(off,on,desired)
  VLAN Membership:       static, dynamic
  Fast Start:            yes
  Queuing:               rx-(N/A), tx-(1p3q1t, Sharing/Shaping)
  CoS rewrite:           yes
  ToS rewrite:           yes
  Inline power:          no
  SPAN:                  source/destination
  UDLD:                  yes
  Link Debounce:         no
  Link Debounce Time:    no
  Port Security:         yes
  Dot1x:                 yes
  Maximum MTU:           9198 bytes (Jumbo Frames)
  Multiple Media Types:  no
  Diagnostic Monitoring: N/A

Switch# show flowcontrol interface gigabitEthernet 5/5
Port      Send FlowControl  Receive FlowControl  RxPause TxPause
          admin    oper     admin    oper
--------- -------- -------- -------- --------     ------- -------
Gi5/5     off      off      desired  off          0       0
```

This example shows the output of the **show interfaces** and **show flowcontrol** commands on an unsupported Fast Ethernet port 3/5:

```
Switch# show interfaces fa3/5 capabilities
FastEthernet3/5
  Model:                 WS-X4148-RJ-45
```

```
Type:                   10/100BaseTX
Speed:                  10,100,auto
Duplex:                 half,full,auto
Trunk encap. type:      802.1Q,ISL
Trunk mode:             on,off,desirable,nonegotiate
Channel:                yes
Broadcast suppression:  percentage(0-100), sw
Flowcontrol:            rx-(none),tx-(none)
VLAN Membership:        static, dynamic
Fast Start:             yes
Queuing:                rx-(N/A), tx-(1p3q1t, Shaping)
CoS rewrite:            yes
ToS rewrite:            yes
Inline power:           no
SPAN:                   source/destination
UDLD:                   yes
Link Debounce:          no
Link Debounce Time:     no
Port Security:          yes
Dot1x:                  yes
Maximum MTU:            1552 bytes (Baby Giants)
Multiple Media Types:   no
Diagnostic Monitoring:  N/A

Switch# show flowcontrol interface fa3/5
Port      Send FlowControl  Receive FlowControl  RxPause TxPause
          admin    oper     admin    oper

--------- -------- -------- -------- --------   ------- -------
Fa3/5     Unsupp.  Unsupp.  Unsupp.  Unsupp.       0       0
```

# Configuring Jumbo Frame Support

These subsections describe jumbo frame support:

## Ports and Modules that Support Jumbo Frames

The following ports and modules support jumbo frames:

- Supervisor uplink ports

- WS-X4306-GB: all ports

- WS-X4232-GB-RJ: ports 1-2

- WS-X4418-GB: ports 1-2

- WS-X4412-2GB-TX: ports 13-14

- WS-X4506-GB-T

- the 4648-GB-RJ45V

- WS-X4648-GB+RJ45V

- WS-X4648-RJ45V-E

- WS-X4648-RJ45V+E

- WS-X4706-10GE

# Jumbo Frame Support

These sections describe jumbo frame support:

## Maximum Transmission Units

The Catalyst 4500 series switch allows you to configure a maximum of 32 different maximum transmission unit (MTU) sizes systemwide. This means that the maximum number of different MTU sizes that you can configure with the **system mtu**, **mtu**, **ip mtu**, and **ipv6 mtu** command on all Layer 2 and Layer 3 interfaces combined is 32.

Also, the system stores the ipv4 and ipv6 MTU sizes configured on an interface separately. So, for every **system mtu** command or per interface **mtu** command, two separate MTU values are stored, one for ipv4 and one for ipv6. This further reduces the number of slots available (out of 32). However, only a single MTU value is stored for each **ip mtu** and **ipv6 mtu** commands.

If the new MTU value you are configuring is already present in the system (that is, configured on some other interface), then no new slot(s) will be allocated to store it again.

If the maximum limit of 32 is reached and an attempt is made to configure a new MTU size on a new interface, the system will only allow configuration to proceed if the new MTU size has previously been configured on some interface. Otherwise, an error message will be displayed and the default MTU size will be assigned to the interface being configured.

## Jumbo Frame Support Overview

A jumbo frame is a frame larger than the default Ethernet size. Enable jumbo frame support by configuring a larger-than-default MTU size on a port or interface.

Catalyst 4500 series switch Ethernet LAN ports configured with a nondefault MTU size accept frames containing packets with a size between 1500 and 9216 bytes (including Ethernet payload, header and trailer). (The maximum MTU size for a Catalyst 4948 series switch is 9198 bytes (not including header and trailer.)) With a nondefault MTU size configured, the packet size of ingress frames is checked. If the packet is larger than the configured MTU, it is dropped.

For traffic that needs to be routed, the MTU of the egress port is checked. If the MTU is smaller than the packet size, the packet is forwarded to the CPU. If the "do not fragment bit" is not set, it is fragmented. Otherwise, the packet is dropped.

**Note**    Jumbo frame support does not fragment Layer 2 switched packets.

The Catalyst 4500 series switch does not compare the packet size with the MTU at the egress port, but jumbo frames are dropped in ports that do not support them. The frames can be transmitted in ports that do support jumbo frames, even though the MTU is not configured to jumbo size.

> **Note** Jumbo frame support is only configured per interface; jumbo frame support cannot be configured globally.

## Ethernet Ports

These sections describe configuring nondefault MTU sizes on Ethernet ports:

-
-

### Ethernet Port Overview

With Cisco IOS Release 12.2(25)EW, configuring a nondefault MTU size on certain Ethernet ports limits the size of ingress packets. The MTU does not impact the egress packets.

With releases earlier than Cisco IOS Release 12.1(13)EW, you can configure the MTU size only on Gigabit Ethernet.

### Layer 3 and Layer 2 EtherChannels

With Release Cisco IOS Release 12.2(25)EW and later releases, you can configure all the interfaces in an EtherChannel provided that they have the same MTU. Changing the MTU of an EtherChannel changes the MTU of all member ports. If the MTU of a member port cannot be changed to the new value, that port is suspended (administratively shut down). A port cannot join an EtherChannel if the port has a different MTU. If a member port of an EtherChannel changes MTU, the member port is suspended.

## VLAN Interfaces

If switch ports reside in the same VLAN, either configure all of the switch ports to handle jumbo frames and support the same MTU size, or configure none of them. However, such uniformity of MTU size in the same VLAN is not enforced.

When a VLAN has switch ports with different MTU size, packets received from a port with a larger MTU might be dropped when they are forwarded to a port with a smaller MTU.

If the switch ports in a VLAN have jumbo frames enabled, the corresponding SVI can have jumbo frames enabled. The MTU of an SVI should always be smaller than the smallest MTU among all the switch ports in the VLAN, but this condition is not enforced.

The MTU of a packet is not checked on the ingress side for an SVI; it is checked on the egress side of an SVI. If the MTU of a packet is larger than the MTU of the egress SVI, the packet will be sent to the CPU for fragmentation processing. If the "do not fragment" bit is not set, the packet is fragmented. Otherwise, the packet is dropped.

## Configuring MTU Sizes

To configure the MTU size, perform this task:

| | Command | Purpose |
|---|---|---|
| Step 1 | `Switch(config)# ` **`interface`** `{{`**`vlan`** `vlan_ID} \| {{type`[1] ` slot/port} \| {`**`port-channel`** ` port_channel_number} slot/port}}` | Selects the interface to configure. |

|  | Command | Purpose |
|---|---|---|
| Step 2 | Switch(config-if)# **mtu** *mtu_size* | Configures the MTU size. |
|  | Switch(config-if)# **no mtu** | Reverts to the default MTU size (1500 bytes). |
| Step 3 | Switch(config-if)# **end** | Exits configuration interface mode. |
| Step 4 | Switch(config)# **end** | Exits configuration mode. |
| Step 5 | Switch# **show running-config interface** [{**fastethernet** \| **gigabitethernet**} *slot/port*] | Verifies the running configuration. |

1. *type* = **fastethernet**, **gigabitethernet**, or **tengigabitethernet**

**Note** When you remove a linecard, and then reinsert the card, some or all of the MTU values configured on the ports of that linecard may be unconfigured. This will happen if the systemwide limit of 32 different MTUs is reached while the card is removed. Upon reinserting the linecard, the system attempts to reapply the MTU configuration on the ports. If this attempt fails, the MTU values are set to the default.

**Note** When configuring the MTU size for VLAN interfaces and Layer 3 and Layer 2 Ethernet ports, note that the supported MTU values are from 1500 to 9198 bytes.

This example shows how to configure the MTU size on Gigabit Ethernet port 1/1:

```
switch# conf terminal
switch(config)# interface gi1/1
switch(config-if)# mtu 9198
switch(config-if)# end
switch(config)# end
switch# show interface gigabitethernet 1/2
GigabitEthernet1/2 is administratively down, line protocol is down
  Hardware is C6k 1000Mb 802.3, address is 0030.9629.9f88 (bia 0030.9629.9f88)
  MTU 9216 bytes, BW 1000000 Kbit, DLY 10 usec,
<...Output Truncated...>
switch#
```

For details on how to configure IP MTU size, refer to .

# Interacting with Baby Giants

The baby giants feature, introduced in Cisco IOS Release 12.1(12c)EW, uses the global command **system mtu <size>** to set the global baby giant MTU. This feature also allows certain interfaces to support Ethernet payload size of up to 1552 bytes.

Both the **system mtu** command and the per-interface **mtu** command can operate on interfaces that can support jumbo frames, but the per-interface **mtu** command takes precedence.

For example, before setting the per-interface MTU for interface gi1/1, you issue the **system mtu 1550** command to change the MTU for gi1/1 to 1550 bytes. Next, you issue the per-interface **mtu** command to change the MTU for gi1/1 to 9198 bytes. Now, if you change the baby giant MTU to 1540 bytes with the command **system mtu 1540**, the MTU for gi1/1 remains unchanged at 9198 bytes.

# Configuring the Port Debounce Timer

> **Note**  You can configure port debounce only on 10 Gigabit Ethernet ports.

The port debounce timer suppresses notification of short link-down events. Link-down events that are shorter than the port debounce timer are not notified to Layer 2 or Layer 3 protocols, decreasing traffic loss due to network reconfiguration.  You can configure the port debounce timer separately on each LAN port.

> ⚠ **Caution**  Enabling the port debounce timer causes a delay in link down detections, resulting in loss of traffic during the debouncing period. This situation might affect the convergence and reconvergence of some Layer 2 and Layer 3 protocols.

To configure the debounce timer on a port, perform this task:

|  | Command | Purpose |
|---|---|---|
| **Step 1** | Switch(config)# **interface tengigabitethernet** *slot/port* | Selects the port to configure. |
| **Step 2** | Switch(config-if)# **link debounce [time** *debounce_time]* | Configures the debounce timer. |
| | Switch(config-if)# **no link debounce** | Reverts to the default setting. |
| **Step 3** | Switch# **show interfaces debounce** | Verifies the configuration. |

> **Note**  The default time is 10ms for E-series supervisor engines and line cards (including Catalyst 4900M, Catalyst 4948-E, Supervisor Engine 6-E, and Supervior Engine 6L-E.  All other supervisor engines use a default of 100ms.

When configuring the debounce timer on a port, you can increase the port debounce timer value between 10 milliseconds and 5000 milliseconds on the tengigabitethernet ports.

> **Note**  By default, debounce is disabled. If you configure debounce without a time, the value is set at 100ms.

This example shows how to enable the port debounce timer on TenGigabit Ethernet port 2/1 and to accept the default value (10 ms):

```
Switch# config terminal
Enter configuration commands, one per line.  End with CNTL/Z.
Switch(config)# interface tenGigabitEthernet 2/1
Switch(config-if)# link debounce
Warning: Enabling debounce feature causes link down detection to be delayed
Switch(config-if)# exit
```

This example shows how to enable the port debounce timer of 5000 ms on TenGigabit Ethernet port 2/2 and to verify the setting:

```
Switch# config terminal
Enter configuration commands, one per line.  End with CNTL/Z.
Switch(config)# interface tenGigabitEthernet 2/2
```

```
Switch(config-if)# link debounce time 5000
Warning: Enabling debounce feature causes link down detection to be delayed
Switch(config-if)# end
Switch#
Switch# show interfaces debounce | include enable
Te2/1    enable          10
Te2/2    enable          5000
Switch#
```

# Configuring auto-MDIX on a Port

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

Auto-MDIX is enabled by default. When you enable auto-MDIX, you must also set the speed on the port to **auto** so that for the feature to operate correctly. auto-MDIX is supported on copper media ports. It is not supported on fiber media ports.

**Note** The following linecards support Auto-MDIX by default, when port auto-negotiation is enabled: WS-X4424-GB-RJ45, WS-X4448-GB-RJ45,WS-X4548-GB-RJ45 and WS-X4412-2GB-T. You cannot disable them with the **mdix** command.

**Note** The following linecards do not support Auto-MDIX, neither by default nor by CLI: WS-X4548-GB-RJ45V, WS-X4524-GB-RJ45V, WS-X4506-GB-T,WS-X4148-RJ, WS-X4248-RJ21V, WS-X4248-RJ45V, WS-X4224-RJ45V and WS-X4232-GB-RJ.

**Note** The following linecards support Auto-MDIX through the CLI on their copper media ports: WS-X4124-RJ45, WS-X4148-RJ45 (hardware revision 3.0 or higher), and WS-X4232-GB-RJ45 (hardware revision 3.0, or higher), WS-X4920-GE-RJ45 and WS-4648-RJ45V+E (Auto-MDIX support when inline power is disabled on the port).

Table 6-1 shows the link states that results from auto-MDIX settings and correct and incorrect cabling.

*Table 6-1        Link Conditions and auto-MDIX Settings*

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

To configure auto-MDIX on a port, perform the following task:

| | Command | Purpose |
|---|---|---|
| **Step 1** | Switch# **configure terminal** | Enters global configuration mode |
| **Step 2** | Switch(config)# **interface** *interface-id* | Enters interface configuration mode for the physical interface to be configured. |
| **Step 3** | Switch(config-if)# **speed auto** | Configures the port to autonegotiate speed with the connected device. |
| **Step 4** | Switch(config-if)# **mdix auto** | Enables auto-MDIX on the port. |
| **Step 5** | Switch(config-if)# **end** | Returns to privileged EXEC mode. |
| **Step 6** | Switch# **show interfaces** *interface-id* | Verifies the configuration of the auto-MDIX feature on the interface. |
| **Step 7** | Switch# **copy running-config startup-config** | (Optional) Saves your entries in the configuration file. |

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

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

```
Switch# configure terminal
Switch(config)# interface fastethernet 6/5
Switch(config-if)# speed auto
Switch(config-if)# mdix auto
Switch(config-if)# end
```

## Displaying the Interface auto-MDIX Configuration

To display the interface speed and duplex mode configuration for an interface, perform this task:

| | Command | Purpose |
|---|---|---|
| **Step 1** | Switch> **enable** | Enables privileged EXEC mode. <br><br> • Enter your password if prompted. |
| **Step 2** | Switch# **show interfaces type** *slot***/***interface* | Displays the interface auto-MDIX configuration setting and operational state. |

Depending on how the **speed auto** and the **mdix auto** commands are configured on a supported linecard interface, the **show interfaces** command displays the following possible auto-MDIX statuses:

Table 6-2 shows the auto-MDIX setting and operational state and the status of auto-MDIX.

*Table 6-2    Auto-MDIX and Operational State*

| Auto-MDIX Setting And Operational State on an Interface | Description |
|---|---|
| Auto-MDIX on (operational: on) | Auto-MDIX is enabled and is fully functioning. |
| Auto-MDIX on (operational: off) | Auto-MDIX is enabled on this interface but it is not functioning. To allow auto-MDIX feature to function properly, you must also set the interface speed to be autonegotiated. |
| Auto-MDIX off | Auto-MDIX has been disabled with the **no mdix auto** command. |

This example show s how to display the auto-MDIX configuration setting and its operational state on Fast Ethernet interface 6/1:

```
Switch# show interfaces fastethernet 6/1
FastEthernet6/1 is up, line protocol is up (connected)
  Hardware is Fast Ethernet Port, address is 0001.64fe.e5d0 (bia 0001.64fe.e5d0)
  MTU 1500 bytes, BW 100000 Kbit, DLY 100 usec,
     reliability 255/255, txload 1/255, rxload 1/255
  Encapsulation ARPA, loopback not set
  Keepalive set (10 sec)
  Full-duplex, 100Mb/s, link type is auto, media type is 10/100BaseTX
  input flow-control is unsupported output flow-control is unsupported
  Auto-MDIX on (operational: on)
  ARP type: ARPA, ARP Timeout 04:00:00
  Last input 00:00:16, output never, output hang never
  Last clearing of "show interface" counters never
  Input queue: 0/2000/0/0 (size/max/drops/flushes); Total output drops: 0
  Queueing strategy: fifo
  Output queue: 0/40 (size/max)
  5 minute input rate 0 bits/sec, 0 packets/sec
  5 minute output rate 0 bits/sec, 0 packets/sec
     511 packets input, 74464 bytes, 0 no buffer
     Received 511 broadcasts (511 multicasts)
     0 runts, 0 giants, 0 throttles
     0 input errors, 0 CRC, 0 frame, 0 overrun, 0 ignored
     0 input packets with dribble condition detected
     3552 packets output, 269088 bytes, 0 underruns
     0 output errors, 0 collisions, 0 interface resets
     0 babbles, 0 late collision, 0 deferred
     1 lost carrier, 0 no carrier
     0 output buffer failures, 0 output buffers swapped out
Switch#
```

# Understanding Online Insertion and Removal

The online insertion and removal (OIR) feature supported on the Catalyst 4500 series switch allows you to remove and replace modules while the system is online. You can shut down the module before removal and restart it after insertion without causing other software or interfaces to shut down.

You do not need to enter a command to notify the software that you are going to remove or install a module. The system notifies the supervisor engine that a module has been removed or installed and scans the system for a configuration change. The newly installed module is initialized, and each interface type is verified against the system configuration; then the system runs diagnostics on the new interface. There is no disruption to normal operation during module insertion or removal.

If you remove a module and then replace it, or insert a different module of the same type into the same slot, no change to the system configuration is needed. An interface of a type that has been configured previously will be brought online immediately. If you remove a module and insert a module of a different type, the interface(s) on that module will be administratively up with the default configuration for that module.

# Monitoring and Maintaining the Interface

The following sections describe how to monitor and maintain the interfaces:

-

# Monitoring Interface and Controller Status

The Cisco IOS software for the Catalyst 4500 series switch contains commands that you can enter at the EXEC prompt to display information about the interface, including the version of the software and the hardware, the controller status, and statistics about the interfaces. The following table lists some of the interface monitoring commands. (You can display the full list of **show** commands by entering the **show ?** command at the EXEC prompt.) These commands are fully described in the *Interface Command Reference*.

To display information about the interface, perform this task:

|        | Command | Purpose |
|--------|---------|---------|
| **Step 1** | Switch# **show interfaces** [*type slot/interface*] | Displays the status and configuration of all interfaces or of a specific interface. |
| **Step 2** | Switch# **show running-config** | Displays the configuration currently running in RAM. |
| **Step 3** | Switch# **show protocols** [*type slot/interface*] | Displays the global (system-wide) and interface-specific status of any configured protocol. |
| **Step 4** | Switch# **show version** | Displays the hardware configuration, software version, the names and sources of configuration files, and the boot images. |

This example shows how to display the status of Fast Ethernet interface 5/5:

```
Switch# show protocols fastethernet 5/5
FastEthernet5/5 is up, line protocol is up
Switch#
```

# Clearing and Resetting the Interface

To clear the interface counters shown with the **show interfaces** command, enter the following command:

| Command | Purpose |
|---------|---------|
| **Switch# clear counters** {*type slot/interface*} | Clears interface counters. |

This example shows how to clear and reset the counters on Fast Ethernet interface 5/5:

```
Switch# clear counters fastethernet 5/5
Clear "show interface" counters on this interface [confirm] y
Switch#
*Sep 30 08:42:55: %CLEAR-5-COUNTERS: Clear counter on interface FastEthernet5/5
by vty1 (171.69.115.10)
Switch#
```

The **clear counters** command (without any arguments) clears all the current interface counters from all interfaces.

> **Note** The **clear counters** command does not clear counters retrieved with SNMP; it clears only those counters displayed with the EXEC **show interfaces** command.

# Shutting Down and Restarting an Interface

You can disable an interface, which 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 will not be mentioned in any routing updates.

To shut down an interface and then restart it, perform this task:

| | Command | Purpose |
|---|---|---|
| **Step 1** | Switch(config)# **interface** {**vlan** *vlan_ID*} \| {{**fastethernet** \| **gigabitethernet** \| **tengigabitethernet**} *slot*/*port*} \| {**port-channel** *port_channel_number*} | Specifies the interface to be configured. |
| **Step 2** | Switch(config-if)# **shutdown** | Shuts down the interface. |
| **Step 3** | Switch(config-if)# **no shutdown** | Reenables the interface. |

This example shows how to shut down Fast Ethernet interface 5/5:

```
Switch(config)# interface fastethernet 5/5
Switch(config-if)# shutdown
Switch(config-if)#
*Sep 30 08:33:47: %LINK-5-CHANGED: Interface FastEthernet5/5, changed state to a
administratively down
Switch(config-if)#
```

This example shows how to reenable Fast Ethernet interface 5/5:

```
Switch(config-if)# no shutdown
Switch(config-if)#
*Sep 30 08:36:00: %LINK-3-UPDOWN: Interface FastEthernet5/5, changed state to up
Switch(config-if)#
```

To verify whether an interface is disabled, enter the EXEC **show interfaces** command. An interface that has been shut down will appear as "administratively down."

# Configuring Interface Link Status and Trunk Status Events

You can configure interface link status and trunk status events. On the Catalyst 4500 series switch, the following interface logging event notifications are supported both globally and per interface:

- Enable/disable notification on the interface whenever its data link status is changed.
- Enable/disable notification on the trunk interface whenever its trunking status is changed.

Use the **[no]** **logging event link-status [use-global]** command to enable/disable the interface link status event. Use the **[no]** **logging event trunk-status [use-global]** command to enable/disable the interface trunk status event.

Each interface link status logging event can be configured in one of the following states:

- **logging event link-status** - Link status logging event is enabled explicitly on the interface regardless of the switch global setting.

- **no logging event link-status** - Link status logging event is disabled explicitly on the interface regardless of the switch global setting.

- **logging event link-status use-global** - This is the default link status logging event configuration on the interface; its configuration should follow the switch global link status logging event setting.

The interface trunk status logging event can be configured in the same configuration states.

## Configuring Link Status Event Notification for an Interface

To enable/disable a link status logging event, enter one of the following commands:

| Command | Purpose |
|---|---|
| Switch(config-if)# **logging event link-status** | Enables interface link status logging. |
| Switch(config-if)# **no logging event link-status** | Disables interface link status logging. |
| Switch(config-if)# **logging event link-status use-global** | Specifies the global default setting for interface link status logging. |

## Global Settings

You can also provide a global configuration for the corresponding logging event. A global configuration provides default logging settings for all interfaces. The **[no] logging event link-status global** command lets you enable/disable the interface link status logging for the entire switch. The **[no] logging event trunk-status global** command lets you enable/disable interface trunk status logging for the entire switch.

Each interface link status logging event, if not configured at the interface level, will use the following global logging event setting:

- **logging event link-status global** - Link status logging event is enabled, if not configured on the interface.

- **no logging event link-status global** - Link status logging event is disabled, if not configured on the interface.

The interface trunk status logging event has similar global configurations.

## Configuring a Switch Global Link Status Logging Event

To enable/disable the global link status logging event, enter one of the following commands:

| Command | Purpose |
|---|---|
| Switch(config-if)# **logging event link-status global** | Enables global link status logging. |
| Switch(config-if)# **no logging event link-status global** | Disables global link status logging. |

# Result

The following example displays a summary of the operating states for the interface logging event under different combinations of global and interface logging settings:

```
global setting        interface setting        actual logging state
--------------        ----------------        --------------------
    on                    on                        on
    off                   on                        on
    on                    off                       off
    off                   off                       off
    on                    default(use-global)       on
    off                   default(use-global)       off
```

The following example displays the configuration and logging message output for link status and trunk status logging events:

```
//
// The global link status and trunk status logging events are enabled.
//
Switch# show running | include logging
show running | include logging
logging event link-status global
logging event trunk-status global
Switch#

//
// The interface link status and trunk status logging settings
// are set to default values, which follow regardless of the global
// setting.
//
Switch# show running interface g1/4
Building configuration...

Current configuration: 97 bytes
!
interface GigabitEthernet1/4
 switchport trunk encapsulation dot1q
 switchport mode trunk
end
Switch#

//
// The trunk status logging messages for the interface are
// displayed whenever the interface trunking status is changed.
// Here we change the other end node's trunking encapsulation
// from dot1q to isl.
//
3d00h: %DTP-5-ILGLCFG: Illegal config(on,isl--on,dot1q) on Gi1/4
3d00h: %DTP-5-ILGLCFG: Illegal config(on,isl--on,dot1q) on Gi1/4
3d00h: %DTP-5-ILGLCFG: Illegal config(on,isl--on,dot1q) on Gi1/4

//
// The link and trunk status logging message for the interface
// are displayed whenever the interface link status is changed.
// Here we do a "shut" and "no shut" on the other end link node.
//
3d00h: %DTP-5-NONTRUNKPORTON: Port Gi1/4 has become non-trunk
3d00h: %LINEPROTO-5-UPDOWN: Line protocol on Interface
GigabitEthernet1/4, changed state to down
3d00h: %LINK-3-UPDOWN: Interface GigabitEthernet1/4, changed state to
down
3d00h: %LINK-3-UPDOWN: Interface GigabitEthernet1/4, changed state to up
```

```
3d00h: %DTP-5-TRUNKPORTON: Port Gi1/4 has become dot1q trunk
3d00h: %LINEPROTO-5-UPDOWN: Line protocol on Interface
GigabitEthernet1/4, changed state to up
```

# Resetting the Interface to the Default Configuration

If you have configured a interface with many command lines and you want to clear all the configuration on that interface, you can use the **default interface** global configuration command, as follows:

```
Switch(config)# default interface fastEthernet 3/5
Interface FastEthernet3/5 set to default configuration
```

This command will clear all the configurations and shutdown the interface:

```
Switch# show run interface fastethernet 3/5
Building configuration...

Current configuration : 58 bytes
!
interface FastEthernet3/5
 no ip address
 shutdown
end
```