Configuring Basic Interface Parameters

About the Basic Interface Parameters

Description

For the Ethernet and management interfaces, you can configure the description parameter to provide a recognizable name for the interface. Using a unique name for each interface allows you to quickly identify the interface when you are looking at a listing of multiple interfaces.

For information about setting the description parameter for port-channel interfaces, see the "Configuring a Port-Channel Description" section. For information about configuring this parameter for other interfaces, see the "Configuring the Description" section.

Beacon

The beacon mode allows you to identify a physical port by flashing its link state LED with a green light. By default, this mode is disabled. To identify the physical port for an interface, you can activate the beacon parameter for the interface.

For information about configuring the beacon parameter, see the "Configuring the Beacon Mode" section.

Error Disabled

A port is in the error-disabled (err-disabled) state when the port is enabled administratively (using the **no shutdown** command) but disabled at runtime by any process. For example, if UDLD detects a unidirectional link, the port is shut down at runtime. However, because the port is administratively enabled, the port status

displays as err-disable. Once a port goes into the err-disable state, you must manually reenable it or you can configure a timeout value that provides an automatic recovery. By default, the automatic recovery is not configured, and by default, the err-disable detection is enabled for all causes.

When an interface is in the err-disabled state, use the **errdisable detect cause** command to find information about the error.

You can configure the automatic error-disabled recovery timeout for a particular error-disabled cause and configure the recovery period.

The **errdisable recovery cause** command provides an automatic recovery after 300 seconds.

You can use the **errdisable recovery interval** command to change the recovery period within a range of 30 to 65535 seconds. You can also configure the recovery timeout for a particular err-disable cause.

If you do not enable the error-disabled recovery for the cause, the interface stays in the error-disabled state until you enter the **shutdown** and **no shutdown** commands. If the recovery is enabled for a cause, the interface is brought out of the error-disabled state and allowed to retry operation once all the causes have timed out. Use the **show interface status err-disabled** command to display the reason behind the error.

MDIX

The medium dependent interface crossover (MDIX) parameter enables or disables the detection of a crossover connection between devices. This parameter applies only to copper interfaces. By default, this parameter is enabled. The **no mdix auto** command is supported only on N9K-C93108TC-EX, N9K-C93108TC-FX, N9K-X9788TC-FX, and N9K-C9348GC-FXP devices.

For information about configuring the MDIX parameter, see the Configuring the MDIX Parameter section.

Interface Status Error Policy

Cisco NX-OS policy servers such as Access Control List (ACL) Manager and Quality of Service (QoS) Manager, maintain a policy database. A policy is defined through the command-line interface.

Policies are pushed when you configure a policy on an interface to ensure that policies that are pushed are consistent with the hardware policies. To clear the errors and to allow the policy programming to proceed with the running configuration, enter the **no shutdown** command. If the policy programming succeeds, the port is allowed to come up. If the policy programming fails, the configuration is inconsistent with the hardware policies and the port is placed in an error-disabled policy state. The error-disabled policy state remains and the information is stored to prevent the same port from being brought up in the future. This process helps to avoid unnecessary disruption to the system.

Modifying Interface MTU Size

The maximum transmission unit (MTU) size specifies the maximum frame size that an Ethernet port can process. For transmissions to occur between two ports, you must configure the same MTU size for both ports. A port drops any frames that exceed its MTU size.

By default, the Cloud-Scale ASIC NX-OS system always allows an extra 166B in the MTU on top of the configured value in order to fully support/accept different types of encapsulations in the hardware.

Cisco NX-OS allows you to configure MTU on an interface, with options to configure it on different level in the protocol stack. By default, each interface has an MTU of 1500 bytes, which is the IEEE 802.3 standard

for Ethernet frames. Larger MTU sizes are possible for more efficient processing of data to allow different application requirements. The larger frames, are also called jumbo frames, can be up to 9216 bytes in size.

MTU is configured per interface, where an interface can be a Layer 2 or a Layer 3 interface. For a Layer 2 interface, you can configure the MTU size with one of two values, the value system default MTU value or the system jumbo MTU value. The system default MTU value is 1500 bytes. Every Layer 2 interface is configured with this value by default. You can configure an interface with the default system jumbo MTU value, that is 9216 bytes. To allow an MTU value from 1500 through 9216, you must adjust the system jumbo MTU to an appropriate value where interface can be configured with the same value.

> **Note**   You can change the system jumbo MTU size. When the value is changed, the Layer 2 interfaces that use the system jumbo MTU value, will automatically changes to the new system jumbo MTU value.

A Layer 3 interface, can be Layer 3 physical interface (configure with no switchport), switch virtual interface (SVI), and sub-interface, you can configure an MTU size between 576 and 9216 bytes.

For the Cisco Nexus 9372 switch, the following applies:

- The 10-G interfaces are mapped to specific hardware ports where the default MTU is 1500.

- The 40-G interfaces are mapped as a HiGiG port where the default MTU is 3FFF and the MTU limit check is disabled.

- In the case of 40-G interfaces, since the MTU limit check is disabled, it ignores the packet size and traffic flows irrespective of its MTU.

- When the configured MTU of all interfaces on the switch do not match, the switch's behavior may vary depending on the specific port that is mismatched as well as the traffic flow. The following are examples of the switch's behavior in various scenarios:

  - When a Layer 3 port receives a frame whose length exceeds the port's MTU size, the port will drop the frame.

  - When a Layer 3 port receives a frame whose length is less than the ingress port's MTU size, but greater than the egress Layer 3 port's MTU size, then the frame is punted to the supervisor of the switch.

    1. If the frame is an IP packet that has the Don't Fragment (DF) bit set, then the frame will be dropped in software. Otherwise, the frame will be fragmented in software.

    2. Otherwise, the frame will be fragmented in software.

    3. This can cause performance issues (such as increased latency or packet loss for affected traffic flows) due to Control Plane Policing (CoPP) enabled by default on Cisco Nexus switches. For more information about Control Plane Policing, refer to the **Configuring Control Plane Policing** chapter of the **Cisco Nexus 9000 Series NX-OS Security Configuration Guide**.

  - When a Layer 2 port receives a frame whose length exceeds the port's MTU size, the port will drop the frame.

  - When a Layer 2 port receives a frame whose length is less than the ingress port's MTU size, but greater than the egress Layer 2 port's MTU size, and the frame is routed between VLANs by the switch, then the frame is punted to the supervisor of the switch.

1. If the frame is an IP packet that has the Don't Fragment (DF) bit set, then the frame will be dropped in software. Otherwise, the frame will be fragmented in software.

2. Otherwise, the frame will be fragmented in software.

3. This can cause performance issues (such as increased latency or packet loss for affected traffic flows) due to Control Plane Policing (CoPP) enabled by default on Cisco Nexus switches. For more information about Control Plane Policing, refer to the **Configuring Control Plane Policing** chapter of the **Cisco Nexus 9000 Series NX-OS Security Configuration Guide**.

   • When a Layer 2 port receives a frame whose length is less than the ingress port's MTU size, but greater than the egress Layer 2 port's MTU size, and the frame is switched within the same VLAN by the switch, then the switch will drop the frame.

The maximum transmission unit (MTU) size specifies the maximum frame size that an Ethernet port can process. For transmissions to occur between two ports, you must configure the same MTU size for both ports. A port drops any frames that exceed its MTU size. By default, each port has an MTU of 1500 bytes, which is the IEEE 802.3 standard for Ethernet frames. Larger MTU sizes are possible for more efficient processing of data with less overhead. The larger frames, called jumbo frames, can be up to 9216 bytes in size, which is also the default system jumbo MTU size. On a Layer 3 interface, you can configure an MTU size between 576 and 9216 bytes.

Note   The global LAN port MTU size applies to the traffic through a Layer 3 Ethernet LAN port that is configured with a nondefault MTU size.

For a Layer 2 port, you can configure an MTU size that is either the system default (1500 bytes) or the system jumbo MTU size (initially 9216 bytes).

Note   If you change the system jumbo MTU size, Layer 2 ports automatically use the system default MTU size (1500 bytes) unless you specify the new system jumbo MTU size for some or all of those ports.

For information about setting the MTU size, see the *Configuring the MTU Size* section.

Note   On Cisco Nexus 9300-FX2 and 9300-GX devices, if ingress interface is configured with an MTU less than 9216, FTE does not capture input errors and does not display any events. However, if the ingress interface is configured with an MTU of 9216, FTE displays all the events.

Bandwidth

Ethernet ports have a fixed bandwidth of 1,000,000 Kb at the physical layer. Layer 3 protocols use a bandwidth value that you can set for calculating their internal metrics. The value that you set is used for informational purposes only by the Layer 3 protocols—it does not change the fixed bandwidth at the physical layer. For example, the Enhanced Interior Gateway Routing Protocol (EIGRP) uses the minimum path bandwidth to determine a routing metric, but the bandwidth at the physical layer remains at 1,000,000 Kb.

For information about configuring the bandwidth parameter for port-channel interfaces, see the "Configuring the Bandwidth and Delay for Informational Purposes" section. For information about configuring the bandwidth parameter for other interfaces, see the "Configuring the Bandwidth" section.

Throughput Delay

Specifying a value for the throughput-delay parameter provides a value used by Layer 3 protocols; it does not change the actual throughput delay of an interface. The Layer 3 protocols can use this value to make operating decisions. For example, the Enhanced Interior Gateway Routing Protocol (EIGRP) can use the delay setting to set a preference for one Ethernet link over another, if other parameters such as link speed are equal. The delay value that you set is in the tens of microseconds.

For information about configuring the bandwidth parameter for port-channel interfaces, see the "Configuring the Bandwidth and Delay for Informational Purposes" section. For information about configuring the throughput-delay parameter for other interfaces, see the "Configuring the Throughput Delay" section.

Administrative Status

The administrative-status parameter determines whether an interface is up or down. When an interface is administratively down, it is disabled and unable to transmit data. When an interface is administratively up, it is enabled and able to transmit data.

For information about configuring the administrative status parameter for port-channel interfaces, see the "Shutting Down and Restarting the Port-Channel Interface" section. For information about configuring the administrative-status parameter for other interfaces, see the "Shutting Down and Activating the Interface" section.

Unidirectional Link Detection Parameter

UDLD Overview

The Cisco-proprietary Unidirectional Link Detection (UDLD) protocol allows devices that are connected through fiber-optic or copper (for example, Category 5 cabling) Ethernet cables to monitor the physical configuration of the cables and detect when a unidirectional link exists. When a device detects a unidirectional link, UDLD shuts down the affected LAN port and alerts the user. Unidirectional links can cause a variety of problems.

UDLD performs tasks that autonegotiation cannot perform, such as detecting the identities of neighbors and shutting down misconnected LAN ports. When you enable both autonegotiation and UDLD, Layer 1 detections work to prevent physical and logical unidirectional connections and the malfunctioning of other protocols.

A unidirectional link occurs whenever traffic transmitted by the local device over a link is received by the neighbor but traffic transmitted from the neighbor is not received by the local device. If one of the fiber strands in a pair is disconnected, as long as autonegotiation is active, the link does not stay up. In this case, the logical link is undetermined, and UDLD does not take any action. If both fibers are working normally at Layer 1, UDLD determines whether those fibers are connected correctly and whether traffic is flowing bidirectionally between the correct neighbors. This check cannot be performed by autonegotiation, because autonegotiation operates at Layer 1.

The Cisco Nexus 9000 Series device periodically transmits UDLD frames to neighbor devices on LAN ports with UDLD enabled. If the frames are echoed back within a specific time frame and they lack a specific acknowledgment (echo), the link is flagged as unidirectional and the LAN port is shut down. Devices on both
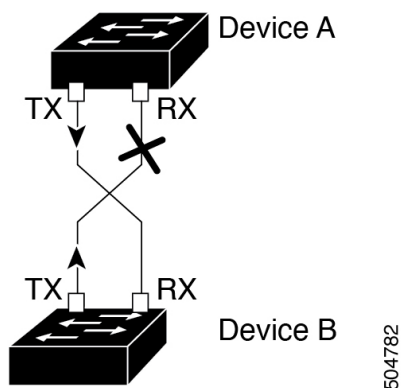
ends of the link must support UDLD in order for the protocol to successfully identify and disable unidirectional links. You can configure the transmission interval for the UDLD frames, either globally or for the specified interfaces.

Note   By default, UDLD is locally disabled on copper LAN ports to avoid sending unnecessary control traffic on this type of media.

The figure shows an example of a unidirectional link condition. Device B successfully receives traffic from device A on the port. However, device A does not receive traffic from device B on the same port. UDLD detects the problem and disables the port.

Figure 1: Unidirectional Link

Default UDLD Configuration

The following table shows the default UDLD configuration.

Table 1: UDLD Default Configuration

Feature	Default Value
UDLD global enable state	Globally disabled
UDLD per-port enable state for fiber-optic media	Enabled on all Ethernet fiber-optic LAN ports
UDLD per-port enable state for twisted-pair (copper) media	Disabled on all Ethernet 10/100 and 1000BASE-TX LAN ports
UDLD aggressive mode	Disabled
UDLD message interval	15 seconds

For information about configuring the UDLD for the device and its port, see the "Configuring the UDLD Mode" section.

UDLD Normal and Aggressive Modes

UDLD supports Normal and Aggressive modes of operation. By default, Normal mode is enabled.

In Normal mode, UDLD detects the following link errors by examining the incoming UDLD packets from the peer port:

- Empty echo packet

- Uni-direction

- TX/RX loop

- Neighbor mismatch

By default, UDLD aggressive mode is disabled. You can configure UDLD aggressive mode only on point-to-point links between network devices that support UDLD aggressive mode.

If UDLD aggressive mode is enabled, when a port on a bidirectional link that has a UDLD neighbor relationship established stops receiving UDLD frame, UDLD tries to re-establish the connection with the neighbor. After eight failed retries, the port is disabled.

In the following scenarios, enabling the UDLD aggressive mode disables one of the ports to prevent the discarding of traffic.

- One side of a link has a port stuck (both transmission and receive)

- One side of a link remains up while the other side of the link is down

Note You enable the UDLD aggressive mode globally to enable that mode on all the fiber ports. You must enable the UDLD aggressive mode on copper ports on specified interfaces.

Tip When a line card upgrade is being performed during an in-service software upgrade (ISSU) and some of the ports on the line card are members of a Layer 2 port channel and are configured with UDLD aggressive mode, if you shut down one of the remote ports, UDLD puts the corresponding port on the local device into an error-disabled state. This behavior is correct.

To restore service after the ISSU has completed, enter the **shutdown** command followed by the **no shutdown** command on the local port.

Port-Channel Parameters

A port channel is an aggregation of physical interfaces that comprise a logical interface. You can bundle up to 32 individual interfaces into a port channel to provide increased bandwidth and redundancy. Port channeling also load balances traffic across these physical interfaces. The port channel stays operational if at least one physical interface within the port channel is operational.

You can create Layer 3 port channels by bundling compatible Layer 3 interfaces.

Any configuration changes that you apply to the port channel are applied to each interface member of that port channel.

For information about port channels and for information about configuring port channels, see Chapter 6, "Configuring Port Channels."

Port Profiles

Beginning with 7.0(3)I4(1) on Cisco Nexus 9300 Series switches, you can create a port profile that contains many interface commands and apply that port profile to a range of interfaces. Each port profile can be applied only to a specific type of interface; the choices are as follows:

On Cisco Nexus 9300 Series switches, you can create a port profile that contains many interface commands and apply that port profile to a range of interfaces. Each port profile can be applied only to a specific type of interface; the choices are as follows:

- Ethernet

- VLAN network interface

- Loopback

- Port channel

- Tunnel

When you choose Ethernet or port channel as the interface type, the port profile is in the default mode which is Layer 3. Enter the **switchport** command to change the port profile to Layer 2 mode.

You inherit the port profile when you attach the port profile to an interface or range of interfaces. When you attach, or inherit, a port profile to an interface or range of interfaces, the system applies all the commands in that port profile to the interfaces. Additionally, you can have one port profile inherit the settings from another port profile. Inheriting another port profile allows the initial port profile to assume all of the commands of the second, inherited, port profile that do not conflict with the initial port profile. Four levels of inheritance are supported. The same port profile can be inherited by any number of port profiles.

The system applies the commands inherited by the interface or range of interfaces according to the following guidelines:

- Commands that you enter under the interface mode take precedence over the port profile's commands if there is a conflict. However, the port profile retains that command in the port profile.

- The port profile's commands take precedence over the default commands on the interface, unless the port-profile command is explicitly overridden by the default command.

- When a range of interfaces inherits a second port profile, the commands of the initial port profile override the commands of the second port profile if there is a conflict.

- After you inherit a port profile onto an interface or range of interfaces, you can override individual configuration values by entering the new value at the interface configuration level. If you remove the individual configuration values at the interface configuration level, the interface uses the values in the port profile again.

- There are no default configurations associated with a port profile.

- On Cisco Nexus C9232E-B1 switch, the ports will be in 2x400G profile by default. To change to other breakout mode, you must configure **no interface breakout module 1 port <port#> map 400g-2x** " and then to "**interface breakout module 1 port <port#> map <map name>**.

A subset of commands are available under the port-profile configuration mode, depending on which interface type you specify.

To apply the port-profile configurations to the interfaces, you must enable the specific port profile. You can configure and inherit a port profile onto a range of interfaces prior to enabling the port profile. You would then enable that port profile for the configurations to take effect on the specified interfaces.

If you inherit one or more port profiles onto an original port profile, only the last inherited port profile must be enabled; the system assumes that the underlying port profiles are enabled.

When you remove a port profile from a range of interfaces, the system undoes the configuration from the interfaces first and then removes the port-profile link itself. Also, when you remove a port profile, the system checks the interface configuration and either skips the port-profile commands that have been overridden by directly entered interface commands or returns the command to the default value.

If you want to delete a port profile that has been inherited by other port profiles, you must remove the inheritance before you can delete the port profile.

You can also choose a subset of interfaces from which to remove a port profile from among that group of interfaces that you originally applied the profile. For example, if you configured a port profile and configured ten interfaces to inherit that port profile, you can remove the port profile from just some of the specified ten interfaces. The port profile continues to operate on the remaining interfaces to which it is applied.

If you delete a specific configuration for a specified range of interfaces using the interface configuration mode, that configuration is also deleted from the port profile for that range of interfaces only. For example, if you have a channel group inside a port profile and you are in the interface configuration mode and you delete that port channel, the specified port channel is also deleted from the port profile as well.

Just as in the device, you can enter a configuration for an object in port profiles without that object being applied to interfaces yet. For example, you can configure a virtual routing and forward (VRF) instance without it being applied to the system. If you then delete that VRF and related configurations from the port profile, the system is unaffected.

After you inherit a port profile on an interface or range of interfaces and you delete a specific configuration value, that port-profile configuration is not operative on the specified interfaces.

If you attempt to apply a port profile to the wrong type of interface, the system returns an error.

When you attempt to enable, inherit, or modify a port profile, the system creates a checkpoint. If the port-profile configuration fails, the system rolls back to the prior configuration and returns an error. A port profile is never only partially applied.

Cisco QSFP+ to SFP+ Adapter Module Support

The Cisco QSFP+ to SFP+ Adapter (QSA) module provides 10G support for the 40G uplink ports that are a part of the Cisco Nexus M6PQ and Cisco Nexus M12PQ uplink modules of specific Cisco Nexus 9300 devices.

A group of six consecutive ports in the M6PQ or M12PQ uplink module must be operating at the same speed (40G or 10G) to use the QSA/QSFP modules.

- For Cisco Nexus 9396PX devices, 2/1-6 ports form the first port speed group and the remaining 2/7-12 ports form the second port speed group.

- For Cisco Nexus 93128PX/TX devices, 2/1-6 ports form the first port speed group and the remaining 2/7-8 ports form the second port speed group.

- For Cisco Nexus 937xPX/TX devices, 1/49-54 ports form the only port speed group.

- For Cisco Nexus 93120TX devices, 1/97-102 ports form the only port speed group.

- For Cisco Nexus 9332PQ devices, 1/27-32 ports form the only port speed group.

Use the **speed-group 10000** command to configure the first port of a port speed group for the QSA. This command specifies the administrator speed preference for the port group. (The default port speed is 40G.)

- The **speed-group 10000** command specifies a speed of 10G.

- The **no speed-group 10000** command specifies a speed of 40G.

- If you add, modify, or remove the speed-group command on any interface in the speed group, all interfaces in the speed group will change to the newly configured speed. This may cause link state changes based on the transceiver that is plugged in. This also includes the removal of an existing speed-group command using the default interface command.

- Uplink modules should not be removed from a Cisco Nexus 9300 platform switch that is running Cisco NX-OS Release 7.0(3)I7(5). The ports on uplink modules should be used only for uplinks.

- Beginning with Cisco NX-OS Release 9.2(2), CWDM4 is supported on the 36-port 100-Gigabit Ethernet QSFP28 line cards (N9K-X9636C-R), the 36-port 40-Gigabit Ethernet QSFP+ line cards (N9K-X9636Q-R), the 36-port 100-Gigabit QSFP28 line cards (N9K-X9636C-RX) and the 52-port 100-Gigabit QSFP28 line cards (N9K-X96136YC-R).

After the speed has been configured, the compatible transceiver modules are enabled. The remaining transceiver modules in the port group (incompatible transceiver modules) become error disabled with a reason of "check speed-group config".

Note The Cisco QSFP+ to SFP+ Adapter (QSA) module does not provide 10G support for the 40G line cards for Cisco Nexus 9500 devices.

Beginning with Cisco NX-OS Release 7.0(3)I5(2), you can use a QSFP-to-SFP adapter on Cisco Nexus 9200 and 9300-EX Series switches and Cisco Nexus 3232C and 3264Q Series switches.

You can use a QSFP-to-SFP adapter on Cisco Nexus 9200 and 9300-EX Series switches and Cisco Nexus 3232C and 3264Q Series switches.

Cisco SFP+ Adapter Module Support

Beginning with Cisco NX-OS Release 7.0(3)I4(2), you can use the CVR-2QSFP28-8SFP adapter for 25-Gigabit optics support on 100-Gigabit ports of the Cisco Nexus 9236C switch.

You can use the CVR-2QSFP28-8SFP adapter for 25-Gigabit optics support on 100-Gigabit ports of the Cisco Nexus 9236C switch.

The **interface breakout module** command can be used to split this switch's 100G interfaces into four 25G interfaces. After you enter this command, you must copy the running configuration to the startup configuration.

Beginning with Cisco NX-OS Release 9.2(3), 10/25 LR is supported on N9K-C93180YC-EX, N9K-X97160YC-EX, N9K-C93180YC-FX, N9K-C93240YC-FX2 and N3K-C34180YC switches. This dual speed optical transceiver operates at 25G by default and it seamlessly interoperates with other 25G LR transceivers. Because auto speed sensing is not supported on this device, to interoperate with a 10G transceiver, you must manually configure it to use 10G speed.

Beginning with Cisco NX-OS Release 9.3(5), 10G BASE-T SFP+ (RJ-45) is supported on N9K-C93240YC-FX2, N9K-C93180YC-EX, N9K-C93180YC-FX and N9K-C93360YC-FX2 devices. This copper transceiver operates at 10G by default.

> **Note**
> When you connect a SFP-10G-T-X device into a port, all the neighboring ports of this device must be either empty, or be connected to copper devices only.

Table 2: Default Port Mapping

Device Name	Port Map
Cisco Nexus N9K-C93180YC-EX and N9K-C93180YC-FX	PI/PE: 1,4-5,8-9,12-13, 16,37,40-41,44-45, 48
Cisco Nexus N9K-C93240YC-FX2	W/ PI Fan/PS: 2,6,8,12,14,18,20,24,26,30,32,36,38,42,44,48 W/ PE Fan/PS: 6,12,18,24,30,36,42,48
Cisco Nexus N9K-C93360YC-FX2	PI/PE 1,4-5,8,41,44-45,48-49,52-53,56-57, 60-61,64-65,68-69,72-73,76-77,80-81,84-85,88-89,92-93,96

Cisco SFP-10G-T-X Module Support

Beginning with Cisco NX-OS Release 9.3(5), 10G BASE-T SFP+ (RJ-45) is supported on N9K-C93240YC-FX2, N9K-C93180YC-EX, N9K-C93180YC-FX and N9K-C93360YC-FX2 devices. This copper transceiver operates at 10G by default.

> **Note**
> When you connect a SFP-10G-T-X device into a port, all the neighboring ports of this device must be either empty, or be connected to passive copper links only.

> **Note**
> Interface configured with media-type 10G-TX while in admin up state will remain errdisabled under Unsupported media-type. To remove this condition, use the following commands on the interface:
>
> • **shutdown**
>
> • **no shutdown**

Table 3: Default Port Mapping

Device Name	Port Map
Cisco Nexus N9K-C93180YC-EX, N9K-C93180YC-FX, N9K-C93180YC-FX3 and N9K-C93180YC-FX3S	PI/PE: 1, 4-5, 8-9, 12-13, 16, 37, 40-41, 44-45, 48

Device Name	Port Map
Cisco Nexus N9K-C93240YC-FX2	W/ PI Fan/PS: 2, 6, 8, 12, 14, 18, 20, 24, 26, 30, 32, 36, 38, 42, 44, 48 W/ PE Fan/PS: 6, 12, 18, 24, 30, 36, 42, 48
Cisco Nexus N9K-C93360YC-FX2	PI/PE 1, 4-5, 8, 41, 44-45, 48-49, 52-53, 56-57, 60-61, 64-65, 68-69, 72-73, 76-77, 80-81, 84-85, 88-89, 92-93, 96

Guidelines and Limitations

Basic interface parameters have the following configuration guidelines and limitations:

- MDIX is enabled by default on copper ports. It is not possible to disable it.

- **show** commands with the **internal** keyword are not supported.

- Fiber-optic Ethernet ports must use Cisco-supported transceivers. To verify that the ports are using Cisco-supported transceivers, use the **show interface transceivers** command. Interfaces with Cisco-supported transceivers are listed as functional interfaces.

- A port can be either a Layer 2 or a Layer 3 interface; it cannot be both simultaneously.

  By default, each port is a Layer 3 interface.

  You can change a Layer 3 interface into a Layer 2 interface by using the **switchport** command. You can change a Layer 2 interface into a Layer 3 interface by using the **no switchport** command.

- Flow control using pause frames is not supported.

- You usually configure Ethernet port speed and duplex mode parameters to auto to allow the system to negotiate the speed and duplex mode between ports. If you decide to configure the port speed and duplex modes manually for these ports, consider the following:

  - Before you configure the speed and duplex mode for an Ethernet or management interface, see the Default Settings section for the combinations of speeds and duplex modes that can be configured at the same time.

  - If you set the Ethernet port speed to auto, the device automatically sets the duplex mode to auto.

  - If you enter the **no speed** command, the device automatically sets both the speed and duplex parameters to auto (the **no speed** command produces the same results as the **speed auto** command).

  - If you configure an Ethernet port speed to a value other than auto (for example, 1G, 10G, or 40G), you must configure the connecting port to match. Do not configure the connecting port to negotiate the speed.

  - To configure speed, duplex, and automatic flow control for an Ethernet interface, you can use the **negotiate auto** command. To disable automatic negotiation, use the **no negotiate auto** command.

  - Beginning with Cisco NX-OS Release 9.3(6), Cisco Nexus N9K-C92348GC-X switches support 10M full-duplex and 100M half-duplex modes mode on ports 1 through 48.

✎

Note   The device cannot automatically negotiate the Ethernet port speed and duplex mode if the connecting port is configured to a value other than auto.

⚠

Caution   Changing the Ethernet port speed and duplex mode configuration might shut down and reenable the interface.

- When using a QSFP-40G-CR4 cable to connect between N9K-C9332PQ non-ALE ports and N9K-C9372PX ALE ports, you need to manually set the speed to 40000.

- For BASE-T copper ports, auto-negotiation is enabled even when fixed speed is configured.

- The port profile feature is not supported.

- Using a regular expression to address a set of interfaces is supported with the **regex** command option. The **regex** command option is an extension that is available for all interface commands.

Example:

```
switch(config-if-range)# interface ethernet regex [2]/
switch(config-if-range)# where
  conf; interface Ethernet2/1-8      admin@switch%default
switch(config-if-range)# interface ethernet regex [1]/2[2-4]
switch(config-if-range)# where
  conf; interface Ethernet1/22-24      admin@switch%default
```

- The source-interface command option provides support for management applications to configure an IPv4 and/or IPv6 inband or outband source IP address for the copy command and other processes (such as tacacs, ntp, ping/ping6, icmp-error and traceroute).

  - Configuration commands

    ip *services* **source-interface** *interface* **vrf** *vrf name*

    Examples:

      - ip ftp source-interface ethernet 8/1 vrf management

      - ip http source-interface loopback 1 vrf blue

      - ip ssh source-interface ethernet ethernet 5/1

        /*This command executes in the VRF context.*/

      - ip ping source-interface ethernet 8/1 vrf blue

      - ip traceroute source-interface ethernet 8/1 vrf red

      - ip icmp-errors source-interface ethernet 8/1

        /*This command executes in the VRF context.*/

  - Show commands:

    show ip copy *services* **source-interface** *interface* **vrf** *vrf name*

      - show ip ftp source-interface ethernet 8/1 vrf management

- `show ip http source-interface loopback 1 vrf blue`

- `show ip ssh source-interface ethernet ethernet 5/1`

  `/*This command executes in the VRF context.*/`

- `show ip ping source-interface ethernet 8/1 vrf blue`

- `show ip traceroute source-interface ethernet 8/1 vrf red`

- `show ip icmp-errors source-interface ethernet 8/1`

  `/*This command executes in the VRF context.*/`

- Service commands:

  copy *service*://username@hostname/*path* **file source-interface** *interface name*

  Examples:

  - `copy ftp://username@hostname/usr/local/bin file source-interface ethernet 8/1`

  - `copy scp://username@hostname/usr/local/bin file source-interface ethernet 8/1`

  - `copy tftp://username@hostname/usr/local/bin file source-interface ethernet 8/1`

  - `copy http://username@hostname/usr/local/bin file source-interface ethernet 8/1`

  - `copy sftp://username@hostname/usr/local/bin file source-interface ethernet 8/1`

- Port profiles are supported on Cisco Nexus 9300 Series switches and Cisco Nexus 9500 Series switches.

- Auto-negotiation is not supported on 25-Gigabit Ethernet Transceiver Modules on Cisco Nexus 9200 and 9300-EX platform switches; and Cisco Nexus 9500 platform switches that uses N9K-X9700-EX line cards.

- Beginning with Cisco NX-OS Release 9.2(1), autonegotiation on native 25G ports is supported on Cisco Nexus N9K-X97160YC-EX, N9K-C93180YC-FX, N9K-C93240YC-FX2 and N9K-C93240YC-FX2-Z switches.

- Autonegotiation is not supported on Cisco Nexus N9K-C92300YC switch.

- Autonegotiation is not supported on 25G breakout ports.

- Beginning with Cisco NX-OS Release 10.4(2)F, you must configure same FEC on both 50Gx2 breakout ports for the links to be active. FEC type is not supported for auto negotiation on the ports. Make sure that you configure same configuration on both ports if the default configurations are different on the ports.

- Autonegotiation is not supported when N9K-C93108TC-FX3P switch is connected to either of the following switches:

  - N9K-C9236C, N9K-C92300YC, N9K-C93180YC-EX, N9K-C93180YC-EXU, N9K-C9232C, N9K-C92300YC, and N9K-C93180YC-FX.

  - N3K-C3172TQ-XL, N3K-C3172TQ-10GT, N3K-C3172PQ-10GE, and N3K-C3132Q-40GE.

- Beginning with Cisco NX-OS Release 9.2(2), Autonegotiation (40 G/100 G) is supported on the following ports:

  - Cisco Nexus 9336C-FX2 switch: Ports 1-6 and 33-36

  - Cisco Nexus 9364C switch: Ports 49-64

  - Cisco Nexus 93240YC-FX2 switch: Ports 51-54

  - Cisco Nexus 9788TC line card: Ports 49-52

- Beginning with Cisco NX-OS Release 9.2(2), 10 GB with QSA is supported on the following ports:

  - Cisco Nexus 9336C-FX2 switch: Ports 1-36

  - Cisco Nexus 9364C switch: Ports 49-64

  - Cisco Nexus 9788TC line card: Ports 49-52

- Beginning with Cisco NX-OS Release 9.2(2), 1 GB with QSA is supported on the following ports:

  - Cisco Nexus 9336C-FX2 switch: Ports 7-32

  - Cisco Nexus 9364C switch: Ports 65 and 66 only

- Beginning with Cisco NX-OS Release 10.1(1), 1 GB with QSA is supported on the following ports:

  - Cisco Nexus 9336C-FX2-E switch: Ports 9-36.

- From Cisco NX-OS Release 10.4(1)F, below limitations are on Cisco Nexus C9348GC-FX3PH switch:

  - On front ports of switch, ports from 41 to 48, control plane may be impacted, in case of congestion or line rate traffic.

  - No drop at 99.98% of line rate traffic.

  - Below mentioned interface counters are supported on front ports 41 to 48:

    Interface Packets - Ingress Packets, Rx Unicast Packets, Rx Multicast Packets, Rx Broadcast packets, Tx Unicast Packets, Egress Packets, Tx Multicast Packets, and Tx Broadcast Packets

    Interface Errors - Ingress Runt Errors, Ingress FCS Error, Input Errors, Symbol Error, Ingress crc, and Output Errors

    Interface Collision - Collision, Single Collision, Multi Collision, and Late Collision

    Interface Bytes - Rx Bytes, and Tx Bytes

    Other supported Interface counters - Tx Dropped, Short Frame, Jumbo Frames, Input Discard, Deferred, and Jabber

- Beginning with Cisco NX-OS Release 10.4(1)F, 10 GB with QSA and 25G with QSA28 is supported on the following:

  - Cisco Nexus C9348GC-FX3PH switch: Ports 53 and 54 only

  - Cisco Nexus C9348GC-FX3 switch: Ports 53 and 54 only

- Beginning with Cisco NX-OS Release 10.4(1)F, 25G with QSA28 is supported on the following:

- Cisco Nexus 93180YC-FX3 switch: Ports 49 to 54

- Cisco Nexus 93108TC-FX3P switch: Ports 49 to 54

- Beginning with Cisco NX-OS Release 10.4(2)F, 10 GB with QSA and 25G with QSA28 is supported on Cisco Nexus Switch C93108TC-FX3 for Ethernet Ports 49 to 54 only (6x40/100G Ethernet Module).

- Beginning with Cisco NX-OS Release 9.3(1), only MTU 9216 can be configured on FEX fabric ports. Trying to configure any other value generates an error. If the MTU value on a FEX fabric port-channel was set to 9216 before the switch was upgraded to Cisco NX-OS Release 9.3(1), the **show running config** command does not display the MTU value, but the **show running-config diff** command does.

- Beginning with Cisco NX-OS Release 9.3(1), FEX fabric port-channels support only MTU 9216 by default.

- The following line cards do not support Link Training:

Nexus 9300 Modules:

- N9K-M12PQ (C9396PX, C9396TX, C93128PX, C93128TX)

Nexus 9500 Modules:

- X9536PQ

- X9564PX

- X9564TX

The N9K-X9436PQ, N9K-X9564PX, N9K-C9396PX, and N9K-X9564TX line cards do not support Link Training.

- If cable length is more than 5 meters, Auto Negotiation is not supported. This cable length limitation is applicable only to copper cables and not applicable to optical cables.

- When you use a backslash (\) at end of a valid interface description, the parser identifies the backslash as a continuation character and appends an extra line break in command output by adding a new line character '\n' to the command string. This is a Day-1 behavior.

- Cisco Nexus 9000 series EX TOR switches and line cards do not support RS-FEC, and CONS16-RS-FEC.

- Beginning with Cisco NX-OS Release 10.2(3)F, the **link-flap error-disable count** command can be configured on all physical ports.

- The **link-flap error-disable count** command is supported on all Cisco Nexus 9000 Series switches.

- Beginning with Cisco NX-OS Release 10.3(1)F, Interface Consistency Checker support is provided on Cisco Nexus 9800 platform switches.

- Beginning with Cisco NX-OS Release 10.3(1)F, native (400G, 100G, 40G) and breakout (4x100G) ports support is provided on N9K-X9836DM-A line card of Cisco Nexus 9800 platform switches.

- Beginning with Cisco NX-OS Release 10.4(1)F, the breakout (4x10G and 4x25G) ports support is provided on N9K-X9836DM-A line card of Cisco Nexus 9808/9804 platform switches.

- Beginning with Cisco NX-OS Release 10.3(1)F, 10G Optics support using CVR-QSFP-SFP10G adapter is provided for N9K-X9836DM-A line card of Cisco Nexus 9800 platform switches.

- Beginning with Cisco NX-OS Release 10.3(1)F, Auto negotiation is not supported for 40G, 100G copper based links for N9K-X9836DM-A line card of Cisco Nexus 9800 platform switches.

- Beginning with Cisco NX-OS Release 10.4(1)F, Auto negotiation for 100G/40G is supported on the Cisco Nexus 9332D-H2R platform switches. However, 400G is not supported.

- Beginning with Cisco NX-OS Release 10.4(2)F, Auto negotiation for 100G/40G ports is supported on the last four ports of Cisco Nexus 93400LD-H1 platform switches.

- Beginning with Cisco NX-OS Release 10.4(3)F, Auto negotiation for 100G/40G ports is supported on Cisco Nexus N9K-C9364C-H1 platform switches.

- Beginning with Cisco NX-OS Release 10.3(1)F, statistics support for Physical Interface is provided on Cisco Nexus 9808 platform switches.

- Beginning with Cisco NX-OS Release 10.4(1)F, statistics support for Physical Interface is provided on Cisco Nexus 9804 platform switches.

- Cisco Nexus 9808/9804 platform switches have the following limitations for Physical Interface statistics support:

  - Port-channel is not supported

  - Broadcast counters/statistics are not supported for interface counters.

  - Locally generated/injected packets will not be classified into unicast, multicast or broadcast. However, these will be accounted under total packets and bytes. For example: cdp packets.

  - For **show interface ethernet 1/1 counters detailed snmp** command, Cisco Nexus 9800 platform supports different frame size range as below:

```
This platform counter Range
===============
TX Frame octet Range
TX legal frames with 1519-2500 bytes.
TX legal frames with 2501-9000 bytes.
Nexus existing platform
================
TX Length=1519-2047
TX Length=2048-4095
TX Length=4096-8191
TX Length=8192-9215
TX Length>=9216
Similar frame size suppport exists for Rx direction also.

show interface ethernet 1/1 counters detailed snmp
Ethernet1/1
Rx Packets: 4004
Rx Unicast Packets: 4000
Rx Jumbo Packets: 4000
Rx Bytes: 7031737
Rx Packets from 65 to 127 bytes: 1
Rx Packets from 128 to 255 bytes: 1
Rx Packets from 512 to 1023 bytes: 1
Rx Packets from 1024 to 1518 bytes: 1
Rx Packets from 1519 to 2500 bytes: 4000 >>>> New range supported
Tx Packets: 17
Tx Bytes: 4948
Tx Packets from 0 to 64 bytes: 2
Tx Packets from 65 to 127 bytes: 3
Tx Packets from 128 to 255 bytes: 10
Tx Packets from 512 to 1023 bytes: 1
```

```
Tx Packets from 1024 to 1518 bytes: 1
Tx Packets from 1519 to 2500 bytes: 2 >>>>> New range
```

- In case of interface error counters, Align-Err, Runts, Giants, Input discards and Output Discards counters are not supported and will be shown as 0.

For example:

```
show interface ethernet 1/1 counters errors

------------------------------------------------------------------------------
Port Align-Err FCS-Err Xmit-Err Rcv-Err UnderSize OutDiscards
------------------------------------------------------------------------------
Eth1/1 0 0 0 0 0 0

------------------------------------------------------------------------------
Port Single-Col Multi-Col Late-Col Exces-Col Carri-Sen Runts
------------------------------------------------------------------------------
Eth1/1 0 0 0 0 0 0

------------------------------------------------------------------------------
Port Giants SQETest-Err Deferred-Tx IntMacTx-Er IntMacRx-Er Symbol-Err
------------------------------------------------------------------------------
Eth1/1 0 -- 0 0 0 0

------------------------------------------------------------------------------
Port InDiscards
------------------------------------------------------------------------------
Eth1/1 0

------------------------------------------------------------------------------
Port Stomped-CRC
------------------------------------------------------------------------------
Eth1/1 0
```

- Beginning with Cisco NX-OS Release 10.3(1)F, UDLD support is provided on Cisco Nexus 9808 platform switches.

- Beginning with Cisco NX-OS Release 10.4(1)F, UDLD support is provided on the following Cisco Nexus switches:

  - Cisco Nexus 9804 Platform switches

  - Cisco Nexus X98900CD-A and X9836DM-A line cards with Cisco Nexus 9808 and 9804 switches.

Default Settings

The following lists the default settings for the basic interface parameters.

Parameter	Default
Description	Blank
Beacon	Disabled
Bandwidth	Data rate of interface
Throughput delay	100 microseconds

Parameter	Default
Administrative status	Shutdown
MTU	1500 bytes
UDLD global	Globally disabled
UDLD per-port enable state for fiber-optic media	Enabled on all Ethernet fiber-optic LAN ports
UDLD per-port enable state for copper media	Disabled on all Ethernet 1G, 10G, or 40G LAN ports
UDLD message interval	Disabled
UDLD aggressive mode	Disabled
Error disable	Disabled
Error disable recovery	Disabled
Error disable recovery interval	300 seconds
Buffer-boost	Enabled **Note** Feature available on N9K-X9564TX and N9K-X9564PX line cards and Cisco Nexus 9300 series devices.

Configuring the Basic Interface Parameters

When you configure an interface, you must specify the interface before you can configure its parameters.

Specifying the Interfaces to Configure

Before you begin

Before you can configure the parameters for one or more interfaces of the same type, you must specify the type and the identities of the interfaces.

The following table shows the interface types and identities that you should use for specifying the Ethernet and management interfaces.

Table 4: Information Needed to Identify an Interface for Configurations

Interface Type	Identity
Ethernet	I/O module slot numbers and port numbers on the module
Management	0 (for port 0)

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

You enter a range of interfaces using dashes (-) and commas (,). Dashes separate contiguous interfaces and commas separate noncontiguous interfaces. When you enter noncontiguous interfaces, you must enter the media type for each interface.

This example shows how to configure a contiguous interface range:

```
switch(config)# interface ethernet 2/29-30
switch(config-if-range)#
```

This example shows how to configure a noncontiguous interface range:

```
switch(config)# interface ethernet 2/29, ethernet 2/33, ethernet 2/35
switch(config-if-range)#
```

You can specify subinterfaces in a range only when the subinterfaces are on the same port, for example, 2/29.1-2. But you cannot specify the subinterfaces in a range of ports, for example, you cannot enter 2/29.2-2/30.2. You can specify two of the subinterfaces discretely, for example, you can enter 2/29.2, 2/30.2.

This example shows how to configure a a breakout cable:

```
switch(config)# interface ethernet 1/2/1
switch(config-if-range)#
```

SUMMARY STEPS

1. **configure terminal**
2. **interface** *interface*

DETAILED STEPS

	Command or Action	Purpose

Step 1	**configure terminal**
	
	Example:
	
	`switch# configure terminal`
	`switch(config)#`
		Enters global configuration mode.

Step 2	**interface** *interface*
	
	Example:
	
	`switch(config)# interface ethernet 2/1`
	`switch(config-if)#`
	
	Example:
	
	`switch(config)# interface mgmt0`
	`switch(config-if)#`
		Specifies the interface that you are configuring. You can specify the interface type and identity. For an Ethernet port, use **ethernet** *slot/port*. For the management interface, use **mgmt0**.
		
		Examples:
		
		• The 1st example shows how to specify the slot 2, port 1 Ethernet interface.
		
		• The 2nd example shows how to specify the management interface.

	Command or Action	Purpose
		Note You do not need to add a space between the interface type and identity (port or slot/port number) For example, for the Ethernet slot 4, port 5 interface, you can specify either "ethernet 4/5" or "ethernet4/5." The management interface is either "mgmt0" or "mgmt 0."
		When you are in the interface configuration mode, the commands that you enter configure the interface that you specified for this mode.

Configuring the Description

You can provide textual interface descriptions for the Ethernet and management interfaces.

SUMMARY STEPS

1. **configure terminal**
2. **interface** *interface*
3. **description** *text*
4. **show interface** *interface*
5. (Optional) **show interface** *mgmt*
6. **exit**
7. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `switch# `**`configure terminal`** `switch(config)#`	Enters global configuration mode.
Step 2	**interface** *interface* **Example:** `switch(config)# `**`interface ethernet 2/1`** `switch(config-if)#` **Example:** `switch(config)# `**`interface mgmt0`** `switch(config-if)#`	Specifies the interface that you are configuring. You can specify the interface type and identity. For an Ethernet port, use **ethernet** *slot/port*. For the management interface, use **mgmt0**. Examples: • The 1st example shows how to specify the slot 2, port 1 Ethernet interface. • The 2nd example shows how to specify the management interface.

	Command or Action	Purpose
Step 3	**description** *text* **Example:** `switch(config-if)#` **`description Ethernet port 3 on module 1`** `switch(config-if)#`	Specifies the description for the interface.
Step 4	**show interface** *interface* **Example:** `switch(config)#` **`show interface ethernet 2/1`**	(Optional) Displays the interface status, which includes the description parameter.
Step 5	(Optional) **show interface** *mgmt* **Example:** `switch(config)#` **`show interface mgmt 0`**	(Optional) Displays the management interface, which includes the description parameter.
Step 6	**exit** **Example:** `switch(config-if)#` **`exit`** `switch(config)#`	Exits the interface mode.
Step 7	**copy running-config startup-config** **Example:** `switch(config)#` `copy running-config startup-config`	(Optional) Copies the running configuration to the startup configuration.

Example

This example shows how to set the interface description to Ethernet port 24 on module 3:

```
switch# configure terminal
switch(config)# interface ethernet 3/24
switch(config-if)# description server1
switch(config-if)#
```

The output of the **show interface eth** command is enhanced as shown in the following example:

```
Switch# show version
Software
BIOS: version 06.26
NXOS: version 6.1(2)I2(1) [build 6.1(2)I2.1]
BIOS compile time: 01/15/2014
NXOS image file is: bootflash:///n9000-dk9.6.1.2.I2.1.bin
NXOS compile time: 2/25/2014 2:00:00 [02/25/2014 10:39:03]

switch# show interface ethernet 6/36
Ethernet6/36 is up
admin state is up, Dedicated Interface
Hardware: 40000 Ethernet, address: 0022.bdf6.bf91 (bia 0022.bdf8.2bf3)
Internet Address is 192.168.100.1/24
MTU 9216 bytes, BW 40000000 Kbit, DLY 10 usec
```

The output of the **show interface mgmt** command is enhanced as shown in the following example:

```
switch# show interface mgmt 0mgmt0 is up
admin state is up,
  Hardware: GigabitEthernet, address: d009.c863.6660 (bia d009.c863.6660)
  Internet Address is 10.10.1.1
  MTU 1500 bytes, BW 1000000 Kbit , DLY 10 usec
  reliability 255/255, txload 1/255, rxload 1/255
  Encapsulation ARPA, medium is broadcast
  full-duplex, 1000 Mb/s
  Auto-Negotiation is turned on
  Auto-mdix is turned off
  EtherType is 0x0000
  1 minute input rate 208920 bits/sec, 146 packets/sec
  1 minute output rate 514648 bits/sec, 144 packets/sec
  Rx
    11890676 input packets 11773213 unicast packets 97704 multicast packets
    19759 broadcast packets 2089190866 bytes
  Tx
    11776034 output packets 11774699 unicast packets 1323 multicast packets
    12 broadcast packets 5228573079 bytes
  Management transceiver: Present
  Active connector: SFP
```

The **active connector** will show SFP when RJ45 connector is removed.

Configuring the Beacon Mode

You can enable the beacon mode for an Ethernet port to flash its LED to confirm its physical location.

SUMMARY STEPS

1. **configure terminal**
2. **interface ethernet** *slot/port*
3. [**no**] **beacon**
4. **show interface ethernet** *slot/port*
5. **exit**
6. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose

Step 1	**configure terminal**
	
	Example:
	`switch# configure terminal`
	`switch(config)#`
		Enters global configuration mode.

Step 2	**interface ethernet** *slot/port*
	
	Example:
	`switch(config)# interface ethernet 3/1`
	`switch(config-if)#`
		Specifies an interface to configure, and enters interface configuration mode.

	Command or Action	Purpose

Step 3	[**no**] **beacon**
	
	Example:
	
	`switch(config)# beacon`
	`switch(config-if)#`
		Enables the beacon mode or disables the beacon mode. The default mode is disabled.

Step 4	**show interface ethernet** *slot/port*
	
	Example:
	
	`switch(config)# show interface ethernet 2/1`
	`switch(config-if)#`
		(Optional) Displays the interface status, which includes the beacon mode state.

Step 5	**exit**
	
	Example:
	
	`switch(config-if)# exit`
	`switch(config)#`
		Exits the interface mode.

Step 6	**copy running-config startup-config** **Example:** `switch(config)# copy running-config startup-config`	(Optional) Copies the running configuration to the startup configuration.

Example

This example shows how to enable the beacon mode for the Ethernet port 3/1:

```
switch# configure terminal
switch(config)# interface ethernet 3/1
switch(config-if)# beacon
switch(config-if)#
```

This example shows how to disable the beacon mode for the Ethernet port 3/1:

```
switch# configure terminal
switch(config)# interface ethernet 3/1
switch(config-if)# no beacon
switch(config-if)#
```

This example shows how to configure the dedicated mode for Ethernet port 4/17 in the group that includes ports 4/17, 4/19, 4/21, and 4/23:

```
switch# configure terminal
switch(config)# interface ethernet 4/17, ethernet 4/19, ethernet 4/21, ethernet 4/23
switch(config-if)# shutdown
switch(config-if)# interface ethernet 4/17
switch(config-if)# no shutdown
switch(config-if)#
```

Configuring the Error-Disabled State

You can view the reason that an interface moves to the error-disabled state and configure automatic recovery.

Enabling the Error-Disable Detection

You can enable error-disable detection in an application. As a result, when a cause is detected on an interface, the interface is placed in an error-disabled state, which is an operational state that is similar to the link-down state.

SUMMARY STEPS

1. **configure terminal**
2. **errdisable detect cause** {**acl-exception** | **all** | **link-flap** | **loopback**}
3. **shutdown**
4. **no shutdown**
5. **link-flap error-disable count** *<number_of_link_flaps>* **interval** *<time_in_seconds>*
6. **show interface status err-disabled**
7. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** switch# **configure terminal** switch(config)#	Enters global configuration mode.
Step 2	**errdisable detect cause** {**acl-exception**	**all**	**link-flap**	**loopback**} **Example:** switch(config)# **errdisable detect cause all** switch(config-if)#	Specifies a condition under which to place the interface in an error-disabled state. The default is enabled.
Step 3	**shutdown** **Example:** switch(config-if)# **shutdown** switch(config)#	Brings the interface down administratively. To manually recover the interface from the error-disabled state, enter this command first.
Step 4	**no shutdown** **Example:** switch(config-if)# **no shutdown** switch(config)#	Brings the interface up administratively and enables the interface to recover manually from the error-disabled state.
Step 5	**link-flap error-disable count** *<number_of_link_flaps>* **interval** *<time_in_seconds>* **Example:** switch(config-if)# link-flap error-disable count **10** interval **30**	Specifies the link-flap error-disable count and interval that you are configuring. • **count** configures the number of flaps that the port can tolerate. The maximum number of flaps that can be tolerated is 30 and minimum is 2. • **interval** configures the time in seconds for the configured flap count to occur after which action is

	Command or Action	Purpose
		taken. The maximum interval is 420 seconds and minimum is 30 seconds.
Step 6	**show interface status err-disabled** **Example:** `switch(config)# `**`show interface status err-disabled`**	(Optional) Displays information about error-disabled interfaces.
Step 7	**copy running-config startup-config** **Example:** `switch(config)# `**`copy running-config startup-config`**	(Optional) Copies the running configuration to the startup configuration.

Example

This example shows how to enable the error-disabled detection in all cases:

```
switch(config)# errdisable detect cause all
switch(config)#
```

Enabling the Error-Disabled Recovery

You can specify the application to bring the interface out of the error-disabled state and retry coming up. It retries after 300 seconds, unless you configure the recovery timer (see the **errdisable recovery interval** command).

SUMMARY STEPS

1. **configure terminal**
2. **errdisable recovery cause** {**all** | **bpduguard** | **failed-port-state** | **link-flap** | **loopback** | **miscabling** | **psecure-violation** | **security-violation** | **storm-control** | **udld** | **vpc-peerlink**}
3. **show interface status err-disabled**
4. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `switch# `**`configure terminal`** `switch(config)#`	Enters global configuration mode.
Step 2	**errdisable recovery cause** {**all**	**bpduguard**	**failed-port-state**	**link-flap**	**loopback**	**miscabling**	**psecure-violation**	**security-violation**	**storm-control**	**udld**	**vpc-peerlink**}	Specifies a condition under which the interface automatically recovers from the error-disabled state, and the device retries bringing the interface up. The device waits 300 seconds to retry. The default is disabled.

	Command or Action	Purpose

	Example:
	`switch(config)# errdisable recovery cause all`
	`switch(config-if)#`	

Step 3	**show interface status err-disabled**
	Example:
	`switch(config)# show interface status err-disabled`
	`switch(config-if)#`
		(Optional) Displays information about error-disabled interfaces.

Step 4	**copy running-config startup-config** **Example:** `switch(config)# copy running-config startup-config`	(Optional) Copies the running configuration to the startup configuration.

Example

This example shows how to enable error-disabled recovery under all conditions:

```
switch(config)# errdisable recovery cause all
switch(config)#
```

Configuring the Error-Disabled Recovery Interval

You can configure the error-disabled recovery timer value.

SUMMARY STEPS

1. **configure terminal**
2. **errdisable recovery interval** *interval*
3. **show interface status err-disabled**
4. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose

Step 1	**configure terminal**
	Example:
	`switch# configure terminal`
	`switch(config)#`
		Enters global configuration mode.

Step 2	**errdisable recovery interval** *interval*
	Example:
	`switch(config)# errdisable recovery interval 32`
	`switch(config-if)#`
		Specifies the interval for the interface to recover from the error-disabled state. The range is from 30 to 65535 seconds, and the default is 300 seconds.

Step 3	**show interface status err-disabled** **Example:**	(Optional) Displays information about error-disabled interfaces.

	Command or Action	Purpose
	switch(config)# **show interface status err-disabled** switch(config-if)#	
Step 4	**copy running-config startup-config** **Example:** switch(config)# **copy running-config startup-config**	(Optional) Copies the running configuration to the startup configuration.

Example

This example shows how to configure the error-disabled recovery timer to set the interval for recovery to 32 seconds:

```
switch(config)# errdisable recovery interval 32
switch(config)#
```

Configuring the MDIX Parameter

To detect the type of connection (crossover or straight) with another copper Ethernet port, enable the medium dependent independent crossover (MDIX) parameter for the local port. By default, this parameter is enabled.

Before you begin

Enable MDIX for the remote port.

SUMMARY STEPS

1. **configure terminal**
2. **interface ethernet** *slot* / *port*
3. {**mdix auto** | **no mdix**}
4. **show interface ethernet** *slot* / *port*
5. **exit**
6. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** switch# **configure terminal** switch(config)#	Enters global configuration mode.
Step 2	**interface ethernet** *slot* / *port* **Example:** switch(config)# **interface ethernet 3/1** switch(config-if)#	Specifies an interface to configure, and enters into interface configuration mode.

	Command or Action	Purpose
Step 3	{**mdix auto** | **no mdix**} **Example:** ``` switch(config)# mdix auto switch(config-if)# ``` ``` switch(config)# no mdix switch(config-if)# ```	Specifies whether to enable or disable MDIX detection for the port. **Note** The **no mdix auto** command is supported only on N9K-C93108TC-EX, N9K-C93108TC-FX, N9K-X9788TC-FX, and N9K-C9348GC-FXP devices.
Step 4	**show interface ethernet** *slot* / *port* **Example:** ``` switch(config)# show interface ethernet 3/1 switch(config-if)# ```	Displays the interface status, which includes the MDIX status.
Step 5	**exit** **Example:** ``` switch(config)# exit ```	Exits the interface mode.
Step 6	**copy running-config startup-config** **Example:** ``` switch(config)# copy running-config startup-config ```	(Optional) Copies the running configuration to the startup configuration.

Example

This example shows how to enable MDIX for Ethernet port 3/1:

```
switch# configure terminal
switch(config)# interface ethernet 3/1
switch(config-if)# mdix auto
switch(config-if)#
```

This example shows how to enable MDIX for Ethernet port 3/1:

```
switch# configure terminal
switch(config)# interface ethernet 3/1
switch(config-if)# no mdix
switch(config-if)#
```

Configuring Media-Type for SFP-10G-T-X

To specify the SFP-10G-T-X device connection on an interface, use the **media-type 10g-tx** command in interface configuration mode. To restore the default value, use the **no** form of this command.

SUMMARY STEPS

1. configure terminal
2. interface *interface-id*
3. media-type 10g-tx

DETAILED STEPS

	Command or Action	**Purpose**
Step 1	configure terminal **Example:** Switch# **configure terminal**	Enters global configuration mode.
Step 2	interface *interface-id* **Example:** Switch (config)# **interface ethernet 1/5**	Specify the media port to be configured, and enter interface configuration mode.

Step 3	media-type 10g-tx
	
	Example:
	
	Switch (Config)# **[no] media-type 10g-tx**
		Configure the SFP-10G-T-X device connection on an interface.
		
		Note Interface configured with media-type 10G-TX while in admin up state will remain errdisabled under Unsupported media-type. To remove this condition, use the following commands on the interface:
		
		        • **shutdown**
		
		        • **no shutdown**

Verifying Media-Type

This example shows how to verify the media-type configuration:

Note   The ports that support SFP-10G-T-X may vary in different devices. This example displays the port numbers that support SFP-10G-T-X on a Cisco Nexus N9K-C93240YC-FX2 switch.

```
switch# sh running-config interface ethernet 1/2

!Command: show running-config interface Ethernet1/2
!Running configuration last done at: Mon Jun  1 10:16:46 2020
!Time: Mon Jun  1 10:16:54 2020

version 9.3(5) Bios:version 05.41

interface Ethernet1/2
  switchport
  switchport access vlan 10
  mtu 9216
  media-type 10g-tx
  no shutdown

Supported ports in Switch 01:
switch# sh interface status | i i SFP-10
Eth1/2        --              connected 10      full   10G     SFP-10G-T-X
Eth1/6        --              connected 11      full   10G     SFP-10G-T-X
Eth1/8        --              connected 11      full   10G     SFP-10G-T-X
Eth1/12       --              connected 12      full   10G     SFP-10G-T-X
```

```
Eth1/14      --              connected 12     full   10G     SFP-10G-T-X
Eth1/18      --              connected 13     full   10G     SFP-10G-T-X
Eth1/20      --              connected 13     full   10G     SFP-10G-T-X
Eth1/24      --              connected 14     full   10G     SFP-10G-T-X
Eth1/26      --              connected 14     full   10G     SFP-10G-T-X
Eth1/30      --              connected 15     full   10G     SFP-10G-T-X
Eth1/32      --              connected 15     full   10G     SFP-10G-T-X
Eth1/36      --              connected 16     full   10G     SFP-10G-T-X
Eth1/38      --              connected 16     full   10G     SFP-10G-T-X
Eth1/42      --              connected 20     full   10G     SFP-10G-T-X
Eth1/44      Connect_to_Sw_01 connected 202   full   10G     SFP-10G-T-X
Eth1/48      Connect_to_Sw_02 connected 202   full   10G     SFP-10G-T-X



switch# sh mod
Mod Ports          Module-Type                     Model            Status
--- ----- ------------------------------------- -------------------- ---------
1   60    48x10/25G + 12x40/100G Ethernet Modul N9K-C93240YC-FX2     active *

Mod Sw                    Hw    Slot
--- ---------------------- ------ ----
1   9.3(4.104)            0.3020 NA


Mod MAC-Address(es)                    Serial-Num
--- ------------------------------------ ----------
1   b4-de-31-94-4e-c8 to b4-de-31-94-4f-0f FDO2143306S

Mod Online Diag Status
--- ------------------
1   Pass
```

Configuring the MTU Size

You can configure the maximum transmission unit (MTU) size for Layer 2 and Layer 3 Ethernet interfaces. For Layer 3 interfaces, you can configure the MTU to be between 576 and 9216 bytes (even values are required). For Layer 2 interfaces, you can configure the MTU to be either the system default MTU (1500 bytes) or the system jumbo MTU size (which has the default size of 9216 bytes).

MTU is configured per interface, where the interface can be a Layer 2 or a Layer 3 interface. Every interface has default MTU of 1500 bytes. This value is called system default MTU. You can configure a Layer 2 interface, with a a value of 9216 bytes, which is the default value of the system jumbo MTU. To allow an MTU value that is between 1500 and 9216, system jumbo MTU needs to be adjusted to appropriate value where interface can be configured with the same value.

✎

Note   You can change the system jumbo MTU size. When the value is changed, the Layer 2 interfaces that use the system jumbo MTU value, will automatically changes to the new system jumbo MTU value.

A Layer 3 interface, can be Layer 3 physical interface switch virtual interface (SVI), and subinterface, you can configure an MTU size between 576–9216 bytes.

By default, Cisco NX-OS configures Layer 3 parameters. If you want to configure Layer 2 parameters, you need to switch the port mode to Layer 2. You can change the port mode by using the **switchport** command.

After changing the port mode to Layer 2, you can return to configuring Layer 3 interfaces by changing the port mode again, by using the **no switchport** command.

Configuring the Interface MTU Size

For Layer 3 interfaces, you can configure an MTU with keyword MTU and value in bytes where value is between 576–9216 bytes. Beginning with Cisco NX-OS Release 9.3(1), you can configure the MTU size up to 9216 bytes on the management interfaces on all Cisco Nexus 9000 switches. The change in the configuration may trigger a temporary link flap at the end device.

For Layer 2 interfaces, you can configure an interface using the keyword MTU with value in bytes. The value can be a system default MTU size of 1500 bytes or the system jumbo MTU value that can be adjusted to the default size of 9216 bytes.

If you need to use a different system jumbo MTU size for Layer 2 interfaces, see the *Configuring the System Jumbo MTU Size* section.

SUMMARY STEPS

1. **configure terminal**
2. **interface ethernet** *slot/port*, vlan *vlan-id* mgmt *0*
3. **mtu** *size*
4. **exit**

DETAILED STEPS

	Command or Action	Purpose

Step 1	**configure terminal**
	
	Example:
	
	`switch# configure terminal`
	`switch(config)#`
		Enters global configuration mode.

Step 2	**interface ethernet** *slot/port*, vlan *vlan-id* mgmt *0*
	
	Example:
	
	`switch(config)# interface ethernet 3/1`
	`switch(config-if)#`
	`switch(config)# interface vlan 100`
	`switch(config-if)#`
	`switch(config)# interface mgmt 0`
	`switch(config-if)#`
		Specifies an Ethernet interface to configure, and enters interface configuration mode.

Step 3	**mtu** *size*
	
	Example:
	
	`switch(config-if)# mtu 9216`
	`switch(config-if)#`
		Configure the MTU value on an interface.
		
		For a Layer 3 interface, a physical Layer 3 interface, an SVI or sub-interface, then the value can be between 576-9216 bytes. If the interface is a physical Layer 2 interface, then the value can be 1500 or system jumbo MTU value.

Step 4	**exit**
	
	Example:
	
	`switch(config-if)# exit`
	`switch(config)#`
		Exits the interface mode.

Example

This example shows how to configure the Layer 2 Ethernet port 3/1 with the default MTU size (1500):

```
switch# configure terminal
switch(config)# interface ethernet 3/1
switch(config-if)# switchport
switch(config-if)# mtu 1500
switch(config-if)#
```

This example displays the output of show running-config interface command:

```
switch# show run int mgmt0
!Command: show running-config interface mgmt0
!Running configuration last done at: Fri May 31 11:32:28 2019
!Time: Fri May 31 11:32:33 2019
version 9.3(1) Bios:version 07.65
interface mgmt0
mtu 9216
vrf member management
ip address 168.51.170.73/82
```

Configuring the System Jumbo MTU Size

You can configure and use the system jumbo MTU for a Layer 2 interfaces MTU value. The system jumbo MTU must be specified as an even number between 1500 and 9216. The default value of system jumbo MTU is 9216 bytes.

Beginning with Cisco NX-OS Release 9.3(1), you can configure the MTU size up to 9216 bytes on the management interfaces on all Cisco Nexus 9000 switches. The change in the configuration might trigger a temporary link flap at the end device.

Beginning with Cisco NX-OS Release 9.3(1), FEX fabric port-channels support only MTU 9216 by default.

Note　Beginning with Cisco NX-OS Release 9.3(1), only MTU 9216 can be configured on FEX fabric ports. Trying to configure any other value generates an error. If the MTU value on a FEX fabric port-channel was set to 9216 before the switch was upgraded to Cisco NX-OS Release 9.3(1), the **show running config** command does not display the MTU value, but the **show running-config diff** command does.

SUMMARY STEPS

1. **configure terminal**
2. **system jumbomtu** *size*
3. **interface** *type slot/port*
4. **mtu** *size*
5. **exit**
6. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `switch# `**`configure terminal`** `switch(config)#`	Enters global configuration mode.
Step 2	**system jumbomtu** *size* **Example:** `switch(config)# `**`system jumbomtu 8000`** `switch(config)#`	Specifies the system jumbo MTU size. Use an even number between 1500 and 9216. **Note** In general accepted practice, a jumbo frame is considered to have an MTU size greater than 9000 bytes.
Step 3	**interface** *type slot/port* **Example:** `switch(config)# `**`interface ethernet 2/1`** `switch(config-if)#`	Specifies an interface to configure and enters interface configuration mode.
Step 4	**mtu** *size* **Example:** `switch(config-if)# `**`mtu 8000`** `switch(config-if)#`	System jumbo MTU is added to a Layer 2 interface.
Step 5	**exit** **Example:** `switch(config-if)# `**`exit`** `switch(config)#`	Exits the interface mode.
Step 6	**copy running-config startup-config** **Example:** `switch(config)# `**`copy running-config startup-config`**	(Optional) Copies the running configuration to the startup configuration.

Example

This example shows how to configure the system jumbo MTU as 8000 bytes and how to change the MTU specification for a Layer 2 interface that was configured with the previous jumbo MTU size:

```
switch# configure terminal
switch(config)# system jumbomtu 8000
switch(config)# interface ethernet 2/2
switch(config-if)# mtu 8000
```

This example displays the output of **show running-config interface** command:

```
switch# show run int mgmt0
!Command: show running-config interface mgmt0
!Running configuration last done at: Fri May 31 11:32:28 2019
!Time: Fri May 31 11:32:33 2019
```

```
version 9.3(1) Bios:version 07.65
interface mgmt0
mtu 9216
vrf member management
ip address 178.41.170.93/22
```

Configuring the Bandwidth

You can configure the bandwidth for Ethernet interfaces. The physical layer uses an unchangeable bandwidth of 1G, 10G, or 40G, but you can configure a value of 1 to 100,000,000 KB for Level 3 protocols.

SUMMARY STEPS

1. **configure terminal**
2. **interface ethernet** *slot/port*
3. **bandwidth** *kbps*
4. **show interface ethernet** *slot/port*
5. **exit**
6. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	**Purpose**
Step 1	**configure terminal** **Example:** `switch# `**`configure terminal`** `switch(config)#`	Enters global configuration mode.
Step 2	**interface ethernet** *slot/port* **Example:** `switch(config)# `**`interface ethernet 3/1`** `switch(config-if)#`	Specifies an Ethernet interface to configure, and enters interface configuration mode.
Step 3	**bandwidth** *kbps* **Example:** `switch(config-if)# `**`bandwidth 1000000`** `switch(config-if)#`	Specifies the bandwidth as an informational-only value between 1 and 100,000,000.
Step 4	**show interface ethernet** *slot/port* **Example:** `switch(config)# `**`show interface ethernet 2/1`**	(Optional) Displays the interface status, which includes the bandwidth value.
Step 5	**exit** **Example:** `switch(config-if)# `**`exit`** `switch(config)#`	Exits the interface mode.

	Command or Action	Purpose
Step 6	**copy running-config startup-config** **Example:** `switch(config)# `**`copy running-config startup-config`**	(Optional) Copies the running configuration to the startup configuration.

Example

This example shows how to configure an informational value of 1,000,000 Kb for the Ethernet slot 3, port 1 interface bandwidth parameter:

```
switch# configure terminal
switch(config)# interface ethernet 3/1
switch(config-if)# bandwidth 1000000
switch(config-if)#
```

Configuring the Throughput Delay

You can configure the interface throughput delay for Ethernet interfaces. The actual delay time does not change, but you can set an informational value between 1 and 16777215, where the value represents the number of tens of microseconds.

SUMMARY STEPS

1. **configure terminal**
2. **interface ethernet** *slot/port*
3. **delay** *value*
4. **show interface ethernet** *slot/port*
5. **exit**
6. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `switch# `**`configure terminal`** `switch(config)#`	Enters global configuration mode.
Step 2	**interface ethernet** *slot/port* **Example:** `switch(config)# `**`interface ethernet 3/1`** `switch(config-if)#`	Specifies an Ethernet interface to configure, and enters interface configuration mode.

	Command or Action	Purpose

Step 3	**delay** *value*
	
	Example:
	
	`switch(config-if)# delay 10000`
	`switch(config-if)#`
		Specifies the delay time in tens of microseconds. You can set an informational value range between 1 and 16777215 tens of microseconds.

Step 4	**show interface ethernet** *slot/port*
	
	Example:
	
	`switch(config)# show interface ethernet 3/1`
	`switch(config-if)#`
		(Optional) Displays the interface status, which includes the throughput-delay time.

Step 5	**exit**
	
	Example:
	
	`switch(config-if)# exit`
	`switch(config)#`
		Exits the interface mode.

Step 6	**copy running-config startup-config** **Example:** `switch(config)# copy running-config startup-config`	(Optional) Copies the running configuration to the startup configuration.

Example

This example shows how to configure the throughput-delay time so that one interface is preferred over another. A lower delay value is preferred over a higher value. In this example, Ethernet 7/48 is preferred over 7/47. The default delay for 7/48 is less than the configured value on 7/47, which is set for the highest value (16777215):

```
switch# configure terminal
switch(config)# interface ethernet 7/47
switch(config-if)# delay 16777215
switch(config-if)# ip address 192.168.10.1/24
switch(config-if)# ip router eigrp 10
switch(config-if)# no shutdown
switch(config-if)# exit
switch(config)# interface ethernet 7/48
switch(config-if)# ip address 192.168.11.1/24
switch(config-if)# ip router eigrp 10
switch(config-if)# no shutdown
switch(config-if)#
```

Note You must first ensure the EIGRP feature is enabled by running the **feature eigrp** command.

Shutting Down and Activating the Interface

You can shut down and restart Ethernet or management interfaces. When you shut down interfaces, they become disabled and all monitoring displays show them as being down. This information is communicated

to other network servers through all dynamic routing protocols. When the interfaces are shut down, the interface is not included in any routing updates. To activate the interface, you must restart the device.

SUMMARY STEPS

1. **configure terminal**
2. **interface** *interface*
3. **shutdown**
4. **show interface** *interface*
5. **no shutdown**
6. **show interface** *interface*
7. **exit**
8. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose

Step 1	**configure terminal**
	
	Example:
	`switch# configure terminal`
	`switch(config)#`
		Enters global configuration mode.

Step 2	**interface** *interface*
	
	Example:
	`switch(config)# interface ethernet 2/1`
	`switch(config-if)#`
	
	`switch(config)# interface mgmt0`
	`switch(config-if)#`
		Specifies the interface that you are configuring. You can specify the interface type and identity. For an Ethernet port, use *ethernet slot/port*. For the management interface, use *mgmt0*.
		
		Examples:
		• The 1st example shows how to specify the slot 2, port 1 Ethernet interface.
		• The 2nd example shows how to specify the management interface.

Step 3	**shutdown**
	
	Example:
	`switch(config-if)# shutdown`
	`switch(config-if)#`
		Disables the interface.

Step 4	**show interface** *interface*
	
	Example:
	`switch(config-if)# show interface ethernet 2/1`
	`switch(config-if)#`
		(Optional) Displays the interface status, which includes the administrative status.

Step 5	**no shutdown**
	
	Example:
	`switch(config-if)# no shutdown`
	`switch(config-if)#`
		Reenables the interface.

	Command or Action	Purpose
Step 6	**show interface** *interface* **Example:** ``` switch(config-if)# show interface ethernet 2/1 switch(config-if)# ```	(Optional) Displays the interface status, which includes the administrative status.
Step 7	**exit** **Example:** ``` switch(config-if)# exit switch(config)# ```	Exits the interface mode.
Step 8	**copy running-config startup-config** **Example:** ``` switch(config)# copy running-config startup-config ```	(Optional) Copies the running configuration to the startup configuration.

Example

This example shows how to change the administrative status for Ethernet port 3/1 from disabled to enabled:

```
switch# configure terminal
switch(config)# interface ethernet 3/1
switch(config-if)# shutdown
switch(config-if)# no shutdown
switch(config-if)#
```

Configuring the UDLD Mode

You can configure normal unidirectional link detection (UDLD) modes for Ethernet interfaces on devices configured to run UDLD.

Before you can enable the aggressive UDLD mode for an interface, you must make sure that UDLD is already enabled globally on the device and on the specified interfaces.

Note If the interface is a copper port, you must use the command enable UDLD to enable the UDLD. If the interface is a fiber port you need not explicitly enable UDLD on the interface. However if you attempt to enable UDLD on a fiber port using the enable UDLD command, you may get an error message indicating that is not a valid command.

The following table lists CLI details to enable and disable UDLD on different interfaces

Table 5: CLI Details to Enable or Disable UDLD on Different Interfaces

Description	Fiber port	Copper or Nonfiber port
Default setting	Enabled	Disabled

Description	Fiber port	Copper or Nonfiber port
Enable UDLD command	**no udld disable**	**udld enable**
Disable UDLD command	**udld disable**	**no udld enable**

Before you begin

You must enable UDLD for the other linked port and its device.

SUMMARY STEPS

1. **configure terminal**
2. [**no**] **feature udld**
3. **udld message-time** *seconds*
4. **udld aggressive**
5. **interface ethernet** *slot/port*
6. **udld** [**enable** | **disable**]
7. **show udld** [**ethernet** *slot/port* | **global** | **neighbors**]
8. **exit**
9. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** switch# **configure terminal** switch(config)#	Enters global configuration mode.
Step 2	[**no**] **feature udld** **Example:** switch(config)# **feature udld** switch(config)# switch(config)# **no feature udld** switch(config)#	Enables/Disables UDLD for the device.
Step 3	**udld message-time** *seconds* **Example:** switch(config)# **udld message-time 30** switch(config)#	(Optional) Specifies the interval between sending UDLD messages. The range is from 7 to 90 seconds, and the default is 15 seconds.
Step 4	**udld aggressive** **Example:** switch(config)# **udld aggressive** switch(config)#	Enables UDLD in aggressive mode by default on all fiber interfaces. Use the **no** form to disable aggressive mode UDLD on all fibers ports by default.

Command or Action	Purpose
	Note For copper interfaces, you enter the interface command mode for those interfaces you want to configure for UDLD aggressive mode and issue this command in interface command model.
	Note Use the udld aggressive command to configure the ports to use a UDLD mode: • To enable fiber interfaces for the aggressive mode, enter the **udld aggressive** command in the global command mode and all the fiber interfaces will be in aggressive UDLD mode. • To enable the copper interfaces for the aggressive mode, you must enter the **udld aggressive** command in the interface mode, specifying each interface you want in aggressive UDLD mode. To use the aggressive UDLD mode, you must configure the interfaces on both ends of the link for the aggressive UDLD mode.
Step 5 **interface ethernet** *slot/port* **Example:** `switch(config)# `**`interface ethernet 3/1`** `switch(config-if)#`	(Optional) Specifies an interface to configure, and enters interface configuration mode.
Step 6 **udld** [**enable** | **disable**] **Example:** `switch(config-if)# `**`udld enable`** `switch(config-if)#`	(Optional) Enables UDLD on the specified copper port or disables UDLD on the specified fiber port. To enable UDLD on copper ports, enter the **udld enable** command. To enable UDLD on fiber ports, enter the **no udld disable** command. Enables UDLD in normal mode by default on all fiber interfaces. Use the **no** form to disable normal mode UDLD on all fibers ports by default.
Step 7 **show udld** [**ethernet** *slot/port* | **global** | **neighbors**] **Example:** `switch(config)# `**`show udld`** `switch(config)#`	(Optional) Displays the UDLD status.
Step 8 **exit** **Example:**	Exits the interface mode.

	Command or Action	Purpose
	```switch(config-if-range)# exit``` ```switch(config)#```	
Step 9	**copy running-config startup-config** **Example:** ```switch(config)# copy running-config startup-config```	(Optional) Copies the running configuration to the startup configuration.

Example

This example shows how to enable the UDLD for the device:

```
switch# configure terminal
switch(config)# feature udld
switch(config)#
```

This example shows how to set the UDLD message interval to 30 seconds:

```
switch# configure terminal
switch(config)# feature udld
switch(config)# udld message-time 30
switch(config)#
```

This example shows how to disable UDLD for Ethernet port 3/1:

```
switch# configure terminal
switch(config)# interface ethernet 3/1
switch(config-if-range)# no udld enable
switch(config-if-range)# exit
```

This example shows how to disable UDLD for the device:

```
switch# configure terminal
switch(config)# no feature udld
switch(config)# exit
```

This example shows how to enable fiber interfaces for the aggressive UDLD mode:

```
switch# configure terminal
switch(config)# udld aggressive
```

This example shows how to enable the aggressive UDLD mode for the copper Ethernet interface3/1:

```
switch# configure terminal
switch(config)# interface ethernet 3
switch(config-if)# udld aggressive
```

This example shows how to check if aggressive mode is enabled.

```
switch# sh udld global

UDLD global configuration mode: enabled-aggressive
UDLD global message interval: 15
switch#
```

This example shows how to check if udld aggressive mode is operational for a given interface.

```
switch# sh udld ethernet 8/2

Interface Ethernet8/2
-------------------------------
Port enable administrative configuration setting: device-default
```

```
Port enable operational state: enabled-aggressive
Current bidirectional state: bidirectional
Current operational state:  advertisement - Single neighbor detected
Message interval: 15
Timeout interval: 5
<>
```

Configuring Debounce Timers

You can enable the debounce timer for Ethernet ports by specifying a debounce time (in milliseconds) or disable the timer by specifying a debounce time of 0.

Note The link state of 10G and 100G ports may change repeatedly when connected to service provider network. As a part of *link reset* or *break-link* functionality, it is expected that the Tx power light on the SFP to change to N/A state, at an event of link state change.

However, to prevent this behavior during the link state change, you may increase the link debounce timer to start from 500ms and increase it in 500ms intervals until the link stabilizes. On the DWDM, UVN, and WAN network, it is recommended to disable automatic link suspension (ALS) whenever possible. ALS suspends the link on the WAN when the Nexus turn off the link.

Note

The **link debounce time** and **link debounce link-up time** commands can only be applied to a physical Ethernet interface.

Use the **show interface debounce** command to display the debounce times for all Ethernet ports.

The **link debounce time** command is not supported on 10G and 40G ports on the Cisco Nexus 93300YC-FX and Cisco Nexus 9336C-FX switches.

The **link debounce time** command is supported on 1G, 10G, 40G, 25G and 100G SFP/QSFP ports on the Cisco Nexus 9000 series switches.

The **link debounce time** is supported on 1G, 10G, 25G, 40G and 100G ports on Cisco Nexus N9K-C9732C-FX, N9K-C9364C, N9K-X97160YC-EX, N9K-C9336C-FX2, and N9K-C93240YC-FX2 platform switches.

The **link debounce time** command is supported on 1G, 10G, 40G, 25G and 100G SFP/QSFP ports on the Cisco Nexus 9000 series switches.

The **link debounce time** is supported on 1G, 10G, 25G, 40G and 100G ports on Cisco Nexus N9K-C9732C-FX, N9K-C9364C, N9K-X97160YC-EX, N9K-C9336C-FX2, and N9K-C93240YC-FX2 platform switches.

The **link debounce time** is not supported on RJ-45 ports on Cisco Nexus 9500 platform switches with N9K-X97160TC-FX line cards.

Beginning with Cisco NX-OS Release 10.2(3)F, the **link debounce time** command is supported on N9K-C93180YC-FX3S, N9K-C93180YC-FX3, N9K-C93108TC-FX3P and N9K-X9716D-GX platform switches.

Beginning with Cisco NX-OS Release 10.2(3)F, the **link debounce time** command is supported on this following ports and platform switches:

Ports	Switches
1G	Cisco Nexus N9K-C9364C, N9K-C93300YC-FX2, N9K-C93240YC-FX2, N9K-C93240YC-FX2-Z, N9K-X97160YC-EX, N9K-C9316D-GX, N9K-C93600CD-GX, N9K-C9364C-GX, N9K-C9232C, N9K-C93180YC-EX, N9K-C93180YC-EXU, N9K-C93180YC-EX-24, N9K-C93180YC-FX3S, N9K-C93180YC-FX3, N9K-C93108TC-FX3P, and N9K-X9716D-GX
10G	Cisco Nexus N9K-C9364C, N9K-C93300YC-FX2, N9K-C93240YC-FX2, N9K-C93240YC-FX2-Z, N9K-X97160YC-EX, N9K-C9316D-GX, N9K-C93600CD-GX, N9K-C9364C-GX, N9K-C9232C, N9K-C93180YC-EX, N9K-C93180YC-EXU, N9K-C93180YC-EX-24, N9K-C93180YC-FX3S, N9K-C93180YC-FX3, N9K-C93108TC-FX3P, and N9K-X9716D-GX
25G	Cisco Nexus N9K-C93300YC-FX2, N9K-C93240YC-FX2, N9K-C93240YC-FX2-Z, N9K-X97160YC-EX, N9K-C9316D-GX, N9K-C93600CD-GX, N9K-C9364C-GX, N9K-C9232C, N9K-C93180YC-EX, N9K-C93180YC-EXU, N9K-C93180YC-EX-24, N9K-C93180YC-FX3S, N9K-C93180YC-FX3, N9K-C93108TC-FX3P, and N9K-X9716D-GX
40G	Cisco Nexus N9K-C9364C, N9K-X9732C-FX, N9K-C9336C-FX2, N9K-C93300YC-FX2, N9K-C93240YC-FX2, N9K-C93240YC-FX2-Z, N9K-X97160YC-EX, N9K-C9316D-GX, N9K-C93600CD-GX, N9K-C9364C-GX, N9K-C9232C, N9K-C93180YC-EX, N9K-C93180YC-EXU, N9K-C93180YC-EX-24, N9K-C93180YC-FX3S, N9K-C93180YC-FX3, N9K-C93108TC-FX3P, and N9K-X9716D-GX
100G	
400G	

SUMMARY STEPS

1. **configure terminal**
2. **interface ethernet** *slot/port*
3. **link debounce time** *time*
4. **link debounce link-up** *time*

DETAILED STEPS

	Command or Action	Purpose

Step 1	**configure terminal**
	
	Example:
	
	```switch# configure terminal```
	 ```switch(config)#```
		Enters global configuration mode.

Step 2	**interface ethernet** *slot/port*
	
	Example:
	
	```switch(config)# interface ethernet 3/1```
	 ```switch(config-if)#```
		Specifies an Ethernet interface to configure, and enters interface configuration mode.

Step 3	**link debounce time** *time*
	
	Example:
	
	```switch(config-if)# link debounce time 1000```
	 ```switch(config-if)#```
		Enables the debounce timer for the specified time (1 to 5000 milliseconds).
		
		If you specify 0 milliseconds, the debounce timer is disabled.

Step 4	**link debounce link-up** *time*
	
	Example:
	
	```switch(config-if)# link debounce link-up 1000```
	 ```switch(config-if)#```
		Enables the link-up timer for the specified time (1000 to 10000 milliseconds). This command applies only if the port speeds are 10G, 25G, 40G and 100G.
		
		The default value of the timer is 0. If the value is set to 0 the interface will be up without any delay.
		
		Note The **no link debounce link-up** command also resets the value to 0.
		
		Note This command is supported only on Cisco Nexus N9K-X9732C-FX , N9K-C93300YC-FX, N9K-C9336C-FX2, N9K-C9364C and N9K-X97160YC-EX switches.

Example

- The following example enables the debounce timer and sets the debounce time to 1000 milliseconds for an Ethernet interface:

```
switch# configure terminal
switch(config)# interface ethernet 1/4
switch(config-if)# link debounce time 1000
```

• The following example disables the debounce timer for an Ethernet interface:

```
switch# configure terminal
switch(config)# interface ethernet 1/4
switch(config-if)# link debounce time 0
```

• The following example sets the debounce link-up timer to 1000 milliseconds for an Ethernet interface:

```
switch# configure terminal
switch(config)# interface ethernet 1/4
switch(config-if)# link debounce link-up time 1000
```

Configuring Port Profiles

Beginning with 7.0(3)I4(1), you can apply several configuration parameters to a range of interfaces simultaneously. All the interfaces in the range must be the same type. You can also inherit the configurations from one port profile into another port profile. The system supports four levels of inheritance.

You can apply several configuration parameters to a range of interfaces simultaneously. All the interfaces in the range must be the same type. You can also inherit the configurations from one port profile into another port profile. The system supports four levels of inheritance.

Creating a Port Profile

You can create a port profile on the device. Each port profile must have a unique name across types and the network.

Note Port profile names can include only the following characters:

  • a-z

  • A-Z

  • 0-9

  • The special characters - and _ are allowed. (No other special characters are allowed.)

  • No special characters are allowed, except for the following:

    • .

    • -

    • _

SUMMARY STEPS

  1. **configure terminal**
  2. **port-profile** [**type** {**ethernet** | **interface-vlan** | **loopback** | **port channel** | **tunnel**}] *name*
  3. **port-profile** [**type** {**ethernet** | **interface-vlan** | **port-channel**}] *name*

    4. **exit**

    5. (Optional) **show port-profile**

    6. (Optional) **copy running-config startup-config**

DETAILED STEPS

	Command or Action	**Purpose**
Step 1	**configure terminal**	Enters the global configuration mode.
Step 2	**port-profile** [**type** {**ethernet** | **interface-vlan** | **loopback** | **port channel** | **tunnel**}] *name*	Creates and names a port profile for the specified type of interface and enters the port-profile configuration mode.
Step 3	**port-profile** [**type** {**ethernet** | **interface-vlan** | **port-channel**}] *name*	Creates and names a port profile for the specified type of interface and enters the port-profile configuration mode.
Step 4	**exit**	Exits the port-profile configuration mode.
Step 5	(Optional) **show port-profile**	Displays the port-profile configuration.
Step 6	(Optional) **copy running-config startup-config**	Copies the running configuration to the startup configuration.

Example

This example shows how to create a port profile named test for ethernet interfaces:

```
switch# configure terminal
switch(config)# port-profile type ethernet test
switch(config-ppm)#
```

Entering Port-Profile Configuration Mode and Modifying a Port Profile

You can enter the port-profile configuration mode and modify a port profile. To modify the port profile, you must be in the port-profile configuration mode.

SUMMARY STEPS

    1. **configure terminal**

    2. **port-profile** [**type** {**ethernet** \| **interface-vlan** \| **loopback** \| **port channel** \| **tunnel**}] *name*

    3. **port-profile** [**type** {**ethernet** \| **interface-vlan** \| **port-channel**}] *name*

    4. **exit**

    5. (Optional) **show port-profile**

    6. (Optional) **copy running-config startup-config**

DETAILED STEPS

	Command or Action	**Purpose**
Step 1	**configure terminal**	Enters the global configuration mode.

	Command or Action	**Purpose**
Step 2	**port-profile** [**type** {**ethernet**	**interface-vlan**	**loopback**	**port channel**	**tunnel**}] *name*	Enters the port-profile configuration mode for the specified port profile and allows you to add or remove configurations to the profile.
Step 3	**port-profile** [**type** {**ethernet**	**interface-vlan**	**port-channel**}] *name*	Enters the port-profile configuration mode for the specified port profile and allows you to add or remove configurations to the profile.
Step 4	**exit**	Exits the port-profile configuration mode.
Step 5	(Optional) **show port-profile**	Displays the port-profile configuration.
Step 6	(Optional) **copy running-config startup-config**	Copies the running configuration to the startup configuration.

Example

This example shows how to enter the port-profile configuration mode for the specified port profile and bring all the interfaces administratively up:

```
switch# configure terminal
switch(config)# port-profile type ethernet test
switch(config-ppm)# no shutdown
switch(config-ppm)#
```

Assigning a Port Profile to a Range of Interfaces

You can assign a port profile to an interface or to a range of interfaces. All the interfaces must be the same type.

SUMMARY STEPS

1. **configure terminal**
2. **interface** [**ethernet** *slot/port* | **interface-vlan** *vlan-id* | **loopback** *number* | **port-channel** *number* | **tunnel** *number*]
3. **interface** [**ethernet** *slot/port* | **interface-vlan** *vlan-id* | **port-channel** *number*]
4. **inherit port-profile** *name*
5. **exit**
6. (Optional) **show port-profile**
7. (Optional) **copy running-config startup-config**

DETAILED STEPS

	Command or Action	**Purpose**
Step 1	**configure terminal**	Enters the global configuration mode.
Step 2	**interface** [**ethernet** *slot/port*	**interface-vlan** *vlan-id*	**loopback** *number*	**port-channel** *number*	**tunnel** *number*]	Selects the range of interfaces.

	Command or Action	**Purpose**
Step 3	**interface** [**ethernet** *slot/port* | **interface-vlan** *vlan-id* | **port-channel** *number*]	Selects the range of interfaces.
Step 4	**inherit port-profile** *name*	Assigns the specified port profile to the selected interfaces.
Step 5	**exit**	Exits the port-profile configuration mode.
Step 6	(Optional) **show port-profile**	Displays the port-profile configuration.
Step 7	(Optional) **copy running-config startup-config**	Copies the running configuration to the startup configuration.

Example

This example shows how to assign the port profile named adam to Ethernet interfaces 7/3 to 7/5, 10/2, and 11/20 to 11/25:

```
switch# configure terminal
switch(config)# interface ethernet7/3-5, ethernet10/2, ethernet11/20-25
switch(config-if)# inherit port-profile adam
switch(config-if)#
```

Enabling a Specific Port Profile

To apply the port-profile configurations to the interfaces, you must enable the specific port profile. You can configure and inherit a port profile onto a range of interfaces before you enable that port profile. You would then enable that port profile for the configurations to take effect on the specified interfaces.

If you inherit one or more port profiles onto an original port profile, only the last inherited port profile must be enabled; the system assumes that the underlying port profiles are enabled.

You must be in the port-profile configuration mode to enable or disable port profiles.

SUMMARY STEPS

1. **configure terminal**
2. **port-profile** [**type** {**ethernet** \| **interface-vlan** \| **loopback** \| **port channel** \| **tunnel**}] *name*
3. **port-profile** [**type** {**ethernet** \| **interface-vlan** \| **port-channel**}] *name*
4. **state enabled**
5. **exit**
6. (Optional) **show port-profile**
7. (Optional) **copy running-config startup-config**

DETAILED STEPS

	Command or Action	**Purpose**
Step 1	**configure terminal**	Enters the global configuration mode.

	Command or Action	**Purpose**
Step 2	**port-profile** [**type** {**ethernet** | **interface-vlan** | **loopback** | **port channel** | **tunnel**}] *name*	Creates and names a port profile for the specified type of interface and enters the port-profile configuration mode.
Step 3	**port-profile** [**type** {**ethernet** | **interface-vlan** | **port-channel**}] *name*	Creates and names a port profile for the specified type of interface and enters the port-profile configuration mode.
Step 4	**state enabled**	Enables that port profile.
Step 5	**exit**	Exits the port-profile configuration mode.
Step 6	(Optional) **show port-profile**	Displays the port-profile configuration.
Step 7	(Optional) **copy running-config startup-config**	Copies the running configuration to the startup configuration.

Example

This example shows how to enter the port-profile configuration mode and enable the port profile:

```
switch# configure terminal
switch(config)# port-profile type ethernet test
switch(config-ppm)# state enabled
switch(config-ppm)#
```

Inheriting a Port Profile

You can inherit a port profile onto an existing port profile. The system supports four levels of inheritance.

SUMMARY STEPS

1. **configure terminal**
2. **port-profile** *name*
3. **inherit port-profile** *name*
4. **exit**
5. (Optional) **show port-profile**
6. (Optional) **copy running-config startup-config**

DETAILED STEPS

	Command or Action	**Purpose**
Step 1	**configure terminal**	Enters the global configuration mode.
Step 2	**port-profile** *name*	Enters the port-profile configuration mode for the specified port profile.
Step 3	**inherit port-profile** *name*	Inherits another port profile onto the existing one. The original port profile assumes all the configurations of the inherited port profile.

	Command or Action	**Purpose**
Step 4	**exit**	Exits the port-profile configuration mode.
Step 5	(Optional) **show port-profile**	Displays the port-profile configuration.
Step 6	(Optional) **copy running-config startup-config**	Copies the running configuration to the startup configuration.

Example

This example shows how to inherit the port profile named adam onto the port profile named test:

```
switch# configure terminal
switch(config)# port-profile test
switch(config-ppm)# inherit port-profile adam
switch(config-ppm)#
```

Removing a Port Profile from a Range of Interfaces

You can remove a port profile from some or all of the interfaces to which you have applied the profile. You do this configuration in the interfaces configuration mode.

SUMMARY STEPS

1. **configure terminal**
2. **interface** [**ethernet** *slot/port* | **interface-vlan** *vlan-id* | **loopback** *number* | **port-channel** *number* | **tunnel** *number*]
3. **interface** [**ethernet** *slot/port* | **interface-vlan** *vlan-id* | **port-channel** *number*]
4. **no inherit port-profile** *name*
5. **exit**
6. (Optional) **show port-profile**
7. (Optional) **copy running-config startup-config**

DETAILED STEPS

	Command or Action	**Purpose**
Step 1	**configure terminal**	Enters the global configuration mode.
Step 2	**interface** [**ethernet** *slot/port*	**interface-vlan** *vlan-id*	**loopback** *number*	**port-channel** *number*	**tunnel** *number*]	Selects the range of interfaces.
Step 3	**interface** [**ethernet** *slot/port*	**interface-vlan** *vlan-id*	**port-channel** *number*]	Selects the range of interfaces.
Step 4	**no inherit port-profile** *name*	Un-assigns the specified port profile to the selected interfaces.
Step 5	**exit**	Exits the port-profile configuration mode.
Step 6	(Optional) **show port-profile**	Displays the port-profile configuration.

	Command or Action	Purpose
Step 7	(Optional) **copy running-config startup-config**	Copies the running configuration to the startup configuration.

Example

This example shows how to unassign the port profile named adam to Ethernet interfaces 7/3 to 7/5, 10/2, and 11/20 to 11/25:

```
switch# configure terminal
switch(config)# interface ethernet 7/3-5, 10/2, 11/20-25
switch(config-if)# inherit port-profile adam
switch(config-if)#


switch# configure terminal
switch(config)# interface ethernet 7/3-5, 10/2, 11/20-25
switch(config-if)# no inherit port-profile adam
switch(config-if)#
```

Removing an Inherited Port Profile

You can remove an inherited port profile. You do this configuration in the port-profile mode.

SUMMARY STEPS

1. **configure terminal**
2. **port-profile** *name*
3. **no inherit port-profile** *name*
4. **exit**
5. (Optional) **show port-profile**
6. (Optional) **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal**	Enters the global configuration mode.
Step 2	**port-profile** *name*	Enters the port-profile configuration mode for the specified port profile.
Step 3	**no inherit port-profile** *name*	Removes an inherited port profile from this port profile.
Step 4	**exit**	Exits the port-profile configuration mode.
Step 5	(Optional) **show port-profile**	Displays the port-profile configuration.
Step 6	(Optional) **copy running-config startup-config**	Copies the running configuration to the startup configuration.

Example

This example shows how to remove the inherited port profile named adam from the port profile named test:

```
switch# configure terminal
switch(config)# port-profile test
switch(config-ppm)# no inherit port-profile adam
switch(config-ppm)#
```

Configuring link mac-up timer

This procedure describes how to configure mac up timers on DWDM/Dark fiber circuits.

SUMMARY STEPS

1. **configure terminal**
2. **interface** *type slot/port*
3. **link mac-up timer** *seconds*

DETAILED STEPS

	Command or Action	Purpose

Step 1	**configure terminal**
	
	Example:
	```switch# configure terminal```
	```switch(config)#```
		Enters global configuration mode.

Step 2	**interface** *type slot/port*
	
	Example:
	```switch(config)# interface ethernet1/2```
	```switch(config-if)#```
		Configures an interface and enters interface configuration mode.

Step 3	**link mac-up timer** *seconds* **Example:** ```switch(config-if)# link mac-up timer 10```	Enables modification of the link mac-up timer. The link mac-up timer range is 0-120. **Note** This should only be done on DWDM links.

Configuring 25G Autonegotiation

Autonegotiation allows devices to advertise enhanced modes of operation it possesses via the link segment and to detect corresponding enhanced operational modes that the other devices may be advertising. Autonegotiation provides the means to exchange information between two devices that share a link segment and to automatically configure both devices to take maximum advantage of their abilities.

Guidelines and Limitations for 25G Autonegotiation

- Beginning with Cisco NX-OS Release 9.2(1), autonegotiation on native 25G ports with copper cables is supported on Cisco Nexus N9K-X97160YC-EX, N9K-C93180YC-FX, N9K-C93240YC-FX2 and N9K-C93240YC-FX2-Z switches.

- Autonegotiation is not supported on Cisco Nexus N9K-C92300YC switch.

- Autonegotiation is not supported on 25G breakout ports.

FEC selection with 25G Autonegotiation

Table 6: FEC Selection with 25G Autonegotiation

Hardware	FEC based on CR Lengths			
	1m	2m	3m	5m
N9K-C93240YC-FX2	No FEC	No FEC	FC-FEC	RS-IEEE
N9K-C93180YC-FX	No FEC	No FEC	FC-FEC	RS-IEEE
N9K-C93180YC-EX	No FEC	No FEC	FC-FEC	FC-FEC
N9K-X97160YC-EX	No FEC	No FEC	FC-FEC	FC-FEC

Note 25G autonegotiation is not supported on Cisco Nexus N9K-C92300YC switch.

Enabling Autonegotiation

You can enable autonegotiation using the *negotiate auto* command. To enable autonegotiation, follow these steps:

SUMMARY STEPS

1. **configure terminal**
2. **interface ethernet** *port number*
3. **negotiate auto** *port speed*

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `switch# `**`configure terminal`** `switch(config)#`	Enters global configuration mode.

	Command or Action	Purpose

Step 2	**interface ethernet** *port number*
	
	 Example:
	
	 ```switch# int e1/7```
	 ```switch(config-if)#```
		Selects the interface and enters interface mode.

Step 3	**negotiate auto** *port speed*
	
	 Example:
	
	 ```switch(config-if)# negotiate auto 25000```
	 ```switch(config-if)#```
		Enables autonegotiation on the selected interface.
		
		 Note You must apply this command on interfaces at both sides of the 25G native link.

This example shows how to enable autonegotiation on a specified interface:

Example

```
switch# sh int e1/7 st
--------------------------------------------------------------------------------
Port            Name               Status     Vlan      Duplex  Speed   Type
--------------------------------------------------------------------------------
Eth1/7          --                 connected routed     full    25G     SFP-H25GB-CU1M
switch# conf
switch(config)# int e1/7
switch(config-if)# negotiate auto 25000
```

Disabling Autonegotiation

You can disable autonegotiation using the *no negotiate auto* command. To disable autonegotiation, follow these steps:

SUMMARY STEPS

1. **configure terminal**
2. **interface ethernet** *port number*
3. **no negotiate auto** *port speed*

DETAILED STEPS

	Command or Action	Purpose

Step 1	**configure terminal**
	
	 Example:
	
	 ```switch# configure terminal```
	 ```switch(config)#```
		Enters global configuration mode.

Step 2	**interface ethernet** *port number*
	
	 Example:
	
	 ```switch# int e1/7```
	 ```switch(config-if)#```
		Selects the interface and enters interface mode.

	Command or Action	Purpose

Step 3	**no negotiate auto** *port speed*
	
	Example:
	
	`switch(config-if)# no negotiate auto 25000`
	`switch(config-if)#`
		Disables autonegotiation on the selected interface.
		
		Note You must apply this command on interfaces at both sides of the link.

This example shows how to disable autonegotiation on a specified interface.

Example

```
switch# sh int e1/7 st
--------------------------------------------------------------------------------
Port          Name               Status    Vlan     Duplex  Speed   Type
--------------------------------------------------------------------------------
Eth1/7        --                 connected routed    full    25G     SFP-H25GB-CU1M
switch# conf
switch(config)# int e1/7
switch(config-if)# no negotiate auto 25000
```

Verifying the Basic Interface Parameters

You can verify the basic interface parameters by displaying their values. You can also clear the counters listed when you display the parameter values.

To display basic interface configuration information, perform one of the following tasks:

Command	Purpose
show cdp all	Displays the CDP status.
show interface *interface*	Displays the configured states of one or all interfaces.
show interface brief	Displays a table of interface states.
show interface status err-disabled	Displays information about error-disabled interfaces.
show udld *interface*	Displays the UDLD status for the current interface or all interfaces.
show udld global	Displays the UDLD status for the current device.
show port-profile	Displays information about the port profiles.
show system internal pktmgr internal control	Displays the control message for pktmgr.

Command	Purpose
show cdp all	Displays the CDP status.
show interface *interface*	Displays the configured states of one or all interfaces.

Command	Purpose
show interface brief	Displays a table of interface states.
show interface status err-disabled	Displays information about error-disabled interfaces.
show udld *interface*	Displays the UDLD status for the current interface or all interfaces.
show udld global	Displays the UDLD status for the current device.
show system internal pktmgr internal control	Displays the control message for pktmgr.

Monitoring the Interface Counters

You can display and clear interface counters using Cisco NX-OS.

Displaying Interface Statistics

You can set up to three sampling intervals for statistics collections on interfaces.

SUMMARY STEPS

1. **configure terminal**
2. **interface ether** *slot/port*
3. **load-interval counters** [**1** | **2** | **3**] *seconds*
4. **show interface** *interface*
5. **exit**
6. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose

Step 1	**configure terminal**
	
	Example:
	`switch# configure terminal`
	`switch(config)#`
		Enters global configuration mode.

Step 2	**interface ether** *slot/port*
	
	Example:
	`switch(config)# interface ether 4/1`
	`switch(config)#`
		Specifies interface.

Step 3	**load-interval counters** [**1**	**2**	**3**] *seconds*
			
			Example:
			`switch(config)# load-interval counters 1 100`
			`switch(config)#`
				Sets up to three sampling intervals to collect bit-rate and packet-rate statistics. The default values for each counter is as follows:
				
				1—30 seconds (60 seconds for VLAN)

	Command or Action	Purpose
		2—300 seconds
		3—not configured

Step 4	**show interface** *interface*
	
	Example:
	`switch(config)# show interface ethernet 2/2`
	`switch#`
		(Optional) Displays the interface status, which includes the counters.

Step 5	**exit**
	
	Example:
	`switch(config-if-range)# exit`
	`switch(config)#`
		Exits the interface mode.

Step 6	**copy running-config startup-config** **Example:** `switch(config)# copy running-config startup-config`	(Optional) Copies the running configuration to the startup configuration.

Example

This example shows how to set the three sample intervals for the Ethernet port 3/1:

```
switch# configure terminal
switch(config)# interface ethernet 3/1
switch(config-if)# load-interval counter 1 60
switch(config-if)# load-interval counter 2 135
switch(config-if)# load-interval counter 3 225
switch(config-if)#
```

Clearing Interface Counters

You can clear the Ethernet and management interface counters by using the **clear counters interface** command. You can perform this task from the configuration mode or interface configuration mode.

SUMMARY STEPS

1. **clear counters interface** [**all** | **ethernet** *slot/port* | **loopback** *number* | **mgmt** *number* | **port channel** *channel-number*]
2. **show interface** *interface*
3. **show interface** [**ethernet** *slot/port* | **port channel** *channel-number*] **counters**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**clear counters interface** [**all**	**ethernet** *slot/port*	**loopback** *number*	**mgmt** *number*	**port channel** *channel-number*]	Clears the interface counters.

	Command or Action	Purpose

	Example:
	```switch# clear counters ethernet 2/1```
	```switch#```	

Step 2	**show interface** *interface*
	Example:
	```switch# show interface ethernet 2/1```
	```switch#```
		(Optional) Displays the interface status.

Step 3	**show interface** [**ethernet** *slot/port* | **port channel** *channel-number*] **counters**
	Example:
	```switch# show interface ethernet 2/1 counters```
	```switch#```
		(Optional) Displays the interface counters.

Example

This example shows how to clear the counters on Ethernet port 5/5:

```
switch# clear counters interface ethernet 5/5
switch#
```

Configuring DWDM

You can configure DWDM to operate at one of the 96 possible wavelengths available.

SUMMARY STEPS

1. **configure terminal**
2. **interface** *interface*
3. **itu channel** *1-96*
4. **exit**
5. **show run interface**
6. **show** *itu channel all*

DETAILED STEPS

	Command or Action	Purpose

Step 1	**configure terminal**
	Example:
	```switch# configure terminal```
	```switch(config)#```
		Enters global configuration mode.

Step 2	**interface** *interface* **Example:**	Enter interface configuration mode.

	Command or Action	Purpose
	```config)# interface <type slot/port>``` ```switch(config-if)#```	

Step 3	**itu channel** *1-96*
	
	Example:
	```config)# interface <type slot/port>```
	```switch(config-if)# itu channel ?```
		Specify the itu channel value, and enter the configuration.

Step 4	**exit**
	
	Example:
	```switch(config-if)# exit```
	```switch(config)#```
		Exits the interface mode.

Step 5	**show run interface**
	
	Example:
	```switch(config)# show run interface <type slot/port>```
	```switch(config)#```
		Displays the value of itu channel.

Step 6	**show** *itu channel all* **Example:** ```switch(config)# show itu channel [<>	all]``` ```switch# sh itu channel ?```	Displays the mapping of all itu channel, wavelength and frequency.

Configuration Example for QSA

For a Cisco Nexus 9396PX:

- Using the default configuration on port 2/1, all the QSFPs in port group 2/1-6 are brought up with a speed of 40G. If there are any QSA modules in port group 2/1-6, they are error disabled.

- Using the **speed-group** [**10000** | **40000**] command to configure port 2/7, all the QSAs in port group 2/7-12 are brought up with a speed of 10G or 40G. If there are any QSFP modules in port group 2/7-12, they are error disabled.

The QSA configuration Example should be made into its own section under the Configuring the Basic Interface Parameters section and should resemble the following: conf t interface ethernet x/x --- This interface is only the first port in the speed group speed-group [10000|40000

This example shows how to configure QSA for the first port in the speed group for a Cisco Nexus 9396PX:

```
switch# conf t
switch(config)# interface ethernet 2/7
switch(config-if)# speed-group 10000
```