Troubleshoot Port Flaps on Catalyst 9000 Switches

Contents

Issue

Environment

Cause

Resolution

Prerequisites

- Requirements
- Components Used

Background Information

Troubleshooting Workflow

- Network Module Installation Reference
- Verify Cable and Both Sides of the Connection
- Verify SFP and SFP+ Compatibility
- Identify Port Flaps
- Interface show commands
- Verify Cable Status with Time Domain Reflector (TDR)
- TDR Guidelines
- Digital Optic Monitoring (DOM)
  - How to Enable DOM
  - Digital Optic Monitoring Syslog Messages
- Cisco Optics and Forward Error Correction (FEC)
- Debug Commands

Related Cisco Bugs

Related Information

Issue

This document describes how to troubleshoot one or more switch interfaces that repeatedly change state between up and down (port flaps or link flaps), which causes intermittent connectivity.

Environment

Cisco Catalyst 9000 Series switches with copper or fiber uplinks/downlinks. The issue can occur on access ports, uplinks, and (when present) removable network-module ports. The physical path can include patch panels, endpoint devices, and optical transceivers.

- Media applicability: TDR applies to copper interfaces only; DOM applies to supported optical transceivers only.

- Interface types: Standalone fixed ports and module ports are in scope; use the same workflow for either.
- Optics support: Use only optics supported for the platform and software release.

Cause

Common causes include bad or unsupported cabling, unsupported or faulty Small Form-Factor Pluggable (SFP) or Small Form-Factor Pluggable Plus (SFP+) transceivers, duplex or negotiation mismatches, endpoint power-save behavior, and other physical-layer faults.

Resolution

Follow the troubleshooting workflow in this article to confirm the flap in logs, validate physical connectivity and optics, review interface counters, and run copper-only TDR or fiber-only DOM checks as applicable.

Prerequisites

Requirements

There are no specific requirements for this document.

Components Used

The information in this document is based on Cisco Catalyst 9000 Series switches.

The information in this document was created from the devices in a specific lab environment. All of the devices used in this document started with a cleared (default) configuration. If your network is live, ensure that you understand the potential impact of any command.

Background Information

A port flap, also referred to as a link flap, occurs when a physical interface on the switch repeatedly goes up and down. Common causes include bad, unsupported, or nonstandard cabling, unsupported Small Form-Factor Pluggable (SFP) transceivers, and other link-synchronization issues. The condition can be intermittent or permanent.

Since link flaps tend to be a physical issue, this document describes how to diagnose the problem, collect useful logs, and troubleshoot port flaps on Cisco Catalyst 9000 Series switches.

Troubleshooting Workflow

Use this workflow to isolate the most common causes of port flaps. Start at Step 1 and proceed in order. Run optional steps only when the link type and symptoms match.

1. Confirm the **flap in logs**. Use **show logging** to confirm repeated up/down events and capture timestamps.
2. Check **physical connectivity** and **optics**. Verify cabling, patch panels, and supported transceivers; reseat or swap with known-good components.
3. Review **interface counters** and **hardware statistics**. Use **show interfaces** and controller counters to identify Layer 1 errors (CRC, symbol errors, FCS errors).
4. Run **TDR** (copper links only). Use TDR to detect OPEN/SHORT conditions and approximate fault distance.
5. Check **DOM** (fiber links only). Review optical power/temperature/voltage against datasheet thresholds.
6. Verify **FEC** (supported optics only). Confirm **Forward Error Correction settings** and **compatibility** when applicable.
7. Use **debug commands** only if the issue persists. Collect **targeted debug output** during a reproduction window and disable **it** immediately after capture.

Each section includes a short note that indicates when to use it.

Use this section when physical access is available. Verify that network modules, cables, and SFP or SFP+ transceivers are properly installed before reviewing counters or running additional diagnostics.

Network Module Installation Reference

Use this reference when a removable network module is present. The table describes best practices to install a network module in a Cisco Catalyst 9000 Series switch:

Platform	**URL**
Catalyst 9200 Series Switches	Catalyst 9200 Series Switches Hardware Installation Guide
Catalyst 9300 Series Switches	Catalyst 9300 Series Switches Hardware Installation Guide
Catalyst 9400 Series Switches	Catalyst 9400 Series Switches Hardware Installation Guide
Catalyst 9500 Series Switches	Catalyst 9500 Series Switches Hardware Installation Guide

Catalyst 9600 Series Switches	Catalyst 9600 Series Switches Hardware Installation Guide

Verify Cable and Both Sides of the Connection

Use this section after the flap is confirmed in the logs. These tables describe common cable-related causes of link flaps and the recommended recovery actions.

Cause	**Recovery Action**
Bad Cable	Swap the suspect cable with a known-good cable. Look for broken or missing pins on connectors.
Loose connection	Reseat the cable. Remove the connector and reinsert it to ensure it is fully seated.
Patch Panels	Eliminate faulty patch-panel connections. Bypass the patch panel, if possible, to rule it out.
Bad or wrong SFP (fiber specific)	Swap the suspect SFP with a known-good SFP. Verify hardware and software support for this type of SFP.
Bad Port or Module Port	Move the cable to a known-good port to troubleshoot a suspect port or module.
Bad or old endpoint device	Swap the phone, speaker, or other endpoint with a known-good device or a newer device.
Device Sleep Mode	This can be an expected flap. Check the timestamp of the port flap. Compare it to other simultaneous events. Then determine whether a sleep setting is the cause.

Verify SFP and SFP+ Compatibility

Use this section when the flapping link uses a fiber transceiver. The Cisco portfolio of hot-pluggable interfaces offers a broad set of choices for speeds, protocols, reaches, and supported transmission media.

Use any combination of SFP or SFP+ transceiver modules that the Cisco Catalyst 9000 Series switch supports. Each port must match the wavelength specifications on the other end of the cable. The cable must not exceed the supported cable length for reliable communication.

Use only Cisco SFP transceiver modules on the Cisco device. Each SFP or SFP+ transceiver module supports the Cisco Quality Identification (ID) feature. This feature allows a Cisco switch or router to identify and validate that the transceiver module is certified and tested by Cisco.

Tip: Use the Cisco Optics-to-Device Compatibility Matrix to verify optic support.

Identify Port Flaps

Prerequisites: Privileged EXEC access (for example, enable) and access to the device system log buffer or remote syslog.

Expected outcome: Repeated up/down messages for the same interface within a short time window confirm a flap condition.

Use this section first to confirm that a flap is occurring. Run the show logging | include changed command to identify a link flap event. This example shows partial switch system log messages for a link flap event on interface TenGigabitEthernet1/0/40:

<#root>

```
Switch#
```

show logging | include changed

```
August 17 21:06:08.431 UTC: %LINEPROTO-5-UPDOWN: Line protocol on Interface TenGigabitEthernet1/0/40, c
August 17 21:06:39.058 UTC: %LINK-3-UPDOWN: Interface TenGigabitEthernet1/0/40, changed state to down
August 17 21:06:41.968 UTC: %LINK-3-UPDOWN: Interface TenGigabitEthernet1/0/40, changed state to up
August 17 21:06:42.969 UTC: %LINEPROTO-5-UPDOWN: Line protocol on Interface TenGigabitEthernet1/0/40, c
August 17 21:07:20.041 UTC: %LINEPROTO-5-UPDOWN: Line protocol on Interface TenGigabitEthernet1/0/40, c
August 17 21:07:21.041 UTC: %LINK-3-UPDOWN: Interface TenGigabitEthernet1/0/40, changed state to down
August 17 21:07:36.534 UTC: %LINEPROTO-5-UPDOWN: Line protocol on Interface TenGigabitEthernet1/0/40, c
August 17 21:08:06.598 UTC: %LINK-3-UPDOWN: Interface TenGigabitEthernet1/0/40, changed state to up
August 17 21:08:07.628 UTC: %LINEPROTO-5-UPDOWN: Line protocol on Interface TenGigabitEthernet1/0/40, c
August 17 21:08:08.628 UTC: %LINK-3-UPDOWN: Interface TenGigabitEthernet1/0/40, changed state to down
August 17 21:08:10.943 UTC: %LINK-3-UPDOWN: Interface TenGigabitEthernet1/0/40, changed state to up
August 17 21:08:11.944 UTC: %LINEPROTO-5-UPDOWN: Line protocol on Interface TenGigabitEthernet1/0/40, c
```

Tip: When reviewing system message logs, use the port-flap **timestamp** to correlate events.

- Identify the timestamp of each up/down transition.
- Compare the timestamp to other events on the same interface (for example, endpoint power-save behavior).
- Determine whether the flap is expected or indicates a physical-layer issue.

Interface show commands

When to use: After repeated up/down events are confirmed and the physical path is validated.

Command: **show interfaces** and **show controllers ethernet-controller interface** for the affected interface.

Expected result: Error counters (CRC, symbol errors, FCS) remain stable or increase slowly under normal conditions.

Next action if abnormal: Swap cable/optic, move to a known-good port, and re-check counters; proceed to TDR (copper) or DOM (fiber) as applicable.

Use this section after the flap is confirmed and the physical path is checked. The **show interfaces** command provides information that helps identify a possible Layer 1 issue that causes a link flap event:

<#root>

```
Switch#

show interfaces tenGigabitEthernet 1/0/40


TenGigabitEthernet1/0/40 is up, line protocol is up (connected)
Hardware is Ten Gigabit Ethernet, address is 00a5.bf9c.29a8 (bia 00a5.bf9c.29a8)
  MTU 1500 bytes, BW 10000000 Kbit/sec, DLY 10 usec,
     reliability 255/255, txload 1/255, rxload 1/255
  Encapsulation ARPA, loopback not set
  Keepalive not set
  Full-duplex, 10Gb/s, link type is auto, media type is

SFP-10GBase-SR   <-- SFP plugged into the port


  input flow-control is on, output flow-control is unsupported
  ARP type: ARPA, ARP Timeout 04:00:00
  Last input 00:00:03, output 00:00:00, output hang never
  Last clearing of "show interface" counters never
  Input queue: 0/2000/0/0 (size/max/drops/flushes); Total output drops: 0
  Queueing strategy: fifo
  Output queue: 0/40 (size/max)
  5 minute input rate 0 bits/sec, 0 packets/sec
  5 minute output rate 0 bits/sec, 0 packets/sec
     670 packets input, 78317 bytes, 0 no buffer
     Received 540 broadcasts (540 multicasts)
     0 runts, 0 giants, 0 throttles
     0 input errors, 0 CRC, 0 frame, 0 overrun, 0 ignored
     0 watchdog, 540 multicast, 0 pause input
     0 input packets with dribble condition detected
     1766 packets output, 146082 bytes, 0 underruns
     0 Output 0 broadcasts (0 multicasts)
     0 output errors, 0 collisions, 0 interface resets
     0 unknown protocol drops




     0 babbles, 0 late collision, 0 deferred
     0 lost carrier, 0 no carrier, 0 pause output
```

```
     0 output buffer failures, 0 output buffers swapped out
```

The table lists some of the counters from the **show interfaces** command:

Counter	**Issues and Common Causes that Increase Error Counters**
CRC	A high number of cyclic redundancy check (CRC) errors is usually the result of collisions. It can also indicate a physical issue such as cabling, an SFP, a bad interface, or a network interface card (NIC). It can also indicate a duplex mismatch.
Input Errors	This includes runts, giants, no buffer, CRC, frame, overrun, and ignored counts. Other input-related errors can also increase the input errors count.
Output Errors	Output errors can increase when the output queue is undersized or when the interface is oversubscribed.
Total output drops	Output drops are generally the result of interface oversubscription caused by many-to-one traffic patterns or a 10 Gbps to 1 Gbps transfer. Interface buffers are a limited resource and can absorb a burst only up to a point. After that point, packets start to drop. Buffers can be tuned to provide some cushion, but they cannot guarantee zero output drops.
Unknown protocol drops	Unknown protocol drops occur when the receiving interface is not configured for that protocol or when the switch does not recognize the protocol. For example, if two switches are connected and Cisco Discovery Protocol (CDP) is disabled on one switch interface, unknown protocol drops occur on that interface. The CDP packets are no longer recognized, and the switch drops them.

The **history** command allows an interface to maintain utilization history in a graphical format similar to CPU history. History can be maintained as either bits per second (bps) or packets per second (pps), as shown in this example:

<#root>

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

```
history ?
```

```
 bps Maintain history in bits/second
 pps Maintain history in packets/second
```

Along with the rate, various interface counters can be monitored:

<#root>

```
Switch(config-if)#

history [bps|pps] ?

  all                                Include all counters
  babbles                            Include ethernet output babbles - Babbl
  crcs                               Include CRCs - CRCs
  deferred                           Include ethernet output deferred - Defer
  dribbles                           Include dribbles - Dribl
  excessive-collisions               Include ethernet excessive output collisions -
                                     ExCol
  flushes                            Include flushes - Flush
  frame-errors                       Include frame errors - FrErr
  giants                             Include giants - Giant
  ignored                            Include ignored - Ignor
  input-broadcasts                   Include input broadcasts - iBcst
  input-drops                        Include input drops - iDrop
  input-errors                       Include input errors - iErr
  interface-resets                   Include interface resets - IRset
  late-collisions                    Include ethernet late output collisions - LtCol
  lost-carrier                       Include ethernet output lost carrier - LstCr
  multi-collisions                   Include ethernet multiple output collisions -
                                     MlCol
  multicast                          Include ethernet input multicast - MlCst
  no-carrier                         Include ethernet output no-carrier - NoCarr
  output-broadcasts                  Include output broadcasts - oBcst
  output-buffer-failures             Include output buffer failures - oBufF
  output-buffers-swapped-out         Include output buffers swapped out - oBSwO
  output-drops                       Include output drops - oDrop
  output-errors                      Include output errors - oErr
  output-no-buffer                   Include output no buffer - oNoBf
  overruns                           Include overruns - OvrRn
  pause-input                        Include ethernet input pause - PsIn
  pause-output                       Include ethernet output pause - PsOut
  runts                              Include runts - Runts
  single-collisions                  Include ethernet single output collisions - SnCol
  throttles                          Include throttles - Thrtl
  underruns                          Include underruns - UndRn
  unknown-protocol-drops             Include unknown protocol drops - Unkno
  watchdog                           Include ethernet output watchdog - Wtchdg
  <cr>                               <cr>
SW_1(config-if)#
```

As with CPU history, graphs are available for the last 60 seconds, the last 60 minutes, and the last 72 hours. Separate graphs are maintained for input and output histograms:

<#root>

```
Switch#
```

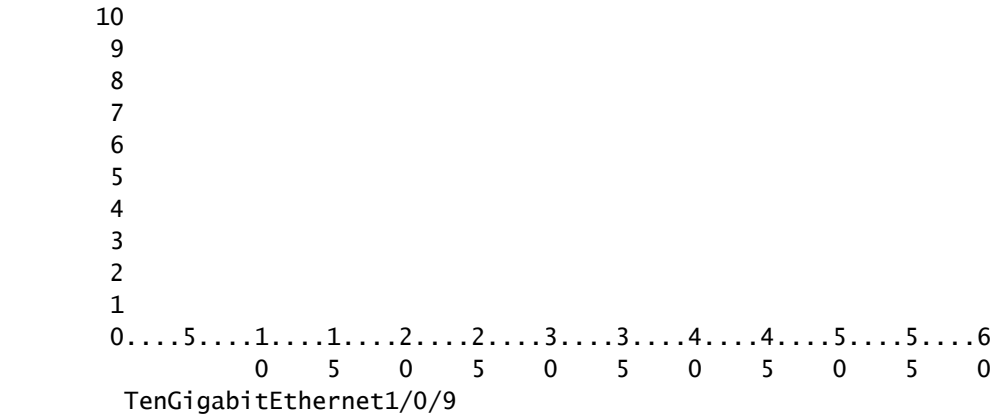
show interfaces gigabitEthernet 1/0/2 history ?

```
 60min    Display 60 minute histograms only
 60sec    Display 60 second histograms only
 72hour   Display 72 hour histograms only
 all      Display all three histogram intervals
 both     Display both input and output histograms
 input    Display input histograms only
 output   Display output histograms only
 |        Output modifiers
 <cr> <cr>
```

```
------ Sample output ---------
```

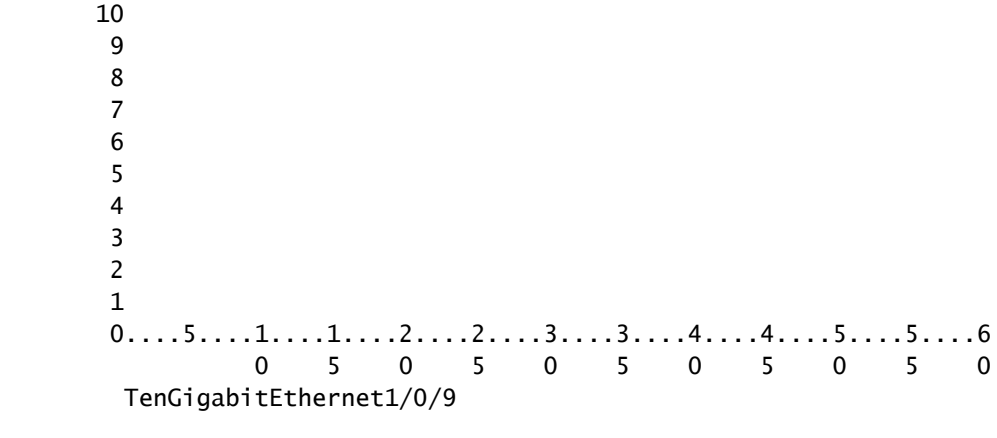
```
Switch#
```

show interfaces tenGigabitEthernet 1/0/9 history 60sec



input

```
 rate(mbits/sec) (last 60 seconds)
```



output

```
 rate(mbits/sec) (last 60 seconds)
```

Use the **show controllers ethernet-controller interface {interface{*interface-number*}}** command to display per-interface (Transmit and Receive) traffic counters and error statistics read from the hardware. Use the **phy** keyword to display the interface internal registers. Use the **port-info** keyword to display information about the port application-specific integrated circuit (ASIC).

This is an example of output from the **show controllers ethernet-controller** for a specific interface:

<#root>

Switch#

show controllers ethernet-controller tenGigabitEthernet 2/0/1

Transmit

TenGigabitEthernet2/0/1

Receive

61572 Total bytes	282909 Total bytes
0 Unicast frames	600 Unicast frames
0 Unicast bytes	38400 Unicast bytes
308 Multicast frames	3163 Multicast frames
61572 Multicast bytes	244509 Multicast bytes
0 Broadcast frames	0 Broadcast frames
0 Broadcast bytes	0 Broadcast bytes
0 System FCS error frames	0 IpgViolation frames
0 MacUnderrun frames	0 MacOverrun frames
0 Pause frames	0 Pause frames
0 Cos 0 Pause frames	0 Cos 0 Pause frames
0 Cos 1 Pause frames	0 Cos 1 Pause frames
0 Cos 2 Pause frames	0 Cos 2 Pause frames
0 Cos 3 Pause frames	0 Cos 3 Pause frames
0 Cos 4 Pause frames	0 Cos 4 Pause frames
0 Cos 5 Pause frames	0 Cos 5 Pause frames
0 Cos 6 Pause frames	0 Cos 6 Pause frames
0 Cos 7 Pause frames	0 Cos 7 Pause frames
0 Oam frames	0 OamProcessed frames
0 Oam frames	0 OamDropped frames
193 Minimum size frames	3646 Minimum size frames
0 65 to 127 byte frames	1 65 to 127 byte frames
0 128 to 255 byte frames	0 128 to 255 byte frames
115 256 to 511 byte frames	116 256 to 511 byte frames
0 512 to 1023 byte frames	0 512 to 1023 byte frames
0 1024 to 1518 byte frames	0 1024 to 1518 byte frames
0 1519 to 2047 byte frames	0 1519 to 2047 byte frames
0 2048 to 4095 byte frames	0 2048 to 4095 byte frames
0 4096 to 8191 byte frames	0 4096 to 8191 byte frames
0 8192 to 16383 byte frames	0 8192 to 16383 byte frames
0 16384 to 32767 byte frame	0 16384 to 32767 byte frame
0 > 32768 byte frames	0 > 32768 byte frames
0 Late collision frames	

```
0 SymbolErr frames



<-- Usually indicates Layer 1 issues. Large amounts of symbol errors can indicate a bad device, cable, c

            0 Excess Defer frames

0 Collision fragments



<-- If this counter increments, this is an indication that the ports are configured at half-duplex.

            0 Good (1 coll) frames                  0 ValidUnderSize frames
            0 Good (>1 coll) frames                 0 InvalidOverSize frames
            0 Deferred frames                       0 ValidOverSize frames
            0 Gold frames dropped

0 FcsErr frames



<--



Are the result of collisions at half-duplex, a duplex mismatch, bad hardware (NIC, cable, or port)

            0 Gold frames truncated
            0 Gold frames successful
            0 1 collision frames
            0 2 collision frames
            0 3 collision frames
            0 4 collision frames
            0 5 collision frames
            0 6 collision frames
            0 7 collision frames
            0 8 collision frames
            0 9 collision frames
            0 10 collision frames
            0 11 collision frames
            0 12 collision frames
            0 13 collision frames
            0 14 collision frames
            0 15 collision frames
            0 Excess collision frames

LAST UPDATE 22622 msecs AGO
```

Tip: The **show interfaces {interface{*interface-number*}} controller** command can also display per-interface Transmit and Receive statistics read from the hardware.

Use the **show platform pm interface-flaps {interface{*interface-number*}}** command to display the number of times an interface goes down:

This is an example of output from the **show platform pm interface-flaps {interface{*interface-number*}}** for a specific interface:

<#root>

Switch#

show platform pm interface-flaps tenGigabitEthernet 2/0/1

```
  Field                      AdminFields          OperFields
================================================================
  Access Mode                Static               Static
  Access Vlan Id             1                    0
  Voice Vlan Id              4096                 0
  VLAN Unassigned                                 0
  ExAccess Vlan Id           32767
  Native Vlan Id             1
  Port Mode                  dynamic              access
  Encapsulation              802.1Q               Native
  disl                       auto
  Media                      unknown
  DTP Nonegotiate            0                    0
  Port Protected             0                    0
  Unknown Unicast Blocked    0                    0
  Unknown Multicast Blocked  0                    0
  Vepa Enabled               0                    0
  App interface              0                    0
  Span Destination           0

  Duplex                     auto                 full
  Default Duplex             auto
  Speed                      auto                 1000
  Auto Speed Capable         1                    1
  No Negotiate               0                    0
  No Negotiate Capable       1024                 1024
  Flow Control Receive       ON                   ON
  Flow Control Send          Off                  Off
  Jumbo                      0                    0
  saved_holdqueue_out        0
  saved_input_defqcount      2000
  Jumbo Size                 1500

  Forwarding Vlans : none
  Current Pruned Vlans : none
  Previous Pruned Vlans : none

  Sw LinkNeg State : LinkStateUp
```

No.of LinkDownEvents :   12

```
<--  Number of times the interface flapped

  XgxsResetOnLinkDown(10GE):

Time Stamp Last Link Flapped(U) : Aug 19 14:58:00.154   <-- Last time the interface flapped



 LastLinkDownDuration(sec) 192



<-- Time in seconds the interface stayed down during the last flap event



LastLinkUpDuration(sec):  2277                          <-- Time in seconds the interface stayed up befo
```

Use this section for fiber links when optical health must be verified. Use the **show idprom interface {interface-number}** command to display the IDPROM information for the transceiver installed in the specified interface. Use the **detail** keyword to display detailed hexadecimal IDPROM fields.

This example shows output from the **show idprom {interface{*interface-number*}}** command for a specific interface. The **High** and **Low Warning|Alarm thresholds** values listed in this command output are the normal operational optical transceiver parameters. These values can be verified in the data sheet for the specific optic. Refer to the Cisco Optics Transceiver Modules Documentation.

<#root>

```
Switch#

show idprom interface Twe1/0/1



IDPROM for transceiver TwentyFiveGigE1/0/1 :
  Description                               = SFP or SFP+ optics (type 3)
  Transceiver Type:                         = GE CWDM 1550 (107)
  Product Identifier (PID)                  =

CWDM-SFP-1550  <--

  Vendor Revision                           = A
  Serial Number (SN)                        =

SERIALNUMBER

  Vendor Name                               = CISCO-FINISAR
  Vendor OUI (IEEE company ID)              = 00.90.65 (36965)
  Common Language Equipment Identifier (CLEI) code = CNTRV14FAB
```

```
  Cisco part number                          = 10-1879-03
  Device State                               = Enabled.
  Date code (yy/mm/dd)                       = 14/12/22
  Connector type                             = LC.
  Encoding                                   = 8B10B (1)
  Nominal bitrate                            = OTU-1 (2700 Mbits/s)
  Minimum bit rate as % of nominal bit rate  = not specified
  Maximum bit rate as % of nominal bit rate  = not specified
  The transceiver type is 107
  Link reach for 9u fiber (km)               = LR-2(80km) (80)
                                               LR-3(80km) (80)
                                               ZX(80km) (80)
  Link reach for 9u fiber (m)                = IR-2(40km) (255)
                                               LR-1(40km) (255)
                                               LR-2(80km) (255)
                                               LR-3(80km) (255)
                                               DX(40KM) (255)
                                               HX(40km) (255)
                                               ZX(80km) (255)
                                               VX(100km) (255)
  Link reach for 50u fiber (m)               = SR(2km) (0)
                                               IR-1(15km) (0)
                                               IR-2(40km) (0)
                                               LR-1(40km) (0)
                                               LR-2(80km) (0)
                                               LR-3(80km) (0)
                                               DX(40KM) (0)
                                               HX(40km) (0)
                                               ZX(80km) (0)
                                               VX(100km) (0)
                                               1xFC, 2xFC-SM(10km) (0)
                                               ESCON-SM(20km) (0)
  Link reach for 62.5u fiber (m)             = SR(2km) (0)
                                               IR-1(15km) (0)
                                               IR-2(40km) (0)
                                               LR-1(40km) (0)
                                               LR-2(80km) (0)
                                               LR-3(80km) (0)
                                               DX(40KM) (0)
                                               HX(40km) (0)
                                               ZX(80km) (0)
                                               VX(100km) (0)
                                               1xFC, 2xFC-SM(10km) (0)
                                               ESCON-SM(20km) (0)
  Nominal laser wavelength                   = 1550 nm.
  DWDM wavelength fraction                   = 1550.0  nm.
  Supported options                          = Tx disable
                                               Tx fault signal
                                               Loss of signal (standard implementation)
  Supported enhanced options                 = Alarms for monitored parameters
  Diagnostic monitoring                      = Digital diagnostics supported
                                               Diagnostics are externally calibrated
                                               Rx power measured is "Average power"
  Transceiver temperature operating range    = -5 C to 75 C (commercial)
  Minimum operating temperature              = 0 C
  Maximum operating temperature              = 70 C
```

High temperature alarm threshold

```
          = +90.000 C
```

High temperature warning threshold

= +85.000 C

Low temperature warning threshold

= +0.000 C

Low temperature alarm threshold

= -4.000 C

High voltage alarm threshold

= 3600.0 mVolts

High voltage warning threshold

= 3500.0 mVolts

Low voltage warning threshold

= 3100.0 mVolts

Low voltage alarm threshold

= 3000.0 mVolts
High laser bias current alarm threshold = 84.000 mAmps
High laser bias current warning threshold = 70.000 mAmps
Low laser bias current warning threshold = 4.000 mAmps
Low laser bias current alarm threshold = 2.000 mAmps

High transmit power alarm threshold

= 7.4 dBm

High transmit power warning threshold

= 4.0 dBm

Low transmit power warning threshold

```
      = -1.7 dBm

Low transmit power alarm threshold

        = -8.2 dBm

High receive power alarm threshold

        = -3.0 dBm

Low receive power alarm threshold

        = -33.0 dBm

High receive power warning threshold

        = -7.0 dBm

Low receive power warning threshold

        = -28.2 dBm
  External Calibration: bias current slope  = 1.000
  External Calibration: bias current offset = 0
```

Tip: Ensure that the hardware and software version of the device are compatible with the installed SFP or SFP+. Use the Cisco Optics-to-Device Compatibility Matrix.

This table lists the commands that can be used to troubleshoot link flaps. Recommended order of use: **show logging**, **show interfaces**, **show controllers ethernet-controller**, **show platform pm interface-flaps**, and then fiber-specific or copper-specific commands as applicable.

Command	Purpose
show interfaces counters errors	Displays the interface error counters.
show interfaces capabilities	Displays the capabilities of the specific interface.
show interface transceivers (fiber or SFP specific)	Displays information about the optical transceivers that have digital optical monitoring (DOM) enabled.

show interface link	Displays link level information.
show interface {interface{*interface-number*}} platform	Displays interface platform information.
show controllers ethernet-controller {interface{*interface-number*}} port-info	Displays additional port information.
show controllers ethernet-controller {interface{*interface-number*}} link status detail	Displays link status.
show errdisable flap-values	Displays the number of flaps that are allowed to occur before the errdisable status.
clear counters	Use this command to zero the traffic and error counters so that you can see if the problem is only temporary, or if the counters continue to increment.
clear controllers ethernet-controller	Use this command to clear the hardware Transmit and Receive counters.

Verify Cable Status with Time Domain Reflector (TDR)

The Time Domain Reflectometer (TDR) feature allows you to determine if a cable is OPEN or SHORT when it is at fault. With TDR, you can check the status of copper cables for the ports on the Catalyst 9000 Series Switches. TDR detects a cable fault with a signal that is sent through the cable and read the signal that is reflected back. All or part of the signal can be reflected back due to defects in the cable

Use the **test cable-diagnostics tdr** {**interface{*interface-number*} }** to start the TDR test, then use the **show cable-diagnostics tdr** {interface*interface-number*}.

Tip: Refer to the Checking Port Status and Connectivity for further details

The example shows a TDR test result for interface Tw2/0/10:

<#root>

```
Switch#
```

```
show cable-diagnostics tdr interface tw2/0/10
```

```
TDR test last run on: November 05 02:28:43
```

```
Interface Speed Local pair  Pair length         Remote pair Pair status
--------- ----- ---------- ------------------ ----------- --------------------
Tw2/0/10 1000M Pair A       1     +/- 5 meters  Pair A      Impedance Mismatch
               Pair B       1     +/- 5 meters  Pair B      Impedance Mismatch
               Pair C       1     +/- 5 meters  Pair C      Open
               Pair D       3     +/- 5 meters  Pair D      Open
```

Tip: On Catalyst 9300 Series Switches, only these cable fault types are detected - OPEN, SHORT, and IMPEDANCE MISMATCH. The Normal status is displayed in case cable is properly terminated and this is done for illustrative purpose.

TDR Guidelines

These guidelines apply to the use of TDR:

- Do not change the port configuration while the TDR test is running.
- If you connect a port during a TDR test to an Auto-MDIX enabled port, the TDR result can be invalid.
- If you connect a port during a TDR test to a 100BASE-T port such as that on the device, the unused pairs (4-5 and 7-8) are reported as faulty because the remote end does not terminate these pairs.
- Due to cable characteristics, you must run the TDR test multiple times to get accurate results.
- Do not change port status (for example, remove the cable at the near or far end) because the results can be inaccurate.
- TDR works best if the test cable is disconnected from the remote port. Otherwise, it can be difficult for you to interpret results correctly.
- TDR operates across four wires. Based on on the cable conditions, the status can show that one pair is OPEN or SHORT while all other wire pairs display as faulty. This operation is acceptable because you can declare a cable faulty provided one pair of wires is either OPEN or SHORT.
- TDR intent is to determine how poorly a cable functions rather than to locate a faulty cable.
- When TDR locates a faulty cable, you can still use an offline cable diagnosis tool to better diagnose the problem.
- TDR results can differ between runs on different switch models of Catalyst 9300 Series Switches because of the resolution difference of TDR implementations. When this occurs, you must refer to an offline cable diagnosis tool.

Digital Optic Monitoring (DOM)

Digital Optical Monitoring (DOM) is an industry wide standard, intended to define a digital interface to access real-time parameters such as:

- Temperature
- Transceiver supply voltage
- Laser bias current
- Optical Tx power
- Optical Rx power

How to Enable DOM

The table list the commands you can used to turn on/off DOM for all transceivers type in the system:

Steps	**Command or Action**	**Purpose**
Step 1	**enable** Example: **switch > enable**	Enables the physical EXEC mode. Enter your password if prompted.
Step 2	**configure terminal** Example: **switch#configure terminal**	Enters Global configuration mode.
Step 3	**transceiver type all** Example: **switch(config)#transceiver type all**	Enters the Transceiver type configuration mode.
Step 4	**monitoring** Example: **switch(config)#monitoring**	Enables monitoring of all optical transceivers.

Use the **show interfaces {interface{*interface-number*}} transceiver detail** command to display transceiver information:

<#root>

```
Switch#

show interfaces hundredGigE 1/0/25 transceiver detail


ITU Channel not available (Wavelength not available),
Transceiver is internally calibrated.
mA: milliamperes, dBm: decibels (milliwatts), NA or N/A: not applicable.
++ : high alarm, + : high warning, - : low warning, -- : low alarm.
A2D readouts (if they differ), are reported in parentheses.
The threshold values are calibrated.
```

High Alarm High Warn Low Warn Low Alarm

Port	Temperature (Celsius)	Threshold (Celsius)	Threshold (Celsius)	Threshold (Celsius)	Threshold (Celsius)
Hu1/0/25	28.8	75.0	70.0	0.0	-5.0

High Alarm High Warn Low Warn Low Alarm

Port	Voltage (Volts)	Threshold (Volts)	Threshold (Volts)	Threshold (Volts)	Threshold (Volts)
Hu1/0/25	3.28	3.63	3.46	3.13	2.97

High Alarm High Warn Low Warn Low Alarm

Port	Lane	Current (milliamperes)	Threshold (mA)	Threshold (mA)	Threshold (mA)	Threshold (mA)
Hu1/0/25	N/A	6.2	10.0	8.5	3.0	2.6

Optical High Alarm High Warn Low Warn Low Alarm

Port	Lane	Transmit Power (dBm)	Threshold (dBm)	Threshold (dBm)	Threshold (dBm)	Threshold (dBm)
Hu1/0/25	N/A	-2.2	1.7	-1.3	-7.3	-11.3

```
  Optical          High Alarm  High Warn   Low Warn   Low Alarm

                 Receive Power    Threshold   Threshold   Threshold  Threshold

Port      Lane   (dBm)            (dBm)       (dBm)       (dBm)      (dBm)
--------- ----   ---------------  ----------  ---------   ---------  ---------
Hu1/0/25  N/A    -16.7                2.0        -1.0        -9.9       -13.9
```

Tip: In order to determine if an optical transceiver operates at the appropriate signal levels, please refer to the Cisco Optics Datasheet

Digital Optic Monitoring Syslog Messages

This section describes the most relevant threshold violation syslog messages:

Temperature Levels of SFP optics

- Explanation: These log messages are generated when temperature is low or exceeds the normal optic operate values:

<#root>

```
%SFF8472-3-THRESHOLD_VIOLATION: Te7/3: T

emperature high alarm

; Operating value: 88.7 C, Threshold value: 74.0 C.
%SFF8472-3-THRESHOLD_VIOLATION: Fo1/1/1:

Temperature low alarm

; Operating value: 0.0 C, Threshold value: 35.0 C.
```

Voltage Levels of SFP Optics

- Explanation: These log messages are generated when voltage is low or exceeds the normal optic operate values:

<#root>

```
%SFF8472-3-THRESHOLD_VIOLATION: Gi1/1/3:

Voltage high warning

; Operating value: 3.50 V, Threshold value: 3.50 V.
%SFF8472-5-THRESHOLD_VIOLATION: Gi1/1:

Voltage low alarm

; Operating value: 2.70 V, Threshold value: 2.97 V.
```

Light Levels of SFP Optics

- Explanation: These log messages are generated when the light power is low or exceeds the optic operate values:

<#root>

```
%SFF8472-3-THRESHOLD_VIOLATION: Gi1/0/1: Rx

power high warning

; Operating value: -2.7 dBm, Threshold value: -3.0 dBm.
%SFF8472-5-THRESHOLD_VIOLATION: Te1/1: Rx

power low warning

; Operating value: -13.8 dBm, Threshold value: -9.9 dBm.
```

Tip: For further information on DOM see Digital Optical Monitoring

Cisco Optics and Forward Error Correction (FEC)

FEC is a technique used to detect and correct a certain number of errors in a bitstream and appends redundant bits and error-checking code to the message block before transmission. As a module manufacturer, Cisco takes care to design our transceivers to comply with specifications. When the optical transceiver is operated in a Cisco host platform, the FEC is enabled by default based on the optical module type that the host software detects (See this downloadable table). In the vast majority of cases, the FEC implementation is dictated by the industry standard that the optic type supports.

For certain custom specifications, FEC implementations vary. Refer to Understanding FEC and its Implementation in Cisco Optics document for detailed information.

The example shows how to configure FEC and some of the available options:

```
<#root>

switch(config-if)#

fec


?
  auto Enable FEC Auto-Neg
  cl108 Enable clause108 with 25G
  cl74 Enable clause74 with 25G
  off Turn FEC off

Use the

show interface


 command to verify FEC configuration:

TwentyFiveGigE1/0/13 is up, line protocol is up (connected)
  Hardware is Twenty Five Gigabit Ethernet, address is 3473.2d93.bc8d (bia 3473.2d93.bc8d)
  MTU 9170 bytes, BW 25000000 Kbit/sec, DLY 10 usec,
      reliability 255/255, txload 1/255, rxload 1/255
  Encapsulation ARPA, loopback not set
  Keepalive set (10 sec)
  Full-duplex, 25Gb/s, link type is force-up, media type is SFP-25GBase-SR

  Fec is auto



< -- The configured setting for FEC is displayed here


  input flow-control is on, output flow-control is off
  ARP type: ARPA, ARP Timeout 04:00:00
--snip--
```

Note: Both sides of a link must have the same FEC encoding algorithm enabled for the link to come up.

Debug Commands

This table lists the various commands that can be used to debug Port Flaps

Caution: Use the **debug** commands with caution. Please be aware many **debug** commands have an impact on live network and only are recommended to use in a lab environment when the issue is reproduced. ;

Command	Purpose
debug pm	Port Manager debugging

debug pm port	Port related events
debug platform pm	NGWC Platform Port Manager Debug Info
debug platform pm l2-control	NGWC L2 Control Infra debug
debug platform pm link-status	Interface link detection events
debug platform pm pm-vectors	Port Manager Vector Functions
debug condition interface <interface name>	Selectively enable debugs for specific interface
debug interface state	States transitions

This is a partial sample output example of the **debug** commands listed in the table:

<#root>

```
SW_2#
```

sh debugging

```
PM (platform):
 L2 Control Infra debugging is on
```

<-- debug platform pm l2-control

```
 PM Link Status debugging is on
```

<-- debug platform pm link-status

```
 PM Vectors debugging is on
```

<-- debug platform pm pm-vectors

```
Packet Infra debugs:

Ip Address Port
------------------------------------------------------|----------

Port Manager:
  Port events debugging is on
```

<-- debug pm port

```
Condition 1: interface Te1/0/2 (1 flags triggered)
Flags: Te1/0/2

------ Sample output ---------
```

***Aug 25 20:01:05.791: link up/down event : link-down on Te1/0/2**

***Aug 25 20:01:05.791: pm_port 1/2: during state access, got event 5(link_down) <-- Link down event (day/**

*Aug 25 20:01:05.791: @@@ pm_port 1/2: access -> pagp
*Aug 25 20:01:05.792: IOS-FMAN-PM-DEBUG-PM-VECTORS: Success sending PM tdl message
*Aug 25 20:01:05.792: IOS-FMAN-PM-DEBUG-PM-VECTORS: Success sending PM tdl message
*Aug 25 20:01:05.792: IOS-FMAN-PM-DEBUG-PM-VECTORS: Success sending PM tdl message
*Aug 25 20:01:05.792: IOS-FMAN-PM-DEBUG-PM-VECTORS: Vp Disable: pd=0x7F1E797914B0 dpidx=10 Te1/0/2
*Aug 25 20:01:05.792: IOS-FMAN-PM-DEBUG-PM-VECTORS: Success sending PM tdl message
*Aug 25 20:01:05.792: IOS-FMAN-PM-DEBUG-PM-VECTORS: Success sending PM tdl message
*Aug 25 20:01:05.792: Maintains count of VP per Interface:delete, pm_vp_counter[0]: 14, pm_vp_counter[1
*Aug 25 20:01:05.792: *** port_modechange: 1/2 mode_none(10)
*Aug 25 20:01:05.792: @@@ pm_port 1/2: pagp -> dtp

***Aug 25 20:01:05.792: stop flap timer : Te1/0/2 pagp**

*Aug 25 20:01:05.792: *** port_bndl_stop: 1/2 : inform yes
*Aug 25 20:01:05.792: @@@ pm_port 1/2: dtp -> present
*Aug 25 20:01:05.792: *** port_dtp_stop: 1/2
*Aug 25 20:01:05.792: stop flap timer : Te1/0/2 pagp
*Aug 25 20:01:05.792: stop flap timer : Te1/0/2 dtp
*Aug 25 20:01:05.792: stop flap timer : Te1/0/2 unknown

***Aug 25 20:01:05.792: *** port_linkchange: reason_link_change(3): link_down(0)1/2**

<-- State link change

*Aug 25 20:01:05.792: pm_port 1/2: idle during state present

***Aug 25 20:01:05.792: @@@ pm_port 1/2: present -> link_down <-- State of the link**

*Aug 25 20:01:06.791: %LINEPROTO-5-UPDOWN: Line protocol on Interface TenGigabitEthernet1/0/2, changed

***Aug 25 20:01:07.792: %LINK-3-UPDOWN: Interface TenGigabitEthernet1/0/2, changed state to down**

***Aug 25 20:01:11.098: IOS-FMAN-PM-DEBUG-LINK-STATUS: Received LINKCHANGE in xcvr message, if_id 10 (TenG**

***Aug 25 20:01:11.098: IOS-FMAN-PM-DEBUG-LINK-STATUS: if_id 0xA, if_name Te1/0/2, link up <-- Link became**

***Aug 25 20:01:11.098: link up/down event: link-up on Te1/0/2**

```
*Aug 25 20:01:11.098: pm_port 1/2: during state link_down, got event 4(link_up)
*Aug 25 20:01:11.098: @@@ pm_port 1/2: link_down -> link_up
*Aug 25 20:01:11.098: flap count for link type : Te1/0/2 Linkcnt = 0
*Aug 25 20:01:11.099: pm_port 1/2: idle during state link_up
*Aug 25 20:01:11.099: @@@ pm_port 1/2: link_up -> link_authentication
*Aug 25 20:01:11.099: pm_port 1/2: during state link_authentication, got event 8(authen_disable)
*Aug 25 20:01:11.099: @@@ pm_port 1/2: link_authentication -> link_ready
*Aug 25 20:01:11.099: *** port_linkchange: reason_link_change(3): link_up(1)1/2
*Aug 25 20:01:11.099: pm_port 1/2: idle during state link_ready
*Aug 25 20:01:11.099: @@@ pm_port 1/2: link_ready -> dtp
*Aug 25 20:01:11.099: IOS-FMAN-PM-DEBUG-PM-VECTORS: Set pm vp mode attributes for Te1/0/2 vlan 1
*Aug 25 20:01:11.099: IOS-FMAN-PM-DEBUG-PM-VECTORS: Success sending PM tdl message
*Aug 25 20:01:11.099: IOS-FMAN-PM-DEBUG-PM-VECTORS: Success sending PM tdl message
*Aug 25 20:01:11.099: IOS-FMAN-PM-DEBUG-PM-VECTORS: Success sending PM tdl message
*Aug 25 20:01:11.099: pm_port 1/2: during state dtp, got event 13(dtp_complete)
*Aug 25 20:01:11.099: @@@ pm_port 1/2: dtp -> dtp
*Aug 25 20:01:11.099: IOS-FMAN-PM-DEBUG-PM-VECTORS: Set pm vp mode attributes for Te1/0/2 vlan 1
*Aug 25 20:01:11.099: IOS-FMAN-PM-DEBUG-PM-VECTORS: Success sending PM tdl message
*Aug 25 20:01:11.099: DTP flapping: flap count for dtp type: Te1/0/2 Dtpcnt = 0
*Aug 25 20:01:11.099: pm_port 1/2: during state dtp, got event 110(dtp_done)
*Aug 25 20:01:11.099: @@@ pm_port 1/2: dtp -> pre_pagp_may_suspend
*Aug 25 20:01:11.099: pm_port 1/2: idle during state pre_pagp_may_suspend
*Aug 25 20:01:11.099: @@@ pm_port 1/2: pre_pagp_may_suspend -> pagp_may_suspend
*Aug 25 20:01:11.099: pm_port 1/2: during state pagp_may_suspend, got event 33(pagp_continue)
*Aug 25 20:01:11.099: @@@ pm_port 1/2: pagp_may_suspend -> start_pagp
*Aug 25 20:01:11.099: pm_port 1/2: idle during state start_pagp
*Aug 25 20:01:11.099: @@@ pm_port 1/2: start_pagp -> pagp
*Aug 25 20:01:11.100: IOS-FMAN-PM-DEBUG-PM-VECTORS: Success sending PM tdl message
*Aug 25 20:01:11.100: IOS-FMAN-PM-DEBUG-PM-VECTORS: Success sending PM tdl message
*Aug 25 20:01:11.100: IOS-FMAN-PM-DEBUG-PM-VECTORS: Set pm vp mode attributes for Te1/0/2 vlan 1
*Aug 25 20:01:11.100: IOS-FMAN-PM-DEBUG-PM-VECTORS: Success sending PM tdl message
*Aug 25 20:01:11.100: IOS-FMAN-PM-DEBUG-PM-VECTORS: Success sending PM tdl message
*Aug 25 20:01:11.100: IOS-FMAN-PM-DEBUG-PM-VECTORS: Success sending PM tdl message
*Aug 25 20:01:11.100: *** port_bndl_start: 1/2
*Aug 25 20:01:11.100: stop flap timer : Te1/0/2 pagp
*Aug 25 20:01:11.100: pm_port 1/2: during state pagp, got event 34(dont_bundle)
*Aug 25 20:01:11.100: @@@ pm_port 1/2: pagp -> pre_post_pagp
*Aug 25 20:01:11.100: pm_port 1/2: idle during state pre_post_pagp
*Aug 25 20:01:11.100: @@@ pm_port 1/2: pre_post_pagp -> post_pagp
*Aug 25 20:01:11.100: IOS-FMAN-PM-DEBUG-PM-VECTORS: Success sending PM tdl message
*Aug 25 20:01:11.100: IOS-FMAN-PM-DEBUG-PM-VECTORS: Success sending PM tdl message
*Aug 25 20:01:11.100: pm_port 1/2: during state post_pagp, got event 14(dtp_access)
*Aug 25 20:01:11.100: @@@ pm_port 1/2: post_pagp -> access
*Aug 25 20:01:11.100: IOS-FMAN-PM-DEBUG-PM-VECTORS: Success sending PM tdl message
*Aug 25 20:01:11.100: IOS-FMAN-PM-DEBUG-PM-VECTORS: Success sending PM tdl message
*Aug 25 20:01:11.100: IOS-FMAN-PM-DEBUG-PM-VECTORS: Success sending PM tdl message
*Aug 25 20:01:11.100: IOS-FMAN-PM-DEBUG-PM-VECTORS: Set pm vp mode attributes for Te1/0/2 vlan 1
*Aug 25 20:01:11.100: IOS-FMAN-PM-DEBUG-PM-VECTORS: Success sending PM tdl message
*Aug 25 20:01:11.100: Maintains count of VP per Interface:add, pm_vp_counter[0]: 15, pm_vp_counter[1]:
*Aug 25 20:01:11.100: IOS-FMAN-PM-DEBUG-PM-VECTORS: vlan vp enable for port(Te1/0/2) and vlan:1
*Aug 25 20:01:11.101: IOS-FMAN-PM-DEBUG-PM-VECTORS: VP ENABLE: vp_pvlan_port_mode:access for Te1/0/2
*Aug 25 20:01:11.101: IOS-FMAN-PM-DEBUG-PM-VECTORS: VP Enable: vp_pvlan_native_vlanId:1 for Te1/0/2
*Aug 25 20:01:11.101: IOS-FMAN-PM-DEBUG-PM-VECTORS: Success sending PM tdl message
*Aug 25 20:01:11.101: IOS-FMAN-PM-DEBUG-PM-VECTORS: Success sending PM tdl message
*Aug 25 20:01:11.101: *** port_modechange: 1/2 mode_access(1)
*Aug 25 20:01:11.101: IOS-FMAN-PM-DEBUG-PM-VECTORS: The operational mode of Te1/0/2 in set all vlans is
```

```
*Aug 25 20:01:11.101: IOS-FMAN-PM-DEBUG-PM-VECTORS: Success sending PM tdl message
*Aug 25 20:01:11.101: IOS-FMAN-PM-DEBUG-PM-VECTORS: vp_pvlan port_mode:access vlan:1 for Te1/0/2
*Aug 25 20:01:11.101: IOS-FMAN-PM-DEBUG-PM-VECTORS: vp_pvlan port_mode:access native_vlan:1 for Te1/0/2
*Aug 25 20:01:11.102: IOS-FMAN-PM-DEBUG-PM-VECTORS: Success sending PM tdl message

*Aug 25 20:01:13.098: %LINK-3-UPDOWN: Interface TenGigabitEthernet1/0/2, changed state to up



*Aug 25 20:01:14.098: %LINEPROTO-5-UPDOWN: Line protocol on Interface TenGigabitEthernet1/0/2, changed s
```

Related Cisco Bugs

Cisco Bug ID	Description
Cisco bug ID CSCvu13029	Intermittent Link Flaps on mGig Cat9300 switches to mGig capable endpoints.
Cisco bug ID CSCvt50788	Cat9400 mGig interop issues with other mGig devices causes link flaps.
Cisco bug ID CSCvu92432	CAT9400: Mgig interface Flaps with Mgig APs.
Cisco bug ID CSCve65787	Autoneg support for 100G/40G/25G Cu xcvr.

Related Information

- Cisco Optics-to-Device Compatilbility Matrix
- Cisco SFP Modules for Gigabit Ethernet Applications Data Sheet
- 25GE and 100GE – Enabling Higher Speeds in Enterprise with Investment Protection White Paper
- Cisco CWDM SFP Solution Data Sheet
- Support Innovation: How Cisco TAC is transforming documentation and simplifying self-service
- Cisco Technical Support & Downloads