



# Port Module Changes

Application Note

CONFIDENTIAL

**AN1290**  
Rev. APPL-2024.06  
2024-06-28

## Table of Contents

1. Introduction .....	3
2. New or Changed CLI Interface Configuration Commands .....	3
2.1. speed Command.....	3
2.2. duplex Command .....	5
2.3. media-type Command .....	6
2.4. fec Command .....	7
3. New or Changed CLI Status Commands .....	9
3.1. Show Interface Status.....	9
3.2. Show Detailed Interface Status .....	12
4. New CLI Debug Commands .....	15
4.1. debug show interface state .....	15
4.2. debug sfp dump .....	16
5. New Debug Trace Groups .....	16

---

# 1. Introduction

This document describes port module changes that have happened between the 2020.12 and 2021.03 release of the WebStaX application software.

It applies to all chip families that run this software.

If a given feature is only meant for one chip family or board layout, it is specifically mentioned in the text.

The primary reason for refactoring the port module has been to be able to implement clause 73 aneg, which is auto-negotiation of speed and FEC (Forward Error Correction) for 10GBASE-KR and 25GBASE-KR(-S), also known as copper backplane (CuBP), and for 25G directly attached, twinaxial cables (25GBASE-CR and 25GBASE-CR-S). The latter are called DAC (Directly Attached Copper) cables in daily speech, though this is the official name only for 10GBASE-CR cables.

Clause 73 aneg is often wrongly called "KR", because it used to be used only on 10GBASE-KR (CuBP), but since it's now also used on 25GBASE-CR(-S), we should get used to calling it clause 73, named after the section in IEEE802.3 where it is specified.

During the refactoring, it was found that some of the CLI commands had to be changed in order for them to work better. One of the oddities with the old module was for instance that if you changed on parameter of a port, it could have effect on other parameters. This has been cleaned up as will be explained in subsequent chapters.

## 2. New or Changed CLI Interface Configuration Commands

### 2.1. speed Command

In 2020.06, the `speed` command looked like:

```
speed {25g | 10g | 5g | 2500 | 1000 | 100 | 10 | auto {[10] [100] [1000] [2500] [5g] [10g]}}
```

In 2020.09, it was changed that to implement some clause 73 features, and it looked like:

```
speed {25g | 10g | 5g | 2500 | 1000 | 100 | 10 | auto {[10] [100] [1000] [2500] [5g] [10g]} | kr {[1000] [2500] [5g] [10g] [25g] [no-r-fec] [no-rs-fec] [no-train] [remote-only]}}
```

Now, it has been re-iterated and resulted in the following command:

```
speed {10 | 100 | 1000 | 2500 | 5g | 10g | 25g | force-clause-73 | auto {[10] [100] [1000] [2500] [5g] [10g] {[no-hdx] | [no-fdx]}}}
```

Firstly, the initial arguments are now ordered in ascending speeds - just as they are in the optional arguments to `auto`. This doesn't cause any functional changes.

Secondly, the `auto` command is much more versatile now. The most noticeable change compared to the 2020.06 version is `no-hdx` and `no-fdx`. These will be explained in the `duplex` Command section below.

Previously, `speed auto` was the default for copper ports and SFP ports with a native speed lower than 10G, which meant that they ran clause 28 and clause 37 aneg (1G auto-negotiation), respectively.

**NOTE**

Clause 28 and clause 37 aneg also refer to sections in IEEE802.3. Clause 28 specifies aneg operation for PHYs and clause 37 specifies aneg operation for SFPs (not CuSFPs, because these are PHYs disguised as SFPs).

Previously, 10G and 25G ports defaulted to forced 10G and 25G, respectively. Even though the speed was "forced", they *could* link up in a lower speed if an SFP with a nominal speed of e.g. 1G was inserted. Now, all sorts of ports default to `speed auto`, whose operation is explained in the following section.

**NOTE**

Throughout this document, the term "SFP's nominal speed" is used to indicate the highest speed the SFP supports.

A final note: In the 2020.09 and 2020.12 releases, `speed kr` took some options indicating what to advertise and request. Now, all speeds are advertised (well, not 25G on a 10G port or if a 10G SFP is inserted in a 25G port), FEC is handled through a new `fec` command (see `fec` Command) and training is always enabled.

### 2.1.1. speed auto

`speed auto` now means something along these lines:

If the port's native speed is < 10G, the port will always use 1G aneg no matter what kind of SFP is inserted. This is as it was before.

If the port's native speed is 2.5G, 5G or 10G and it's a copper port, it will also use 1G aneg with some extensions to be able to auto-negotiate speeds higher than 1G.

If the port's native speed is 10G and it's an SFP port, the operation depends on the inserted SFP's nominal speed: If the inserted SFP's nominal speed is  $\geq 10G$ , the port will run forced 10G. Otherwise, if the SFP's nominal speed is 100 Mbps, the port will run forced 100M, otherwise the port will run 1G aneg.

If the port's native speed is 25G, it also depends on the inserted SFP's nominal speed: If the inserted SFP's nominal speed is 25G and it's a DAC cable, it will run clause 73, otherwise it will run forced 25G. If the inserted SFP's nominal speed is 10G, it will run forced 10G. If the inserted SFP's nominal speed is 100 Mbps, it will run forced 100M. Otherwise (the inserted SFP's nominal speed is  $\geq 1G$  and  $< 10G$ ), it will run 1G aneg.

**NOTE**

We have 2.5G ports as well. Since we don't have 2.5G SFPs, `speed auto` on a 2.5G port will resort to 1G aneg, which at most can link up in 1G. So if you want the 2.5G port to operate in 2.5G mode, you must force it to do so - in both ends. This has always been like this.

### 2.1.2. How to Avoid or Force Clause 73

From the previous section, we can see that clause 73 is auto-enabled whenever a 25G DAC cable is detected in a 25G port. In order to avoid running clause 73 (not recommended), the user must force the port to run a given speed (e.g. `speed 25g`).

One can also see that if a 25G Optical SFP is inserted in a 25G port or a 10G SFP (of any type) is inserted into a 10G or 25G port, the port does not run clause 73. This is in accordance with the standard. However, the user may want to force the use of clause 73, which is why we have the `force-clause-73` keyword to the `speed` command.

The `force-clause-73` keyword is only available if you have enabled debug by issuing `platform debug allow` prior to configuring the port. This is because the software automatically finds out when to really apply clause 73.

This `force-clause-73` keyword can be saved to startup-config and read back from startup-config during boot.

**NOTE**

`force-clause-73` is not available in the Web GUI. When the port is in this mode, it will - in the Web GUI - show itself as `Autonegotiation` in the `Configured Speed` column, and if you click `Save` while it still shows `Autonegotiation`, it will still be in forced clause 73 mode. If you change the speed to a forced speed, the forced clause 73 mode will be disabled.

### 2.1.3. Forced Speeds

If you force a given speed on a port, it will only be able to link up at that speed. This has the following implications for SFP ports:

If you insert an SFP with a nominal speed lower than the forced speed, an operational warning will be issued (see `Operational Warnings` Column) and you will not get link.

If you insert an SFP with a nominal speed at or higher than the forced speed, you may get link. In addition, an operational warning will be issued if the SFP's nominal speed is higher than the forced speed.

## 2.2. `duplex` Command

In 2020.12, the `duplex` command looked like:

```
duplex {half | full | auto [half | full]}
```

Now, it is:

```
duplex {half | full}
```

Previously, the `duplex auto [half | full]` command was used to specify what kind of duplex should be advertised when running aneg.

This had the deficiency that you couldn't at the same time tell what duplex should be used when running a forced speed and which duplex should be advertised when running aneg.

Therefore, the command is changed to only specify what the port should be using when running a forced speed.

For advertised duplex, `speed auto` now takes two additional parameters, `no-hdx` and `no-fdx`, which cannot be specified at the same time, and which are only available if the ports supports both half and full duplex (all ports support the latter).

These two parameters are kind of self-explanatory and are carefully selected to have the `no`-form, so that if they are not specified at all, both will be advertised (hdx won't be advertised if the port doesn't support half duplex, though).

Previously, if `duplex auto` was issued, then also the `speed` was changed behind the user's back from whatever it was before to `auto`, while issuing a message similar to the following on the console:

```
GigabitEthernet 1/2 set to auto mode, speed configuration updated accordingly
```

This is no longer a problem, because the `duplex` command only affects forced speeds.

For backwards compatibility with already saved startup-configs, the old syntax is still allowed during boot, but not interactively at runtime.

## 2.3. media-type Command

In 2020.12, the `media-type` command looked like:

```
media-type {rj45 | sfp | dual | dac-1m | dac-2m | dac-3m | dac-5m}
```

In 2021.03, it is:

```
media-type {rj45 | sfp | dual}
```

The `dac-Xm` options were meant for 10G ports with a DAC cable to select the right serdes configuration based on the DAC cable's configured length.

Since then, a common set of serdes configuration has been found, independent of the DAC cable's length, so we don't need them anymore.

However, for backwards compatibility with already saved startup-config, the old syntax is still allowed during boot, but not interactively at runtime.

Also, previously, the `media-type` command was only available on dual media ports.

This makes a lot of sense, because it's not needed on non-dual media ports, since it cannot be changed.

However, it is very nice to be able to see a port's native media type when doing a:

```
show running-config feature port all-defaults
```

And because whatever comes out of that command must be applicable when used interactively (or copied from a file), the command must also be available on non-dual media ports.

### 2.3.1. media-type on Dual Media Ports

Previously, when issuing `no media-type` on a dual media port, all sorts of warnings could be issued and the port's `speed` and `duplex` may or may not have changed.

Now (still talking dual media ports), if the port is configured as `media-type dual` or `media-type sfp` and you try to set the port to a forced speed, you will get an error.

Vice versa, if you have first configured the port with `media-type rj45`, and then forced the speed to e.g. 100 Mbps ( `speed 100` ), and attempt to change the media type to `sfp` or `dual`, you will also get an adequate error.

SFPs win over RJ45. This means, when `media-type` is set to `dual` and an SFP is plugged in, you will not get link through RJ45. You will need to unplug the SFP or configure `media-type` as `rj45` to get link on RJ45.

## 2.4. fec Command

FEC is short for Forward Error Correction. It is a technique for controlling errors over an unreliable link. The idea is that the sender overlays the frame data signal with error correction data that allows a receiver to correct bit errors in the received frame. FEC will not affect the throughput on the link (however, due to a bug on 25G ports, RS-FEC rate compensation cannot be enabled, causing the maximum throughput to be 24.998 Gbps when RS-FEC is enabled).

It comes in two flavors:

- R-FEC (IEEE802.3 clause 74 - sometimes called Firecode). This is meant for 10G.
- RS-FEC (IEEE802.3 clause 108 - stands for Reed-Solomon-FEC). This is meant for 25G and doesn't work on 10G links.

The command is new in 2021.03 and looks like: `fec {auto | r-fec | rs-fec | none}`

The command is only available on platforms with FEC support, that is, Jaguar-2, Serval-T and FireAnt, and only on ports on these platforms that have FEC support (10G and 25G ports).

The command affects both what is requested during clause 73 aneg and what the port is configured to use if not running clause 73 aneg.

If running clause 73 aneg on 10G ports or on 25G ports loaded with 10G SFPs, we always tell the link partner that we support R-FEC. What the end user can control with the `fec` command is whether we request R-FEC. If either us or the link partner requests R-FEC, the port will end up using R-FEC.

Clause 73 on 25G ports loaded with 25G SFPs is a bit different, because 25G ports must support both R-FEC and RS-FEC. With the `fec` command, the end user can control whether R-FEC and/or RS-FEC is requested or not (all combinations). If at least one end request RS-FEC, the port will end up using RS-FEC, otherwise if at least one end requests R-FEC, the port will end up using R-FEC, otherwise the port will not use any FEC.

The individual settings are:

- `auto` : This is the default and means the following:
  - If a 10G port runs clause 73, R-FEC will be requested.
  - If a 25G port runs clause 73 with a 10G SFP, R-FEC will be requested.
  - If a 25G port runs clause 73 with a 25G SFP, both R-FEC and RS-FEC will be requested.
  - If a 25G port is loaded with a 25G optical SFP and clause 73 is not running or the port is forced to 25G, RS-FEC will be enabled.
  - If a 25G port is loaded with a 25G DAC cable and speed is forced to 25G, RS-FEC will be enabled.
  - If a 25G port is loaded with a 25G DAC cable and speed is forced to 5G or 10G, R-FEC will be enabled.
  - Otherwise, no FEC will be enabled.
- `r-fec` :
  - If a 10G or 25G port runs clause 73, only R-FEC will be requested.
  - If a 10G or 25G port does not run clause 73, but is loaded with at least a 10G SFP and the speed is at least 5G, only R-FEC will be enabled.
  - Otherwise, no FEC will be enabled.
- `rs-fec` :
  - This option is only available on 25G ports.
  - If a 25G SFP is inserted and the port is running clause 73, RS-FEC will be requested.
  - If a 25G SFP is inserted, and the port is not running clause 73, but will end up in 25G (either through forced 25G or `auto`), RS-FEC will be enabled.
  - Otherwise, no FEC will be enabled.
- `none` :
  - If the port is running clause 73, neither R-FEC nor RS-FEC will be requested (but remember that this does not mean that clause 73 aneg will not result in the port running some sort of FEC).
  - Otherwise, the port will not run any FEC.

**NOTE**

All the combinations of `speed` and `fec` per port and per SFP type can be quite difficult to grasp, so the Web GUI help under Port Configuration contains tables that show all combinations along with what you can expect.



## 3. New or Changed CLI Status Commands

### 3.1. Show Interface Status

In 2020.12, the interface status command was:

```
show interface <port_type_list> status [err-disable]
```

In 2021.12, it looks like: `show interface <port_type_list> status [err-disable] [details [clause-73]]`

The `err-disable` part is not visible on any platform, because the error-disable module is not enabled on any platform.

Not only the command, but also what comes out of executing it has changed. Below you will see the old and the new status.

#### Old Interface Status Layout

```
# show interface * status
```

Interface	Mode	Speed/Duplex	Media Type	Flow Control	Max Frame	Excessive	Link
GigabitEthernet 1/1	enabled	Auto	rj45	disabled	9600	Discard	1Gfdx
GigabitEthernet 1/2	enabled	Auto	rj45	disabled	9600	Discard	Down
GigabitEthernet 1/3	enabled	Auto	rj45	disabled	9600	Discard	Down
GigabitEthernet 1/4	enabled	Auto	rj45	disabled	9600	Discard	Down
GigabitEthernet 1/5	enabled	Auto	rj45	disabled	9600	Discard	Down
GigabitEthernet 1/6	enabled	Auto	rj45	disabled	9600	Discard	Down
GigabitEthernet 1/7	enabled	Auto	rj45	disabled	9600	Discard	Down
GigabitEthernet 1/8	enabled	Auto	rj45	disabled	9600	Discard	Down
GigabitEthernet 1/9	enabled	Auto	rj45	disabled	9600	Discard	Down
GigabitEthernet 1/10	enabled	Auto	rj45	disabled	9600	Discard	Down
GigabitEthernet 1/11	enabled	Auto	rj45	disabled	9600	Discard	Down
GigabitEthernet 1/12	enabled	Auto	rj45	disabled	9600	Discard	Down
GigabitEthernet 1/13	enabled	Auto	rj45	disabled	9600	Discard	Down
GigabitEthernet 1/14	enabled	Auto	rj45	disabled	9600	Discard	Down
GigabitEthernet 1/15	enabled	Auto	rj45	disabled	9600	Discard	Down
GigabitEthernet 1/16	enabled	Auto	rj45	disabled	9600	Discard	Down
GigabitEthernet 1/17	enabled	Auto	rj45	disabled	9600	Discard	Down
GigabitEthernet 1/18	enabled	Auto	rj45	disabled	9600	Discard	Down
GigabitEthernet 1/19	enabled	Auto	rj45	disabled	9600	Discard	Down
GigabitEthernet 1/20	enabled	Auto	rj45	disabled	9600	Discard	1Gfdx
GigabitEthernet 1/21	enabled	Auto	dual	disabled	9600	Discard	Down
GigabitEthernet 1/22	enabled	Auto	dual	disabled	9600	Discard	1Gfdx
GigabitEthernet 1/23	enabled	Auto	dual	disabled	9600	Discard	1Gfdx
Fiber							
GigabitEthernet 1/24	enabled	Auto	dual	disabled	9600	Discard	1Gfdx
Fiber							
GigabitEthernet 1/25	enabled	100fdx	sfp	disabled	9600	Discard	Down
2.5GigabitEthernet 1/1	enabled	Auto	sfp	disabled	9600	Discard	Down

*New Interface Status Layout*

```
# show interface * status
```

Interface	Mode	Speed	Aneg	Media Type	SFP Family	Link	Operational Warnings
Gi 1/1	Enabled	Auto	Yes	RJ45	N/A	1Gfdx	
Gi 1/2	Enabled	Auto	Yes	RJ45	N/A	Down	
Gi 1/3	Enabled	Auto	Yes	RJ45	N/A	Down	
Gi 1/4	Enabled	Auto	Yes	RJ45	N/A	Down	
Gi 1/5	Enabled	Auto	Yes	RJ45	N/A	Down	
Gi 1/6	Enabled	Auto	Yes	RJ45	N/A	Down	
Gi 1/7	Enabled	Auto	Yes	RJ45	N/A	Down	
Gi 1/8	Enabled	Auto	Yes	RJ45	N/A	Down	
Gi 1/9	Enabled	Auto	Yes	RJ45	N/A	Down	
Gi 1/10	Enabled	Auto	Yes	RJ45	N/A	Down	
Gi 1/11	Enabled	Auto	Yes	RJ45	N/A	Down	
Gi 1/12	Enabled	Auto	Yes	RJ45	N/A	Down	
Gi 1/13	Enabled	Auto	Yes	RJ45	N/A	Down	
Gi 1/14	Enabled	Auto	Yes	RJ45	N/A	Down	
Gi 1/15	Enabled	Auto	Yes	RJ45	N/A	Down	
Gi 1/16	Enabled	Auto	Yes	RJ45	N/A	Down	
Gi 1/17	Enabled	Auto	Yes	RJ45	N/A	Down	
Gi 1/18	Enabled	Auto	Yes	RJ45	N/A	Down	
Gi 1/19	Enabled	Auto	Yes	RJ45	N/A	Down	
Gi 1/20	Enabled	Auto	Yes	RJ45	N/A	1Gfdx	
Gi 1/21	Enabled	Auto	Yes	SFP	25G DAC	Down	SFP's nominal speed is higher than actual speed, which may cause instability
Gi 1/22	Enabled	Auto	Yes	RJ45	None	1Gfdx	
Gi 1/23	Enabled	Auto	Yes	SFP	1G Unknown	1Gfdx	SFP type cannot be determined on this interface. Instability can be expected
Gi 1/24	Enabled	Auto	Yes	SFP	1G Unknown	1Gfdx	SFP type cannot be determined on this interface. Instability can be expected
Gi 1/25	Enabled	100fdx	No	SFP	None	Down	
2.5G 1/1	Enabled	Auto	Yes	SFP	None	Down	

First of all, to save space, the short interface name is now displayed and some of the configuration columns are gone.

The **Link** column no longer writes the media it links up with (e.g. instead of showing **1Gfdx Fiber**, it just shows **1Gfdx**).

The other columns are:

### 3.1.1. Aneg Column

The **Aneg** column can hold one of the following values:

- **No** :  
The port does not run any kind of aneg.
- **Yes** :  
The port runs clause 28 or 37 (1G aneg) or it's a special 10G aneg (for 10G PHYs).
- **Yes (Cl73)** :  
The port runs clause 73 aneg.
- **Unknown** :  
No SFP is plugged in.

### 3.1.2. 'Media Type' Column

Previously, the Media Type column showed the configured media type ( `rj45` , `sfp` , or `dual` ).

Now, it displays the actually used media type. It can hold one of two values:

- 'RJ45':  
The port is a pure copper port with an RJ45 connector or it is a dual media port with media type configured as 'rj45' or it is a dual media port with media type configured as 'dual', but no SFP is plugged in. So if media type is configured as `dual` and **no** SFP is plugged in, `RJ45` will be shown!
- 'SFP':  
The port is a pure SFP port with an SFP cage or it is a dual media port with media type configured as 'sfp' or it is a dual media port with media type configured as 'dual' and an SFP is plugged in.

### 3.1.3. SFP Family Column

The SFP Family column can hold one of the following values:

- N/A :  
The port is a copper port.
- None :  
No SFP is plugged in.
- <speed> Optical :  
E.g. `10G Optical` means this is a 10G Optical SFP.
- <speed> CuSFP :  
E.g. `1G CuSFP` means this is a 1G copper SFP.
- <speed> DAC :  
E.g. `10G DAC` means this is a 10G DAC cable.
- <speed> CuBP :  
E.g. `10G CuBP` means this is a copper backplane port, so there is no SFP, but Serdes are connected back-to-back.
- <speed> Unknown :  
E.g. `1G Unknown` means an SFP is plugged in to a 1G port, but the application is not able to read the SFP type from the SFP's ROM.

### 3.1.4. Operational Warnings Column

The Operational Warnings column indicates various, *potential* problems on a port. Some may cause the port never to get link whereas others may cause link to become unstable, whereas yet others may cause traffic to be disturbed.

First, let's see which operational warnings that currently can be shown:

- This dual media port does not support CuSFPs :  
Some dual media ports (currently only Luton26) do not support CuSFPs. You will not be able to get link.

- **CuSFPs require speed auto :**  
A port is configured to a forced speed and it is loaded with a CuSFP. This is not allowed, because the application cannot change the configuration of the SFP's internal PHY. You will not be able to get link.
- **This SFP does not support the configured, forced speed :**  
The forced speed is either lower or higher than the speeds the SFP supports. You will not be able to get link.
- **The port's minimum speed is higher than the SFP's maximum speed :**  
The nominal speed of the SFP is smaller than the lowest speed supported by the port. This could e.g. be a 100Mbps SFP in a 25G port, because our 25G ports only support speeds down to 1G.
- **SFP's nominal speed is higher than actual speed, which may cause instability :**  
Many SFPs work in many speeds, but there are also some that only work in their nominal speed. This warning is merely a kind of disclaimer that the SFP is not running at its nominal speed (e.g. a 1G SFP in a port configured to forced 100 Mbps or a 10G SFP that auto-negotiates to 1G).
- **This SFP cannot run clause 73 aneg (which is forced) :**  
You have forced the port to run clause 73 aneg (with `speed force-clause-73`), but the SFP you have inserted is not a 10G or 25G SFP.
- **SFP type cannot be determined on this interface. Instability can be expected :**  
The SFP's ROM cannot be read. This is a typical issue on Luton26's last three front ports (Gi 1/22-24), because of a board layout bug. When this happens, the application loads the same kind of SFP driver that would be loaded in case of a CuBP port. This SFP driver may or may not work, so there is absolutely no guarantee that you will get link and get error free communication.
- **SFP is not readable. Please replace or expect stability :**  
For some reason, the SFP's ROM returns garbage when read, typically because the SFP is of an older brand that doesn't live up to the standards for reading it (through I2C).
- **SFP does not support half duplex :**  
The speed is configured to a fixed speed and duplex is configured to half, but the inserted SFP does not support half duplex (it's a 100Mbps SFP).

**NOTE**

In the Web GUI, a new column called **Warning** is added to the Port Configuration page. If the port has operational warnings, the LED for that port will turn Yellow. Otherwise it will be gray (turned off).

## 3.2. Show Detailed Interface Status

As shown in the previous section, 2021.03's `show interface status` comes with two, nested new optional arguments.

Here's an example of the (current) output you get by showing detailed status on a 25G port loaded with a 25G DAC cable, which is therefore running clause 73.

*Show Detailed Status*

```
# show interface 25g 1/1 status
```

Interface	Mode	Speed	Aneg	SFP Family	Link	Operational Warnings
25G 1/1	Enabled	Auto	Yes (Cl73)	25G DAC	25Gfdx	

```
# show interface 25g 1/1 status details
25GigabitEthernet 1/1:
Configuration:
  Description:
  Mode:                Enabled
  Speed/Duplex:         Auto
  Media-type:           SFP
  Flowcontrol:          Off
  Max. Frame:           10240 bytes
  Excessive Collision:  Discard
  Frame Length Check:   Disabled
  FEC Mode:             Auto
Status:
  Aneg:                Yes (Cl73)
  Link:                25Gfdx
  Operational Warnings: None
  FEC Mode:            RS-FEC (Reed-Solomon/Clause 108)
  SFP Family:          25G DAC
  SFP Vendor Name:     FS
  SFP Vendor P/N:      S28-PC02
  SFP Vendor S/N:      C1904175718-2
  SFP Vendor Revision:
  SFP Date Code:       190504
  SFP Transceiver:     25GBASE-CR(-S)
Clause 73 Aneg Status:
  Completed:           Yes
  Time since start:    18314 seconds
  Training status:     OK
  Training time:       69 ms
```

The configuration that used to be in `show interface status` is now put in the beginning of the details status. This may not be so useful, because you always can see the entire port configuration by doing this:

*Show Port Configuration*

```
# show running-config feature port all-defaults
...
interface 25GigabitEthernet 1/1
media-type sfp
fec auto
speed auto
duplex full
flowcontrol off
mtu 10240
no excessive-restart
no frame-length-check
no shutdown
no description
...
```

The next part of the details status is actual status. It shows many of the same fields as does the overview ( `show interface status` ), but also the type of FEC that is actually being used along with details of the SFP plugged in.

If the port is not an SFP port, the SFP details will not be shown. Also, if the port doesn't support FEC, the current FEC mode will not be shown.

The last section in the detailed status contains Clause 73 aneg status. This section is only shown if the port is currently using clause 73.

Here, you can see whether clause 73 aneg is complete, how long time since clause 73 aneg was started, and how long it took to train it.

From time to time, experts like additional info on the clause 73 training, so these experts can see this additional info by adding the `clause-73` keyword to the command:

#### Show Detailed Clause 73 Status

```
# show interface 25g 1/1 status details clause-73
25GigabitEthernet 1/1:
Configuration:
Description:
Mode: Enabled
Speed/Duplex: Auto
Media-type: SFP
Flowcontrol: Off
Max. Frame: 10240 bytes
Excessive Collision: Discard
Frame Length Check: Disabled
FEC Mode: Auto
Status:
Aneg: Yes (Cl73)
Link: 25Gfdx
Operational Warnings: None
FEC Mode: RS-FEC (Reed-Solomon/Clause 108)
SFP Family: 25G DAC
SFP Vendor Name: FS
SFP Vendor P/N: S28-PC02
SFP Vendor S/N: C1904175718-2
SFP Vendor Revision:
SFP Date Code: 190504
SFP Transceiver: 25GBASE-CR(-S)
Clause 73 Aneg Status:
Completed: Yes
Time since start: 18838 seconds
Training Results:
LP CM1 MAX/END: 9/5
LP C0 MAX/END: 48/42
LP CP1 MAX/END: 17/7
BER Count CM1: 0 0 0 0 0 0 0 0 0
BER Count C0: 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0
BER Count CP1: 0 0 0 0 0 0 0 0 0 0 0 0 0 6 12 83 175 315
Eye Height CM1: 0 5 9 10 10 8 6 9 9
Eye Height C0: 0 11 11 9 9 10 10 12 12 12 11 12 10 12 11 11 9 10 12 11 11 10 12 10 10 10
11 8 11 10 9 10 10 10 10 10 8 11 12 10 11 12 10 11 11 10 10 10
Eye Height CP1: 0 3 7 8 9 11 12 10 11 10 11 11 0 0 0 0 0
Training status: OK
Training time: 69 ms
Current eye height: 11
Tx Equalizer settings:
LD CM (tap_dly): 14
LD C0 (amplitude): 72
LD CP (tap_adv): 4
Rx CTLE settings:
LD VGA: 5
LD EDC: 4
LD EQR: 13
```

These details may come in handy in case of problems linking up with clause 73 aneg running.

## 4. New CLI Debug Commands

Two new debug commands are added that allows for troubleshooting port issues.

### 4.1. debug show interface state

The first one is for showing the entire internal state of a given port. It may look like nonsense, but is gold for MCHP engineers. Please attach the output if you see something fishy happening with the ports.

Example:

*Show Internal Port State*

```
# debug show interface 25GigabitEthernet 1/1 state
25G 1/1:
Static caps:      <AUTONEG 1G_FDX 2_5G_FDX 5G_FDX 10G_FDX 25G_FDX FLOW_CTRL SD_ENABLE SFP_DETECT
SFP_ONLY>
SFP caps:         <AUTONEG 1G_FDX 2_5G_FDX 5G_FDX 10G_FDX 25G_FDX FLOW_CTRL SD_ENABLE SFP_DETECT
SFP_ONLY>
_conf:            {enable = 1, media_type = SFP, speed = AUTO, adv_dis = <HDX 10M 100M>, fdx = 1,
flow_control = 0, pfc = "00000000", max_length = 10240, exc_col_cont = 0, frame_length_chk = 0,
force_clause_73 = 0, fec_mode = auto, power_mode = Nominal, dscr = ""}
_phy_conf:        {enable = 0, speed = UNDEFINED, fdx = 0, flow_control = 0, adv_dis = <>}
_sfp_conf:         {enable = 1, speed = 1G, flow_control = 0}
_port_conf:        {enable = 1, if_type = SFI, speed = 25G, fdx = 1, flow_control = {obey = 0, gen =
0, pfc = "00000000"}, mtu = 10240, frame_length_chk = 0, exc_col_cont = 0, serdes = {media_type = DAC
(Direct attached copper) cable, unspecified length}, pcs = 0}
_port_status:      {link = 0, speed = UNDEFINED, aneg_method = Clause 73, fdx = 1, fiber = 1, sfp_type
= DAC, sfp_speeds = 10M - 25G, aneg = {obey = 0, gen = 0}, has_kr = 1, mac_if = SFI, oper_warnings = <>}
_phy_status:       {link = 0, speed = UNDEFINED, fdx = 0, aneg = {obey = 0, gen = 0}, copper = 0, fiber
= 0}
_sfp_status:       {link = 0, speed = 10G, fdx = 1, los = 0, aneg = {obey = 0, gen = 0}}
_sfp_dev_status:   {tx_fault = 0, los = 0, present = 1}
_kr_conf:          {aneg = {enable = 1, adv_25g = 1, adv_10g = 1, adv_5g = 1, adv_2g5 = 1, adv_1g = 1,
r_fec_req = 0, rs_fec_req = 1, next_page = 0}, train = {enable = 1, no_remote = 0, use_ber_cnt = 0,
test_mode = 0, test_repeat = 0}}
_fec_conf:         {r_fec = 0, rs_fec = 0}
_kr_status:        {aneg = {complete = 0, active = 0, speed_req = UNDEFINED, fec_change_req = 0,
r_fec_enable = 0, rs_fec_enable = 0}, train = {complete = 0, ob_tap_result = {cm = 0, cp = 0, c0 =
80}, frame_sent = 0, frame_errors = 0}, fec = {r_fec_enable = 0, rs_fec_enable = 0,
corrected_block_cnt = 0, uncorrected_block_cnt = 0}}
PHY dev:          No
_phy_media_if:     N/A
SFP dev:           Yes ({name = FS, pn = S28-PC02, rev = , sn = C1904175718-2, date = 190504, tr =
25GBASE-CR(-S), conn = Copper Pigtail (DAC)})
_phy_speed:        UNDEFINED
_sfp_speed:        25G
_port_speeds:      1G-25G
_sfp_must_run_kr:  Yes
_sfp_may_run_kr:   Yes
_phy_turned_on:    No
_sfp_turned_on:    Yes
_may_load_sfp:     Yes
_is_100m_sfp:      No
```

## 4.2. debug sfp dump

The second is for showing the contents of the SFP ROM.

### NOTE

If you think the application misjudges the type of SFP you have inserted, please dump the SFP's ROM and attach it to the bug report.

#### Dump SFP ROM Contents

```
# debug sfp dump interface 25g 1/1
25G 1/1:
A0h:
0000: 03 04 21 00 00 00 00 00-04 00 00 00 ff 00 00 00 ...!.
0010: 00 00 02 00 46 53 20 20-20 20 20 20 20 20 20 20 ....FS
0020: 20 20 20 20 0d 00 00 00-53 32 38 2d 50 43 30 32 ....S28-PC02
0030: 20 20 20 20 20 20 20 20-20 20 20 20 03 52 00 47 .....R.G
0040: 00 00 00 00 43 31 39 30-34 31 37 35 37 31 38 2d ....C1904175718-
0050: 32 20 20 20 31 39 30 35-30 34 20 20 00 00 00 80 2 190504 ....
0060: 80 00 11 cc ff ff ff ff-ff 50 96 70 18 cd e7 0a .....P.p....
0070: 14 e8 b6 00 00 00 00 00-00 00 00 00 f3 fc 52 00 .....R.
0080: 43 4f 50 51 41 41 34 4a-41 41 33 37 2d 30 39 36 COPQAA4JAA37-096
0090: 30 2d 30 32 56 30 32 20-01 00 46 00 00 00 00 c9 0-02V02 ..F....
00a0: 00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00 .....
00b0: 00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00 .....
00c0: 53 46 50 2d 48 32 35 47-42 2d 43 55 32 4d 20 20 SFP-H25GB-CU2M
00d0: 20 20 20 20 30 38 00 00-00 00 00 00 00 00 00 b3 08.....
00e0: 00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00 .....
00f0: 00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00 .....

A2h: Not available
```

## 5. New Debug Trace Groups

In the old implementation, it was very hard to see when a port linked up or down, because there were no explicit trace for enabling that.

With the 2021.03 implementation, a new trace group called 'callbacks' has been added.

With this, it is possible to see e.g. if the port link flaps.

To enable it, do the following:

#### Show Link Up/Down Traces

```
# debug trace module level port callbacks info
I port/callbacks 05:32:10 30.223,929 197/port_listener_change_notify#120: 25G 1/2 (54): link = 0
I port/callbacks 05:32:10 30.554,482 197/port_listener_change_notify#120: 25G 1/1 (53): link = 0
I port/callbacks 05:32:13 33.673,168 197/port_listener_change_notify#120: 25G 1/2 (54): link = 1
I port/callbacks 05:32:14 34.654,443 197/port_listener_change_notify#120: 25G 1/1 (53): link = 1
I port/callbacks 05:32:14 34.677,472 197/port_listener_change_notify#120: 25G 1/2 (54): link = 0
I port/callbacks 05:32:15 35.654,427 197/port_listener_change_notify#120: 25G 1/2 (54): link = 1
```

Another interesting trace group is for clause 73:



*Show Clause 73 Training Time Traces*

```
# debug trace module level kr base info  
I kr/base 05:34:48 196/KR_poll_v3#619: 25G 1/1 (53): Training completed (71 ms)  
I kr/base 05:34:48 196/KR_poll_v3#619: 25G 1/2 (54): Training completed (77 ms)
```

Do not attempt to enable debug trace for clause 73 (only info trace), because that may cause the auto-negotiation to fail.