Configuring the Card Mode

This chapter lists the supported configurations and the procedures to configure the card mode on the line cards.

1.2T and 1.2TL Line Cards

The following section describes the supported configurations and procedures to configure the card modes on the line cards.

Card Modes

The line cards support module and slice configurations.

The line cards have two trunk ports (0 and 1) and 12 client ports (2 through 13) each. You can configure the line card in two modes:

- Muxponder—In this mode, both trunk ports are configured with the same trunk rate. The client-to-trunk mapping is in a sequence.
- Muxponder slice—In this mode, each trunk port is configured independent of the other with different trunk rates. The client-to-trunk mapping is fixed. For Trunk 0, the client ports are 2 through 7. For Trunk 1, the client ports are 8 through 13.

Sub 50G Configuration

You can configure the sub 50G or coupled mode on the line card only in the muxponder mode. The following table displays the port configuration for the supported data rates.

Trunk Data Rate (per trunk)	Total Configured Data rate	Card Support	Trunk Ports	Client Ports for Trunk 0 (100G)	Shared Client Port (50G per trunk)	Client Ports for Trunk 1 (100G)
150G	300G	1.2T, 1.2TL	0, 1	2	3	4
250G	500G	1.2T	0, 1	2, 3	4	5, 6

Trunk Data Rate (per trunk)	Total Configured Data rate	Card Support	Trunk Ports	Client Ports for Trunk 0 (100G)	Shared Client Port (50G per trunk)	Client Ports for Trunk 1 (100G)
350G	700G	1.2T	0, 1	2, 3, 4	5	6, 7, 8
450G	900G	1.2T	0, 1	2, 3, 4, 5	6	7, 8, 9, 10
550G	1.1T	1.2T	0, 1	2, 3, 4, 5, 6	7	8, 9, 10, 11, 12

Note In all x50G configurations, client traffic on the middle port is affected with ODUK-BDI and LF alarms after the **power cycle or link flap** on the trunk side. This issue is raised when the two network lanes work in coupled mode and move from low to high power. To solve this issue, create a new frame either at the near-end or far-end by performing **shut** or **no shut** of the trunk ports.

Coupled Mode Restrictions

The following restrictions apply to the coupled mode configuration:

- Both trunk ports must be configured with the same bits-per-symbol or baud rate and must be sent over same fiber and direction.
- The chromatic dispersion must be configured to the same value for both trunk ports.
- When trunk internal loopback is configured, it must be done for both trunk ports. Configuring internal loopback on only one trunk results in traffic loss.
- Fault on a trunk port of a coupled pair may cause errors on all clients including those running only on the unaffected trunk port.

Supported Data Rates

The following data rates are supported on the line card.

In R7.1.1 and later releases, you can configure the client port to OTU4 in both the muxponder and muxponder slice modes. In muxponder slice mode, both the slices must be configured with either OTU4 or 100GE Ethernet client rates in R7.1.1. LLDP drop, L1 encryption, and AINS are not supported on the OTU4 configuration.

The following table displays the client and trunk ports that are enabled for the muxponder configuration.

Trunk Data Rate	Card Support	Client Data Rate (100GE, OTU4)	Trunk Ports	Client Ports
200	1.2T	100GE, OTU4	0, 1	2, 3, 4, 5
300	1.2T	100GE, OTU4	0, 1	2, 3, 4, 5, 6, 7
400	1.2T	100GE, OTU4	0, 1	2, 3, 4, 5, 6, 7, 8, 9
500	1.2T	100GE, OTU4	0, 1	2, 3, 4, 5, 6, 7, 8, 9, 10, 11

Trunk Data Rate	Card Support	Client Data Rate (100GE, OTU4)	Trunk Ports	Client Ports
600	1.2T	100GE, OTU4	0, 1	2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13

The following table displays the client and trunk ports that are enabled for the muxponder slice 0 configuration.

Trunk Data Rate	Card Support	Client Data Rate	Trunk Ports	Client Ports
100	1.2T, 1.2TL	100, OTU4	0	2
200	1.2T	100, OTU4	0	2, 3
300	1.2T	100, OTU4	0	2, 3, 4
400	1.2T	100, OTU4	0	2, 3, 4, 5
500	1.2T	100, OTU4	0	2, 3, 4, 5, 6
600	1.2T	100, OTU4	0	2, 3, 4, 5, 6, 7

The following table displays the client and trunk ports that are enabled for the muxponder slice 1 configuration.

Trunk Data Rate	Card Support	Client Data Rate	Trunk Ports	Client Ports
100	1.2T, 1.2TL	100, OTU4	1	8
200	1.2T	100, OTU4	1	8, 9
300	1.2T	100, OTU4	1	8, 9, 10
400	1.2T	100, OTU4	1	8, 9, 10, 11
500	1.2T	100, OTU4	1	8, 9, 10, 11, 12
600	1.2T	100, OTU4	1	8, 9, 10, 11, 12, 13

All configurations can be accomplished by using appropriate values for client bitrate and trunk bitrate parameters of the **hw-module** command.

The following table displays the trunk parameter ranges for the 1.2T card.

Trunk Payload	FEC	Min BPS	Max BPS	Min GBd	Max GBd
200G	27%	2	4.40625	31.51	69.43
300G	27%	2.8984375	6	34.7175497	71.8681352
400G	27%	3.8671875	6	46.2900663	71.8197392
500G	27%	4.8281250	6	57.8625828	71.9068991
600G	15%	5.2578125	-	-	71.9552971

To configure the BPS, see Configuring the BPS, on page 8.

Configuring the Card Mode

You can configure the line card in the module (muxponder) or slice configuration (muxponder slice).

To configure the card in the muxponder mode, use the following commands.

configure

hw-module location *location* **mxponder client-rate {100GE** | **OTU4}**

hw-module location *location* **mxponder trunk-rate** {**50G** | **100G150G** | **200G** | **250G** | **300G** | **350G** | **400G** | **450G** | **500G** | **550G** | **600G** }

commit

To configure the card in the muxponder slice mode, use the following commands.

configure

hw-module location *location* **mxponder-slice** *mxponder-slice-number* **client-rate 100GE**

hw-module location *location* **mxponder-slice trunk-rate** { **200G** | **300G** | **400G** | **500G** | **600G** }

commit

Examples

The following is a sample in which the card is configured in the muxponder mode with a 550G trunk payload.

```
RP/0/RP0/CPU0:ios#config
Tue Oct 15 01:24:56.355 UTC
RP/0/RP0/CPU0:ios(config)#hw-module location 0/1 mxponder client-rate 100GE
RP/0/RP0/CPU0:ios(config)#hw-module location 0/1 mxponder trunk-rate 550G
RP/0/RP0/CPU0:ios(config)#commit
```

The following is a sample in which the card is configured in the muxponder mode with a 500G trunk payload.

```
RP/0/RP0/CPU0:ios#config
Sun Feb 24 14:09:33.989 UTC
RP/0/RP0/CPU0:ios(config)#hw-module location 0/2 mxponder client-rate OTU4
RP/0/RP0/CPU0:ios(config)#hw-module location 0/2 mxponder trunk-rate 500G
RP/0/RP0/CPU0:ios(config)#commit
```

The following is a sample in which the card is configured in the muxponder slice 0 mode with a 500G trunk payload.

```
RP/0/RP0/CPU0:ios#config
RP/0/RP0/CPU0:ios(config)#hw-module location 0/1 mxponder-slice 0 client-rate 100GE
RP/0/RP0/CPU0:ios(config)#hw-module location 0/1 mxponder-slice 0 trunk-rate 500G
RP/0/RP0/CPU0:ios(config)#commit
```

The following is a sample in which the card is configured in the muxponder slice 1 mode with a 400G trunk payload.

```
RP/0/RP0/CPU0:ios#config
RP/0/RP0/CPU0:ios(config)#hw-module location 0/1 mxponder-slice 1 client-rate 100GE
RP/0/RP0/CPU0:ios(config)#hw-module location 0/1 mxponder-slice 1 trunk-rate 400G
RP/0/RP0/CPU0:ios(config)#commit
```

Verifying the Card Configuration

```
RP/0/RP0/CPU0:ios#show hw-module location 0/2 mxponder
Fri Mar 15 11:48:48.344 IST

Location:             0/2
Client Bitrate:       100GE
Trunk  Bitrate:       500G
Status:               Provisioned
LLDP Drop Enabled:    FALSE
Client Port            Mapper/Trunk Port   CoherentDSP0/2/0/0  CoherentDSP0/2/0/1
                     Traffic Split Percentage

HundredGigECtrlr0/2/0/2  ODU40/2/0/0/1                100                   0
HundredGigECtrlr0/2/0/3  ODU40/2/0/0/2                100                   0
HundredGigECtrlr0/2/0/4  ODU40/2/0/0/3                100                   0
HundredGigECtrlr0/2/0/5  ODU40/2/0/0/4                100                   0
HundredGigECtrlr0/2/0/6  ODU40/2/0/0/5                100                   0
HundredGigECtrlr0/2/0/7  ODU40/2/0/1/1                  0                 100
HundredGigECtrlr0/2/0/8  ODU40/2/0/1/2                  0                 100
HundredGigECtrlr0/2/0/9  ODU40/2/0/1/3                  0                 100
HundredGigECtrlr0/2/0/10 ODU40/2/0/1/4                  0                 100
HundredGigECtrlr0/2/0/11 ODU40/2/0/1/5                  0                 100
```

The following is a sample ouput of the coupled mode configuration where the shared client port is highlighted.

```
RP/0/RP0/CPU0:ios#show hw-module location 0/1 mxponder
Tue Oct 15 01:25:57.358 UTC

Location:             0/1
Client Bitrate:       100GE
Trunk  Bitrate:       550G
Status:               Provisioned
LLDP Drop Enabled:    FALSE
Client Port           Mapper/Trunk Port    CoherentDSP0/1/0/0 CoherentDSP0/1/0/1
                   Traffic Split Percentage

HundredGigECtrlr0/1/0/2    ODU40/1/0/0/1             100                   0
HundredGigECtrlr0/1/0/3    ODU40/1/0/0/2             100                   0
HundredGigECtrlr0/1/0/4    ODU40/1/0/0/3             100                   0
HundredGigECtrlr0/1/0/5    ODU40/1/0/0/4             100                   0
HundredGigECtrlr0/1/0/6    ODU40/1/0/0/5             100                   0
HundredGigECtrlr0/1/0/7    ODU40/1/0/0/6              50                  50
HundredGigECtrlr0/1/0/8    ODU40/1/0/1/1               0                 100
HundredGigECtrlr0/1/0/9    ODU40/1/0/1/2               0                 100
HundredGigECtrlr0/1/0/10   ODU40/1/0/1/3               0                 100
HundredGigECtrlr0/1/0/11   ODU40/1/0/1/4               0                 100
HundredGigECtrlr0/1/0/12   ODU40/1/0/1/5               0                 100
```

The following is a sample ouput of all the muxponder slice 0 configurations.

```
RP/0/RP0/CPU0:ios#show hw-module location 0/1 mxponder-slice  0
Fri Mar 15 06:04:18.348 UTC

Location:             0/1
Slice ID:             0
Client Bitrate:       100GE
Trunk  Bitrate:       500G
Status:               Provisioned
LLDP Drop Enabled:    FALSE
Client Port                     Mapper/Trunk Port          CoherentDSP0/1/0/0
                                Traffic Split Percentage

HundredGigECtrlr0/1/0/2         ODU40/1/0/0/1                     100
HundredGigECtrlr0/1/0/3         ODU40/1/0/0/2                     100
```

```
HundredGigECtrlr0/1/0/4         ODU40/1/0/0/3                      100
HundredGigECtrlr0/1/0/5         ODU40/1/0/0/4                      100
HundredGigECtrlr0/1/0/6         ODU40/1/0/0/5                      100
```

The following is a sample ouput of all the muxponder slice 1 configurations.

```
RP/0/RP0/CPU0:ios#show hw-module location 0/1 mxponder-slice 1
Fri Mar 15 06:11:50.020 UTC

Location:             0/1
Slice ID:             1
Client Bitrate:       100GE
Trunk  Bitrate:       400G
Status:               Provisioned
LLDP Drop Enabled:    TRUE
Client Port                     Mapper/Trunk Port          CoherentDSP0/1/0/1
                                Traffic Split Percentage

HundredGigECtrlr0/1/0/8         ODU40/1/0/1/1                      100
HundredGigECtrlr0/1/0/9         ODU40/1/0/1/2                      100
HundredGigECtrlr0/1/0/10        ODU40/1/0/1/3                      100
HundredGigECtrlr0/1/0/11        ODU40/1/0/1/4                      100
```

The following is a sample ouput of the muxponder slice 1 configuration with client configured as OTU4.

```
RP/0/RP0/CPU0:ios#sh hw-module location 0/0 mxponder-slice 1

Wed Mar 11 13:59:11.073 UTC

Location:             0/0
Slice ID:             1
Client Bitrate:       OTU4
Trunk  Bitrate:       200G
Status:               Provisioned
Client Port                     Peer/Trunk Port            CoherentDSP0/0/0/1
                              Traffic Split Percentage
OTU40/0/0/8                     ODU40/0/0/1/1                      100
OTU40/0/0/9                     ODU40/0/0/1/2                      100
```

Use the following command to clear alarm statistics on the optics or coherent DSP controller.

clear counters controller *controllertype R/S/I/P*

The following is a sample in which the alarm statistics are cleared on the coherent DSP controller.

```
RP/0/RP0/CPU0:ios#show controller coherentDSP 0/1/0/0
Tue Jun 11 05:15:12.540 UTC

Port                                       : CoherentDSP 0/1/0/0
Controller State                           : Up
Inherited Secondary State                  : Normal
Configured Secondary State                 : Normal
Derived State                              : In Service
Loopback mode                              : None
BER Thresholds                             : SF = 1.0E-5  SD = 1.0E-7
Performance Monitoring                     : Enable

Alarm Information:
LOS = 1 LOF = 1 LOM = 0
OOF = 1 OOM = 1 AIS = 0
IAE = 0 BIAE = 0        SF_BER = 0
SD_BER = 2      BDI = 2 TIM = 0
FECMISMATCH = 0 FEC-UNC = 0
Detected Alarms                            : None
```

```
Bit Error Rate Information
PREFEC  BER                                     : 8.8E-03
POSTFEC BER                                     : 0.0E+00

TTI :
        Remote hostname                         : P2B8
        Remote interface                        : CoherentDSP 0/1/0/0
        Remote IP addr                          : 0.0.0.0

FEC mode                                        : Soft-Decision 15

AINS Soak                                       : None
AINS Timer                                      : 0h, 0m
AINS remaining time                             : 0 seconds
RP/0/RP0/CPU0:ios#clear counters controller coherentDSP 0/1/0/0
Tue Jun 11 05:17:07.271 UTC
All counters are cleared
RP/0/RP0/CPU0:ios#show controllers coherentDSP 0/1/0/1
Tue Jun 11 05:20:55.199 UTC

Port                                            : CoherentDSP 0/1/0/1
Controller State                                : Up
Inherited Secondary State                       : Normal
Configured Secondary State                      : Normal
Derived State                                   : In Service
Loopback mode                                   : None
BER Thresholds                                  : SF = 1.0E-5  SD = 1.0E-7
Performance Monitoring                          : Enable

Alarm Information:
LOS = 0 LOF = 0 LOM = 0
OOF = 0 OOM = 0 AIS = 0
IAE = 0 BIAE = 0        SF_BER = 0
SD_BER = 0      BDI = 0 TIM = 0
FECMISMATCH = 0 FEC-UNC = 0
Detected Alarms                                 : None

Bit Error Rate Information
PREFEC  BER                                     : 1.2E-02
POSTFEC BER                                     : 0.0E+00

TTI :
        Remote hostname                         : P2B8
        Remote interface                        : CoherentDSP 0/1/0/1
        Remote IP addr                          : 0.0.0.0

FEC mode                                        : Soft-Decision 15

AINS Soak                                       : None
AINS Timer                                      : 0h, 0m
AINS remaining time                             : 0 seconds
```

Regeneration Mode

In an optical transmission system, 3R regeneration helps extend the reach of the optical communication links by reamplifying, reshaping, and retiming the data pulses. Regeneration helps to correct any distortion of optical signals by converting it to an electrical signal, processing that electrical signal, and then retransmitting it again as an optical signal.

In Regeneration (Regen) mode, the OTN signal is received on a trunk port and the regenerated OTN signal is sent on the other trunk port of the line card and the other way round. In this mode, only the trunk optics controller and coherentDSP controllers are created.

Configuring the Card in Regen Mode

The supported trunk rates for the different cards are:

To configure regen mode on 1.2T, 1.2TL, and 2-QDD-C cards, use the following commands:

configure

hw-module location *location*

regen

trunk-rate *trunk-rate*

commit

exit

Example

Verifying the Regen Mode

The following is a sample to verify the regen mode.

show hw-module location *location* **regen**

```
RP/0/RP0/CPU0:ios#show hw-module location 0/0 regen
Mon Mar 25 09:50:42.936 UTC

Location:             0/0
Trunk  Bitrate:       400G
Status:               Provisioned
East Port              West Port
CoherentDSP0/0/0/0      CoherentDSP0/0/0/1
```

The terms, East Port and West Port are used to represent OTN signal regeneration at the same layer.

Configuring the BPS

You can configure the Bits per Symbol (BPS) to 3.4375 to support 300G trunk configurations on 75 GHz networks using the following commands:

configure

controller optics *R/S/I/P* **bits-per-symbol 3.4375**

commit

The following is a sample in which the BPS is configured to 3.4375.

```
RP/0/RP0/CPU0:ios#configure
Wed Mar 27 14:12:49.932 UTC
RP/0/RP0/CPU0:ios(config)#controller optics 0/3/0/0 bits-per-symbol 3.4375
RP/0/RP0/CPU0:ios(config)#commit
```

Viewing BPS and Baud Rate Ranges

To view the the BPS for a specific range use the following command:

show controller optics *R/S/I/P* **bps-range** *bps-range* **| include** *data-rate* **| include** *fec-type*

```
RP/0/RP0/CPU0:ios#show controllers optics 0/3/0/0 bps-range 3 3.05 | include 300G | include
 SD27
Thu Mar 28 03:01:39.751 UTC
300G            SD27            3.0000000       69.4350994
300G            SD27            3.0078125       69.2547485
300G            SD27            3.0156250       69.0753320
300G            SD27            3.0234375       68.8968428
300G            SD27            3.0312500       68.7192736
300G            SD27            3.0390625       68.5426174
300G            SD27            3.0468750       68.3668671
```

To view the baud for a specific range use the following command:

show controller optics *R/S/I/P* **baud-rate-range** *baud-range* **| include** *data-rate* **| include** *fec-type*

```
RP/0/RP0/CPU0:ios#show controllers optics 0/3/0/0 baud-rate-range 43 43.4 | include 300G |
 include SD27
Thu Mar 28 03:12:36.521 UTC
300G            SD27            4.8046875       43.3545986
300G            SD27            4.8125000       43.2842178
300G            SD27            4.8203125       43.2140651
300G            SD27            4.8281250       43.1441394
300G            SD27            4.8359375       43.0744397
300G            SD27            4.8437500       43.0049648
```