

Interface Diagnostics

You can use two diagnostic tools to test the physical layer connections of interfaces: Loopback testing and bit error rate test (BERT) testing. Loopback testing enables you to verify the connectivity of a circuit. BERT testing enables you to identify poor signal quality on a circuit. This section contains the following topics:

- Configuring Loopback Testing on page 1
- Interface Diagnostics on page 3

Configuring Loopback Testing

Loopback testing allows you to verify the connectivity of a circuit. You can configure any of the following interfaces to execute a loopback test: Aggregated Ethernet, Fast Ethernet, Gigabit Ethernet, E1, E3, NxDS0, serial, SONET/SDH, T1, and T3.

The physical path of a network data circuit usually consists of segments interconnected by devices that repeat and regenerate the transmission signal. The transmit path on one device connects to the receive path on the next device. If a circuit fault occurs in the form of a line break or a signal corruption, you can isolate the problem by using a loopback test. Loopback tests allow you to isolate segments of the circuit and test them separately.

To do this, configure a *line loopback* on one of the routers. Instead of transmitting the signal toward the far-end device, the line loopback sends the signal back to the originating router. If the originating router receives back its own data link layer packets, you have verified that the problem is beyond the originating router. Next, configure a line loopback farther away from the local router. If this originating router does not receive its own data link layer packets, you can assume the problem is on one of the segments between the local router and the remote routing platform's interface card. In this case, the next troubleshooting step is to configure a line loopback closer to the local router to find the source of the problem.

There are several types of loopback testing supported by the JUNOS software, as follows:

- DCE local—Loops packets back on the local DCE.
- DCE remote—Loops packets back on the remote DCE.
- Local—Useful for troubleshooting physical PIC errors. A local loopback loops packets, including both data and timing information, back on the local routing platform's PIC. When you configure a local loopback, the interface transmits packets to the channel services unit (CSU) built into the interface. These packets are transmitted onto the circuit toward the far-end device. The PIC receives back its own transmission and ignores any data sent from the physical circuit and the CSU. To test a local loopback, issue the **show interfaces *interface-name*** command. If PPP keepalives transmitted on the interface are received by the PIC, the **Device Flags** field contains the output **Loop-Detected**.
- Payload—Useful for troubleshooting the physical circuit problems between the local router and the remote router. A payload loopback loops data only (without

clocking information) on the remote routing platform's PIC. With payload loopback, overhead is recalculated.

- Remote—Useful for troubleshooting the physical circuit problems between the local router and the remote router. A remote loopback loops packets, including both data and timing information, back on the remote routing platform's interface card. A router at one end of the circuit initiates a remote loopback toward its remote partner. When you configure a remote loopback, the packets received from the physical circuit and CSU are received by the interface. Those packets are then retransmitted by the PIC back toward the CSU and the circuit. This loopback tests all the intermediate transmission segments.

Table 1 shows the loopback modes supported on the various interface types.

Table 1: Loopback Modes by Interface Type

Interface	Loopback Modes	Usage Guidelines
Aggregated Ethernet, Fast Ethernet, Gigabit Ethernet	Local	Configuring Ethernet Loopback Capability
E1 and E3	Local and remote	Configuring E1 Loopback Capability and Configuring E3 Loopback Capability
NxDS0	Payload	Configuring Channelized E1 IQ and IQE Interfaces, Configuring T1 and NxDS0 Interfaces, Configuring Channelized OC12/STM4 IQ and IQE Interfaces (SONET Mode), Configuring Channelized STM1 IQ and IQE Interfaces, and Configuring Channelized T3 IQ Interfaces
Serial (V.35 and X.21)	Local and remote	Configuring Serial Loopback Capability
Serial (EIA-530)	DCE local, DCE remote, local, and remote	Configuring Serial Loopback Capability
SONET/SDH	Local and remote	Configuring SONET/SDH Physical Interface Properties
T1 and T3	Local, payload, and remote	Configuring T1 Loopback Capability and Configuring T3 Loopback Capability See also Configuring the T1 Remote Loopback Response

To configure loopback testing, include the **loopback** statement:

```
[Unresolved xref] mode;
```

You can include this statement at the following hierarchy levels:

- [edit interfaces *interface-name* aggregated-ether-options]
- [edit interfaces *interface-name* ds0-options]
- [edit interfaces *interface-name* e1-options]
- [edit interfaces *interface-name* e3-options]
- [edit interfaces *interface-name* fastether-options]
- [edit interfaces *interface-name* gigether-options]
- [edit interfaces *interface-name* serial-options]
- [edit interfaces *interface-name* sonet-options]
- [edit interfaces *interface-name* t1-options]
- [edit interfaces *interface-name* t3-options]

Interface Diagnostics

BERT allows you to troubleshoot problems by checking the quality of links. You can configure any of the following interfaces to execute a BERT when the interface receives a request to run this test: E1, E3, T1, T3; the channelized DS3, OC3, OC12, and STM1 interfaces; and the channelized DS3 IQ, E1 IQ, and OC12 IQ interfaces.

A BERT test requires a line loop to be in place on either the transmission devices or the far-end router. The local router generates a known bit pattern and sends it out the transmit path. The received pattern is then verified against the sent pattern. The higher the bit error rate of the received pattern, the worse the noise is on the physical circuit. As you move the position of the line loop increasingly downstream toward the far-end router, you can isolate the troubled portion of the link.

To configure BERT, you must configure the duration of the test, the bit pattern to send on the transmit path, and the error rate to monitor when the inbound pattern is received.

To configure the duration of the test, the pattern to send in the bit stream, and the error rate to include in the bit stream, include the **bert-period**, **bert-algorithm**, and **bert-error-rate** statements, respectively, at the [edit interfaces *interface-name interface-type-options*] hierarchy level:

```
[edit interfaces interface-name interface-type-options]  
bert-algorithm algorithm;  
bert-error-rate rate;  
bert-period seconds;
```

By default, the BERT period is 10 seconds. You can configure the BERT period to last from 1 through 239 seconds on some PICs and from 1 through 240 seconds on other PICs.

rate is the bit error rate. This can be an integer from 0 through 7, which corresponds to a bit error rate from 10^{-0} (1 error per bit) to 10^{-7} (1 error per 10 million bits).

algorithm is the pattern to send in the bit stream. For a list of supported algorithms, enter a ? after the **bert-algorithm** statement; for example:

```
[edit interfaces t1-0/0/0 t1-options]
user@host# set bert-algorithm ?
Possible completions:
pseudo-2e11-o152    Pattern is 2^11 -1 (per 0.152 standard)
pseudo-2e15-o151    Pattern is 2^15 - 1 (per 0.152 standard)
pseudo-2e20-o151    Pattern is 2^20 - 1 (per 0.151 standard)
pseudo-2e20-o153    Pattern is 2^20 - 1 (per 0.153 standard)
...
```

For specific hierarchy information, see the individual interface types.



NOTE: The 4-port E1 PIC supports only the following algorithms:

pseudo-2e11-o152	Pattern is 2^11 -1 (per 0.152 standard)
pseudo-2e15-o151	Pattern is 2^15 - 1 (per 0.151 standard)
pseudo-2e20-o151	Pattern is 2^20 - 1 (per 0.151 standard)
pseudo-2e23-o151	Pattern is 2^23 (per 0.151 standard)

When you issue the **help** command from the CLI, all BERT algorithm options are displayed, regardless of the PIC type, and no commit check is available. Unsupported patterns for a PIC type can be viewed in system log messages.



NOTE: The IQE PICs support only the following algorithms:

all-ones-repeating	Repeating one bits
all-zeros-repeating	Repeating zero bits
alternating-double-ones-zeros	Alternating pairs of ones and zeros
alternating-ones-zeros	Alternating ones and zeros
pseudo-2e9-o153	Pattern is 2^9 -1 (per 0.153 (511 type) standard)
pseudo-2e11-o152	Pattern is 2^11 -1 (per 0.152 and 0.153 (2047 type) standards)
pseudo-2e15-o151	Pattern is 2^15 -1 (per 0.151 standard)
pseudo-2e20-o151	Pattern is 2^20 -1 (per 0.151 standard)
pseudo-2e20-o153	Pattern is 2^20 -1 (per 0.153 standard)
pseudo-2e23-o151	Pattern is 2^23 -1 (per 0.151 standard)
repeating-1-in-4	1 bit in 4 is set
repeating-1-in-8	1 bit in 8 is set
repeating-3-in-24	3 bits in 24 are set

When you issue the **help** command from the CLI, all BERT algorithm options are displayed, regardless of the PIC type, and no commit check is available. Unsupported patterns for a PIC type can be viewed in system log messages.

Table 2 shows the BERT capabilities for various interface types.

Table 2: BERT Capabilities by Interface Type

Interface	T1 BERT	T3 BERT	Comments
E1 or T1	Yes (port 0–3)	Yes (port 0–3)	<ul style="list-style-type: none"> ■ Single port at a time ■ Limited algorithms
E3 or T3	Yes (port 0–3)	Yes (port 0–3)	<ul style="list-style-type: none"> ■ Single port at a time
Channelized OC12	N/A	Yes (channel 0–11)	<ul style="list-style-type: none"> ■ Single channel at a time ■ Limited algorithms ■ No bit count
Channelized STM1	Yes (channel 0–62)	N/A	<ul style="list-style-type: none"> ■ Multiple channels ■ Only one algorithm ■ No error insert ■ No bit count
Channelized T3 and Multichannel T3	Yes (channel 0–27)	Yes (port 0–3 on channel 0)	<ul style="list-style-type: none"> ■ Multiple ports and channels ■ Limited algorithms for T1 ■ No error insert for T1 ■ No bit count for T1

These limitations do not apply to channelized IQ interfaces. For information about BERT capabilities on channelized IQ interfaces, see [\[Unresolved xref\]](#).

Starting and Stopping a BERT Test

Before you can start the BERT test, you must disable the interface. To do this, include the `disable` statement at the `[edit interfaces interface-name]` hierarchy level:

```
[edit interfaces interface-name]
[Unresolved xref];
```

After you configure the BERT properties and commit the configuration, begin the test by issuing the `test interface interface-name interface-type-bert-start` operational mode command:

```
user@host> test interface interface-name interface-type-bert-start
```

The test runs for the duration you specify with the `bert-period` statement. If you wish to terminate the test sooner, issue the `test interface interface-name interface-type-bert-stop` command:

```
user@host> test interface interface-name interface-type-bert-stop
```

For example:

```
user@host> test interface t3-1/2/0 t3-bert-start
user@host> test interface t3-1/2/0 t3-bert-stop
```

To view the results of the BERT test, issue the `show interfaces extensive | find BERT` command:

```
user@host> show interfaces interface-name extensive | find BERT
```

For more information about running and evaluating the results of the BERT procedure, see the *JUNOS System Basics and Services Command Reference*.



NOTE: To exchange BERT patterns between a local routing platform and a remote routing platform, include the `loopback remote` statement in the interface configuration at the remote end of the link. From the local routing platform, issue the `test interface` command.

Example: Configuring Bit Error Rate Testing

Configure a BERT test on a T3 interface. In this example, the run duration lasts for 120 seconds. The configured error rate is 0, which corresponds to a bit error rate of 10^{-0} (1 error per bit). The configured bit pattern of `all-ones-repeating` means that every bit the interface sends is a set to a value of 1.

```
[edit interfaces]
t3-1/2/0 {
  t3-options {
    bert algorithm all-ones-repeating;
    bert-error-rate 0;
    bert-period 120;
  }
}
```