

SWAN Ethernet Test Result v0.1

2021/04/20

Introduction.

This document specifies the details to validate DUT Ethernet performance.
This test will use lperf3.1.3/Ethtool Test tool and Control NB.

1.1. Auto Negotiation Test

Test Environment

- Configure the test setup as shown in block diagram below

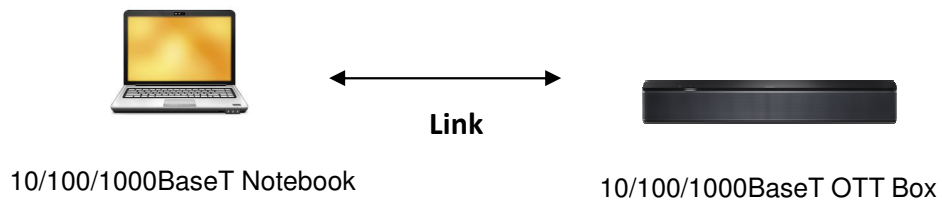


Figure 1: Auto Negotiation Test

Purpose: Verify the DUT responds to management by transmitting the appropriate link signaling.

The performance specified in following paragraphs shall be guaranteed under the following conditions:

HW Version: DVT2 , SW Version: BFX_UA300 10 QT 19.522.17-20201223						
Items #	Description	Result (B6:7E)	Result (B6:37)	Result (B6:3A)	Result (B6:4A)	Result (B6:17)
1	1000BaseT – Full duplex	PASS	PASS	PASS	PASS	PASS
2	1000BaseT – Half duplex	PASS	PASS	PASS	PASS	PASS
3	100BaseT – Full duplex	PASS	PASS	PASS	PASS	PASS
4	100BaseT - Half duplex	PASS	PASS	PASS	PASS	PASS
5	10BaseT – Full duplex	PASS	PASS	PASS	PASS	PASS
6	10BaseT - Half duplex	PASS	PASS	PASS	PASS	PASS

1.2. CRC Error Test

Test Environment

- Configure the test setup as shown in block diagram below

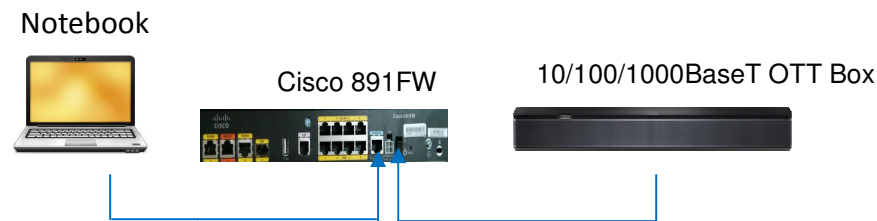


Figure 2: CRC Error Test

The performance specified in following paragraphs shall be guaranteed under the following conditions:

HW Version: DVT2 , SW Version: BFX_UA300 10 QT 19.522.17-20201223						
Items #	Description	Result (B6:7E)	Result (B6:37)	Result (B6:3A)	Result (B6:4A)	Result (B6:17)
1	1000BaseT – Full duplex	PASS	PASS	PASS	PASS	PASS
2	100BaseT – Full duplex	PASS	PASS	PASS	PASS	PASS
3	10BaseT – Full duplex	PASS	PASS	PASS	PASS	PASS

Purpose: Verify the DUT can detect accidental changes to raw data.

■ 1000BaseT :

1. NB:

To run this test , we will open our command processor and input the following information.

“ iperf3.exe -c 192.168.1.2 -t 3600 -w 90k” or

“ iperf3.exe -c 192.168.1.2 -t 3600 -w 1m”

This should look like the following:

```

C:\tool\iperf-3.1.3-win64>iperf3.exe -c 192.168.164.6 -t 3600 -w 1m
Connecting to host 192.168.164.6, port 5201
[ 4] local 192.168.164.9 port 58547 connected to 192.168.164.6 port 5201

[ ID] Interval           Transfer     Bandwidth
[ 4] 0.00-1.00   sec      114 MBytes   957 Mbits/sec
[ 4] 1.00-2.00   sec      113 MBytes   949 Mbits/sec
[ 4] 2.00-3.00   sec      113 MBytes   949 Mbits/sec
[ 4] 3.00-4.00   sec      113 MBytes   949 Mbits/sec
[ 4] 4.00-5.00   sec      113 MBytes   949 Mbits/sec
[ 4] 5.00-6.00   sec      113 MBytes   949 Mbits/sec
[ 4] 6.00-7.00   sec      113 MBytes   949 Mbits/sec
[ 4] 7.00-8.00   sec      113 MBytes   949 Mbits/sec
[ 4] 8.00-9.00   sec      113 MBytes   949 Mbits/sec
[ 4] 9.00-10.00  sec      113 MBytes   949 Mbits/sec
[ 4] 10.00-11.00 sec      113 MBytes   949 Mbits/sec
[ 4] 11.00-12.00 sec      113 MBytes   949 Mbits/sec
[ 4] 12.00-13.00 sec      113 MBytes   949 Mbits/sec

```

2. DUT:

This command will run a iperf test for 72 hours.

“iperf3.exe -s &”

```

-----
Server listening on 5201
-----
Accepted connection from 192.168.164.9, port 58546
[ 7] local 192.168.164.6 port 5201 connected to 192.168.164.9 port 58547
[ ID] Interval           Transfer     Bandwidth
[ 7] 0.00-1.00   sec      112 MBytes   940 Mbits/sec
[ 7] 1.00-2.00   sec      113 MBytes   949 Mbits/sec
[ 7] 2.00-3.00   sec      113 MBytes   949 Mbits/sec
[ 7] 3.00-4.00   sec      113 MBytes   949 Mbits/sec
[ 7] 4.00-5.00   sec      113 MBytes   949 Mbits/sec
[ 7] 5.00-6.00   sec      113 MBytes   949 Mbits/sec
[ 7] 6.00-7.00   sec      113 MBytes   949 Mbits/sec
[ 7] 7.00-8.00   sec      113 MBytes   949 Mbits/sec

```

The test results are as follows:

```

console:/ #
console:/ # ethtool -S eth0 | grep crc
bad_crc_event: 0

```

■ 100MBaseT :

1. NB:

To run this test, we will open our command processor and input the following information.

“iperf3.exe -c 192.168.1.2 -t 3600 -w 20k”

DUT:

The test results are as follows :

```

console:/ #
console:/ # ethtool -S eth0 | grep crc
bad_crc_event: 0

```

■ 10MBaseT :

1. NB:

To run this test, we will open our command processor and input the following information.

“iperf3.exe -c 192.168.1.2 -t 3600 -w 10k”

DUT:

The test results are as follows:

```
console:/ #  
console:/ # ethtool -S eth0 | grep crc  
bad_crc_event: 0
```