



# **Broadcom Ethernet Network Adapter User Guide for Dell Platforms**

## **User Guide**

Broadcom, the pulse logo, NetXtreme, TruTrust, Connecting everything, Avago Technologies, Avago, and the A logo are among the trademarks of Broadcom and/or its affiliates in the United States, certain other countries, and/or the EU.

Copyright © 2020 – 2023 Broadcom. All Rights Reserved.

The term “Broadcom” refers to Broadcom Inc. and/or its subsidiaries. For more information, please visit [www.broadcom.com](http://www.broadcom.com).

Broadcom reserves the right to make changes without further notice to any products or data herein to improve reliability, function, or design. Information furnished by Broadcom is believed to be accurate and reliable. However, Broadcom does not assume any liability arising out of the application or use of this information, nor the application or use of any product or circuit described herein, neither does it convey any license under its patent rights nor the rights of others.

# Table of Contents

<b>1 Regulatory and Safety Approvals</b> .....	8
1.1 Class A Warning Statements.....	9
1.2 Class B Warning Statements.....	10
<b>2 Functional Description</b> .....	11
<b>3 Network Link and Activity Indication</b> .....	24
3.1 BCM957412AXXXX.....	25
3.2 BCM957414AXXXX.....	26
3.3 BCM957416AXXXX.....	27
3.4 BCM957414M4140D.....	28
3.5 BCM957412M4120D.....	29
3.6 BCM957416M4160.....	30
3.7 BCM957412N4120DC.....	31
3.8 BCM957414N4140DC.....	32
3.9 BCM957416N4160DC.....	33
3.10 BCM957504-N425D.....	34
3.11 BCM957504-P425D.....	35
3.12 BCM957508-P2100D.....	36
3.13 BCM957508-N2100D.....	37
3.14 BCM957454-P410SDBT.....	38
3.15 BCM957454-N410SDBT.....	39
<b>4 Features</b> .....	40
4.1 Software and Hardware Features.....	40
4.2 Virtualization Features.....	41
4.3 VXLAN.....	42
4.4 NVGRE/GRE/IP-in-IP/Geneve.....	42
4.5 Stateless Offloads.....	43
4.5.1 IP, TCP, UDP Checksum Offload.....	43
4.5.2 UDP Fragmentation Offload.....	43
4.5.3 TCP Segmentation Offload and Large Send Offload.....	43
4.5.4 Generic Receive Offload (GRO) and Large Receive Offload (LRO).....	43
4.5.5 Header and Data Split.....	43
4.5.6 VLAN Tag Insertion and Removal.....	44
4.5.7 Packet Steering.....	44
4.5.8 Data Center Bridging.....	44
4.6 Priority Flow Control.....	44
4.7 Virtualization Offload.....	45
4.7.1 Multiqueue Support.....	45
4.7.2 KVM/Xen Multiqueue.....	45

4.7.3 Virtual Machine Queue .....	45
4.7.4 Tunneling Offload .....	45
4.8 SR-IOV .....	48
4.8.1 SR-IOV Configuration Support Matrix.....	49
4.9 Network Partitioning (NPAR) .....	49
4.10 Security (BCM575XX Only) .....	49
4.11 RDMA over Converged Ethernet – RoCE.....	50
4.12 VMWare Enhanced Networking Stack (ENS).....	50
4.12.1 Features.....	50
4.12.2 ENS Design Choices .....	50
4.12.3 ENS Performance.....	51
4.12.4 Limitations and Restrictions.....	51
4.13 Supported Combinations .....	51
4.13.1 NPAR, SR-IOV, and RoCE.....	51
4.13.2 NPAR, SR-IOV, and DPDK .....	52
4.14 Unsupported Combinations .....	52
<b>5 Installing the Hardware.....</b>	<b>53</b>
5.1 Safety Precautions.....	53
5.2 System Requirements.....	53
5.2.1 Hardware Requirements.....	53
5.2.2 Memory Requirements .....	53
5.2.3 Preinstallation Checklist.....	54
5.3 Installing the Adapter .....	55
5.4 Connecting the Network Cables .....	55
5.4.1 Validated Cables and Modules .....	55
<b>6 Software Packages and Installation.....</b>	<b>57</b>
6.1 Supported Operating Systems.....	57
6.2 Installing the Linux Driver.....	57
6.2.1 The automated installer installs/updates both the L2 and RoCE drivers. In order to utilize the RoCE feature, see the Manual Driver Installation .....	58
6.2.2 Updating Initramfs.....	58
6.2.3 Linux Ethtool Commands.....	58
6.3 Installing the VMware Driver .....	60
6.4 Installing the Windows Driver.....	60
<b>7 Updating the Firmware .....</b>	<b>61</b>
7.1 Dell Update Package .....	61
7.1.1 Windows .....	61
7.1.2 Linux .....	61
<b>8 Link Aggregation.....</b>	<b>62</b>
8.1 Windows .....	62

8.2 Linux .....	62
8.2.1 Ephemeral Bonds .....	62
8.2.2 Bonding Interface Queries .....	63
<b>9 System-Level Configuration .....</b>	<b>64</b>
9.1 UEFI HII Menu .....	64
9.1.1 Main Configuration Page .....	64
9.1.2 Firmware Image Menu .....	66
9.1.3 Device Configuration Menu .....	67
9.1.4 NIC Configuration .....	70
9.1.5 NIC Partitioning Configuration Menu .....	71
9.2 Auto-Negotiation Configuration .....	75
9.2.1 Operational Link Speed .....	79
9.2.2 Firmware Link Speed .....	79
9.2.3 Auto-Negotiation Protocol .....	79
9.2.4 Windows Driver Settings .....	79
9.2.5 Linux Driver Settings .....	79
9.2.6 ESXi Driver Settings .....	80
9.3 Configuring 200G Link Speeds .....	80
9.3.1 Auto-Negotiation Configuration for 200G .....	81
9.3.2 Forced 200G Configuration .....	82
9.3.3 Configuring Dell Switches .....	82
<b>10 PXE Boot .....</b>	<b>83</b>
10.1 UEFI Mode .....	83
10.1.1 iPXE .....	83
10.1.2 PXE .....	84
10.1.3 Secure Boot .....	85
10.2 PXE Server Configuration .....	86
10.2.1 DHCP Configuration for PXE/iPXE .....	87
10.2.2 TFTP Configuration .....	90
10.2.3 HTTP Configuration .....	92
<b>11 SR-IOV – Configuration and Use Case Examples .....</b>	<b>92</b>
11.1 Enable SR-IOV in BIOS/UEFI and Device .....	92
11.2 Linux Use Case Example: SR-IOV Pass-Through to libvirt Virtual Machine .....	93
11.2.1 Setting MAC Address for the VF .....	93
11.3 Windows SR-IOV Use Case Example .....	94
11.4 VMware SR-IOV Use Case Example .....	94
<b>12 NPAR – Configuration and Use Case Example .....</b>	<b>96</b>
12.1 Features and Requirements .....	96
12.2 Limitations .....	96
12.3 Configuration .....	97

12.4	Reducing NIC Memory Consumption with NPAR .....	100
12.5	Advanced NPAR .....	101
12.5.1	Supported Broadcom Devices .....	101
12.5.2	Supported Operating Systems .....	101
12.5.3	Supported Features and Limitations .....	101
12.5.4	Supported Hardware Configurations .....	102
12.5.5	Required Hardware Settings .....	102
12.6	Ethernet Adapter Operations .....	104
<b>13</b>	<b>Tunneling Configuration Examples</b> .....	<b>105</b>
13.1	Network Diagram .....	105
13.2	VEB and VEPA Modes .....	105
13.2.1	VLAN Configuration .....	105
13.3	VLAN Double Tagging .....	106
13.4	GRE Tunnelling .....	106
13.5	IP-in-IP Tunnelling .....	106
13.6	VXLAN – Configuration and Use Case Examples .....	106
<b>14</b>	<b>RoCE – Configuration and Use Case Examples</b> .....	<b>107</b>
14.1	Enabling RoCE .....	107
14.2	Linux Configuration and Use Case Examples .....	108
14.2.1	Requirements .....	108
14.2.2	Installing Drivers and the RoCE Library .....	108
14.2.3	Verifying RoCE Functionality .....	110
14.2.4	RoCE Connectivity Tests .....	111
14.2.5	RoCE Congestion Control .....	114
14.3	Windows and Use Case Examples .....	127
14.3.1	SMB Direct .....	127
14.4	VMware ESX and Use Case Examples .....	136
14.4.1	Limitations .....	136
14.4.2	BNXT RoCE Driver Requirements .....	136
14.4.3	Installation .....	136
14.4.4	Configuring Paravirtualized RDMA Network Adapters .....	137
14.4.5	Configuring the VM on Linux Guest OS .....	138
<b>15</b>	<b>DCBX – Data Center Bridging</b> .....	<b>139</b>
15.1	QoS Profile – Default QoS Queue Profile .....	139
15.2	DCBX Mode – Enable (IEEE only) .....	140
15.3	DCBX Willing Bit .....	140
<b>16</b>	<b>DPDK – Configuration and Use Case Examples</b> .....	<b>143</b>
16.1	Compiling the Application .....	143
16.2	Running the Application .....	143
16.3	Testpmd Runtime Functions .....	144

---

16.4 Control Functions .....	144
16.5 Display Functions.....	144
16.6 Configuration Functions .....	145
<b>Revision History .....</b>	<b>146</b>

# 1 Regulatory and Safety Approvals

The following sections detail the regulatory approvals, safety approvals, and warning statements for Broadcom Ethernet Network Adapters. See the individual data sheets for the product classification and referenced standard dates.

**Table 1: Regulatory Approvals**

Standard/Country	Certification Type	Compliance
CE/EU	EN 55032 EN 55024/EN 55035 EN 61000-3-2 EN 61000-3-3	CE report and CE sDoC
UKCA/United Kingdom	EN 55032 EN 55024/EN 55035 EN 61000-3-2 EN 61000-3-3	UKCA DoC
FCC/USA	CFR47, Part 15	FCC sDoC and EMC report referencing FCC part 15 regulations
IC/Canada	ICES-003	Report referencing IC standards
ACA/Australia, New Zealand	AS/NZS CISPR 32	sDoC certificate RCM Mark
BSMI/Taiwan	CNS13438, CNS15663	BSMI certificate
MIC/S. Korea	KN 32 and KN 35	Korea certificate MSIP Mark
VCCI/Japan	V-3/VCCI CISPR 32	Copy of VCCI on-line certificate

**Table 2: Safety Approvals**

Item	Applicable Standard	Approval/Certificate
CE/European Union	IEC 62368-1	CB report and certificate
UL/USA	IEC 62368-1 CTUVus UL	UL report and certificate
CSA/Canada	CSA 22.2 No. 950	CSA report and certificate



## 1.1 Class A Warning Statements

### FCC Warning Statement

This device complies with part 15 of the FCC Rules. Operation is subject to the following two conditions:

- This device may not cause harmful interference.
- This device must accept any interference received, including interference that may cause undesired operation.

**NOTE:** This equipment has been tested and found to comply with the limits for a Class A digital device, pursuant to part 15 of the FCC Rules. These limits are designed to provide reasonable protection against harmful interference when the equipment is operated in a commercial environment. This equipment generates, uses, and can radiate radio frequency energy and, if not installed and used in accordance with the instruction manual, may cause harmful interference to radio communications. Operation of this equipment in a residential area is likely to cause harmful interference in which case the user is required to correct the interference at his own expense.

**CAUTION!** Changes or modifications not expressly approved by the manufacturer responsible for compliance could void the user's authority to operate the equipment.

### Japan (VCCI) Warning Statement

この装置は、クラスA機器です。この装置を住宅環境で使用すると電波妨害を引き起こすことがあります。この場合には使用者が適切な対策を講ずるよう要求されることがあります。

VCCI - A

### Korea Warning Statement

이 기기는 업무용(A급)으로 전자파적합기기로서 판매자 또는 사용자는 이 점을 주의하시기 바라며, 가정외의 지역에서 사용하는 것을 목적으로 합니다.

## 1.2 Class B Warning Statements

### FCC Warning Statement

This equipment has been tested and found to comply with the limits for a Class B digital device, pursuant to part 15 of the FCC Rules. These limits are designed to provide reasonable protection against harmful interference in a residential installation. This equipment generates, uses, and can radiate radio frequency energy and, if not installed and used in accordance with the instructions, may cause harmful interference to radio communications. However, there is no guarantee that interference will not occur in a particular installation. If this equipment does cause harmful interference to radio or television reception, which can be determined by turning the equipment off and on, the user is encouraged to try to correct the interference by one or more of the following measures:

- Reorient or relocate the receiving antenna.
- Increase the separation between the equipment and receiver.
- Connect the equipment into an outlet on a circuit different from that to which the receiver is connected.
- Consult the dealer or an experienced radio/TV technician for help.

### Japan (VCCI) Warning Statement

この装置は、クラスB機器です。この装置は、住宅環境で使用することを目的としていますが、この装置がラジオやテレビジョン受信機に近接して使用されると、受信障害を引き起こすことがあります。

取扱説明書に従って正しい取り扱いをして下さい。

VCCI - B

### Korea Warning Statement

이 기기는 가정용 (B급) 전자파적합기기로서 주로 가정에서 사용하는 것을 목적으로 하며, 모든 지역에서 사용할 수 있습니다.

## 2 Functional Description

Broadcom Ethernet Adapters(BCM9574XX and BCM9575XX) family of Ethernet controllers are highly-integrated, full-featured Ethernet LAN controllers optimized for data center and cloud infrastructures. These controllers support 200G/100G/50G/25G/10G/1G in single, dual, or quad-port configurations. These controllers can support up to sixteen lanes of PCIe Gen 3. The BCM9575XX family additionally supports PCIe Gen 4. An extensive set of stateless offloads and virtualization offloads to enhance packet processing efficiency are included to enable low-overhead, high-speed network communications. These NICs are described in [Table 3](#).

**Table 3: Functional Description**

Network Interface Card	Description
<b>BCM957412A4120D/BCM957412M4120D/BCM957412N4120DC</b>	
Speed	Dual-Port 10 Gb/s Ethernet
PCIe	Gen 3 x8 <sup>a</sup>
Interface	SFP+ for 10 Gb/s
Device	Broadcom BCM57412 10 Gb/s MAC controller with integrated dual-channel 10 Gb/s SFI transceiver.
NDIS Name	Broadcom NetXtreme-E Series Dual-Port 10Gb SFP+ Ethernet PCIe Adapter
UEFI Name	Broadcom Dual 10Gb SFP+ Ethernet
<b>BCM957414A4141D/BCM957414M4140D/BCM957414N4140DC</b>	
Speed	Dual-Port 25 Gb/s or 10 Gb/s Ethernet
PCIe	Gen 3 x8 <sup>a</sup>
Interface	SFP28 for 25 Gb/s and SFP+ for 10 Gb/s
Device	Broadcom BCM57414 25 Gb/s MAC controller with integrated dual-channel 25 Gb/s SFI transceiver.
NDIS Name	Broadcom NetXtreme- E Series Dual-Port 25 Gb SFP28 Ethernet PCIe Adapter
UEFI Name	Broadcom Dual 25 Gb SFP 28 Ethernet
<b>BCM957504-NGM250D</b>	
Speed	Quad Port 25 Gb/s Ethernet
PCIe	Gen4 x16
Interface	Fabric connector for 25 Gb/s
Device	Broadcom BCM57504 100G Gb/s MAC controller with integrated quad-channel 25 GB/s SFI transceiver.
NDIS Name	Broadcom NetXtreme E-Series Quad-Port 25 Gb Ethernet Network Daughter Card
UEFI Name	Broadcom Advance Quad 25 GB Ethernet
<b>BCM957416A4160D/BCM957416M4160/BCM957416N4160DC</b>	
Speed	Dual-Port 10GBASE-T Ethernet
PCIe	Gen 3 x8 <sup>a</sup>
Interface	RJ-45 for 10 Gb/s
Device	Broadcom BCM57416 10 Gb/s MAC controller with integrated dual-channel 10GBASE-T transceiver.
NDIS Name	Broadcom NetXtreme-E Series Dual-Port 10GBASE-T Ethernet PCIe Adapter
UEFI Name	Broadcom Dual 10GBASE-T Ethernet

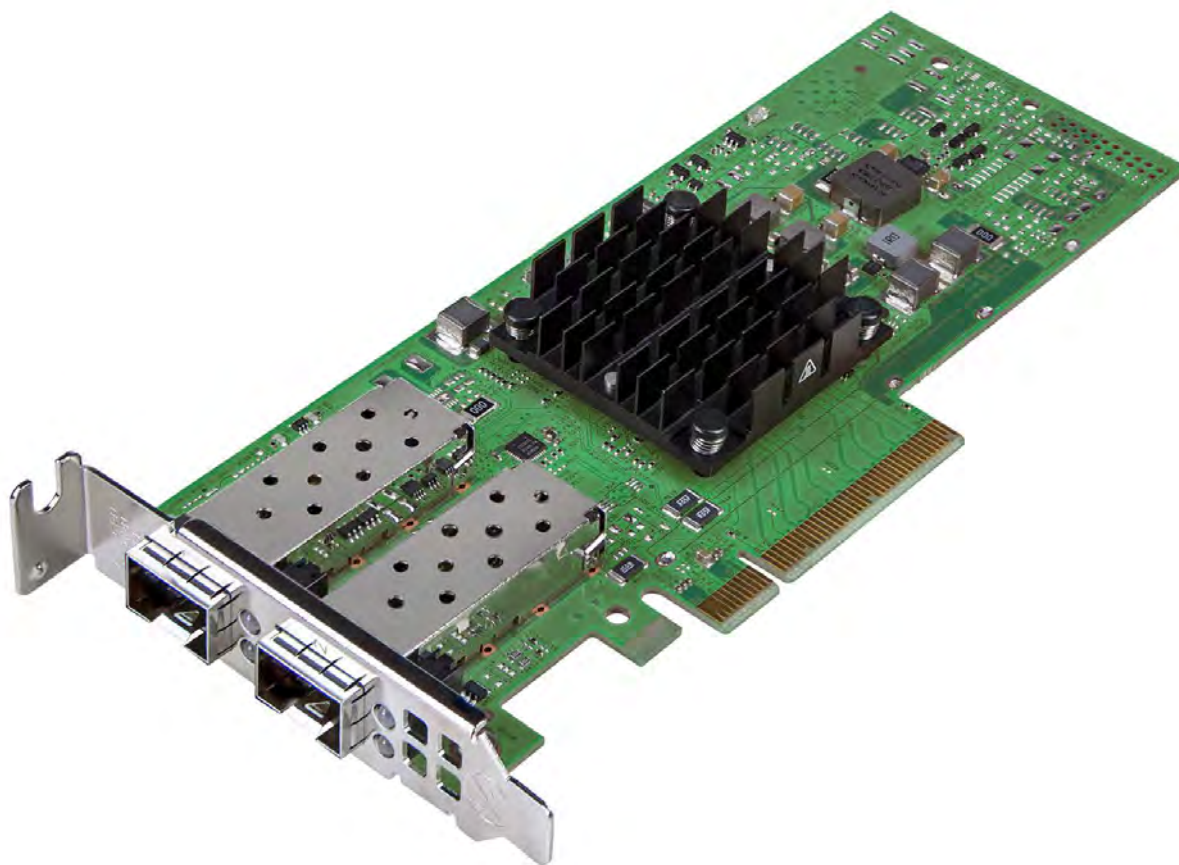
**Table 3: Functional Description (Continued)**

Network Interface Card	Description
<b>BCM957504-P425D/BCM957504-N425D</b>	
Speed	Quad-Port 25 Gb/s Ethernet
PCIe	Gen4 x16
Interface	SFP28 for 25 Gb/s
Device	Broadcom BCM57504 25G Gb/s MAC controller with integrated quad-channel 25 Gb/s SFI transceiver.
NDIS Name	Broadcom NetXtreme-E P425D BCM57504 4x25G SFP28 PCIe Ethernet
UEFI Name	Broadcom BCM57504 4x25G SFP28 PCIe
<b>BCM957508-P2100D/BCM57508-N2100D</b>	
Speed	Dual-Port 100 Gb/s Ethernet
PCIe	Gen4 x16
Interface	QSFP28 for 100 Gb/s
Device	Broadcom BCM57508 100 Gb/s MAC controller with integrated dual-channel 100 Gb/s SFI transceiver.
NDIS Name	Broadcom NetXtreme-E P2100D BCM57508 2x100G QSFP PCIe Ethernet
UEFI Name	BCM57508 2x100G QSFP PCIe
<b>BCM957454-P410SDBT/BCM57545-N410SDBT</b>	
Speed	Quad-Port 10GBASE-T Ethernet
PCIe	Gen 3 x 8
Interface	10GBASE-T
Device	Network adapter with Quad-Port 10GBaseT Ethernet PCIe Gen3 x8
NDIS Name	Broadcom NetXtreme-E P410SDBT BCM57454 4x10G BT PCIE Ethernet
UEFI Name	Broadcom BCM57454 4x10G BT PCIE

- a. The NIC supports PCIe 3, 2, and 1 speeds, however, PCIe Gen 3 is recommended to achieve nominal throughput when 2 ports of 25G links transmit and receive traffic at the same time.

**Figure 1: BCM957412A4120D Network Interface Card**

**NOTE:** [Figure 1](#) shows the standard-profile bracket installed. The surface markings of the component may not reflect the product received. Broadcom reserves the right to change any component on the printed circuit board with the same functionality.

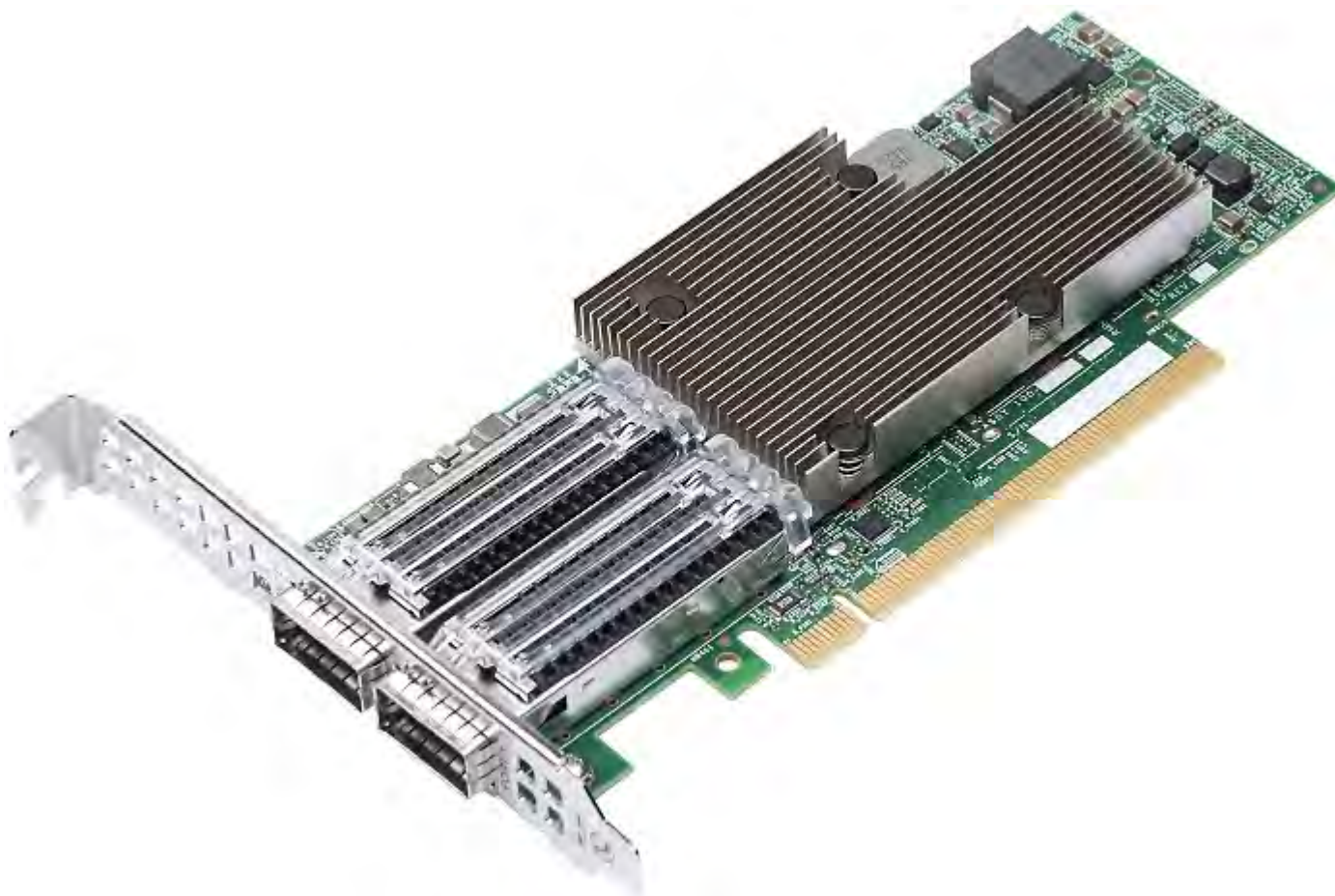
**Figure 2: BCM957414A4141D Network Interface Card**

**NOTE:** [Figure 2](#) shows the low-profile bracket installed. The surface markings of the component may not reflect the product received. Broadcom reserves the right to change any component on the printed circuit board with the same functionality.

**Figure 3: BCM957416A4160D Network Interface Card**

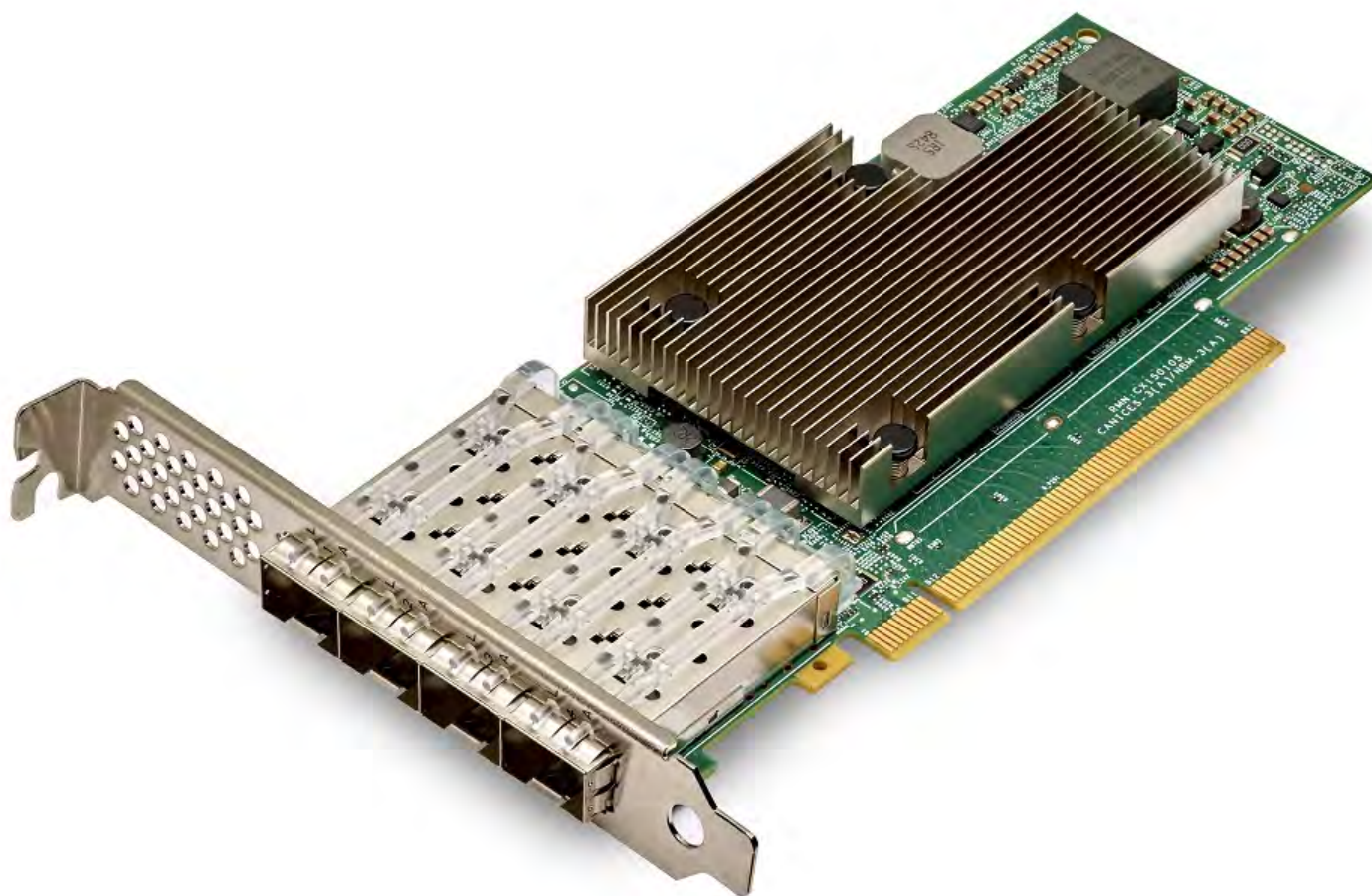
**NOTE:** [Figure 3](#) shows the low-profile bracket installed. The surface markings of the component may not reflect the product received. Broadcom reserves the right to change any component on the printed circuit board with the same functionality.

**Figure 4: BCM957508-P2100D Network Interface Card**



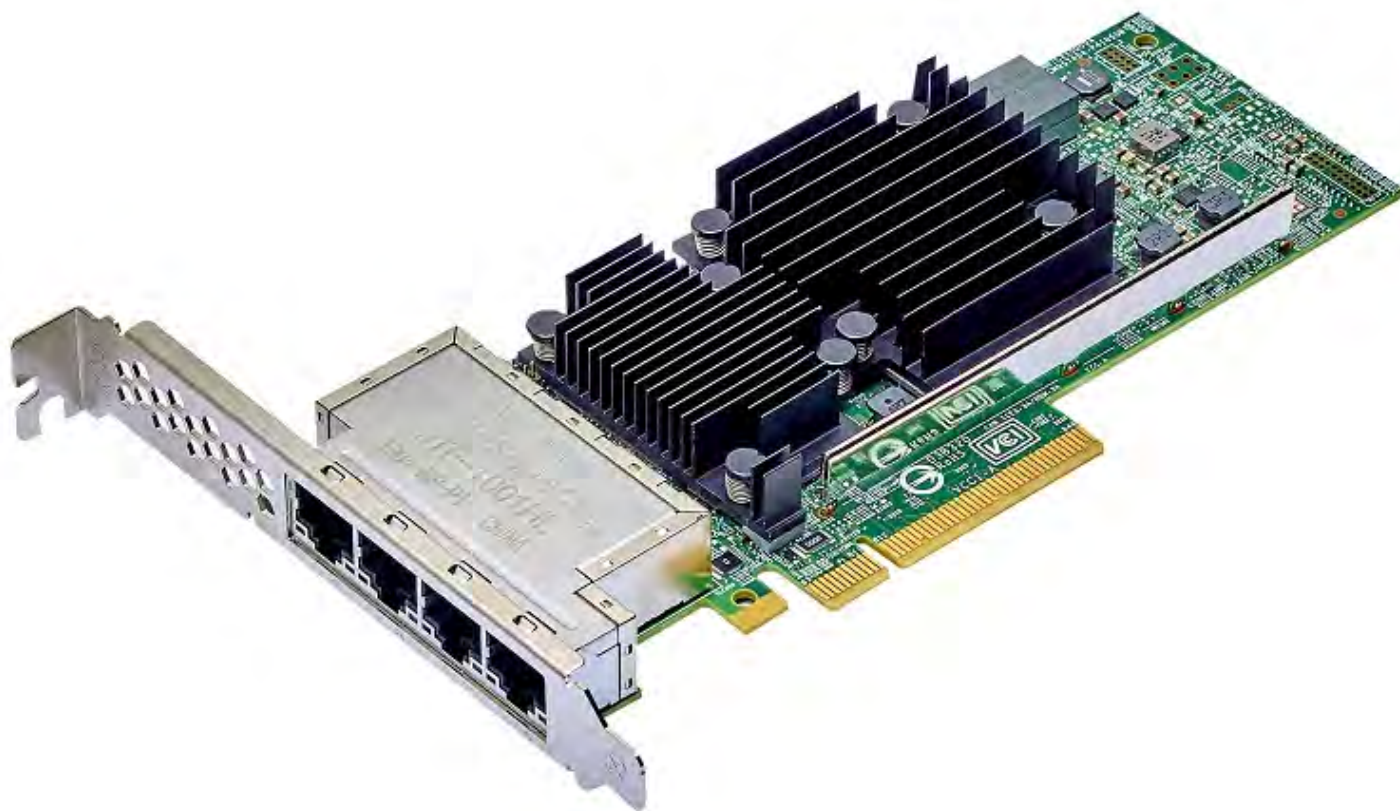
**NOTE:** The surface markings of the component may not reflect the product received. Broadcom reserves the right to change any component on the printed circuit board with the same functionality.



**Figure 5: BCM957504-P425D Network Interface Card**

**NOTE:** [Figure 3](#) shows the standard-profile bracket installed. The surface markings of the component may not reflect the product received. Broadcom reserves the right to change any component on the printed circuit board with the same functionality.

Figure 6: BCM957454-P410SDBT Network Interface Card



**NOTE:** The surface markings of the component may not reflect the product received. Broadcom reserves the right to change any component on the printed circuit board with the same functionality.

Figure 7: BCM957414M4140D Network Daughtercard (rNDC)

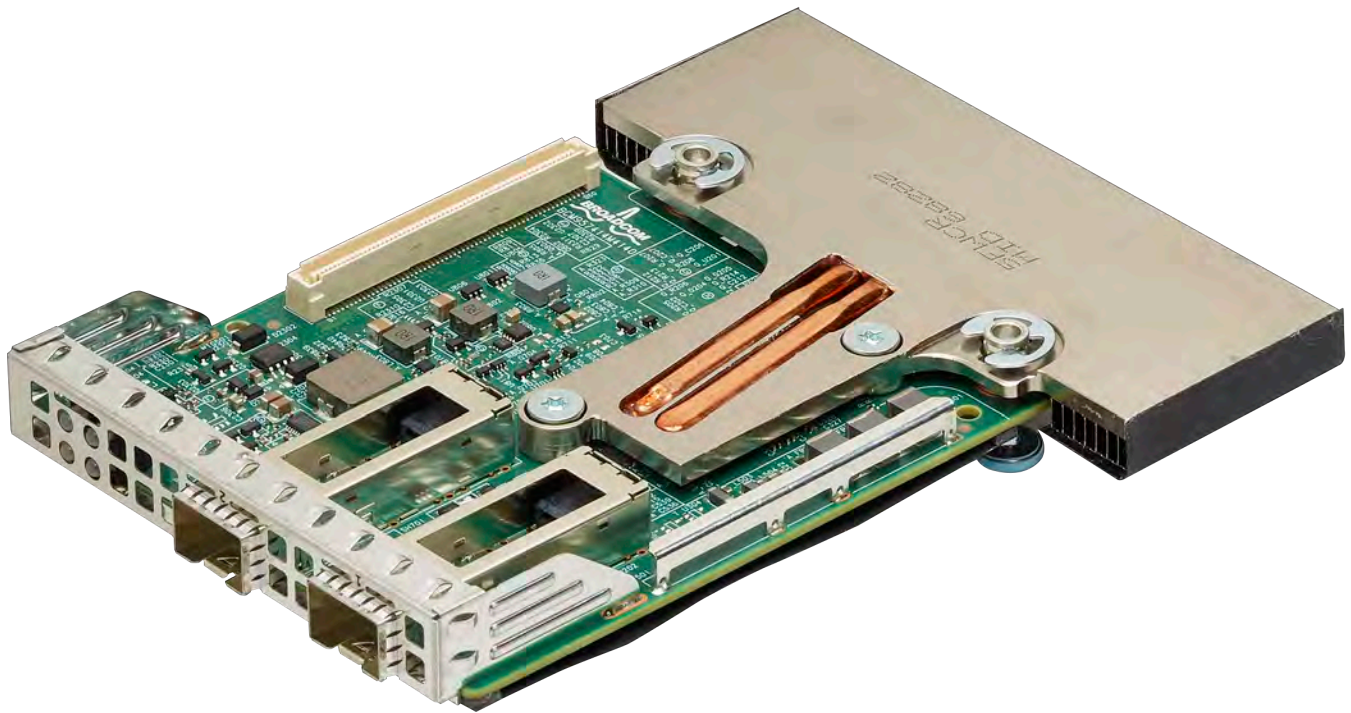
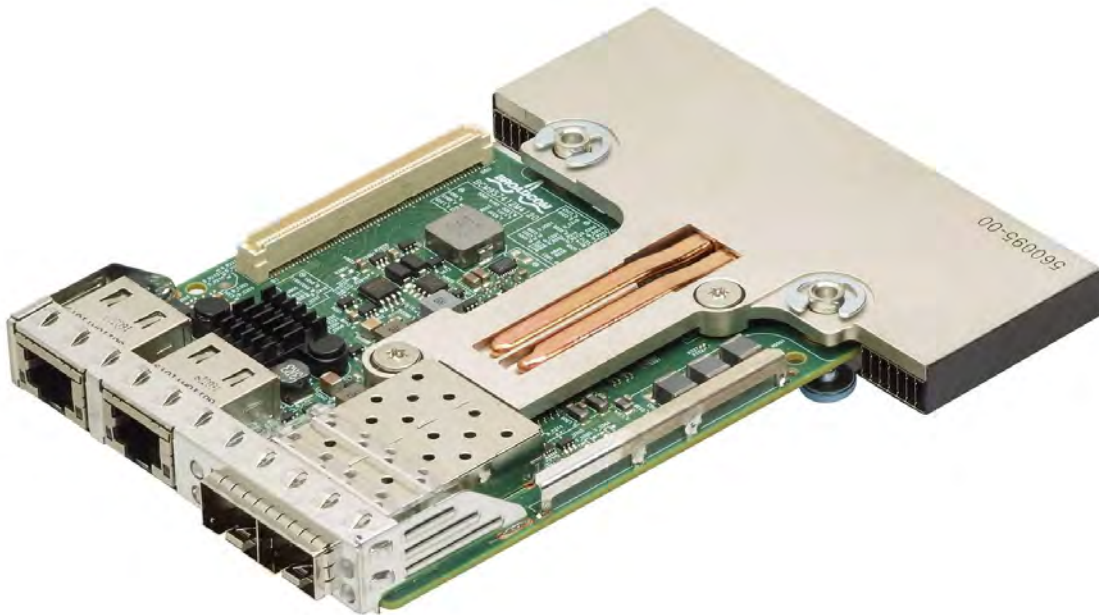
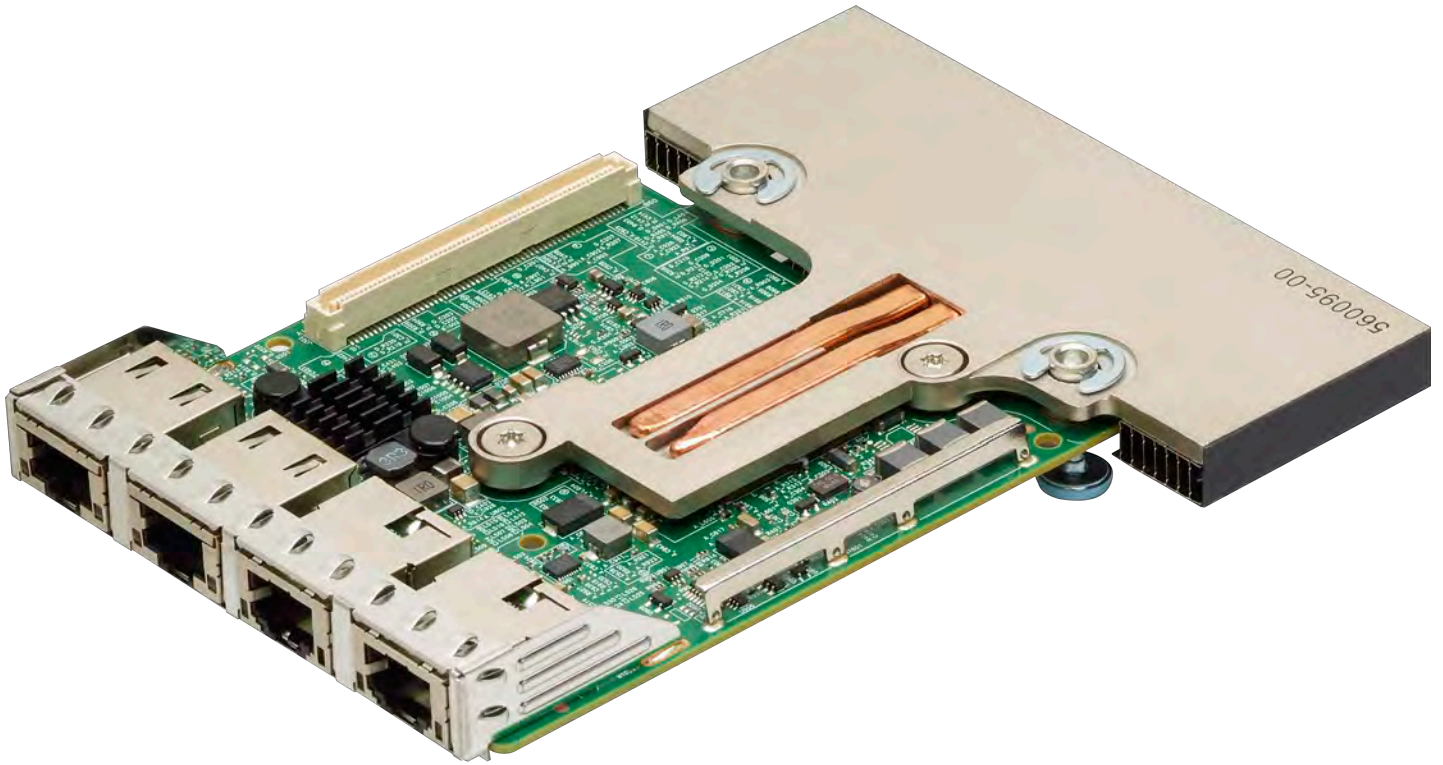


Figure 8: BCM957412M4120D Network Daughtercard (rNDC)



**Figure 9: BCM95416M4160 Network Daughtercard (rNDC)**



**Figure 10: BCM957412N4120 OCP 3.0 SFF Card**



**Figure 11: BCM957414N4140 OCP 3.0 SFF Card**



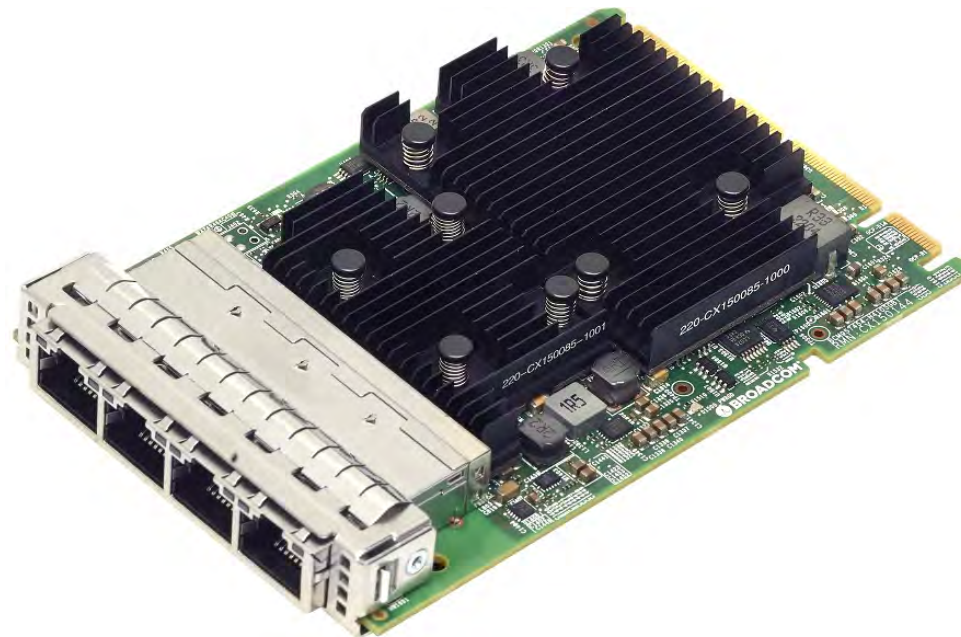
**Figure 12: BCM957416N4160 OCP 3.0 SFF Card**



Figure 13: BCM957504-N425D OCP 3.0 SFF Card



Figure 14: BCM957454-N410SDBT OCP 3.0 SFF Card



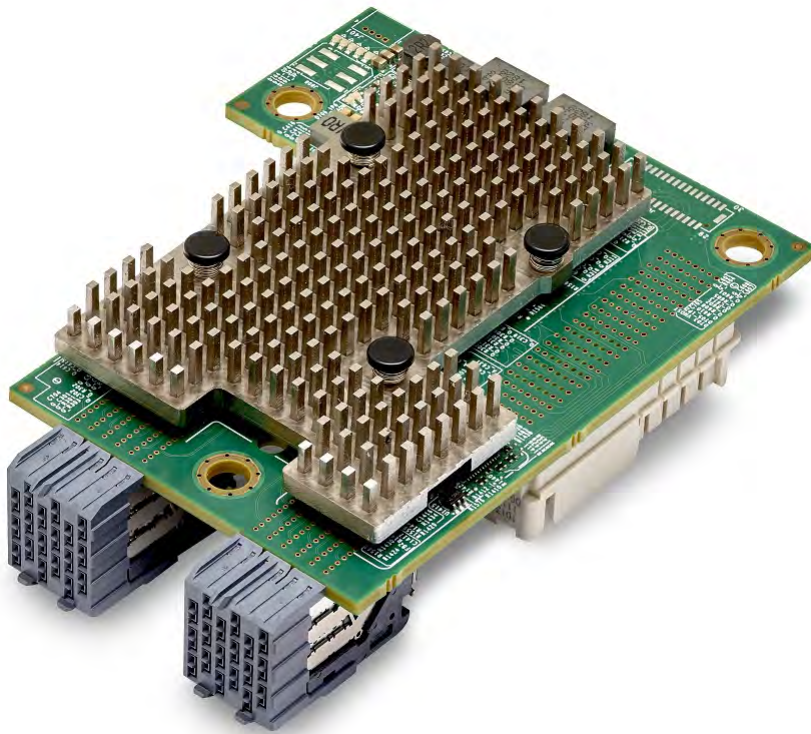
**Figure 15: BCM957508-N2100D OCP 3.0 SFF Card**



**Figure 16: BCM957504-NGM250D Mezzanine Card**



Figure 17: BCM957508-NGM2100D Mezzanine Card



### 3 Network Link and Activity Indication

Ethernet connections, the state of the network link, and activity are indicated by the LEDs on the rear connector as shown in [Table 4](#).

See the individual board data sheets for specific media design.

**Table 4: Network Link and Activity Indicated by Port LEDs**

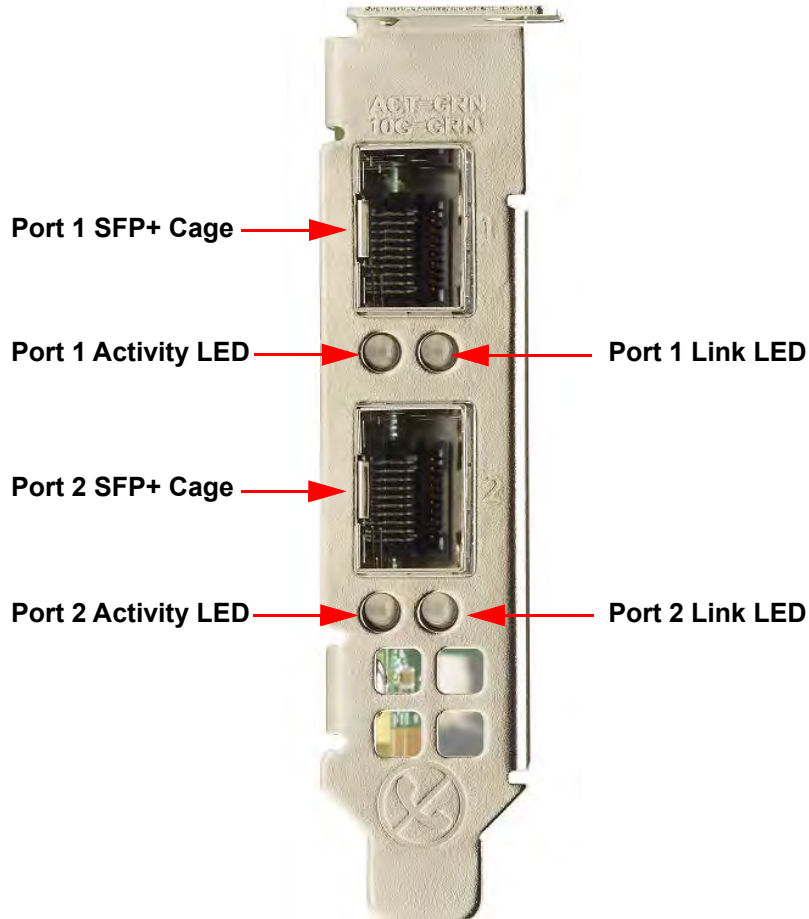
Port LED	LED Appearance	Network State
Link LED	Off	No link (cable disconnected)
	Continuously illuminated	Link
Activity LED	Off	No network activity
	Blinking	Network activity



### 3.1 BCM957412AXXXX

The SFP+ port has two LEDs to indicate traffic activities and link speed. The LEDs are shown in [Figure 18](#) and described in [Table 5](#).

**Figure 18: BCM957412AXXXX Activity and Link LED Locations**



**NOTE:** [Figure 18](#) shows the low-profile bracket installed. The surface markings of the component may not reflect the product upon receipt. Broadcom reserves the right to change any component on the printed circuit board with the same functionality.

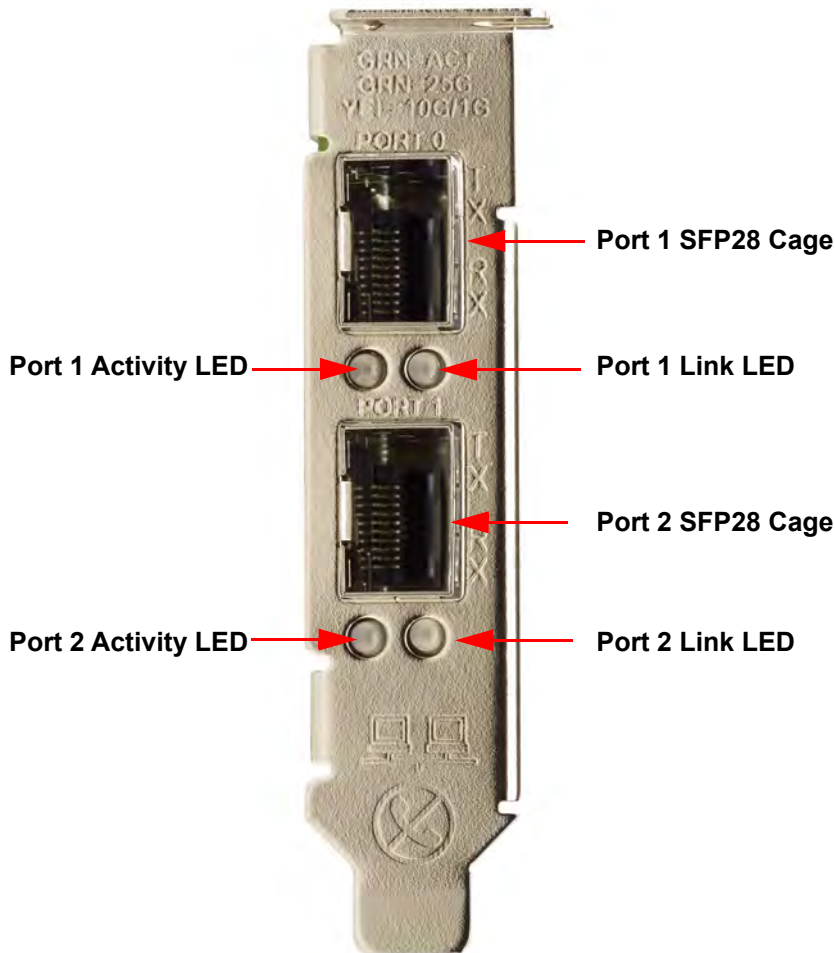
**Table 5: BCM957412AXXXX Activity and Link LED Locations**

LED Type	Color/Behavior	Note
Activity	Off	No Activity
	Green blinking	Traffic Flowing Activity
Link	Off	No Link
	Green	Linked at 10 Gb/s
	Yellow	Linked at 1 Gb/s

## 3.2 BCM957414AXXX

The SFP28 port has two LEDs to indicate traffic activities and link speed. The LEDs are shown in [Figure 19](#) and described in [Table 6](#).

**Figure 19: BCM957414AXXX Activity and Link LED Locations**



**NOTE:** [Figure 19](#) shows the low-profile bracket installed. The surface markings of the component may not reflect the product upon receipt. Broadcom reserves the right to change any component on the printed circuit board with the same functionality.

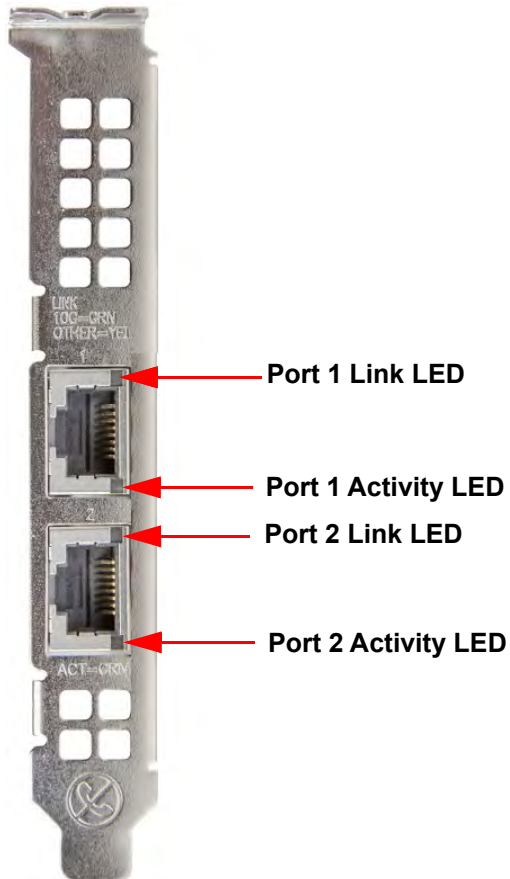
**Table 6: BCM957414AXXX Activity and Link LED Locations**

LED Type	Color/Behavior	Note
Activity	Off	No Activity
	Green blinking	Traffic Flowing Activity
Link	Off	No Link
	Green	Linked at 25 Gb/s
	Yellow	Linked at 10 Gb/s

### 3.3 BCM957416AXXX

Each Ethernet interface has a link LED to indicate Link status and an activity LED to indicate data traffic. The LEDs are shown in [Figure 20](#) and described in [Table 7](#).

**Figure 20: BCM957416AXXX Activity and Link LED Locations**



**Table 7: BCM957416AXXX Activity and Link LED Locations**

LED Type	Color/Behavior	Notes
Activity	Off	No Activity
	Green blinking	Traffic Flowing Activity
Link	Off	No Link
	Green	Linked at 10 Gb/s
	Amber	Linked at 1 Gb/s

### 3.4 BCM957414M4140D

The SFP28 port has two LEDs to indicate traffic activities and link speed. The LEDs are shown in [Figure 21](#) and described in [Table 8](#).

Figure 21: BCM957414M4140D Network Daughtercard (rNDC) Activity and Link LED Locations

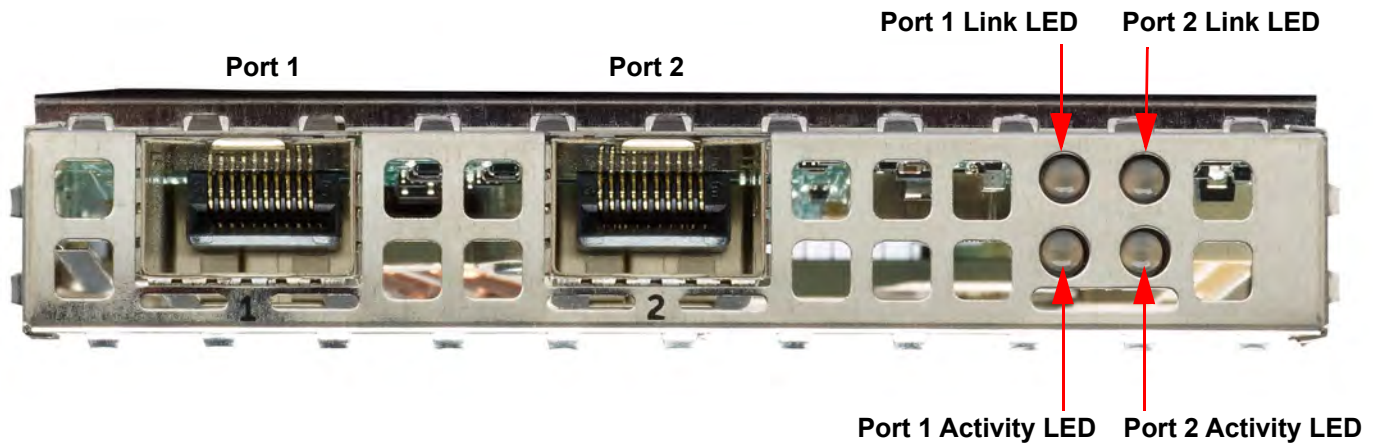


Table 8: BCM957414M4140D Network Daughtercard (rNDC) Activity and Link LED Locations

LED Type	Color/Behavior	Notes
Activity	Off	No Activity
	Green blinking	Traffic Flowing Activity
Link	Off	No Link
	Green	Linked at 25 Gb/s
	Yellow	Linked at 10 Gb/s

**NOTE:** When a XTY28 transceiver is connected to a 25 Gb/s controller and if the link establishes at 1 Gb/s, the link LED is off and the activity LED blinks.

### 3.5 BCM957412M4120D

This rNDC has SFP+ and RJ-45 ports, each with two LEDs to indicate traffic activities and link speed. The LEDs are shown in [Figure 22](#) and described [Table 9](#) (Ports 1 and 2)/[Table 10](#) (Ports 3 and 4).

Figure 22: BCM957412M4120D Network Daughtercard (rNDC) Activity and Link LED Locations

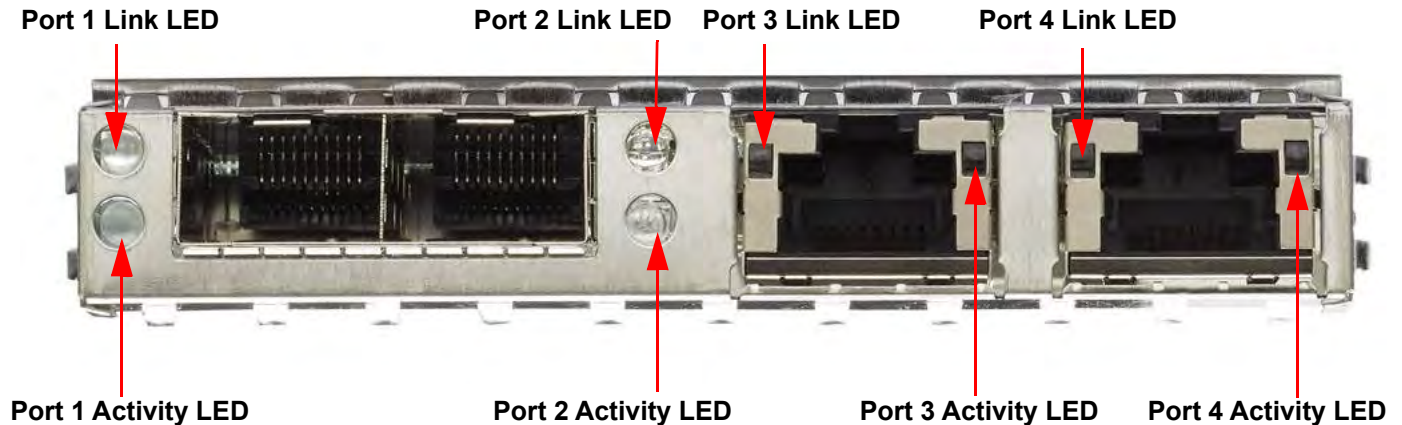


Table 9: BCM957412M4120D Network Daughtercard (rNDC) Activity and Link LED Locations SFP+ Port 1 and 2

LED Type	Color/Behavior	Notes
Activity	Off	No Activity
	Green blinking	Traffic Flowing Activity
Link	Off	No Link
	Green	Linked at 10 Gb/s

**NOTE:** When a XTY28 transceiver is connected to a 10 Gb/s controller and if the link establishes at 1 Gb/s, the link LED is off and the activity LED blinks.

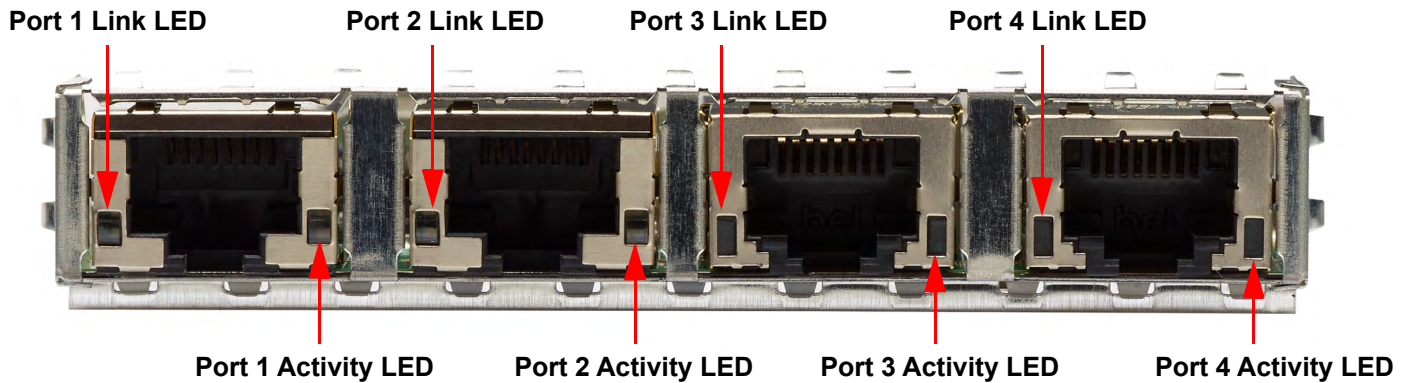
Table 10: 1000BASE-T Port 3 and 4

LED Type	Color/Behavior	Notes
Activity	Off	No Activity
	Green blinking	Traffic Flowing Activity
Link	Off	No Link
	Green	Linked at 1 Gb/s
	Amber	Linked at 10/100 Mb/s

### 3.6 BCM957416M4160

This rNDC has 10GBaseT and 1000BaseT RJ-45 ports, each with two LEDs to indicate traffic activities and link speed. The LEDs are shown in [Figure 23](#) and described [Table 11](#).

**Figure 23: BCM957416M4160 Network Daughtercard (rNDC) Activity and Link LED Locations**



**Table 11: BCM957416M4160 Network Daughtercard (rNDC) Activity and Link LED Locations 10GBASET Port 1 and 2**

LED Type	Color/Behavior	Notes
Activity	Off	No Activity
	Green blinking	Traffic Flowing Activity
Link	Off	No Link
	Green	Linked at 10 Gb/s
	Amber	Linked at 1 Gb/s

### 3.7 BCM957412N4120DC

The SFP+ port supports two LEDs to indicate traffic activities and link speed. The LEDs are shown in [Figure 23](#) and described [Table 12](#). Its locations and form factors conform to the OCP 3.0 Design Specification.

Figure 24: BCM957412N4120DC Network Adapter Activity and Link LED Locations



Table 12: BCM957412N4120DC Network Adapter Activity and Link LED Locations

LED Type	Color/Behavior	Notes
Activity	Off	No Activity
	Green blinking	Traffic Flowing Activity
Link	Off	No Link
	Green	Linked at 10 Gb/s

**NOTE:** When a XTY28 transceiver is connected to a 10 Gb/s controller and if the link establishes at 1 Gb/s, the link LED is off and the activity LED blinks.

### 3.8 BCM957414N4140DC

The SFP+ port supports two LEDs to indicate traffic activities and link speed. The LEDs are shown in [Figure 25](#) and described [Table 13](#). Its locations and form factors conform to the OCP 3.0 Design Specification.

Figure 25: BCM957414N4140DC Network Adapter Activity and Link LED Locations

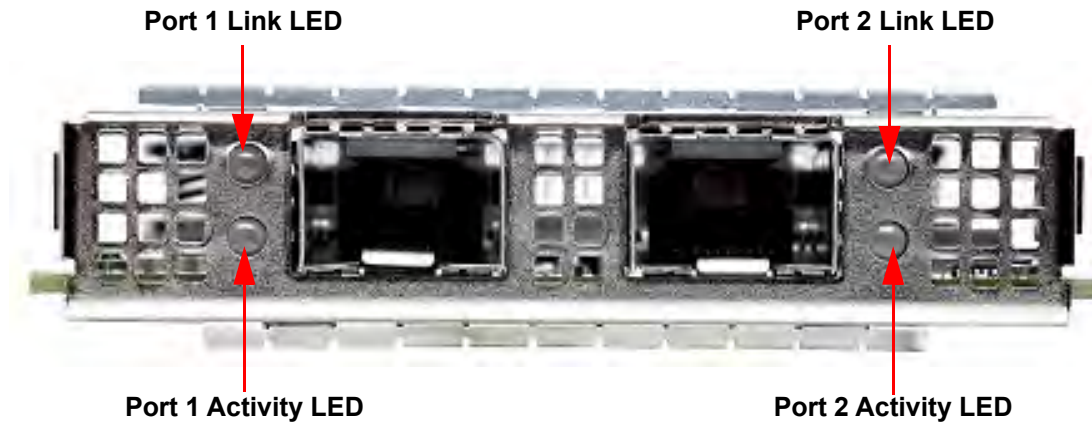


Table 13: BCM957414N4140DC Network Adapter Activity and Link LED Locations

LED Type	Color/Behavior	Notes
Activity	Off	No Activity
	Green blinking	Traffic Flowing Activity
Link	Off	No Link
	Green	Linked at 25 Gb/s
	Amber	Linked at 10 Gb/s

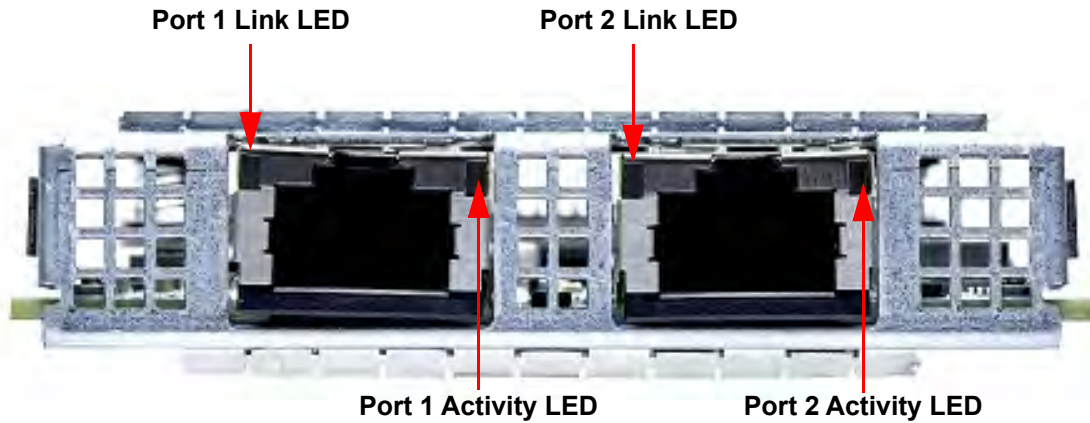
**NOTE:** When a XTY28 transceiver is connected to a 25 Gb/s controller and if the link establishes at 1 Gb/s, the link LED is off and the activity LED blinks.



### 3.9 BCM957416N4160DC

Each Ethernet interface has a link LED to indicate Link status and an activity LED to indicate data traffic. The LEDs are shown in [Figure 26](#) and described [Table 14](#). Its locations and form factors conform to the OCP 3.0 Design Specification.

**Figure 26: BCM957416N4160DC Network Adapter Activity and Link LED Locations**



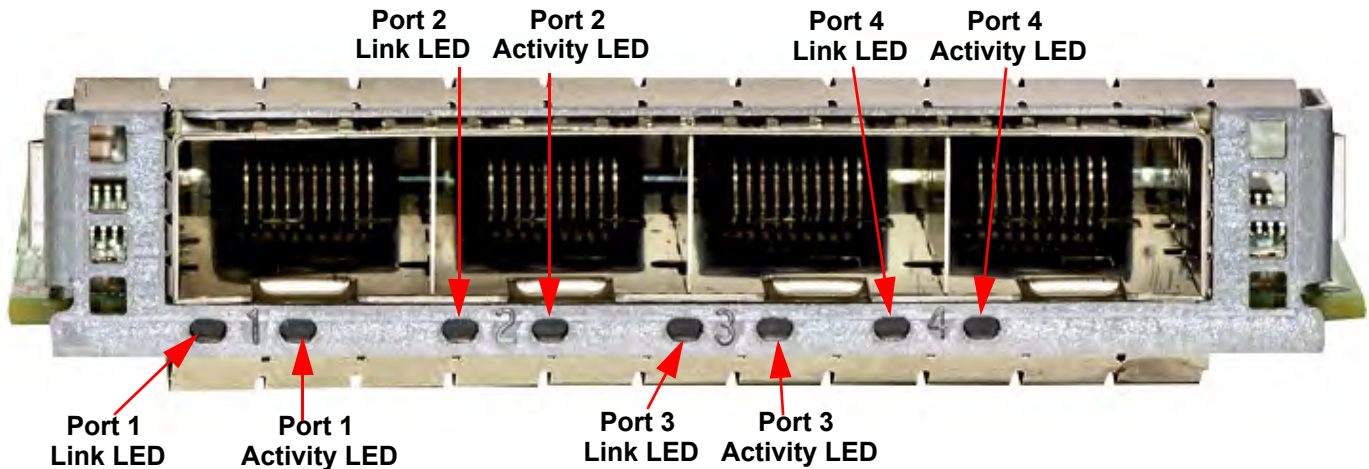
**Table 14: BCM957416N4160DC Network Adapter Activity and Link LED Locations**

LED Type	Color/Behavior	Notes
Activity	Off	No Activity
	Green blinking	Traffic Flowing Activity
Link	Off	No Link
	Green	Linked at 10 Gb/s
	Amber	Linked at 1 Gb/s

### 3.10 BCM957504-N425D

The SFP28 port supports two LEDs to indicate traffic activities and link speed. The LEDs are visible as shown in [Figure 27](#). Its locations and form factors conform to the OCP 3.0 Design Specification. The LED functionality is described in [Table 15](#).

**Figure 27: Activity and Link LED Locations**



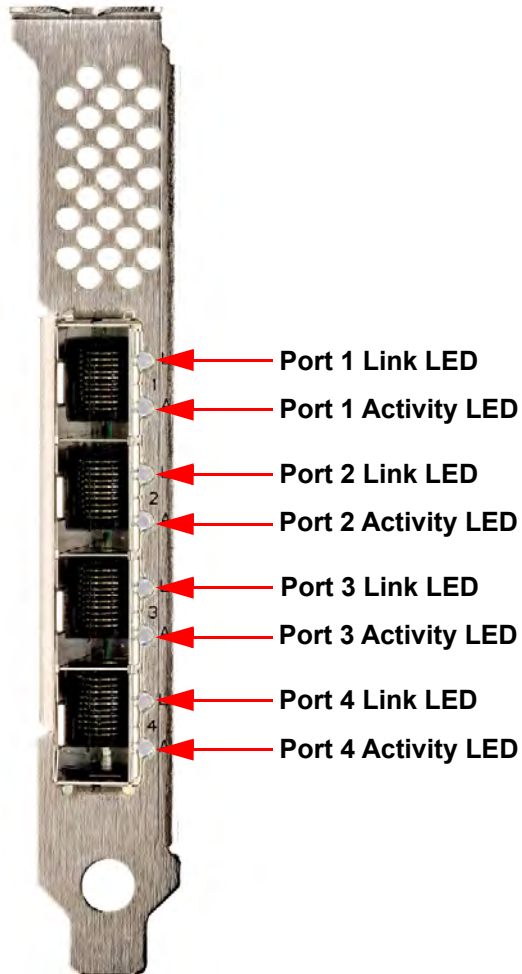
**Table 15: BCM957504-N425D LED Functions**

LED Type	Color/Behavior	Note
Activity	Off	No Link
	Green (blinking)	Link up (traffic flowing)
Link	Off	No Link
	Green	Linked at 25 Gb/s
	Amber	Linked at lower speed

### 3.11 BCM957504-P425D

The SFP28 port supports two LEDs to indicate traffic activities and link speed. The LEDs are visible through the cutout on the bracket as shown in [Figure 28](#).

**Figure 28: Activity and Link LED Locations**



**NOTE:** [Figure 28](#) shows the standard-profile bracket installed. The surface markings of the component may not reflect the product upon receipt. Broadcom reserves the right to change any component on the printed circuit board with the same functionality.

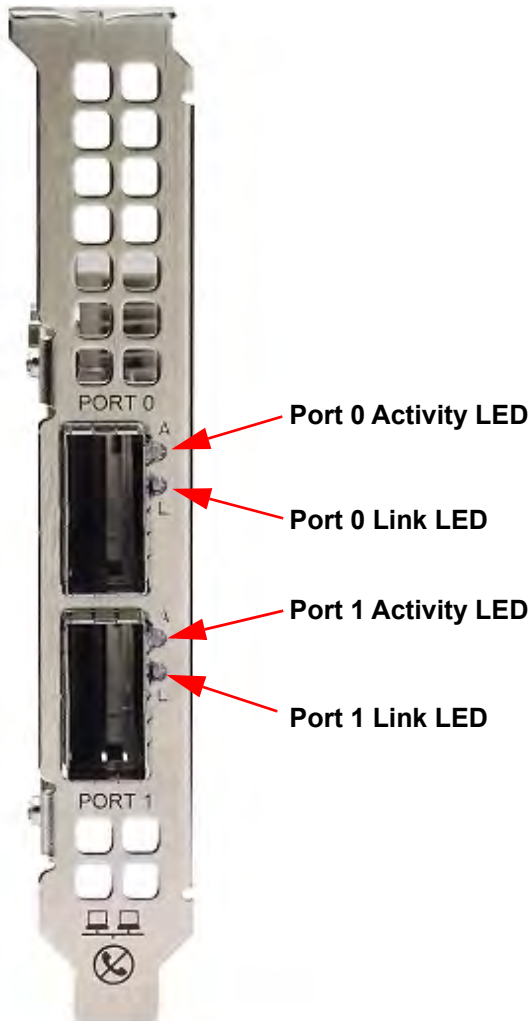
**Table 16: LED Functions**

LED Type	Color/Behavior	Note
Activity	Off	No Link
	Green (blinking)	Link up (traffic flowing)
Link	Off	No Link
	Green	Linked at 25 Gb/s
	Amber	Linked at lower speed

## 3.12 BCM957508-P2100D

The QSFP56 port supports two LEDs to indicate traffic activities and link speed. The LEDs are visible through the cutout on the bracket as shown in [Figure 29](#). The LED functionality is described in [Table 17](#).

**Figure 29: Activity and Link LED Locations**



**NOTE:** [Figure 29](#) shows the standard-profile bracket installed. The surface markings of the component may not reflect the product upon receipt. Broadcom reserves the right to change any component on the printed circuit board with the same functionality.

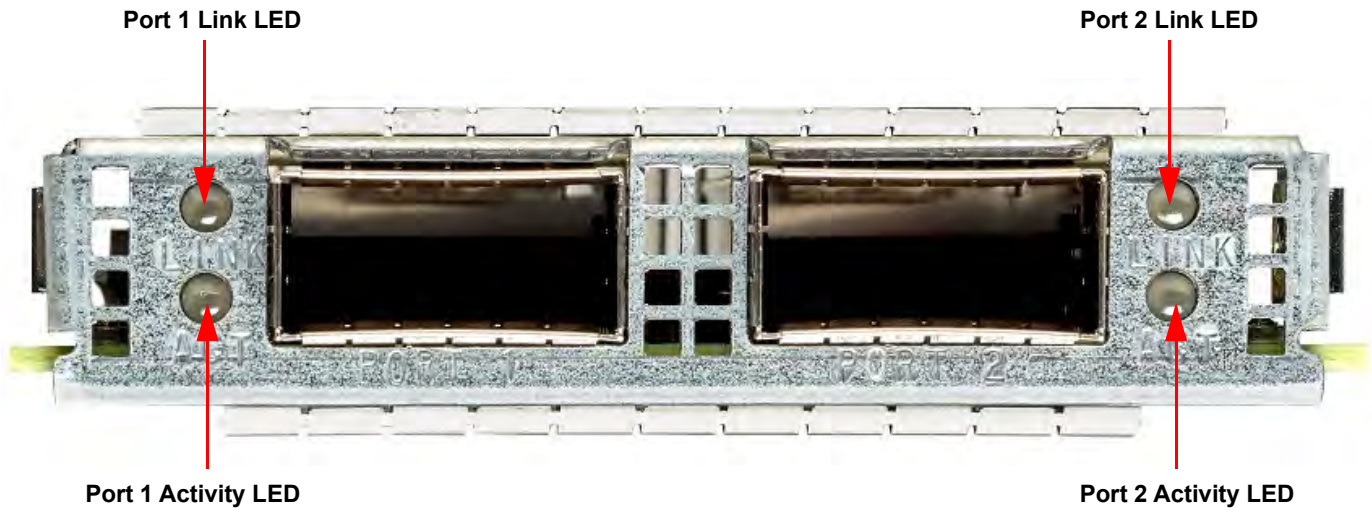
**Table 17: LED Functions**

LED Type	Color/Behavior	Note
Activity	Off	No Link
	Green (blinking)	Link up (traffic flowing)
Link	Off	No Link
	Green	Linked at 100 Gb/s
	Amber	Linked at lower speed

### 3.13 BCM957508-N2100D

The QSFP56 port supports two LEDs to indicate traffic activities and link speed. The LEDs are visible as shown in [Figure 30](#). Its locations and form factors conform to the OCP 3.0 Design Specification.

**Figure 30: Activity and Link LED Locations**



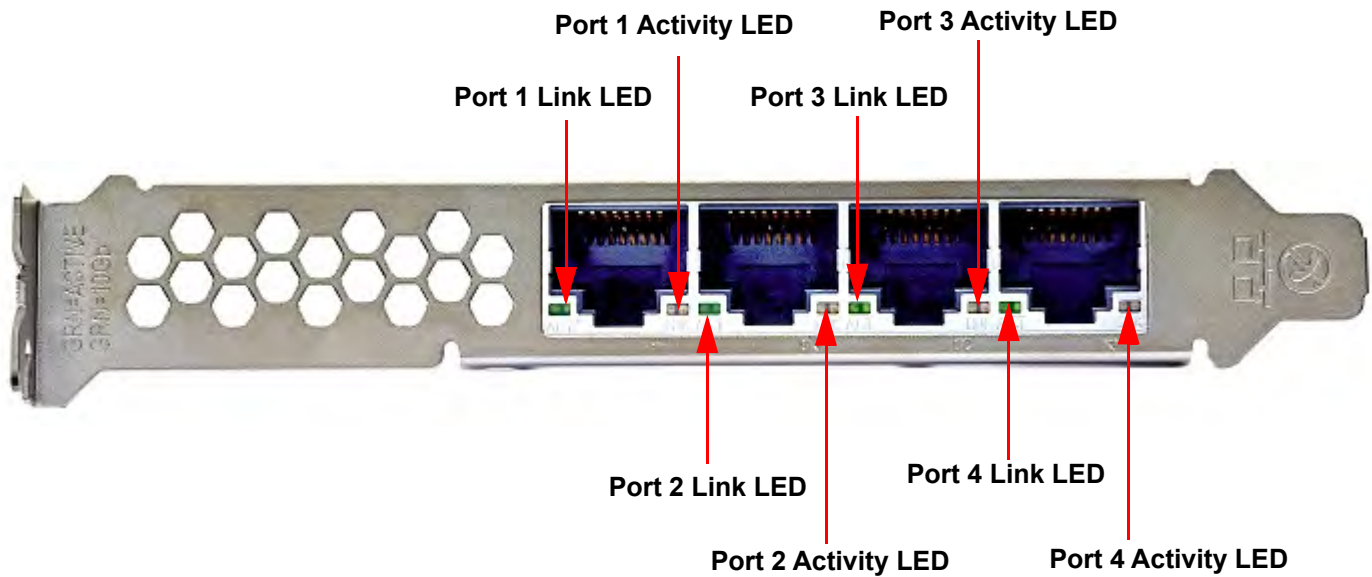
**Table 18: LED Functions**

LED Type	Color/Behavior	Note
Activity	Off	No Activity
	Green (blinking)	Link up (traffic flowing)
Link	Off	No Link
	Green	Linked at 100 Gb/s and 200 Gb/s
	Amber	Linked at lower speed

### 3.14 BCM957454-P410SDBT

The 10GBASE-T port supports two LEDs to indicate traffic activities and link speed. The LEDs are visible through the cutout on the bracket as shown in [Figure 31](#). The LED functionality is described in [Table 19](#).

**Figure 31: Activity and Link LED Locations**



**Table 19: LED Functions**

NVRAM Manufacturer	Device	Mbit
Activity	Off	No activity
	Green blinking	Traffic flowing activity
Link	Off	No link
	Green	Linked at 10 Gb/s
	Orange	Linked at 1 Gb/s

### 3.15 BCM957454-N410SDBT

Each of the four RJ-45 ports supports an LED for traffic activity and a dual-color (G/O) LED for link speed. The LEDs are integrated into the RJ-45 connector as shown in [Figure 31](#). Their locations and form factors conform to the OCP NIC 3.0 design specification.

Figure 32: Activity and Link LED Locations

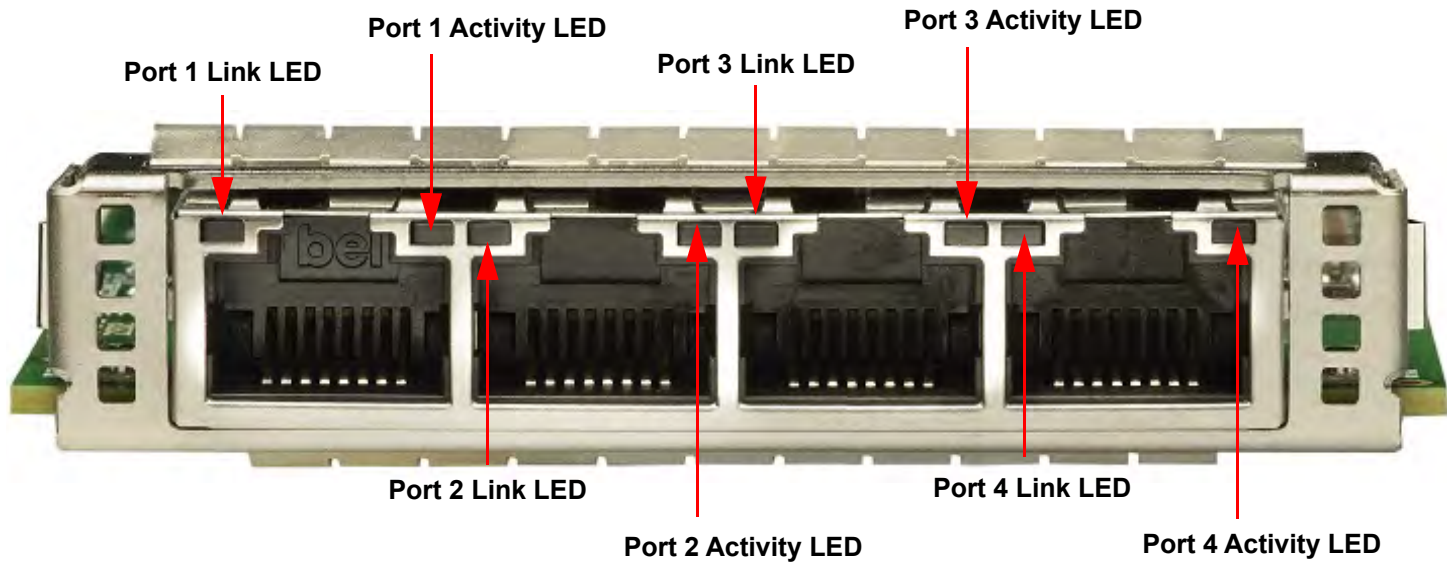


Table 20: LED Functions

NVRAM Manufacturer	Device	Mbit
Activity	Off	No activity
	Green blinking	Traffic flowing activity
Link	Off	No link
	Green	Linked at 10 Gb/s
	Orange	Linked at 1 Gb/s

## 4 Features

See the following sections for device features.

### 4.1 Software and Hardware Features

Table 21 provides a list of host interface features.

**Table 21: Host Interface Features**

Feature	Details
Host Interface	BCM9574XX – PCIe 3.0 (Gen 3: 8 GT/s; Gen 2: 5 GT/s; Gen 1: 2.5 GT/s). BCM9575XX – PCIe 4.0 (Gen 4: 16.0 GT/s; Gen 3: 8 GT/s, Gen 2: 5 GT/s, Gen 1: 2.5 GT/s)
Number of PCIe lanes	PCIe Edge Connectors: <ul style="list-style-type: none"> <li>■ BCM9574XX – x8</li> <li>■ BCM9575XX – x16</li> </ul> <b>NOTE:</b> N series adapters support OCP 3.0 interfaces.
Vital Product Data (VPD)	Supported.
Alternate Routing ID (ARI)	Supported.
Function Level Reset (FLR)	Supported.
Advanced Error Reporting	Supported.
PCIe ECNs	Support for TLP Processing Hints (TPH), Latency Tolerance Reporting (LTR), and Optimized Buffer Flush/Fill (OBFF).
MSI-X Interrupt vector per queue	1 per RSS queue, 1 per NetQueue, 1 per Virtual Machine Queue (VMQ).
IP Checksum Offload	Support for transmit and receive side.
TCP Checksum Offload	Support for transmit and receive side.
UDP Checksum Offload	Support for transmit and receive side.
NDIS TCP Large Send Offload	Support for LSOV1 and LSOV2.
NDIS Receive Segment Coalescing (RSC)	Support for Windows environments.
TCP Segmentation Offload (TSO)	Hardware acceleration support for Linux and VMware environments.
Large Receive Offload (LRO)	Hardware acceleration support for Linux and VMware environments.
Generic Receive Offload (GRO)	Hardware acceleration support for Linux and VMware environments.
Receive Side Scaling (RSS)	Support for Windows, Linux, and VMware environments.
Header-Payload Split	Enables the software TCP/IP stack to receive TCP/IP packets with header and payload data split into separate buffers. Supports Windows, Linux, and VMware environments.
Accelerated Receive Flow Steering (aRFS)	Hardware acceleration support for Linux.
Jumbo Frames	Supported.
NIC Partitioning (NPAR)	Supports up to eight Physical Functions (PFs) per port, or up to 16 PFs per silicon. This option is configurable in NVRAM.
RDMA over Converged Ethernet (RoCE)	BCM9575XX and BCM9574XX support RoCE v2 for Windows, Linux, and VMware.
Data Center Bridging (DCB)	BCM9575XX and BCM9574XX support DCBX (IEEE and CEE specification), PFC, and AVB.
NC-SI (Network Controller Sideband Interface)	Supported.
Wake on LAN (WOL)	Supported in OCP, rNDC, and NGM devices.
PXE boot	Supported.
UEFI boot	Supported.
Pause Flow Control (IEEE 802.3x)	Supported.



**Table 21: Host Interface Features (Continued)**

Feature	Details
Priority Flow Control (IEEE 802.1Qbb)	Supported.
Auto negotiation	Supported.
IEEE 802.1q VLAN	Supported.
Interrupt Aggregation	Supported.
MAC/VLAN filters	Supported.
PTP	Supported only in Linux.

## 4.2 Virtualization Features

Table 22 lists the virtualization features of the Broadcom Ethernet Network Adapter.

**Table 22: Virtualization Features**

Feature	Details
Linux KVM Multiqueue	Supported.
VMware NetQueue	Supported.
NDIS Virtual Machine Queue (VMQ)	Supported.
Virtual eXtensible LAN (VXLAN) – Aware stateless offloads (IP/UDP/TCP checksum offloads, VLAN insertion/removal, NetQueue, VMQ, RSS, TCP segmentation offload, Large Send Offload, Generic Receive Offload).	Supported.
Generic Routing Encapsulation (GRE) – Aware stateless offloads (IP/UDP/TCP checksum offloads, VLAN insertion/removal, VMQ, RSS, TCP segmentation offload, Generic Receive Offload).	Supported.
Network Virtualization using Generic Routing Encapsulation (NVGRE) – Aware stateless offloads (IP/UDP/TCP checksum offloads, VLAN insertion/removal, VMQ, RSS, Large Send Offload).	Supported.
Generic Network Virtualization Encapsulation (Geneve) – Aware Stateless offloads (IP/UDP/TCP checksum offloads, VLAN insertion/removal, NetQueue, RSS, TCP segmentation offload, Generic Receive Offload).	Supported.
IP-in-IP aware stateless offloads (IP/UDP/TCP checksum offloads, VLAN insertion/removal, NetQueue, RSS, TCP segmentation offload, Generic Receive Offload).	Supported.
SR-IOV v1.0	BCM9574XX – 128 Virtual Functions (VFs) for Guest Operating Systems (GOS) per device. BCM9575XX – 128 Virtual Functions (VFs) for Guest Operating Systems (GOS) per device

**Table 22: Virtualization Features (Continued)**

Feature	Details
Edge Virtual Bridging (EVB) (IEEE 802.1Qbg)	<p>BCM9575XX – Edge Virtual Bridging (EVB) enables switching of traffic between PFs/VFs, forwarding of outgoing network traffic from PFs/VFs to appropriate network ports, and steering of incoming network traffic to appropriate PFs/VFs. Both VEB (local switching in the NIC) and VEPA (switching in the adjacent switch) EVB modes of operation are supported. The EVB features supported are:</p> <ul style="list-style-type: none"> <li>■ 1K VFs and up to 128 queues per VF (flexible allocation across PFs/VFs).</li> <li>■ PCIe AER, TPH, FLR support.</li> <li>■ Virtual Ethernet Bridge (VEB)/Virtual Ethernet Port Aggregator (VEPA).</li> <li>■ MAC/VLAN filtering and mirroring.</li> <li>■ VF isolation, source pruning, anti-spoofing checks.</li> <li>■ Stateless and packet steering offloads per VF.</li> <li>■ Forwarding of unicast frames based on {Tunnel ID (optional), Destination MAC, VLAN ID (optional)}.</li> <li>■ Frame replication for multicast, broadcast, and promiscuous mode.</li> <li>■ Source pruning – Provide support for source knockout (prevent sending a multicast or broadcast frame back to the source).</li> <li>■ Mirroring of traffic to a specific PF or VF.</li> <li>■ Packet editing – VLAN insert/swap/delete.</li> <li>■ Anti-spoof checks.</li> </ul>
MSI-X vector port	74 per port default value (two port configuration). 16 per VF and is configurable in HII.

## 4.3 VXLAN

A Virtual eXtensible Local Area Network (VXLAN), defined in IETF RFC 7348, is used to address the need for overlay networks within virtualized data centers accommodating multiple tenants. VXLAN is a Layer 2 overlay or tunneling scheme over a Layer 3 network. Only VMs within the same VXLAN segment can communicate with each other.

## 4.4 NVGRE/GRE/IP-in-IP/Geneve

Network Virtualization using GRE (NVGRE), defined in IETF RFC 7637, is similar to a VXLAN.

**NOTE:** Checksum offload must be enabled when using NVGRE.

## 4.5 Stateless Offloads

### 4.5.1 IP, TCP, UDP Checksum Offload

Host software can configure the Ethernet controller to calculate IP, TCP, and UDP checksums as described in RFC 791, RFC 793, and RFC 768 respectively. The first step in checksum calculation is determining the start of an IP and UDP datagram and TCP segment within a frame, which could vary depending on whether the frame is tagged (VLAN) or encapsulated with an LLC/SNAP header. Then the checksum is computed from the start to the end of the datagram and inserted into the appropriate location in the protocol header. The Ethernet controller is designed to support checksum calculation on all frame types and also on IP datagram and TCP segments containing options.

### 4.5.2 UDP Fragmentation Offload

UDP Fragmentation Offload (UFO) is a feature that enables the software stack to offload fragmentation of large UDP/IP datagrams into multiple UDP/IP packets of size suitable for transmission. Enabling UFO can result in reduced CPU load for UDP applications. Support for this feature is only available in the Linux environment.

### 4.5.3 TCP Segmentation Offload and Large Send Offload

Large Segment Offload (LSO) is a feature that enables the software stack to offload segmentation of large TCP messages into multiple TCP/IP packets of size suitable for transmission. Enabling LSO can result in reduced CPU load for TCP applications. This is also called TCP Segmentation Offload (TSO).

### 4.5.4 Generic Receive Offload (GRO) and Large Receive Offload (LRO)

Generic Receive Offload (GRO) and Large Receive Offload (LRO) are hardware acceleration for TCP data reception. Both GRO and LRO modes of TCP receive offload are supported by the Ethernet Controller's Transparent Packet Aggregation (TPA) feature. Enabling GRO and LRO can significantly reduce CPU load and increase throughput for TCP applications by reducing the number of received messages, interrupts, and DMA operations.

TPA aggregates TCP streams by managing context entries. Each entry in the TPA context is identified by the 4-tuple: Source IP, destination IP, source TCP port, and destination TCP port.

GRO is the preferred TPA mode as packet boundaries are preserved for network routing applications, which may enable LSO for transmission.

### 4.5.5 Header and Data Split

Header-payload split is a feature that enables the software TCP/IP stack to receive TCP/IP packets with header and payload data split into separate buffers. The support for this feature is available in both Windows and Linux environments. The following are potential benefits of header-payload split:

- The header-payload split enables compact and efficient caching of packet headers into host CPU caches. This can result in a receive side TCP/IP performance improvement.
- Header-payload splitting enables page flipping and zero copy operations by the host TCP/IP stack. This can further improve the performance of the receive path.

## 4.5.6 VLAN Tag Insertion and Removal

On the TX Path, the Ethernet controller is capable of inserting IEEE 802.1Q-compliant VLAN tags into transmitted frames and extracting the VLAN tags from received frames.

On the RX path, receiving VLAN-tagged (IEEE 802.1q-compliant) packets is supported by the Ethernet controller. If a function is configured to strip VLAN tag, then the VLAN tag is stripped from the IEEE 802.1q-compliant packet at reception and placed in a receive completion record.

## 4.5.7 Packet Steering

### 4.5.7.1 Receive Side Scaling (RSS)

RSS is a scalable networking technology that enables receive packet processing to be balanced across multiple processors in the system while maintaining in-order delivery of the data. RSS enables different packets, received by a single network adapter, to be processed on different CPUs/cores in parallel while preserving in-order delivery of TCP connections.

Receive Side Scaling (RSS) uses a Toeplitz algorithm which uses 4-tuple match on the received frames and forwards it to a deterministic CPU for frame processing. This allows streamlined frame processing and balances CPU utilization. An indirection table is used to map the stream to a CPU.

Symmetric RSS allows the mapping of packets of a given TCP or UDP flow to the same receive queue.

### 4.5.7.2 Accelerated Receive Flow Steering

Accelerated RFS (aRFS, or RFS) is an Ethernet controller feature that improves packet reception efficiency by delivering packets to queues based on CPU locality of the application. This reduces memory access latency and improves performance. Accelerated RFS takes precedence over RSS when enabled and configured. If the incoming flow does not match any existing n-tuple filters, it is steered according to the RSS hash.

## 4.5.8 Data Center Bridging

Data Center Bridging (DCB) is a set of protocols and capabilities (for example, DCBX, LLDP, ETS, and PFC) for use in a data center environment. The Broadcom Ethernet Network Adapter family of Ethernet controllers support for priority flow control is described in section on Priority Flow Control.

## 4.6 Priority Flow Control

Priority Flow Control (PFC) is a standard-compliant backpressure mechanism implemented in Broadcom Ethernet Network Adapters. The goal of PFC is to backpressure congested priority traffic flow without affecting the traffic flows of uncongested priorities and to ensure that packets are not dropped in burst or transient scenerios. PFC can be used in a network with real-time or time-sensitive traffic because of its capability to provide differential treatment to Traffic Classes. For example, using PFC lower priority Internet traffic can be backpressured leaving the higher priority traffic like VOIP and Streaming Video flowing through the link without flow control.

## 4.7 Virtualization Offload

### 4.7.1 Multiqueue Support

Broadcom Ethernet Network Adapters support Multiqueue in the hardware

### 4.7.2 KVM/Xen Multiqueue

KVM/Multiqueue returns the frames to different queues of the host stack by classifying the incoming frame by processing the received packet's destination MAC address and or IEEE 802.1Q VLAN tag. The classification combined with the ability to DMA the frames directly into a virtual machine's memory allows scaling of virtual machines across multiple processors.

### 4.7.3 Virtual Machine Queue

The NDIS Virtual Machine Queue (VMQ) is a feature that is supported by Microsoft to improve Hyper-V network performance. The VMQ feature supports packet classification based on the destination MAC address to return received packets on different completion queues. This packet classification combined with the ability to DMA packets directly into a virtual machine's memory allows the scaling of virtual machines across multiple processors.

#### 4.7.3.1 VMware NetQueue

The VMware NetQueue is a feature that is similar to Microsoft's NDIS VMQ feature. The NetQueue feature supports packet classification based on the destination MAC address and VLAN to return received packets on different NetQueues. This packet classification combined with the ability to DMA packets directly into a virtual machine's memory allows the scaling of virtual machines across multiple processors.

#### 4.7.3.2 Xen Multiqueue

Xen multiqueue enables network device drivers to dedicate each Rx queue to a specific guest operating system. This means the network device drivers should be able to allocate physical memory from the set of memory pages assigned to a specific guest operating system.

### 4.7.4 Tunneling Offload

Stateless Transport Tunnel Offload (STT) is a tunnel encapsulation that enables overlay networks in virtualized data centers. STT uses IP-based encapsulation with a TCP-like header. There is no TCP connection state associated with the tunnel and that is why STT is stateless. Open Virtual Switch (OVS) uses STT.

An STT frame contains the STT frame header and payload. The payload of the STT frame is an untagged Ethernet frame. The STT frame header and encapsulated payload are treated as the TCP payload and TCP-like header. The IP header (IPv4 or IPv6) and Ethernet header are created for each STT segment that is transmitted.

Broadcom Ethernet Network Adapters support Network Overlays or Tunneling, specifically VXLAN, variants of GRE, and IP-in-IP. Both VXLAN and NVGRE are defined to support larger scale than basic IEEE 802.1Q VLANs, using a 24-bit label space rather than 12-bit VID. Both are L2-in-L3 tunneling methods with NVGRE using GRE to carry the tunnel label and VXLAN using UDP to identify the tunnel label.

Stateless offload for tunneling and encapsulated frames in Broadcom Ethernet Network Adapters applies to VXLAN, Geneve, L2GRE, NVGRE, and also IP-in-IP scheme. The section below describes VXLAN as an example to discuss the general support for this feature. The difference between different tunneling/encapsulation schemes is noted when it is applicable.

All the offloads described in this section are supported on both physical and virtual functions. (PFs and VFs).

#### 4.7.4.1 VXLAN

A Virtual eXtensible Local Area Network (VXLAN), defined in IETF RFC 7348, is used to address the need for overlay networks within virtualized data centers accommodating multiple tenants. The VXLAN scheme and related protocols are defined in IETF RFC 7348.

VXLAN is a Layer 2 overlay or tunneling scheme over a Layer 3 network. Each overlay is termed a VXLAN segment. Only VMs within the same VXLAN segment can communicate with each other. Each VXLAN segment is scoped through a 24-bit segment ID, VXLAN Network Identifier (VNI). This allows up to 16M VXLAN segments to coexist within the same administrative domain. The UDP destination port identifies the presence of a VXLAN tunnel.

#### 4.7.4.2 GRE and NVGRE

Broadcom Ethernet Network Adapters support Generic Routing Encapsulation (GRE) per RFC 2784 and RFC 2890. Network Virtualization using GRE (NVGRE) is a Layer 2 overlay, used to address the need of subnets for overlay networks with larger numbers of VLANs. As with the VXLAN scheme, each NVGRE segment is scoped through a 24-bit identifier, called Virtual Subnet Identifier (VSID), in a GRE header to support up to 16M virtual network segments.

NVGRE is a method for network virtualization, similar to VXLAN in purpose. NVGRE uses a GRE header. The key field defined in RFC 2890 is used to carry a virtual subnet identifier. NVGRE is identified using EtherType = 0x6558, the EtherType for Transparent Ethernet Bridging (carrying a full Ethernet frame as the payload).

#### 4.7.4.3 Geneve

Broadcom Ethernet Network Adapters support Generic Network Virtualization Encapsulation (Geneve, also known as Next Generation Encapsulation). It leverages the same concepts as VXLAN, using UDP destination port to identify the presence of the Geneve tunnel header. A primary goal for Geneve is to enable transport of metadata (system state) between source and destination.

#### 4.7.4.4 IP-in-IP

IP-in-IP is a Layer 3 overlay or tunneling scheme over a Layer 3 network. It is a method by which an IP datagram may be encapsulated (carried as payload) within another IP datagram for the purpose of altering the routing of the inner IP packets and allowing them to be delivered to an intermediate destination that would otherwise not be selected by the destination Address field in the inner IP header. IP Encapsulation with IP is defined in RFC 2003.

#### 4.7.4.5 Checksum Offload

The following checksums are computed on transmit path and then computed and verified on the receive path.

- Outer IPv4 checksum if the outer IP datagram is an IPv4 datagram.
- Outer UDP checksum (if non-zero): The current VXLAN IETF draft suggests that the outer UDP checksum should be transmitted as zero. If the outer UDP checksum field in the VXLAN frame received is zero, then the frame is accepted without computing outer UDP checksum. If the outer UDP checksum field in the VXLAN frame received is non-zero, then the outer UDP checksum should be computed. (Note: The current IETF draft allows the receiver to ignore outer UDP checksum when it is set to non-zero.) This item is not available in L2GRE/NVGRE/IP-in-IP -aware offload.
- Inner IPv4 checksum if the inner IP datagram is an IPv4 datagram.
- Inner UDP or TCP checksum.

#### 4.7.4.6 VLAN Tagging

A VXLAN-frame supports the following tagging options:

- No IEEE 802.1Q tag in inner and outer datagrams.
- IEEE 802.1Q tag in the outer IP datagram only.
- IEEE 802.1Q tag in the inner IP datagram only.
- IEEE 802.1Q tags in both the inner and outer IP datagrams.

**NOTE:** The device supports insertion and removal of outer IEEE 801.Q Tag for VXLAN/GRE/IP-in-IP frames. It also supports insertion and removal of inner IEEE 801.Q Tag for VXLAN.frames. The device retains inner IEEE 801.Q Tag for NVGRE frames in the inner packet.

#### 4.7.4.7 VMQ

For VXLAN frames, the NetQueue uses the following fields for the queue selection:

- Inner destination MAC address.
- Outer destination MAC address.
- VXLAN Network Identifier (VNI).

The device supports NetQueue selection based on any combination of the fields above.

**NOTE:** For GRE/IP-in-IP frames, the VMQ selection is performed using the Ethernet header of the encapsulated packet (inner packet) that includes inner destination MAC address and inner 802.1Q Tag (optional).

#### 4.7.4.8 RSS

For VXLAN frames, there are two options for RSS queue selection:

- RSS hash computation based on outer UDP/IP headers: The VXLAN IETF draft recommends that the source port is set based on the hash of inner headers. This allows the RSS hash computation based on outer UDP/IP headers a viable option.
- RSS hash computation based on inner UDP/IP or TCP/IP headers: This option requires 2-tuple or 4-tuple hash computation based on inner headers. The inner header is parsed for RSS hash computation. The RSS hash computation is performed in parallel with other checksum computations. In some exceptional cases, this may lead to inaccurate hash computations where one or more checksum validation fails.

For GRE/IP-in-IP frames, the RSS queue selection is performed using inner headers. The following are possible combinations:

- GRE/IP-in-IP frame with inner TCP/IP or UDP/IP headers: The RSS is performed using four-tuple (src IP, dst IP, src port, dst port) hash on the inner IP header and TCP or UDP header.
- GRE/IP-in-IP frame without inner TCP or UDP header: The RSS is performed using 2-tuple (src IP, dst IP) hash on the inner IP header.
- Other encapsulated frames (for example, GRE/IP-in-IP frames that cannot be parsed): The RSS is performed using 2-tuple (src IP, dst IP) hash on the outer IP header.

### 4.7.4.9 TCP Segmentation Offload

For VXLAN, the TCP Segmentation Offload (TSO) algorithm is performed on the inner TCP segment. The hypervisor provides template TCP/IP headers for the inner TCP segment as well as template VXLAN/UDP/IP headers for the outer UDP datagram. For every inner TCP segment generated by the VXLAN-aware TSO, the outer VXLAN, UDP, IP, and MAC headers are inserted and outer IPv4 checksum, outer UDP checksum (not for GRE frames), inner IP checksum (for inner IPv4 datagram only), and inner TCP checksum is computed and inserted. The device updates the IP ID field for every inner TCP segment.

For GRE/IP-in-IP, the LSO (Large Send offload) algorithm is performed on the inner TCP segment. The hypervisor provides template TCP/IP headers for the inner TCP segment as well as template GRE/IP/Ethernet headers for the outer IP datagram. For every inner TCP segment generated by the GRE/IP-in-IP-aware LSO, the outer GRE (not applicable for IP-in-IP frames), IP, and MAC headers is inserted and outer IPv4 checksum (for outer IPv4 datagrams only), inner IP checksum (for inner IPv4 datagram only), and inner TCP checksum is computed and inserted. The device updates the IP ID field for every inner TCP segment.

#### 4.7.4.10 Large Receive Offload

Tunneling Offload support for LRO, RSC, TSO, LSO, GSO, and GRO.

## 4.8 SR-IOV

The PCI-SIG defines optional support for Single-Root IO Virtualization (SR-IOV). SR-IOV is designed to allow access of the VM directly to the device using Virtual Functions (VFs). The NIC Physical Function (PF) is divided into multiple virtual functions and each VF is presented as a PF to VMs.

SR-IOV uses IOMMU functionality to translate PCIe virtual addresses to physical addresses by using a translation table.

The number of Physical Functions (PFs) and Virtual Functions (VFs) are managed through the UEFI HII menu and through NVRAM configurations. SR-IOV can be supported in combination with NPAR mode.

The SR-IOV feature requires corresponding SR-IOV support in the BIOS (Intel VT-d or AMD IOMMU) and Operating System/Hypervisor as well as the PCIe endpoint device (the NIC in this case).

The Broadcom Ethernet Network Adapter family of Ethernet Controllers offers the following offload functionality to the VF that are available to the PF:

- TX and RX IP/TCP/UDP checksum offload
- Large Send offload (LSO) or TCP segmentation offload (TSO, GSO)
- Receive Segmentation offload (RSC) or Large Receive offload (LRO, GRO)
- Receive Side Scaling (RSS) – up to 64 queues per VF
- Multiple COS queues – up to 4 queues per VF
- Network Virtualization Generic Routing Encapsulation, Virtual Extensible LAN – NVGRE/VXLAN



## 4.8.1 SR-IOV Configuration Support Matrix

Table 23 provides a SR-IOV support matrix.

Table 23: SR-IOV Support Matrix

SR-IOV Support	Guest OS – VF					
	Win2k19	Win2k22	RH8.0+	RH9.x	SLES12.2+	SLES15.x
Windows 2019	Yes	Yes	Yes	Yes	Yes	Yes
Windows 2022	Yes	Yes	Yes	Yes	Yes	Yes
RH8.x+	Yes	Yes	Yes	Yes	Yes	Yes
RH9.x	Yes	Yes	Yes	Yes	Yes	Yes
SLES15.x	Yes	Yes	Yes	Yes	Yes	Yes
ESX7.x+	Yes	Yes	Yes	Yes	Yes	Yes
ESX8.x	Yes	Yes	Yes	Yes	Yes	Yes

## 4.9 Network Partitioning (NPAR)

The Network Partitioning (NPAR) feature allows a single physical network interface port to appear to the system as multiple network device functions. When NPAR mode is enabled, Broadcom Ethernet Network Adapters are enumerated as multiple PCIe physical functions (PF). Each PF or partition is assigned a separate PCIe function ID on initial power on. Each partition is assigned its own configuration space, BAR address, and MAC address allowing it to operate independently. Partitions support direct assignment to VMs, VLANs, and so on, just as any other physical interface.

The original PCIe definition allowed for eight PFs per device. For Alternative Routing-ID (ARI) capable systems, Broadcom Ethernet Network adapters support up to 16 PFs per device.

## 4.10 Security (BCM575XX Only)

The BCM575XX TruTrust™ technology is capable of secure boot meaning it only executes boot images authenticated by the secure boot loader (SBL). Secure boot functionality is the cornerstone of a security enabled system since it is the root of trust from which all subsequent applications are run. The secure boot capability provides the following functionality:

- Secure boot Core Root of Trust – The Secure Boot Loader is based in device ROM, and outside the scope of modification. It functions as the Core Root of Trust for software, meaning that the system is in a trusted state from reset to when a secure image has been authenticated.
- Boot Image Authentication – Only Images authenticated by the SBL are executed by the system.
- Boot Image Integrity – The SBL cryptographically validates the integrity of the Secure Boot Image before it is executed to ensure that it has not been tampered with maliciously or errantly.
- Boot Image Confidentiality – The secure processor has the hardware support to execute encrypted images which ensures that device images are never in the clear and protected from reverse engineering or used in device cloning.

Secure devices can be delivered to the customer in a state pending final customization. This customization step is executed by the customer, and once complete, only customer signed images execute on the device. Customization provides the following capabilities:

- Customer takes responsibility for the creation and management of keys used in signing their code. This allows the customer to apply their own security policies in managing their keys and ensures that no code can be signed for their devices by a third party.
- Only code signed by the customer runs on their customized device. This ensures that device code cannot be tampered with in the field and verifies the authenticity of the image.

- Customer signed images do not run on other customers secure devices. This prevents piracy of customer images. However, it does not relieve the customer of the responsibility for protecting unsecured binaries from reverse engineering.
- Device or customer specific encrypted execution images can be generated. This prevents piracy of customer images and cloning of devices.

## 4.11 RDMA over Converged Ethernet – RoCE

Remote Direct Memory Access (RDMA) over Converged Ethernet (RoCE) is a complete hardware offload feature in the Broadcom Ethernet Network adapters allow RDMA functionality over an Ethernet network. RoCE functionality is available for both user mode and kernel mode applications. RoCE is supported under Linux, Windows and VMware operating systems.

See the following links for RDMA support for each operating system:

### Windows

[Microsoft SMB Direct](#)

### Linux

[Red Hat Infiniband and RDMA Networking](#)

### VMware

[VMware Network Requirements for RDMA](#)

## 4.12 VMWare Enhanced Networking Stack (ENS)

VMware with Intel designed ENS to support DPDK (Data Plane Development Kit) based NFV (Network Functions Virtualization) applications.

### 4.12.1 Features

This section contains the supported features of ENS:

- New and faster vSphere networking stack targeted for NFV applications.
- DPDK techniques employed with new vmxnet3 virtual device backend.
- New poll mode and interrupt mode physical device drivers.
- Faster switching using flow cache.
- Deliver improved performance while supporting vSphere features.

### 4.12.2 ENS Design Choices

This section contains ENS design choices for improved and deterministic performance:

- Dedicated CPU allocation to system thread and polling.
- NUMA-aware placement of VM and system threads.
- NUMA-aware allocation with large pages.
- Simplified packet representation.
- Use of flow cache.
- Lockless datapath.

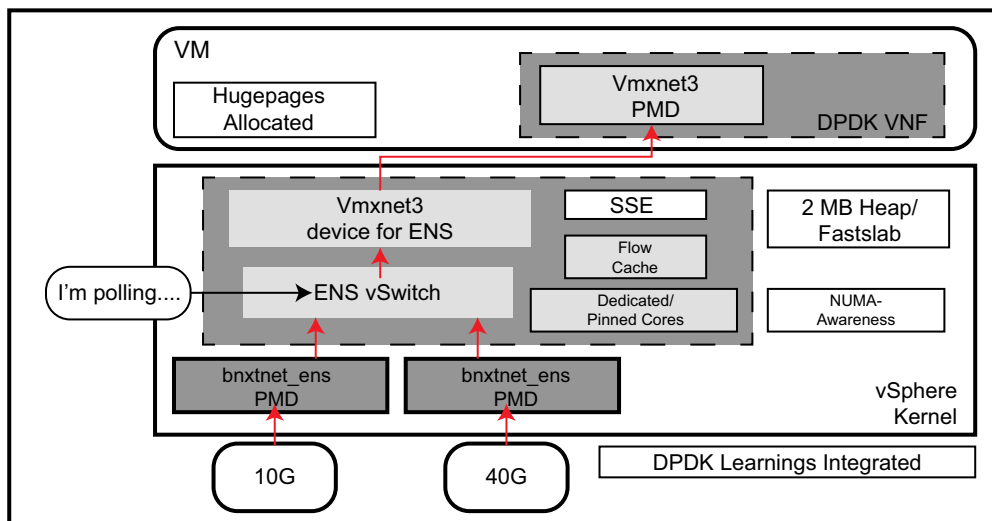
- Vmxnet3 optimizations.
- Streaming SIMD Extension(SSE) instructions faster packet processing.

### 4.12.3 ENS Performance

This section contains ENS performance information:

- 3-5x improvement in packet rate over the existing vSphere networking stack performance scales with the number of system threads.
- Acceptable packet loss.
- Low jitter and latency 1x 3-5x Guest: DPDK API + App Host: Default vSwitch Guest: DPDK API + App Host: ENS vSwitch.

Figure 33: VMware ENS stack



For additional information, see <https://docs.vmware.com/en/VMware-NSX-T-Data-Center/2.5/administration/GUID-668EB7EF-3E39-46C8-AF2F-43B7DAB6D42E.html>

### 4.12.4 Limitations and Restrictions

This section contains the limitations and Broadcom support restrictions:

- LRO is not supported by the current ENS stack. RO is not supported in the ENS path.
- RSS is not supported by the current ENS stack. RSS is not supported in the ENS path.
- SR-IOV is disabled when ENS is enabled on the PF.
- ENS is supported with ESXi 7.0 or higher.

## 4.13 Supported Combinations

The following sections describe the supported feature combinations for this device.

### 4.13.1 NPAR, SR-IOV, and RoCE

Table 24 shows the supported feature combinations of NPAR, SR-IOV, and RoCE.

**Table 24: NPAR, SR-IOV, and RoCE**

SW Feature	Notes
NPAR	Up to 8 PFs or 16 PFs
SR-IOV	Up to 128 VFs (total per chip)
RoCE on PFs	Up to 4 PFs for the BCM5741X devices and 16 PFs for BCM575XX devices
RoCE on VFs	BCM575XX supports RoCE SRIOV over up to 128 VFs in the 219.x Release. The 218.x release does not support RoCE SRIOV for BCM575XX. BCM541X does not support RoCE on VFs.
Host OS	Linux, Windows, ESXi (no vRDMA support)
Guest OS	Linux and Windows
DCB	Up to two COS per port with non-shared reserved memory

**NOTE:** Certain 4 port BCM9575XX adapters support up to 32 VF and 4 NPAR per port. When NPAR and SR-IOV are enabled, certain ESXi OS are not able to configure more than 8 VFs on partitions 3 and above.

### 4.13.2 NPAR, SR-IOV, and DPDK

Table 25 shows the supported feature combinations for NPAR, SR-IOV, and DPDK.

**Table 25: NPAR, SR-IOV, and DPDK**

SW Feature	Notes
NPAR	Up to 8 PFs or 16 PFs
SR-IOV	Up to 128 VFs (total per chip)
DPDK	Supported over all VFs
Host OS	Linux
Guest OS	Linux

## 4.14 Unsupported Combinations

RoCE SRIOV + NPAR is not supported. BCM5741X does not support RoCE SRIOV. BCM575XX does not support RoCE SRIOV in 218.x release. 2.19 release supports RoCE SRIOV for BCM575XX.

## 5 Installing the Hardware

### 5.1 Safety Precautions

**CAUTION!** Server class system power supplies with higher current may be hazardous when normal operating procedures are not used. Before removing the cover of the system, observe the following precautions to protect yourself and to prevent damage to the system components:

- Remove any metallic objects or jewelry from your hands and wrists.
- Make sure to use only insulated or nonconducting tools.
- Verify that the system is powered OFF and unplugged before you touch internal components.
- Install or remove adapters in a static-free environment. The use of a properly grounded wrist strap or other personal antistatic devices and an antistatic mat is strongly recommended.

### 5.2 System Requirements

Before installing the Broadcom Ethernet Network Adapter adapter, verify that the system meets the requirements listed for the operating system.

#### 5.2.1 Hardware Requirements

See the following list of hardware requirements:

- One open PCIe Gen 3 or Gen 4 slot in x8, x16, or x32 mode.
  - 574XXA41XX – PCIe Gen 3 slot in X8 mode.
  - 574XXM41XX – OCP 2.0 or rNDC slots.
  - 57XXXNXXXX – OCP 3.0 slot
- 16 GB memory or more (32 GB or more is recommended for virtualization applications and nominal network throughput performance).

#### 5.2.2 Memory Requirements

The Broadcom Ethernet Network Adapter driver and firmware use PCIe DMA transactions between the host memory and Broadcom Ethernet Network Adapter devices. The amount of host memory required for traffic varies for each operating system. The memory requirements also change when features are enabled such as NPAR and RoCE. This section provides the maximum memory requirements for the Windows operating system when enabling features that require memory allocations.

##### 5.2.2.1 Maximum Memory Requirements for Windows

The following NIC features and OS functions are enabled during the maximum memory requirement calculations:

- Broadcom Ethernet Network Adapter features – NPAR-EP, RDMA, SR-IOV, and 16 RSS rings.
  - Windows 2019 – Disable VMMQ on each NPAR function (VMQ is used in this mode). Each function is bound to a vSwitch. Do not assign VM to vSwitch. Disable **Virtual Switch RSS**.
  - Memory requirements – 3.2 GB per function.

### 5.2.2.2 Minimum Memory Requirements for Windows

The following NIC features and OS functions are enabled during the minimum memory requirement calculations:

- Broadcom Ethernet Network Adapter features – Single physical function, features such as RDMA or SR-IOV are not enabled.
  - Windows 2019 – Driver load only.
  - Memory requirements – 1 GB per function.

### 5.2.3 Preinstallation Checklist

See the following list before installing the Broadcom Ethernet Network Adapter device:

1. Verify that the server meets the hardware and software requirements listed in System Requirements.
2. Verify that the server is using the latest BIOS.
3. If the system is active, shut it down.
4. When the system shutdown is complete, turn off the power and unplug the power cord.
5. Holding the adapter card by the edges, remove it from its shipping package and place it on an antistatic surface.
6. Check the adapter for visible signs of damage, particularly on the card edge connector. Never attempt to install a damaged adapter.

## 5.3 Installing the Adapter

The following instructions apply to installing the Broadcom Broadcom Ethernet Network Adapter Ethernet adapter (add-in NIC) into most servers. See the manuals that are supplied with the server for details about performing these tasks on this particular server.

1. Review the Safety Precautions and Preinstallation Checklist before installing the adapter. Ensure that the system power is OFF and unplugged from the power outlet, and that proper electrical grounding procedures have been followed.
2. Open the system case and select any empty PCIe 3 or 4 x8 or x16 slot.
3. Remove the blank cover plate from the slot.
4. Align the adapter connector edge with the connector slot in the system.
5. Secure the adapter with the adapter clip or screw.
6. Close the system case and disconnect any personal antistatic devices.

## 5.4 Connecting the Network Cables

The Broadcom Broadcom Ethernet Network Adapter adapters support SFP+/SFP28/SFP56/QSFP+/QSFPcables with speeds up to 200 Gb/s. The various Broadcom network adapters interoperate with a wide range of cables. See the [Ethernet Cable and Transceiver Interoperability Testing Report](#) located on Broadcom.com.

### 5.4.1 Validated Cables and Modules

Table 26: Validated Cables and Modules

Modules and Cables	Dell Part Number	Adapters	Descriptions
SFP+ to 1000BASE-T Transceiver	XTY28	BCM957412 BCM957414	SFP+ to 1000BASE-T Transceiver
FTLX1471D3BCL-FC	RN84N	BCM957412 BCM957414 BCM957504	10 Gb/s LR SFP+ Transceiver
FTLX8574D3BNL	N8TDR	BCM957412 BCM957414 BCM957504	85°C extended temperature range 10 Gb/s SFP+ Transceiver
FTLF8536P4BNL-FC	HHHHC	BCM957414 BCM957504	85°C extended temperature range 25 Gb/s SFP+ Transceiver
FTLX8574D3BCL-FC or PLRXPLSCS43811	WTRD1 C5RNH	BCM957412 BCM957414 BCM957504	10 Gb/s-SR SFP+ Transceiver
Dual-rate 10/25G SFP28	M14MK	BCM957414 BCM957504	Dual-rate 10/25G SFP28
25G SFP28 LR Optic	OYR96	BCM957414 - OCP3.0 BCM957504 - OCP3.0	25G SFP28 LR Optic
SFP28 25Gb	W4GPP	BCM957414 BCM95750X	SFP28 25Gb

**Table 26: Validated Cables and Modules (Continued)**

Modules and Cables	Dell Part Number	Adapters	Descriptions
10G and 25G Active Optical Cable	YJF03, P9GND, T1KCN, 1DXKP, MT7R2, K0T7R, W5G04, HHFK0, 3YWG7, 5CMT2, RCVP5, X5DH4	BCM957412 BCM957414 BCM957504	10G and 25G Active Optical Cable
Power-Edge QSFP28 SR4 Optic	4WGYD	BCM957508	Power-Edge QSFP28 SR4 Optic

**NOTE:**

1. Direct Attach Cables (DAC) that conform to IEEE standards can be connected to the adapter.

**5.4.1.1 Copper**

The BCM957416AXXXX and BCM957416XXXX adapters have two RJ-45 connectors used for attaching the system to a CAT 6E Ethernet copper-wire segment.

**5.4.1.2 SFP+**

The BCM957412AXXXX and BCM957412MXXXX adapters have two SFP+ connectors used for attaching the system to a 10 Gb/s Ethernet switch.

**5.4.1.3 SFP28**

The BCM957414XXXX, BCM957414AXXXX, BCM957504 adapters have two SFP28 connectors used for attaching the system to a 25 Gb/s Ethernet switch.

**5.4.1.4 QSFP**

The BCM957508 adapters has QSFP connectors for attaching the system to a 100 Gb/s, 50 Gb/s PAM-4, or 100 Gb/s PAM-4 Ethernet switch.



## 6 Software Packages and Installation

The software is comprised of the driver, firmware, library, and utility components. All components are bundled in files available from <http://support.dell.com>.

### 6.1 Supported Operating Systems

Table 27 provides a list of supported operating systems.

Table 27: Supported Operating Systems

Operating System	Distribution
Windows Server	Windows 2019 and 2022
Redhat	8.6, 8.7, 9.0, and 9.1
SuSE SLES	15 SP3 and 15 SP4
VMware	ESXi 7.0 U3 and ESXi 8.0 U1

### 6.2 Installing the Linux Driver

This section is based on a ZIP file (Broadcom Ethernet Network adapter Linux Installer) which is extracted to a directory referenced as follows as \$REL\_DIR.

Included with the installation package is an automation tool for software installation and system configuration. This is the preferred installation method. The installer provides the following: Installs the included drivers, firmware, RDMA library, and utility tools. Additional system packages are automatically installed as needed:

- ansible, libibverbs-utils, rdma-cm-utils, perfest
- gcc, make, rpmbuild, kernel headers: if necessary to compile drivers or library

To start the automated installation with default values and all above components for an interface that is already up and has an IP address:

```
cd $REL_DIR/Linux/Linux_Installer
sudo bash install.sh -i <IFACE>
```

**NOTE:** <IFACE> must be replaced with the interface name or the device's PCIe address: For example, p1p1 or 41:00.0

To install only the L2/Ethernet driver without installing/configuring RoCE:

```
cd $REL_DIR/Linux/Linux_Installer
sudo bash install.sh -i <IFACE> -2
```

To start the automated installation with IP address and MTU specified with verbose output:

```
cd $REL_DIR/Linux/Linux_Installer
sudo bash install.sh -v -i <IFACE> -a <IP> -n <NETMASK> -m <MTU>
```

For a complete explanation of the automated installer including all options and modes of use, see the `README` file in `$REL_DIR/Linux/Linux_Installer`.

## 6.2.1 The automated installer installs/updates both the L2 and RoCE drivers. In order to utilize the RoCE feature, see the **Manual Driver Installation**

The `bnxt_en` and `bnxt_re` modules must come from the same package. Extract, compile, and install both drivers using the commands in the following sections.

### 6.2.1.1 Compile and Install from the Source RPM

```
cd $REL_DIR/Linux/KMP-L2-RoCE/KMP/<Distro>/<Distro Version>
sudo rpmbuild --rebuild bnxt_en-x.y.z.src.rpm
sudo rpm -i /root/rpmbuild/RPMS/x86_64/kmod-bnxt_en-x.y.z.rpm
sudo depmod -a
sudo modprobe bnxt_en
sudo modprobe bnxt_re
```

### 6.2.1.2 Compile and Install Directly from the Source

```
cd $REL_DIR/Linux/Linux_Driver
tar xf netxtreme-bnxt_en-x.y.z.tar.gz
cd netxtreme-bnxt_en-x.y.z
sudo make
sudo make install
sudo depmod -a
sudo modprobe bnxt_en
sudo modprobe bnxt_re
```

## 6.2.2 Updating Initramfs

Most Linux distributions use a ramdisk image to store the drivers for boot-up. These kernel modules take precedence, therefore, the initramfs must be updated after installing the new `bnxt_en/bnxt_re` modules:

### 6.2.2.1 Debian/Ubuntu

```
sudo update-initramfs -u
```

### 6.2.2.2 Red Hat/CentOS/Fedora

```
sudo dracut -f
```

## 6.2.3 Linux Ethtool Commands

In [Table 28](#), `ethX` should be replaced with the actual interface name.

**Table 28: Linux Ethtool Commands**

Command	Description
<code>ethtool -s ethX speed 25000 autoneg off</code>	Force the speed to 25G. If the link is up on one port, the driver does not allow the other port to be set to a different speed.
<code>ethtool -i ethX</code>	Output includes driver, firmware, and package version.
<code>ethtool -k ethX</code>	Show offload features.
<code>ethtool -K ethX tso off</code>	Turn off TSO.
<code>ethtool -K ethX gro off lro off</code>	Turn off GRO/LRO.
<code>ethtool -g ethX</code>	Show ring sizes.
<code>ethtool -G ethX rx N</code>	Set Ring sizes.
<code>ethtool -S ethX</code>	Get statistics.
<code>ethtool -l ethX</code>	Show number of rings.
<code>ethtool -L ethX rx 0 tx 0 combined M</code>	Set number of rings.
<code>ethtool -C ethX rx-frames N</code>	Set interrupt coalescing. Other parameters supported are: rx-usecs, rx-frames, rx-usecs-irq, rx-frames-irq, tx-usecs, tx-frames, tx-usecs-irq, tx-frames-irq.
<code>ethtool -x ethX</code>	Show RSS flow hash indirection table and RSS key.
<code>ethtool -s ethX autoneg on speed 10000 duplex full</code>	Enable Autoneg (see <a href="#">Auto-Negotiation Configuration</a> for additional information).
<code>ethtool --show-eee ethX</code>	Show EEE state.
<code>ethtool --set-eee ethX eee off</code>	Disable EEE.
<code>ethtool --set-eee ethX eee on tx-lpi off</code>	Enable EEE, but disable LPI.
<code>ethtool -L ethX combined 1 rx 0 tx 0</code>	Disable RSS. Set the combined channels to 1.
<code>ethtool -K ethX ntuple off</code>	Disable Accelerated RFS by disabling ntuple filters.
<code>ethtool -K ethX ntuple on</code>	Enable Accelerated RFS.
<code>ethtool -t ethX</code>	Performs various diagnostic self-tests.
<code>echo 32768 &gt; /proc/sys/net/core/rps_sock_flow_entries</code> <code>echo 2048 &gt; /sys/class/net/ethX/queues/rx-X/rps_flow_cnt</code>	Enable RFS for Ring X.
<code>sysctl -w net.core.busy_read=50</code>	This sets the time to read the device's receive ring to 50 µsecs. For socket applications waiting for data to arrive, using this method can decrease latency by 2 or 3 usecs typically at the expense of higher CPU utilization.
<code>echo 4 &gt; /sys/class/net/&lt;NAME&gt;/device/sriov_numvfs</code>	Enable SR-IOV with four VFs on named interface.
<code>ip link set ethX vf 0 mac 00:12:34:56:78:9a</code>	Set VF MAC address.
<code>ip link set ethX vf 0 state enable</code>	Set VF link state for VF 0.
<code>ip link set ethX vf 0 vlan 100</code>	Set VF 0 modprobe 8021q; ip link add link <NAME> name <VLAN Interface Name> type vlan id <VLAN ID> <b>Example:</b> modprobe 8021q; ip link add link ens3 name ens3.2 type vlan id 2

## 6.3 Installing the VMware Driver

The ESX drivers are provided in VMware standard VIB format and can be downloaded from [VMware.com](http://VMware.com).

1. To install the Ethernet and RDMA driver, issue the following commands:

```
$ esxcli software vib install -v <bnxtnet>--<driver version>.vib
```

```
$ esxcli software vib install -v <bnxtroce>--<driver version>.vib
```

2. A system reboot is required for the new driver to take effect.

Other useful VMware commands are shown in [Table 29](#).

**NOTE:** In [Table 29](#), replace vmnicX with the actual interface name.

**NOTE:** `$ kill -HUP $(cat /var/run/vmware/vmkdevmgr.pid)`  
This command is required after `vmkload_mod bnxtnet` for successful module bring up.

**NOTE:** NPAR + SR-IOV and NPAR + MultiRSS are currently not supported due to resource constraints.

**Table 29: VMware Commands**

Command	Description
<code>esxcli software vib list  grep bnx</code>	List the VIBs installed to see whether the bnxt driver installed successfully.
<code>esxcfg-module -I bnxtnet</code>	Print module info on to screen.
<code>esxcli network get -n vmnicX</code>	Get vmnicX properties.
<code>esxcfg-module -g bnxtnet</code>	Print module parameters.
<code>esxcfg-module -s 'multi_rx_filters=2 disable_tap=0 max_vfs=0,0 RSS=0'</code>	Set the module parameters.
<code>vmkload_mod -u bnxtnet</code>	Unload bnxtnet module.
<code>vmkload_mod bnxtnet</code>	Load bnxtnet module.
<code>esxcli network nic set -n vmnicX -D full -S 25000</code>	Set the speed and duplex of vmnicX.
<code>esxcli network nic down -n vmnicX</code>	Disable vmnicX.
<code>esxcli network nic up -n vmnic6</code>	Enable vmnicX.

## 6.4 Installing the Windows Driver

The driver installer can be downloaded from the following Dell Support website:

- <http://support.dell.com>

Download the installer for your platform and adapter using the URL and follow the instructions in the readme.txt file to install the driver.

## 7 Updating the Firmware

This section provides information for updating the adapter firmware.

### 7.1 Dell Update Package

DUP packages can be downloaded from <http://support.dell.com>. See the following sections to use the Dell Update Package (DUP):

#### 7.1.1 Windows

Broadcom Ethernet Network adapter firmware can be upgraded using the Dell DUP package. The executable is provided in standard Windows x64 executable format. Double-click on the file to execute it.

**NOTE:** During a hot firmware upgrade or error recovery, there may be a period of time that the firmware can not respond to a command sent from the drivers while the firmware is performing a reset. The driver logs this as an error in the event log. During normal operation, the firmware continues the reset and returns to normal operation. As long as the firmware hot upgrade/error recovery is completed successfully, this error can be ignored. This is verified by observing the *Firmware became responsive* and *Firmware reset sequence completed* events in the event log.

**NOTE:** DUP packages can be downloaded from <http://support.dell.com>

#### 7.1.2 Linux

The Dell Linux DUP is provided in `x86_64` executable format. Use the standard Linux `chmod` to update the execute permission and run the executable. See the following example:

1. Login to Linux.
2. `scp` or `cp` the DUP executable on to file system. A typical example is:

```
cp /var/run/media/usb/Network_Firmware_<version>.BIN /root/
```

3. Execute the following command:

```
chmod 755 Network_Firmware_<version>.BIN
```

4. Execute the following command:

```
./Network_Firmware_<version>.BIN
```

A reboot is needed to activate the new firmware.

The installer script updates the firmware along with the drivers

To start the automated installation with the device interface that is already up and has an IP address:

```
cd $REL_DIR/Linux/Linux_Installer
sudo bash install.sh -i <IFACE>
```

**NOTE:** `<IFACE>` must be replaced with the interface name or the device's PCIe address: For example, `p1p1` or `41:00.0`.

## 8 Link Aggregation

The following sections provide information on link aggregation.

### 8.1 Windows

Broadcom Ethernet Network adapters can aggregate network links using the Microsoft teaming feature. For more information on the NIC teaming functionality, see the Microsoft public documentation on [Microsoft.com](https://www.microsoft.com).

Microsoft LBFO is a native teaming driver that can be used in the Windows OS. The teaming driver also provides VLAN tagging capabilities.

### 8.2 Linux

The Linux bonding module is used for link aggregation under Linux. For additional documentation on Linux bonding, see <https://www.kernel.org/doc/Documentation/networking/bonding.txt>.

#### 8.2.1 Ephemeral Bonds

Use the following steps as an example to create a bond interface that is ephemeral:

1. Load the bonding module using the following command:

```
sudo modprobe bonding
```

2. Create a bond interface named bond0 and mode balance-alb using the following command:

```
sudo ip link add bond0 type bond
```

3. Bring down the first interface that will be added to the bond interface using the following command:

```
sudo ip link set enp9s0f0np0 down
```

4. Add the first interface to the bond interface using the following command:

```
sudo ip link set enp9s0f0np0 master bond0
```

5. Bring down the second interface that will be added to the bond interface using the following command:

```
sudo ip link set enp9s0f1np1 down
```

6. Add the second interface to the bond interface using the following command:

```
sudo ip link set enp9s0f1np1 master bond0
```

7. Assign an IP address to the bond interface using the following command:

```
sudo ip addr add 192.168.2.35/16 dev bond0
```

8. Bring up the bond interface using the following command:

```
sudo ip link set bond0 up
```

## 8.2.2 Bonding Interface Queries

Details of the bonding interface can be queried using the following command

```
cat /proc/net/bonding/bond0
```

Use the following steps to setup basic Linux bonding:

1. Load the bonding module using the following command:

```
modprobe bonding mode="balance-alb"
```

2. Add physical network interfaces to the bond interface using the following commands:

```
ifenslave bond0 ethX  
ifenslave bond0 ethY
```

3. Assign an IP address to bond the interface. IPV4Address and NetMask are an IPv4 address and the associated network mask.

```
ifconfig bond0 <IP address> netmask <netmask>
```

## 9 System-Level Configuration

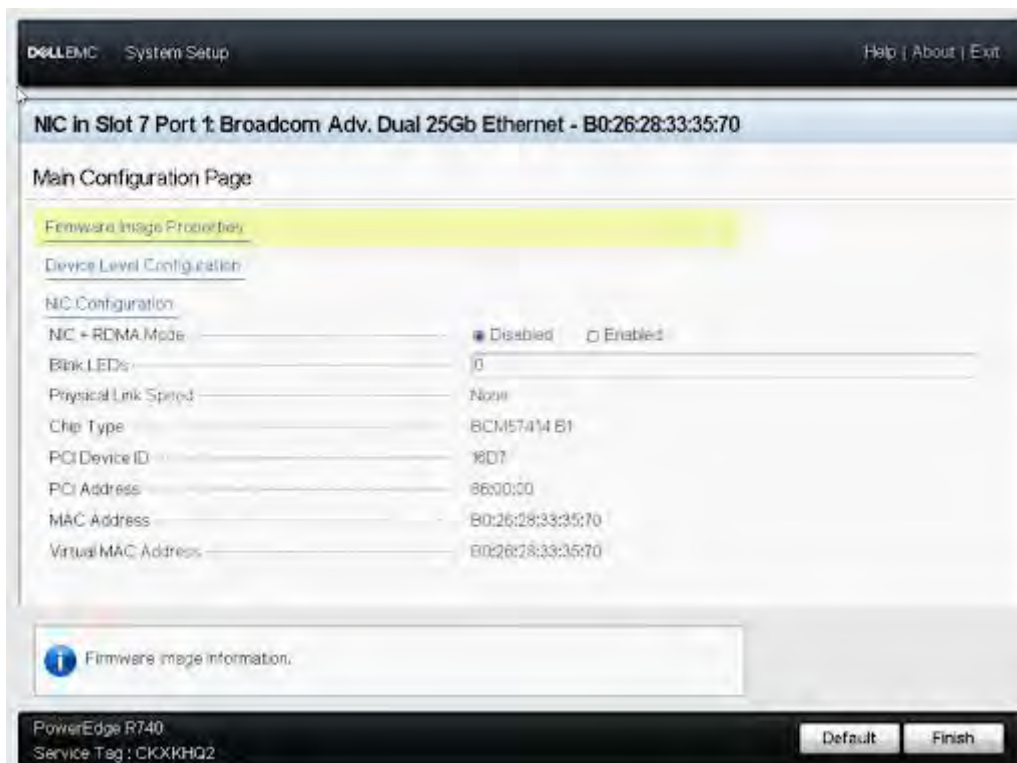
The following sections provide information on system-level NIC configuration.

### 9.1 UEFI HII Menu

Broadcom Ethernet Network adapters can be configured using the HII (Human Interface Infrastructure) menu at boot time. This menu system allows configuration of all persistent settings such as boot protocol (PXE) and virtualization modes (SR-IOV, NPAR), and so on. To enter the HII configuration menu, follow boot-time prompts to BIOS, then device configuration. The layout of the menus of an adapter may not look the same as the others and some settings may not be available or found on the same menu for a different adapter type.

#### 9.1.1 Main Configuration Page

Figure 34: Main Configuration Page



This page displays the following information (see [Figure 34](#)):

- **Firmware Image Menu** – This menu presents the various component versions present in the current firmware package.
- **Device Configuration Menu** – This menu presents adapter specific parameters for configuration.
- **NIC Configuration** – This menu presents PXE boot related parameters for configuration.
- **NIC Partitioning Configuration Menu** – This menu presents NIC partition related parameters for configuration.



- **NIC + RDMA Mode** – This setting configures Remote Direct Memory Access (RDMA) support on the port. RDMA is a technology that permits computers on a network to exchange data in main memory without the involvement of the processor, cache or operating system of either computer. RDMA allows high throughput and low-latency networking. This setting is available only when RDMA is supported on the adapter. This setting will be displayed on the Device Configuration menu in SF mode and in the NIC Partition Configuration menu in NPAR mode.
  - **Enabled** – Turn on RDMA
  - **Disabled** – Turn off RDMA

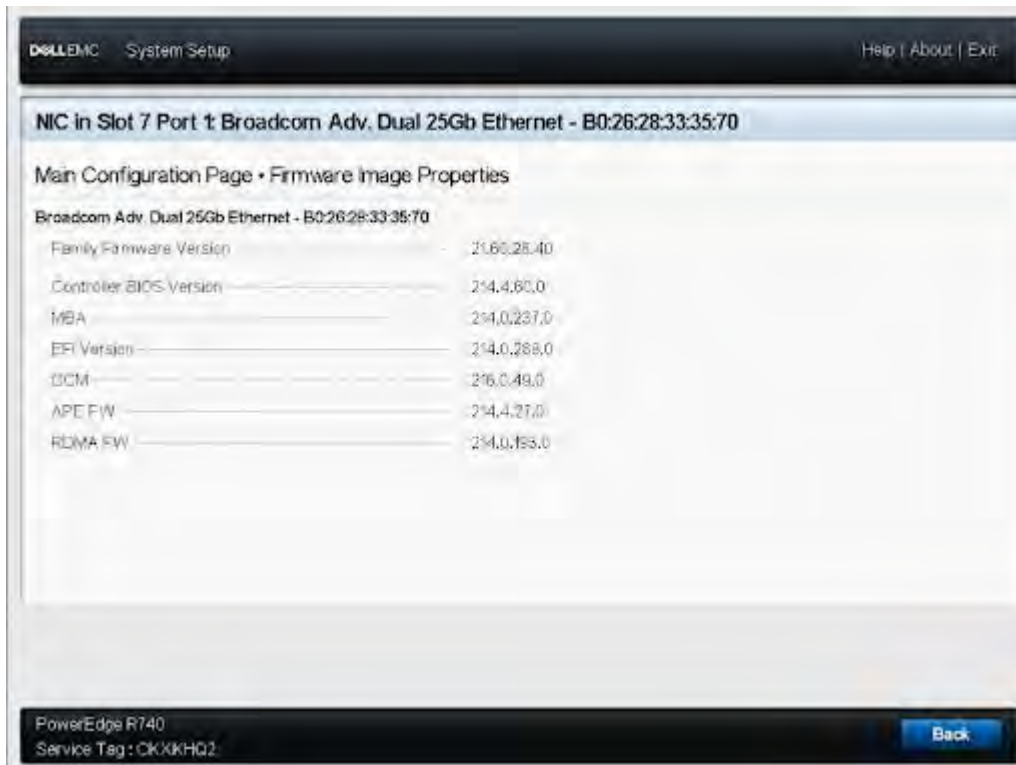
**NOTE:** When NIC + RDMA Mode is enabled, UDP port 4791 is reserved for RDMA use. Therefore, is not available to clients and services running on the host operating system.

- **Blink LEDs** – This setting allows the user to configure the duration for which the LEDs on the physical network port should blink to assist with port identification. This is a numeric setting. The value must be specified in the range 0 to 15 seconds.
- **Link Status** – This field displays the physical link status of the network port as reported by the controller. This is a read-only field.
  - **Connected** – Link is up
  - **Disconnected** – Link is down
- **Physical Link Speed** – This field displays the current link speed of the network port as reported by the controller. This is a read-only field. Speed is reported in Mb/s/Gb/s.
- **Chip Type** – This field displays the Broadcom specific identifier which denoted the adapter family to which the chip belongs and the revision. This is a read-only field.
- **PCI Device ID** – This field displays the 16 bit PCI Device ID reported by the controller. This is a vendor defined ID which varies across non-NPAR, NPAR and RDMA mode. Refer to MF mode and Support RDMA sections for more information on these modes. This is a read-only field.
- **Bus Device Function** – This field displays the BIOS assigned PCI Bus:Device:Function identifier of the card. This is a read-only field.
- **Permanent MAC Address** – This field displays the Permanent MAC address assigned during manufacturing. This is a read-only field.
- **Virtual MAC Address** – This field displays the Virtual MAC address assigned to the device. This is a read-only field from the HII menu. The value for this parameter can be configured using remote utilities.

## 9.1.2 Firmware Image Menu

This menu presents the various component versions present in the current firmware build. Depending on the adapter type, some components may not be available. All fields in this menu are read-only.

**Figure 35: Firmware Image Menu**



This page displays the following information (see [Figure 35](#)):

- **Family Firmware Version** – This field displays the family firmware version. This field may be displayed as Firmware Bundle on some adapters.
- **Boot Code** – This field displays the firmware boot code version.
- **MBA** – This field displays the legacy pre-boot driver version.
- **EFI** – This field displays the UEFI pre-boot driver version.
- **NC-SI** – This field displays the NCSI firmware version.
- **RDMA FW** – This field displays the RoCE firmware version.

### 9.1.3 Device Configuration Menu

This menu presents adapter specific parameters for configuration. Depending on the adapter type, some settings may not be available.

Figure 36: Device Configuration Menu

**Integrated NIC 1 Port 1: Broadcom Adv. Dual 25Gb Ethernet - 00:0A:F7:31:43:40**

Main Configuration Page • Device Level Configuration

**Broadcom Adv. Dual 25Gb Ethernet - 00:0A:F7:31:43:40**

Virtualization Mode	None
NParEP Mode	<input type="radio"/> Disabled <input checked="" type="radio"/> Enabled
PCI Virtual Functions Advertised	8
Number of MSI-X Vectors per VF	16
Maximum Number of PF MSI-X Vectors	74
Link FEC	Disabled
Operational Link Speed	Auto Negotiated
DCBX Mode	Disabled
LLDP nearest bridge	<input type="radio"/> Disabled <input checked="" type="radio"/> Enabled
LLDP nearest non-TPMR bridge	<input type="radio"/> Disabled <input checked="" type="radio"/> Enabled
Auto-negotiation Protocol	IEEE and Consortium
Media Auto Detect	<input type="radio"/> Disabled <input checked="" type="radio"/> Enabled
Default EVB Mode	<input checked="" type="radio"/> VEB <input type="radio"/> VEPA <input type="radio"/> None
Flow Offload	<input checked="" type="radio"/> Disabled <input type="radio"/> Enabled
Adapter Error Recovery	<input checked="" type="radio"/> Disabled <input type="radio"/> Enabled

This page allows the user to configure the following items (see [Figure 36](#)):

- **Virtualization Mode** – This setting configures the type of virtualization to be used by the controller on all ports. This is available only when NPAR is supported on the adapter.
  - **Single Function Mode (SF)** – In this mode, a single PCIe PF is assigned to each network port.
  - **Network Partitioning Mode (NPAR)** – This mode allows a single physical network port to appear to the system as multiple network device functions. Each PF or partition is assigned a separate PCIe function ID on initial power on, and a menu is made visible in setup to configure each partition. The 16 configurable partitions are distributed equally across the network ports.
  - **SR-IOV** – This setting configures Single Root - I/O Virtualization (SR-IOV) which allows different virtual machines (VMs) in a virtual environment to share a single PCI Express hardware interface.
    - **Enabled** – Enable support for SR-IOV
    - **Disabled** – Disable support for SR-IOV

**NOTE:** This setting is available only on adapters which support SR-IOV.

- **PCI Virtual Functions Advertised** – This setting allows the user to configure the number of PCI Virtual Functions Advertised by the port in PCI config space when SR-IOV is enabled in non-NPAR mode. This setting is available for configuration only in non-NPAR mode. This is a numeric setting. The value must be specified in multiples of 8.

**NOTE:** The maximum number of VFs supported by software is 128 VFs per device.

■ **Number of MSI-X Vectors per VF**

This setting configures the MSI-X Vectors per VF. Message Signaled Interrupts (MSI) are an alternative in-band method of signaling an interrupt using special in-band messages to replace traditional out-of-band assertion of dedicated interrupt lines. This is a numeric setting. The maximum number of virtual functions supported by the adapter are shared across the number of physical ports on the adapter. The default value for BCM574XX is 16 and 8 for the BCM5750X.

■ **Maximum Number of PF MSI-X Vectors**

This setting allows the user to configure the Maximum Number of MSI-X Vectors for a physical function. This is a numeric setting. The minimum value for this setting is 0. The maximum value varies across adapters. The default value for this option is 74.

**NOTE:** See [Table 30](#) for a list of recommended settings for maximum VF support. Exceeding the recommended settings may result in undesirable behavior, such as failure of network interfaces to start.

**Table 30: Recommended Configurations (BCM5750X Only)**

	PCIe Physical Functions	VFs per PF	MSI-X Vector per PF (absolute maximum)	MSI-X Vector per VF
Single-Port Adapter	1	128	240	8
Dual-Port Adapter	2	64	120	8
Quad-Port Adapter	4	32	60	8
With NPAR Enabled (any port count)	8	16	94	4
With NPAR EP Enabled (any port count)	16	8	47	4

- **Link FEC** – This setting configures the Forward Error Correction (FEC) mode which is a technique used for controlling errors in data transmission over unreliable or noisy communication channels. This option is useful when longer fiber cables are utilized. This setting is not available on 10G BaseT controllers. Only a subset of the possible values display on some adapters, based on the configuration. Possible values are:
- Disabled
  - CL74 – Fire Code
  - CL91 – Reed Solomon
  - RS544 – RS544, using 1 x N RS
  - RS272 – RS272, using 1 x N RS
  - RS544 – RS544, using 2 x N RS
  - RS272 – RS272, using 2 x N RS

**NOTE:** When Media Auto Detect and Auto-negotiation is enabled, FEC is negotiated based on the link partner advertisement.

- **Energy Efficient Ethernet** – This setting configures the Energy Efficient Ethernet (EEE) mode that allow for less power consumption during periods of low-data activity. This setting is available only on 10GBASE-T controllers.
- **Enabled** – Turn on EEE mode
  - **Disabled** – Turn off EEE mode
- **Operational Link Speed** – This setting configures the default link speed for pre-OS environment in full-power (D0) state. The possible values for this setting depends on the link speeds supported by the adapter.

**NOTE:** The value for this setting is fixed on some adapters based on the configuration.

- **DCB Protocol** – This setting configures the Data Center Bridging (DCB) settings for the controller. Some of the following options may not be available depending on the adapter in use.
  - **Enabled (IEEE only)**
  - **CEE (only)**
  - **Both (IEEE preferred with fallback to CEE)**
- **LLDP nearest bridge** – This setting configures the Link Layer Discovery Protocol (LLDP) which is a vendor-neutral link layer protocol used by network devices for advertising their identity, capabilities, and neighbors on a local area network based on IEEE 802 technology, principally wired Ethernet. An LLDP agent is a mapping of an entity where LLDP runs.
  - **Enabled** – Turn on LLDP nearest bridge
  - **Disabled** – Turn off LLDP nearest bridge
- **LLDP nearest non-TPMR bridge** – This setting configures the Link Layer Discovery Protocol (LLDP) which is a vendor-neutral link layer protocol used by network devices for advertising their identity, capabilities, and neighbors on a local area network based on IEEE 802 technology, principally wired Ethernet. An LLDP agent is a mapping of an entity where LLDP runs. This setting enables LLDP on the nearest non-TPMR bridge agent.
  - **Enabled** – Turn on LLDP nearest non-TPMRbridge
  - **Disabled** – Turn off LLDP nearest non-TPMRbridge
- **Auto-negotiation Protocol** – This setting configures the Auto-negotiation protocol for the adapter. Auto-negotiation is a feature that allows a port on a switch, router, server, or other device to communicate with the device on the other end of the link to determine the optimal duplex mode and speed for the connection. This setting is not available on 10GBASE-T controllers.
  - **IEEE and BAM**
  - **IEEE and Consortium**
  - **BAM Only**
  - **Consortium Only**
  - **IEEE 802.3by**
- **Media Auto Detect** – This setting allows the user to configure the Media Auto Detect feature. This setting is not available on 10GBASE-T controllers.
  - **Enabled** – Turn on Media Auto Detect.
  - **Disabled** – Turn off Media Auto Detect.
- **Default EVB Mode** – This setting configures the Edge Virtual Bridging (EVB) mode which is an IEEE standard that involves the interaction between virtual switching environments in a hypervisor and the first layer of the physical switching infrastructure.
  - **VEB**
  - **VEPA**
  - **None**
- **Port Link Training** – This setting configures port link training when using force link speed. Link training should be enabled when using PAM-4. Link training should be disabled when the attached switch does not support link training.
  - **Enabled** – Port Link Training
  - **Disabled** – Port Link Training
- **Adapter Error Recovery** – This feature enables the recovery of firmware from fatal errors without manual intervention, host reboot, or power cycle. This feature is supported on Linux only.
  - **Enabled** – Adapter Error Recovery
  - **Disabled** – Adapter Error Recovery

## 9.1.4 NIC Configuration

This menu presents legacy boot related parameters for configuration. Depending on the adapter type, some settings may not be available.

**Figure 37: NIC Configuration**

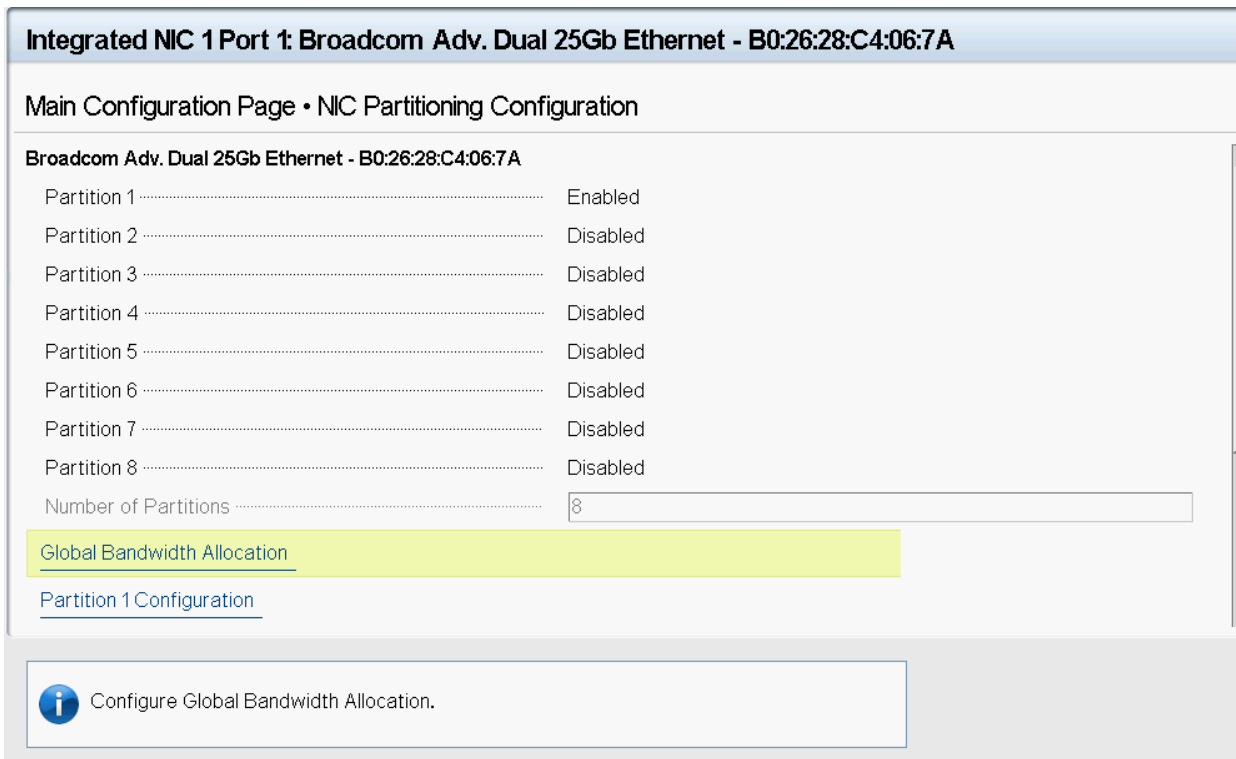
The NIC Configuration menu consists of the following items (see [Figure 37](#)):

- **Option ROM** – This setting allows the user to control whether the Broadcom legacy option ROM driver should be advertised or not.
  - **Enabled** – Load legacy option ROM driver
  - **Disabled** – Do not load legacy option ROM driver
- **Boot Strap Type** – This setting configures the bootstrap protocol for legacy PXE boot.
  - **Auto Detect**
  - **BBS**
  - **Int 18h**
  - **Int 19h**
- **Banner Message Timeout** – This setting configures the duration for which the Broadcom legacy option ROM banner is to be displayed on the screen during POST. This is a numeric setting. The value must be specified in the range 0 to 15 seconds.
- **Pre-boot Wake on LAN** – This setting configures Wake on LAN (WoL) which is the ability to remotely power on a server or to wake it up from sleep mode. This setting is available only on adapters that support the Wake on LAN feature. This setting is available only on adapters that support the **Wake on LAN** feature.
  - **Enabled** – Turn on WoL

- **Disabled** – Turn off WoL
- **VLAN Mode** – This setting configures a virtual LAN.
  - **Enabled** – Turn on VLAN mode
  - **Disabled** – Turn off VLAN mode
- **VLAN ID** – This setting configures a VLAN tag when VLAN mode is enabled. This is a numeric setting. The value must be specified in the range 1 to 4094.
- **Boot Retry Count** – This setting configures the number of times legacy boot must be attempted in case of failure.
  - **No Retry**
  - **1 Retry**
  - **2 Retries**
  - **3 Retries**
  - **4 Retries**
  - **5 Retries**
  - **6 Retries**
  - **Indefinite Retries**
- **Permit Total Port Shutdown** – This feature is supported on Linux OS only. This setting allows the port to be completely disabled when a port down command is received from the host OS or drive. This feature is not supported when virtualization mode is NPAR or NPAR + SRIOV.
  - **Enabled** – Permit Total Port Shutdown
  - **Disabled** – Permit Total Port Shutdown

## 9.1.5 NIC Partitioning Configuration Menu

Figure 38: NIC Partitioning Configuration Menu



**Integrated NIC 1 Port 1: Broadcom Adv. Dual 25Gb Ethernet - B0:26:28:C4:06:7A**

Main Configuration Page • NIC Partitioning Configuration


Broadcom Adv. Dual 25Gb Ethernet - B0:26:28:C4:06:7A

Partition 1 .....	Enabled
Partition 2 .....	Disabled
Partition 3 .....	Disabled
Partition 4 .....	Disabled
Partition 5 .....	Disabled
Partition 6 .....	Disabled
Partition 7 .....	Disabled
Partition 8 .....	Disabled

Number of Partitions .....

[Global Bandwidth Allocation](#)

[Partition 1 Configuration](#)

 Configure Global Bandwidth Allocation.

The **NIC Partitioning Configuration** screen has the following sub-menus (see [Figure 38](#)):

- **Number of Partitions Per Port** – This field displays the number of PCI Physical functions currently enabled on the current network port when MF mode is set to NPAR. This is a read-only field.
- **Partition <n> Configuration** – This menu presents configuration parameters for the ‘n’th partition on the network port in NPar mode. The number of menus presented depends on the value of the **Number of Partitions** field.



### 9.1.5.1 Partition <n> Configuration Menu

This menu presents partition related parameters for configuration. Depending on the adapter type, some settings may not be available (see [Figure 39](#)).

Figure 39: Partition <n> Configuration

**Integrated NIC 1 Port 1: Broadcom Adv. Dual 25Gb Ethernet - B0:26:28:C4:06:7A**

Main Configuration Page • NIC Partitioning Configuration • Partition 1 Configuration

---

**Broadcom Adv. Dual 25Gb Ethernet - B0:26:28:C4:06:7A**

BW Reservation Valid .....	<input checked="" type="radio"/> False <input type="radio"/> True
BW Limit Valid .....	<input checked="" type="radio"/> False <input type="radio"/> True
NIC + RDMA Mode .....	<input checked="" type="radio"/> Disabled <input type="radio"/> Enabled
MAC Address .....	B0:26:28:C4:06:7A
Virtual MAC Address .....	B0:26:28:C4:06:7A
PCI Device ID .....	16D7
PCI Address .....	01:00

Configure BW Reservation Valid. Functions as an on/off switch for the BW Reservation setting.

---

**Integrated NIC 1 Port 1: Broadcom Adv. Dual 25Gb Ethernet - B0:26:28:C4:06:7A**

Main Configuration Page • NIC Partitioning Configuration • Global Bandwidth Allocation

---

**Broadcom Adv. Dual 25Gb Ethernet - B0:26:28:C4:06:7A**

Partition 1 Minimum TX Bandwidth .....	<input type="text" value="0"/>
Partition 1 Maximum TX Bandwidth .....	<input type="text" value="100"/>
Partition 2 Minimum TX Bandwidth .....	<input type="text" value="0"/>
Partition 2 Maximum TX Bandwidth .....	<input type="text" value="100"/>
Partition 3 Minimum TX Bandwidth .....	<input type="text" value="0"/>
Partition 3 Maximum TX Bandwidth .....	<input type="text" value="100"/>
Partition 4 Minimum TX Bandwidth .....	<input type="text" value="0"/>
Partition 4 Maximum TX Bandwidth .....	<input type="text" value="100"/>
Partition 5 Minimum TX Bandwidth .....	<input type="text" value="0"/>
Partition 5 Maximum TX Bandwidth .....	<input type="text" value="100"/>
Partition 6 Minimum TX Bandwidth .....	<input type="text" value="0"/>

Configure BW Reservation. Percentage of total available bandwidth that should be reserved for this partition. Combined values for all active ... (Press <F1> for more help)

- **BW Reservation** – This setting configures the percentage of total available bandwidth that should be reserved for this partition. The total Bandwidth Reservation assigned for all active partitions cannot exceed 100. A value of 0 on all partitions indicates equal division of bandwidth between all partitions. This setting is available only on adapters that support the Bandwidth Reservation feature. This is a numeric setting. The value must be specified in the range 0 to 100.
- **BW Limit** – This setting configures the maximum percentage of available bandwidth this partition is allowed. This is a numeric setting. The value must be specified in the range 0 to 100.
- **BW Reservation Valid** – This setting configures whether BW Reservation is applicable on the current partition. When this setting is disabled, the BW Reservation value for the current partition will be ignored. This setting is available only on adapters that support the Bandwidth Reservation feature.
  - **Enabled** – Turn on BW Reservation
  - **Disabled** – Turn off BW Reservation
- **BW Limit Valid** – This setting configures whether BW Limit is applicable on the current partition. When this setting is disabled, the BW Limit value for the current partition will be ignored.
  - **Enabled** – Turn on BW Limit
  - **Disabled** – Turn off BW Limit

**NOTE:** Not all devices have all of the options available.

- **NIC + RDMA Mode** – This setting configures the RoCE support for the current partition. This setting is available only when RDMA is supported on the current partition. This setting will be displayed on the Device Configuration menu in SF mode and in the **NIC Partition Configuration** menu in NPar mode.
  - **Enabled** – Turn on RDMA
  - **Disabled** – Turn off RDMA

**NOTE:** When NIC + RDMA Mode is enabled, UDP port 4791 is reserved for RDMA use. Therefore, is not available to clients and services running on the host operating system.

- **MAC Address** – This field displays the Permanent MAC address assigned during manufacturing for the current partition. This is a read-only field.
- **Virtual MAC Address** – This field displays the Virtual MAC address assigned to the current partition. This is a read-only field from the HII menu. The value for this parameter can be configured using remote utilities.

## 9.2 Auto-Negotiation Configuration

**NOTE:** In NPAR (NIC partitioning) devices where one port is shared by multiple PCI functions, the port speed is preconfigured and cannot be changed by the driver.

Broadcom Ethernet Network adapters support the following auto-negotiation features:

- Link speed auto-negotiation
- FEC auto-negotiation
- Pause/Flow Control auto-negotiation

**NOTE:** When using SFP+, SFP28 connectors, use DAC or multimode optical transceivers capable of supporting auto-negotiation. Ensure that the link partner port has been set to the matching auto-negotiation protocol. For example, if the local Broadcom port is set to IEEE 802.3by AN protocol, the link partner must support auto-negotiation and must be set to IEEE 802.3by auto-negotiation protocol. When Media Auto Detect and IEEE 802.by + Consortium are enabled, the controller auto detects the Ethernet cables and transceivers to start auto-negotiation with the link partner. FEC is negotiated during auto-negotiation and forced FEC does not take effect in this mode. The default setting enables IEEE 802.by + Consortium and Media Auto Detect.

The supported combination of link speed settings for two port Broadcom Ethernet Network adapters are shown in [Table 31](#).

**Table 31: Supported Link Speeds for the BCM95741X and BCM95750X**

Dual-Port	BCM95741X	BCM9575XX
1G, 1G	Yes	No
1G, 10G	Yes	No
1G, 25G	Yes	No
10G, 10G	Yes	Yes
10G, 25G	No	Yes
25G, 25G	Yes	Yes
50G, 50G	No	Yes
100G, 100G	No	Yes
Quad-Port		
10G, 10G, 10G, 10G	N/A	Yes
10G, 10G, 10G, 25G	N/A	Yes
10G, 10G, 25G, 25G	N/A	Yes
10G, 25G, 25G, 25G	N/A	Yes
25G, 25G, 25G, 25G	N/A	Yes

**NOTE:** 1 Gb/s link speed for SFP+/SFP28 is currently not supported in this release.

- P1 – Port 1 setting.
- P2 – Port 2 setting.
- AN – Auto-negotiation.
- No AN – Forced speed.
- {link speed} – Expected link speed.
- The BCM57414 does not support independent link speed. All ports must operate at the same speed. For example, the ports cannot be set for port1 = 10G and port2 = 25G. Only the BCM5750X supports independent port speeds.

#### BCM957504 Supported Combinations of Link Speed Settings (10/25G-NRZ)

- All port link speeds are independent of each other.
- Each port can be configured as forced for any speed supported by the device.
- AN {link speeds} – Advertised supported auto-negotiation link speeds.

#### BCM957508 Supported Combinations of Link Speed Settings (25/50/100G NRZ and 50/100G PAM-4)

- All port link speeds are independent of each other.
- Each port can be configured as forced or auto-negotiate (NRZ Only) for any speed supported by the device.
- AN {link speeds} – Advertised supported auto-negotiation link speeds.
- PAM-4 support is enabled at a feature preview level with the following notes/restrictions:
  - PAM-4 operation on both ports is supported. NRZ operation on both ports is supported. PAM-4 operation on one port with NRZ operation on the other is not supported.
  - PAM-4 supports fixed speed mode only and must be configured through UEFI/HII.
  - PAM-4 support in this release is limited to DAC/Twinax cables.
  - PAM-4 operation on DAC/Twinax cables requires link training to be enabled. This requires a switch that supports link training an PAM-4.
  - PAM-4 requires FEC mode as RS544\_1xN for 50G and 100G operation.
  - PAM-4 has been tested with the following DAC:
    - DAC Cable: Vendor: Amphenol QSFP56, Part num: NDAAXJ-0003

- AOC Cable: Vendor: Optomind Inc, Part num: C4R448GA005AZZ
- PAM-4 on Linux requires that 50/100G PAM-4 is enabled via UEFI menus only.
- Currently the Linux kernel and associated tools have infrastructure to report link mode. From the reported link mode one can derive the encoding or lanes that may be used for that link. In a case where this is auto-negotiated, this works well. A device reporting a link mode of 100000baseCR2/Full uses a different number of lanes (and therefore encoding) than a device reporting 100000baseCR4/Full. Unfortunately when auto-negotiation is not used there is not ample infrastructure to specify speed and encoding and differentiate between a card using NRZ vs PAM-4 encoding. Currently ethtool supports setting speed, but does not have support for setting encoding or lanes used:

- `ethtool -s|--change DEVNAME` Change generic options

```
[ speed %d ]
[ duplex half|full ]
[ port tp|aui|bnc|mii|fibre ]
[ mdix auto|on|off ]
[ autoneg on|off ]
[ advertise %x ]
[ phyad %d ]
[ xcvr internal|external ]
[ wol p|u|m|b|a|g|s|d... ]
[ sopass %x:%x:%x:%x:%x:%x ]
[ msglvl %d | msglvl type on|off ... ]
```

Recent discussions have been occurring on the upstream Linux networking mailing list to address this problem.

- PAM-4 operation on VMware requires that 50/100G PAM-4 is configured via HII menu and also configured via ESXCLI commands. A reboot is required to ensure proper persistence of these parameters.
  - The command is: `esxcli network nic set -n <interface> -S <speed> -D full`

**Example:** `esxcli network nic set -n vmnic4 -S 200000 -D full`

- PAM-4 operation with 50/100G on Windows Server requires registry key modification.

Windows does not contain a PAM-4 signaling configuration parameter in adapter device advanced properties. For speeds supported by both NRZ and PAM-4, Windows will default to NRZ.

The following work around is required for PAM-4:

- a. Examine the NXE device in Windows Device Manager, Network Adapters, Broadcom xxx.
- b. Right mouse click and select properties, select Details tab, select Driver Key from the Property drop down, and record the last 4 digits after the backslash.
- c. Enter the registry editor (regedit.exe) and navigate to:
  - `HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Control\Class\{4d36e972-e325-11ce-bfc1-08002be10318}\xxx` where xxx is the number recorded previously.
  - Add a DWORD type value named “PreferredSignalingMode”, and set the value to 1.
  - Save and exit the registry editor.
  - Disable and Enable device or Reboot server.

The expected link speeds based on the local and link partner settings are shown in [Table 32](#) and [Table 33](#).

**Table 32: Expected Link Speeds (Forced)**

Local Speed Settings	Link Partner Speed Settings						
	Forced 1G	Forced 10G	Forced 25G	Forced 50G	Forced 100G	Forced 200G	AN Enabled
Forced 1G	<b>1G</b>	No link	No link	No link	No link	No link	No link
Forced 10G	No link	<b>10G</b>	No link	No link	No link	No link	No link
Forced 25G	No link	No link	<b>25G</b>	No link	No link	No link	No link
Forced 50G	No link	No link	No link	<b>50G</b>	No link	No link	No link
Forced 100G	No link	No link	No link	No link	<b>100G</b>	No link	No link
Forced 200G	No link	No link	No link	No link	No link	<b>200G</b>	No link

**Table 33: Expected Link Speeds (Auto-Negotiate)**

Local Speed Settings	Link Partner Speed Settings								
	AN Enabled 1G	AN Enabled 10G	AN Enabled 25G	AN Enabled 1/10G	AN Enabled 1/25G	AN Enabled 10/25G	AN Enabled 1/10/25G	AN Enabled 10/25/50	AN Enabled 10/25/50/100
AN 1G	<b>1G</b>	No link	No link	<b>1G</b>	<b>1G</b>	No link	<b>1G</b>	No link	No link
AN 10G	No link	<b>10G</b>	No link	<b>10G</b>	No link	<b>10G</b>	<b>10G</b>	<b>10G</b>	<b>10G</b>
AN 25G	No link	No link	<b>25G</b>	No link	<b>25G</b>	<b>25G</b>	<b>25G</b>	<b>25G</b>	<b>25G</b>
AN 1/10G	<b>1G</b>	<b>10G</b>	No link	<b>10G</b>	<b>1G</b>	<b>10G</b>	<b>10G</b>	<b>10G</b>	<b>10G</b>
AN 1/25G	<b>1G</b>	No link	<b>25G</b>	<b>1G</b>	<b>25G</b>	<b>25G</b>	<b>25G</b>	<b>25G</b>	<b>25G</b>
AN 10/25G	No link	<b>10G</b>	<b>25G</b>	<b>10G</b>	<b>25G</b>	<b>25G</b>	<b>25G</b>	<b>25G</b>	<b>25G</b>
AN 1/10/25G	<b>1G</b>	<b>10G</b>	<b>25G</b>	<b>10G</b>	<b>25G</b>	<b>25G</b>	<b>25G</b>	<b>25G</b>	<b>25G</b>
AN 10/25/50G	No link	<b>10G</b>	<b>25G</b>	<b>10G</b>	<b>25G</b>	<b>25G</b>	<b>25G</b>	<b>50G</b>	<b>50G</b>
AN 10/25/50/100G	No link	<b>10G</b>	<b>25G</b>	<b>10G</b>	<b>25G</b>	<b>25G</b>	<b>25G</b>	<b>50G</b>	<b>100G</b>

To enable link speed auto-negotiation, the following options can be enabled in system BIOS HII menu:

**System BIOS**→**NetXtreme-E NIC**→**Device Level Configuration**

## 9.2.1 Operational Link Speed

This option configures the link speed used by the OS driver and firmware. This setting is overridden by the driver setting in the OS present state.

## 9.2.2 Firmware Link Speed

This option configures the link speed used by the firmware when the device is in D3.

## 9.2.3 Auto-Negotiation Protocol

This is the supported auto-negotiation protocol used to negotiate the link speed with the link partner. This option must match the AN protocol setting in the link partner port. Broadcom Ethernet Network adapters support the following auto-negotiation protocols: IEEE 802.3by, 25G/50G consortiums and 25G/50G BAM. By default, this option is set to IEEE 802.3by + Consortium and Media Auto Detect enabled.

Link speed and flow control/pause must be configured in the driver in the host OS.

## 9.2.4 Windows Driver Settings

To access the Windows driver settings:

- Open **Windows Manager**→**Broadcom NetXtreme E Series adapter**→**Advanced Properties**→**Advanced** tab

To enable Flow Control/Pause frame AN:

- Flow Control = Auto-Negotiation

To enable link speed AN:

- Speed and Duplex = Auto-Negotiation

## 9.2.5 Linux Driver Settings

**NOTE:** For 10GBASE-T Broadcom Ethernet Network adapters, auto-negotiation must be enabled.

- `ethtool -s eth0 speed 25000 autoneg off` – This command turns off auto-negotiation and forces the link speed to 25 Gb/s.
- `ethtool -s eth0 autoneg on advertise 0x0` – This command enables auto-negotiation and advertises that the device supports all speeds: 1G, 10G, 25G (and 40G, 50G if applicable).

The following are supported advertised speeds.

- `0x020` – 1000BASE-T Full
- `0x1000` – 1000BASE-T Full
- `0x80000000` – 25000BASE-CR Full
- `ethtool -A eth0 autoneg on|off` – Use this command to enable/disable pause frame auto-negotiation.
- `ethtool -a eth0` – Use this command to display the current flow control auto-negotiation setting.

## 9.2.6 ESXi Driver Settings

**NOTE:** For 10GBASE-T Broadcom Ethernet Network adapters, auto-negotiation must be enabled. Using forced speed on a 10GBASE-T adapter results in ESXCLI command failure.

- `$ esxcli network nic get -n <iface>` – This command shows the current speed, duplex, driver version, firmware version and link status.
- `$ esxcli network nic set -S 10000 -D full -n <iface>` – This command sets the forced speed to 10 Gb/s.
- `$ esxcli network nic set -a -n <iface>` – This enables link speed auto-negotiation on interface <iface>.
- `$ esxcli network nic pauseParams list` – Use this command to get pause Parameters list.
- `$ esxcli network nic pauseParams set --auto <1/0> --rx <1/0> --tx <1/0> -n <iface>` – Use this command to set pause parameters.

**NOTE:** Flow control/pause auto-negotiation can be set only when the interface is configured in link speed auto-negotiation mode.

## 9.3 Configuring 200G Link Speeds

Provides information for manually configuring 200G link speeds on Ethernet network adapters and switches.

The BCM57508 network adapter supports 200 gigabit per second link speed using QSFP56 optical transceivers or direct attach copper (DAC) cables. By default, 200G speed is disabled, allowing the more common 2 x 100G configuration to operate as expected. To use 200G speeds, the BCM57508 network adapter must be updated and special care must be taken with configuration of the network switch to enable the required PAM-4 modulation parameters and Forward Error Correction (FEC) mode.

**NOTE:** 200G link speeds are only supported on BCM57508 network adapters.

By default, the BCM57508 network adapter is configured as a 2-port device, supporting up to 100G on each port. To use 200G speed, the device must be reconfigured as a 1-port device and the 200G link speed must be enabled.



## 9.3.1 Auto-Negotiation Configuration for 200G

Provides information for auto-negotiating 200G link speeds on Ethernet network adapters.

### 9.3.1.1 UEFI Configuration

1. Set the Autodetect Speed Exclude Mask to 0.
2. Set Port Enablement to Disable port 2 as shown in the following figure.



A system reboot is required for this change to take effect.

## 9.3.2 Forced 200G Configuration

Provides information for manually configuring 200G link speeds on Ethernet network adapters.

### 9.3.2.1 UEFI Configuration

To reconfigure the BCM57508 network adapter as a 1-port, 200G device.

1. Set the port speed to 200Gbps PAM-4.
2. Set Port Enablement to Disable port 2 as shown in the following figure.
3. Set Port Link Training to match the switch settings (Dell switch default setting is enabled).



## 9.3.3 Configuring Dell Switches

When using a Dell OS10 switch for 200G operation with DAC cables, the default interface configuration must be changed to set the port group profile to 200x2, enable Link Training, and Forward Error Correction. When using 200G optical transceivers only the speed must be set.

**NOTE:** OS10 version 10.5.3.4 or later must be used.

```
enable
configure
port-group <Group ID>
port <Port ID> mode Eth 200g-2x
exit
```

```
interface range ethernet <Port ID>:1-ethernet<Port ID>:5
fec CL119-RS
negotiation off
```

## 10 PXE Boot

To serve PXE requests from PXE clients, a PXE server must be configured. The PXE server can be configured to run regular PXE or iPXE. Regular PXE is a network boot program that downloads config files over TFTP from the PXE server. iPXE is an enhanced implementation of the PXE client firmware and a network boot program which uses iPXE scripts rather than config files and can download scripts and images with HTTP.

**NOTE:** UEFI mode PXE boot is supported on physical functions as well as NIC partitions, whereas legacy mode PXE boot is supported only on physical functions.

The procedure for configuring the PXE server is discussed in [PXE Server Configuration](#).

### 10.1 UEFI Mode

Enable PXE for the Broadcom adapter interface in **Network Configuration** under **System Setup**. Refer to the documentation from the server manufacturer on how to enable PXE on the particular server model.

#### 10.1.1 iPXE

Download the `ipxe.efi` boot loader file using TFTP from the PXE server.

```
Booting from PXE Device 1: NIC in Slot 1 Port 1 Partition 1

>>Start PXE over IPv4.
  Station IP address is 174.30.10.18

  Server IP address is 174.30.10.10
  NBP filename is ipxe/ipxe.efi
  NBP filesize is 974496 Bytes
  Downloading NBP file...

  NBP file downloaded successfully.
  iPXE initialising devices...ok

iPXE 1.0.0+ -- Open Source Network Boot Firmware -- http://ipxe.org
Features: DNS HTTP iSCSI TFTP SRP ULAN AoE EFI Menu

net4: b0:26:28:cb:12:70 using 14e4-16d7 on 0000:3b:00.0 (open)
  Link:up, TX:0 TXE:0 RX:0 RXE:01
Configuring (net4 b0:26:28:cb:12:70) ..... ok
net4: 174.30.10.12/255.255.0.0 gw 174.30.10.10
Next server: 174.30.10.10
Filename: ipxe/legacymenu.ipxe
tftp://174.30.10.10/ipxe/menu.ipxe...
```

## 10.1.2 PXE

The boot loader file is downloaded via TFTP from the PXE server automatically.

```
Booting from PXE Device 1: NIC in Slot 1 Port 1 Partition 1

>>Start PXE over IPv4.
  Station IP address is 174.30.10.18

  Server IP address is 174.30.10.10
  MBP filename is BOOTX64.EFI
  MBP filesize is 1301312 Bytes
  Downloading MBP file...

  MBP file downloaded successfully.
  Fetching Netboot Image
  -
```

The menu items from the `grub.cfg` file are listed.

```
RHEL 6.5
RHEL 7.3
SLES11 SP3
SLES 12
CentOS 7
Windows 2019 Datacentre
ESX 6.7

Use the ▲ and ▼ keys to change the selection.
Press 'e' to edit the selected item, or 'c' for a command prompt.
```

Additional files are downloaded from the PXE server over TFTP based on the menu selection and the PXE boot continues.

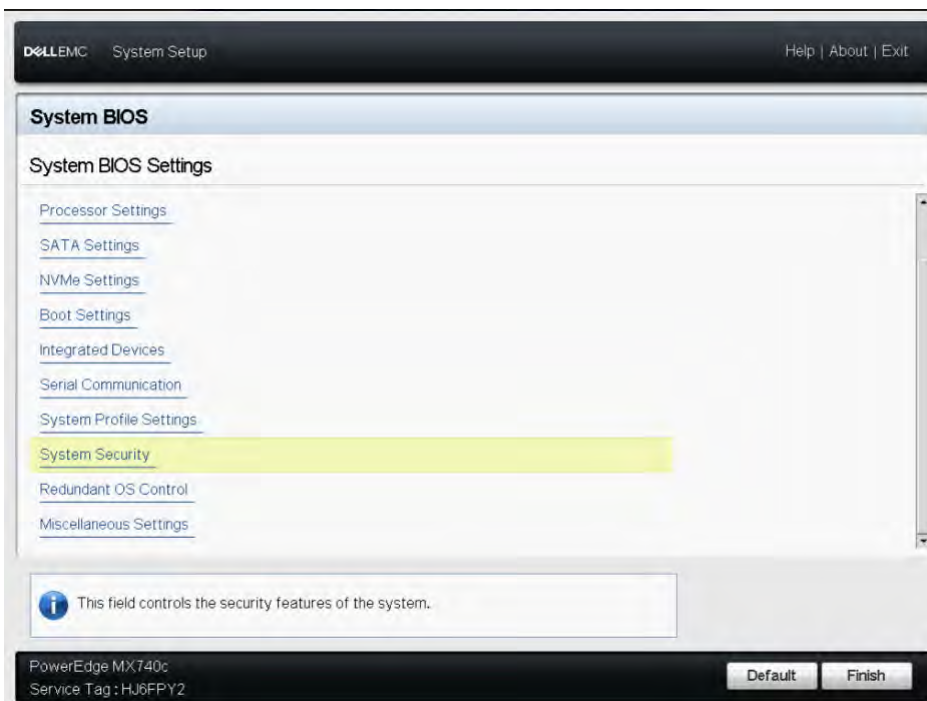
### 10.1.3 Secure Boot

Secure boot is a UEFI based feature developed by UEFI forum to increase security in the pre-boot environment. Secure boot minimizes the threat of potential attacks between firmware initiation and operating system loading.

This feature halts the execution of unsigned code before the operating system boots. Any unsigned firmware running on the adapter is not allowed to run. Only adapters running digitally signed firmware images are listed in preboot configuration menus and in network boot options.

In order to enable this security feature, select **Enable Secure Boot** in the **System Setup Menu**. This option is only available on server platforms that support secure boot. The menu under which the secure boot option is listed varies between server vendors.

**Figure 40: Security Menu Example**



**Figure 41: Enable Secure Boot Example**

## 10.2 PXE Server Configuration

The PXE server is operating system independent. A PXE server is any host that can lease out DHCP IPs to any requesting PXE Client, point to a boot loader file, and transfer further configuration and boot files to the PXE client over any application level protocol.

This section explains how to setup a PXE server on a RHEL/CentOS-based OS. For Windows deployment services, refer to documentation provided on the Microsoft Technet website. For setting up PXE servers on any other operating system, refer to the documentation provided by the respective operating system.

## 10.2.1 DHCP Configuration for PXE/iPXE

To add the DHCP feature on the PXE server, use the following command:

```
yum install dhcp
```

**NOTE:** All IP addresses mentioned in this section are for illustration purposes only. They can be modified to match the subnet that the PXE server is configured in.

### 10.2.1.1 IPv4 DHCP Configuration

To configure an interface with static network settings using `ifcfg` files, for an interface with the name `p1p1`, create a file with name `ifcfg-p1p1` in the `/etc/sysconfig/network-scripts/` directory as follows:

```
DEVICE=p1p1
TYPE=Ethernet
BOOTPROTO=none
ONBOOT=yes
IPADDR=174.30.10.10
NETMASK=255.255.0.0
```

#### 10.2.1.1.1 iPXE

To run iPXE, in the `/etc/dhcp/dhcpd.conf` file, make the following changes:

**NOTE:** The IP addresses used in this section are for examples only.

```
option ipxe.no-pxedhcp 1;

subnet 174.30.10.0 netmask 255.255.0.0 {
    option routers 174.30.10.10;
    range 174.30.10.11 174.30.10.50;
    next-server 174.30.10.10;
    option subnet-mask 255.255.0.0;
    default-lease-time 3600;
    max-lease-time 4800;
    class "pxeclients" {
        match if substring (option vendor-class-identifier, 0, 9) = "PXEClient";
        if not exists ipxe.bus-id {
            next-server 174.30.10.10;
            if option arch = 00:06 {
                filename "ipxe/ipxe-x86.efi";
            } elseif option arch = 00:07 {
                filename "ipxe/ipxe.efi"; # iPXE.efi built with support for Broadcom adapters
            } elseif option arch = 00:00 {
                filename "ipxe/ipxe.pxe"; # iPXE.pxe built with support for Broadcom adapters
            }
        } else {
            next-server 174.30.10.10;
            filename "ipxe/menu.ipxe"; # iPXE boot menu
        }
    }
}
```

### 10.2.1.1.2 PXE

To run PXE, in the `/etc/dhcp/dhcpd.conf` file, make the following changes:

```
ddns-update-style none;
default-lease-time 600;

option space PXE;
option PXE.mtftp-rcode 1 = ip-address;
option PXE.mtftp-cport code 2 = unsigned integer 16;
option PXE.mtftp-sport code 3 = unsigned integer 16;
option PXE.mtftp-tmout code 4 = unsigned integer 8;
option PXE.mtftp-delay code 5 = unsigned integer 8;
option arch code 93 = unsigned integer 16;
allow booting;
allow bootp;
allow unknown-clients;

subnet 174.30.0.0 netmask 255.255.0.0 {
    default-lease-time 600;
    max-lease-time 6000;
    range 174.30.10.11 174.30.10.50; #
    class "pxeclients" {
        match if substring (option vendor-class-identifier, 0, 9) = "PXEClient";
        next-server 174.30.10.10;
        if option arch = 00:06 {
            filename "bootia32.efi";
        } else if option arch = 00:07 {
            filename "BOOTX64.EFI";
        } else {
            filename "pxelinux/pxelinux.0";
        }
    }
}
```

In order to bind a certain hardware MAC address to an IP in the DHCP range, add the following to the `dhcpd.conf` file.

```
host server1-adapter1-port1 {
    hardware ethernet 00:0A:F7:94:F7:A4;
    fixed-address 174.30.10.15;
}
```

To restart the network service, run the following command:

```
service network restart
```

To restart the DHCP service, run the following command:

```
service dhcpd restart
```



### 10.2.1.2 IPv6 DHCP Configuration

In `/etc/sysconfig/network`, make the following changes:

```
NETWORKING_IPV6=yes
IPV6FORWARDING=no
IPV6_AUTOCONF=no
IPV6_AUTOTUNNEL=no
```

In `/etc/sysconfig/network-scripts/ifcfg-<interface_name>` make the following changes:

```
IPV6_AUTOCONF=no
IPV6INIT=yes
IPV6ADDR=2015:9:19:ffff::10/64 # Replace with your static address
```

#### 10.2.1.2.1 iPXE

In `/etc/dhcpd/dhcpd6.conf` file, make the following changes:

```
default-lease-time 2592000;
preferred-lifetime 604800;
option dhcp-renewal-time 3600;
option dhcp-rebinding-time 7200;
allow leasequery;
option dhcp6.info-refresh-time 21600;
dhcpv6-lease-file-name "/var/lib/dhcpd/dhcpd6.leases";

option dhcp6.user-class code 15 = string;
option dhcp6.bootfile-url code 59 = string;
option dhcp6.client-arch-type code 61 = array of unsigned integer 16;
option dhcp6.name-servers 2015:9:19:ffff::10;

subnet6 2015:9:19:ffff::/64 {
    range6 2015:9:19:ffff::11 2015:9:19:ffff::500;
    if exists dhcp6.client-arch-type and option dhcp6.client-arch-type = 00:07 {
        option dhcp6.bootfile-url "tftp://[2015:9:19:ffff::10]/ipxe/ipxe.efi";
    } elsif exists dhcp6.client-arch-type and option dhcp6.client-arch-type = 00:00 {
        option dhcp6.bootfile-url "tftp://[ 2015:9:19:ffff::10]/pxelinux/pxelinux.0";
    } elsif exists dhcp6.user-class and substring(option dhcp6.user-class, 2, 4) = "iPXE" {
        option dhcp6.bootfile-url "tftp://[2015:9:19:ffff::10]/ipxe/menu.ipxe";
    }
}
```

#### 10.2.1.2.2 PXE

In `/etc/dhcpd/dhcpd6.conf` file, make the following changes:

```
default-lease-time 2592000;
preferred-lifetime 604800;
option dhcp-renewal-time 3600;
option dhcp-rebinding-time 7200;
allow leasequery;
option dhcp6.info-refresh-time 21600;
dhcpv6-lease-file-name "/var/lib/dhcpd/dhcpd6.leases";

option dhcp6.user-class code 15 = string;
option dhcp6.bootfile-url code 59 = string;
```

```
option dhcp6.client-arch-type code 61 = array of unsigned integer 16;
option dhcp6.name-servers 2015:9:19:ffff::10;

subnet6 2015:9:19:ffff::/64 {
    range6 2015:9:19:ffff::11 2015:9:19:ffff::500;
    if exists dhcp6.client-arch-type and option dhcp6.client-arch-type = 00:07 {
        option dhcp6.bootfile-url "tftp://[2015:9:19:ffff::10]/BOOTX64.EFI ";
    } elsif exists dhcp6.client-arch-type and option dhcp6.client-arch-type = 00:00 {
        option dhcp6.bootfile-url "tftp://[ 2015:9:19:ffff::10]/pxelinux/pxelinux.0";
    }
}
```

To restart the network service, run the following command:

```
service network restart
```

To restart the DHCPv6 service, run the following command:

```
service dhcpd6 restart
```

### 10.2.1.3 DHCP with VLAN

In `/etc/sysconfig/network-scripts`, make a copy of the interface file for the interface over which VLAN must be configured.

**Example:** `ifcfg-p1p1` and `ifcfg-p1p1.5`

**NOTE:** 5 indicates that the interface will use VLAN ID 5.

In the new VLAN interface file, add the parameter `VLAN=yes`.

UUID can be generated for this new interface using the command `uuidgen p1p1.5`.

In the file `/etc/dhcp/dhcpd`, add a scope for the VLAN interface so that the DHCP server leases DHCP IPs on the VLAN interface.

## 10.2.2 TFTP Configuration

To configure TFTP:

1. Add the TFTP, XINET packages.

```
yum install xinetd tftp-server
```

2. In the `/etc/xinetd.d/tftp` file, make the following changes.

```
service tftp
{
    socket_type= dgram
    protocol= udp
    wait      = yes
    user      = root
    server    = /usr/sbin/in.tftpd
    server_args= -s /var/lib/tftpboot -6
    disable   = no
    per_source= 11
    cps       = 100 2
    flags     = IPv6
}
```

```
}
```

The base path for the TFTP server is `/var/lib/tftpboot/`.

### 10.2.2.1 iPXE

Under the TFTP root directory, create a new folder called `ipxe` and add the `ipxe.efi` and `ipxe.pxe` files to it. Also add the `menu.ipxe` file to it, which can chain more iPXE files to list the menu based on the boot mode.

```
#!/ipxe
iseq ${platform} efi && goto uefibios || goto legacybios

:uefibios
echo Loading the UEFI Menu
chain --replace --autofree ${menu-url}efimenu.ipxe

:legacybios
echo Loading the LEGACY Menu
chain --replace --autofree ${menu-url}biosmenu.ipxe
```

All the chained files are to be present in the `<TFTP_root>/ipxe` directory.

### 10.2.2.2 PXE

Create a new directory `pxelinux` in the TFTP root. Extract the Syslinux package and move the contents to the `<TFTP_root>/pxelinux` directory.

The TFTP root must contain the boot loader file from the target OS. The boot loader file can be copied from the `/boot/efi/EFI/<OS_flavor>` directory on the target OS. It should be renamed to `BOOTX64.EFI`.

Create `grub.cfg` file in the TFTP root and add UEFI mode PXE boot menu items to the same.

```
default=0
timeout=10

title RedHat 7u5
    root (nd)
    kernel /images/rhel75/vmlinuz inst.driver=vesa nomodeset method=http:// 174.30.10.10/images/
RHEL75linux dd
    initrd /images/RHEL7.2/initrd.img ip=dhcp

title SLES12SP4
    root (nd)
    kernel images/sles12sp4/vmlinuz inst.driver=vesa nomodeset inst.repo=http:// 174.30.10.10/
images/SLES12SP4
    initrd /images/sles12sp4/initrd.img ip=dhcp
```

Create `pxelinux.cfg/default` file and add BIOS mode PXE boot menu items to it.

```
DEFAULT menu.c32
PROMPT 0
TIMEOUT 10
MENU TITLE BIOS mode PXE menu

LABEL localdisk
MENU LABEL ^Local Hard Drive
LOCALBOOT 0
```

```

LABEL RHEL75
MENU LABEL ^RedHat 7u5
KERNEL images/rhel75/vmlinuz
APPEND initrd=images/rhel75/initrd.img ramdisk_size=200000 ip=dhcp inst.xdriver=vesa nomodeset
inst.repo=http://174.30.10.10/images/RHEL75

```

```

LABEL SLES12SP4
MENU LABEL ^SLES 12 SP 4
KERNEL images/sles12sp4/vmlinuz
APPEND initrd=images/sles12sp4/initrd.img ramdisk_size=200000 ip=dhcp inst.xdriver=vesa nomodeset
inst.repo=http://174.30.10.10/images/SLES12SP4

```

Create a new directory `images` in the TFTP root. Sub-directories can be created to match the kernel image path specified in `grub.cfg` and `pxelinux.cfg/default` files. Copy the `vmlinuz` and `initrd.img` files from the `/boot` directory of the target OS's to these sub-directories.

To restart the TFTP service, run the following command:

```
service xinetd restart
```

## 10.2.3 HTTP Configuration

To configure HTTP:

1. Add the HTTP feature using the following command:

```
yum install httpd
```

`/var/www/html/` is the base path for the HTTP server. The conf file is present in `/etc/httpd/conf/httpd.conf`.

2. To restart the HTTP service, run the following command:

```
service httpd restart
```

To specifically make HTTP listen on certain interfaces, in the `/etc/httpd/conf/httpd.conf` file, add the `LISTEN` directive for all the required interfaces.

```

Listen 192.0.2.1:80
Listen 192.0.2.5:8000
Listen 174.30.10.10:80

```

If this is not specified, the server listens on all interfaces.

3. Create a new directory `images` in the HTTP root. Subdirectories can be created to match the installation repository path specified in `grub.cfg` and `pxelinux.cfg/default` files. Extract the contents of the installation media of the target OS's to these sub-directories.

# 11 SR-IOV – Configuration and Use Case Examples

## 11.1 Enable SR-IOV in BIOS/UEFI and Device

To enable SR-IOV in BIOS/UEFI and Device:

1. Enable SR-IOV in the NIC cards:

- a. SR-IOV in the NIC card can be enabled using the HII menu. During system boot, access the system **BIOS**→**NetXtreme-E NIC**→**Device Level Configuration** menu.
  - b. Set the **Virtualization** mode to **SR-IOV**.
  - c. Set the number of virtual functions per physical function.
  - d. Set the number of MSI-X vectors per the VF and Max number of physical function MSI-X vectors. If the VF is running out of resources, balance the number of MSI-X vectors per VM.
2. Enable virtualization in the BIOS:
    - a. During system boot, enter the system **BIOS** →**Processor settings**→**Virtualization Technologies** and set it to **Enabled**.
    - b. During system boot, enter the system **BIOS** →**SR-IOV Global** and set it to **Enabled**.

## 11.2 Linux Use Case Example: SR-IOV Pass-Through to libvirt Virtual Machine

1. Install the desired Linux version with Virtualization enabled (libvirt and Qemu).
2. Enable the IOMMU kernel parameter.
  - a. The IOMMU kernel parameter is set by appending `intel_iommu=on` to the kernel command line
3. Use in-box driver, or install the driver as shown in [Installing the Linux Driver](#).
4. Enable Virtual Functions through kernel parameters:
  - a. Once the driver is installed, `lspci` displays the Broadcom Ethernet Network adapter physical interfaces present in the system.
  - b. To activate Virtual functions, use the following command:

```
vi /etc/default/grub (add "intel_iommu=on" to GRUB_CMDLINE_LINUX
grub2-mkconfig -o /boot/grub2/grub.cfg (or /boot/efi/<system type>/grub.cfg.
```

```
echo X > /sys/class/net/<ifname>/device/sriov_numvfs
```

**NOTE:** Ensure that the physical interface (<interface>) is up. VFs are only created if PFs are up. X is the number of VFs that are exported to the OS.

**Example:** `echo 4 > /sys/class/net/eth1/device/sriov_numvfs`

5. Check the PCIe virtual functions exist with `lspci`.
6. Create new virtual machine from installation ISO file.
7. Select **Customize configuration before install**.
8. Select **Add Hardware** → **PCI Host Device** → **Virtual Function**.
9. Finish installation. The VM detects Broadcom Ethernet Network adapter. Use either the in-box driver from the installed operating system or the install driver as shown in [Installing the Linux Driver](#).

### 11.2.1 Setting MAC Address for the VF

The MAC address of each VFs can set using the `ip link set MAC` command.

Example:

```
ip link set <pf ifname> vf 0 mac xx:xx:xx:xx:xx:xx
```

"ip link show" can show the VFs and associated MAC addresses.

Use "virsh start <VM name>"

Optional: `virsh vcpupin <VM-Name> 0 <CPU Number>` can be used to run the VM on a specific CPU.

## 11.3 Windows SR-IOV Use Case Example

**NOTE:** To obtain the maximum number of virtual functions per PF, disable **Virtual Switch RSS** and set the **Maximum Number of RSS Queues** to 1 from the driver advanced properties.

1. Install the latest KB update for your Windows 2012 R2 or Windows 2016 OS.
2. Install the appropriate Virtualization (Hyper-V) options. For more detail requirements and steps on setting up Hyper-V, Virtual Switch, and Virtual Machine, visit [Microsoft.com](http://Microsoft.com).
3. Install the latest Broadcom Ethernet Network adapter driver on the Hyper-V as shown in [Installing the Windows Driver](#).
4. Enable SR-IOV in the NDIS miniport driver advanced properties.
5. In Hyper-V Manager, create your Virtual Switch with the selected Broadcom Ethernet Network adapter interface.
6. Check the **Enable Single-Root I/O Virtualization (SR-IOV)** box while creating the Hyper-V Virtual Adapter.
7. Create a Virtual Machine (VM) and add the desired number of Virtual Adapters.
8. Under the Virtual Machine's Network Adapter settings for each Virtual Adapter, check **Enable SR-IOV** under the **Hardware Acceleration** section.
9. Launch your VM and install the desired guest OS.
10. Use the in-box driver from the VM operating system, or install the Broadcom Ethernet Network adapter driver as shown in [Installing the Windows Driver](#).

**NOTE:** The Virtual Function (VF) driver for Broadcom Ethernet Network adapters is the same driver as the base driver. For example, if the guest OS is Windows 2012 R2, the user needs to install `Bnxtnd64.sys` in VM. The user can do this by running the Broadcom Ethernet Network adapter driver installer executable. Once the driver has been installed in the guest OS, the user can see the VF driver interface(s) appearing in Device Manager of the guest OS in the VM.

**NOTE:** When updating the network adapter settings on a Hyper-Visor VM that is bound to a VF on a network adapter, the properties in the Microsoft Synthetic Network Adapter on the VM should be changed first, followed by a corresponding change in the properties on the VF.

## 11.4 VMware SR-IOV Use Case Example

1. On ESXi, install the driver as shown in [Installing the VMware Driver](#).

2. Enable SR-IOV VFs:

Only the physical functions (PFs) are automatically enabled. If a PF supports SR-IOV, the PF (`vmknicX`) is part of the output of the command shown below.

```
esxcli network sriovnic list
```

To enable one or more virtual functions (VFs), the driver uses the module parameter `max_vfs` to enable the desired number of VFs per PF. For example, to enable four VFs on PF1:

```
esxcfg-module -s 'max_vfs=4' bnxtnet (reboot required)
```

To enable VFs on a set of PFs, use the command format shown below. For example, to enable four VFs on PF 0 and 2 VFs on PF 2:

```
esxcfg-module -s 'max_vfs=4,2' bnxtnet (reboot required)
```

The required VFs of each supported PF are enabled in order during the PF bring up. See the VMware documentation for information on how to map a VF to a VM.

**NOTE:** When using NPAR + SR-IOV, every NPAR function (PF) is assigned a maximum of eight VFs.

**NOTE:** For a VF to be in promiscuous mode, one of the following conditions must be true:

- The VF should be associated with a default VLAN.
- The VF should be a trusted VF.

If none of the above conditions are true, the VF will not be in promiscuous mode and, therefore, will not see packets received by the PF.

## 12 NPAR – Configuration and Use Case Example

### 12.1 Features and Requirements

- OS/BIOS Agnostic – The partitions are presented to the operating system as real network interfaces so no special BIOS or OS support is required like SR-IOV.
- Additional NIC functions without requiring additional switch ports, cabling, PCIe expansion slots.
- Traffic Shaping – The allocation of bandwidth per partition can be controlled so as to limit or reserve as needed.
- Can be used in a Switch Independent manner – The switch does not need any special configuration or knowledge of the NPAR enablement.
- Can be used in conjunction with RoCE and SR-IOV.
- Supports stateless offloads such as LSO, TPA, RSS/TSS, and RoCE (two PFs per port only).
- Alternative Routing-ID support for greater than eight functions per physical device.

**NOTE:** In the **UEFI HII Menu** page, the NXE adapters support up to 16 PFs per device on an ARI capable system. For a 2-port device, this means up to 8 PFs for each port.

### 12.2 Limitations

**NOTE:** An NPAR configuration where a team, virtual switch (simple or distributed), or a logical network switch (N-VDS) contains more than one partition from the same physical port is not supported.

- Shared settings must be suppressed to avoid contention. For example: Speed, Duplex, Flow Control, and similar physical settings are hidden by the device driver to avoid contention.
- Non-ARI systems enable only eight partitions per physical device.
- RoCE + SRIOV + NPAR combination is not supported.
- RoCE for BCM5741X adapters, is only supported on the first two partitions of each physical port, or a total of four partitions per physical device. BCM9575XX adapters can support RoCE on all partitions.
- NPAR + SR-IOV/MultiRSS is not supported with ESXi 7.0 for the BCM5741X.
- Teaming partitions within the same physical port are not supported.
- LACP teaming is not supported.



## 12.3 Configuration

NPAR can be configured using BIOS configuration HII menus on legacy boot systems. Some vendors also expose the configuration via additional proprietary interfaces.

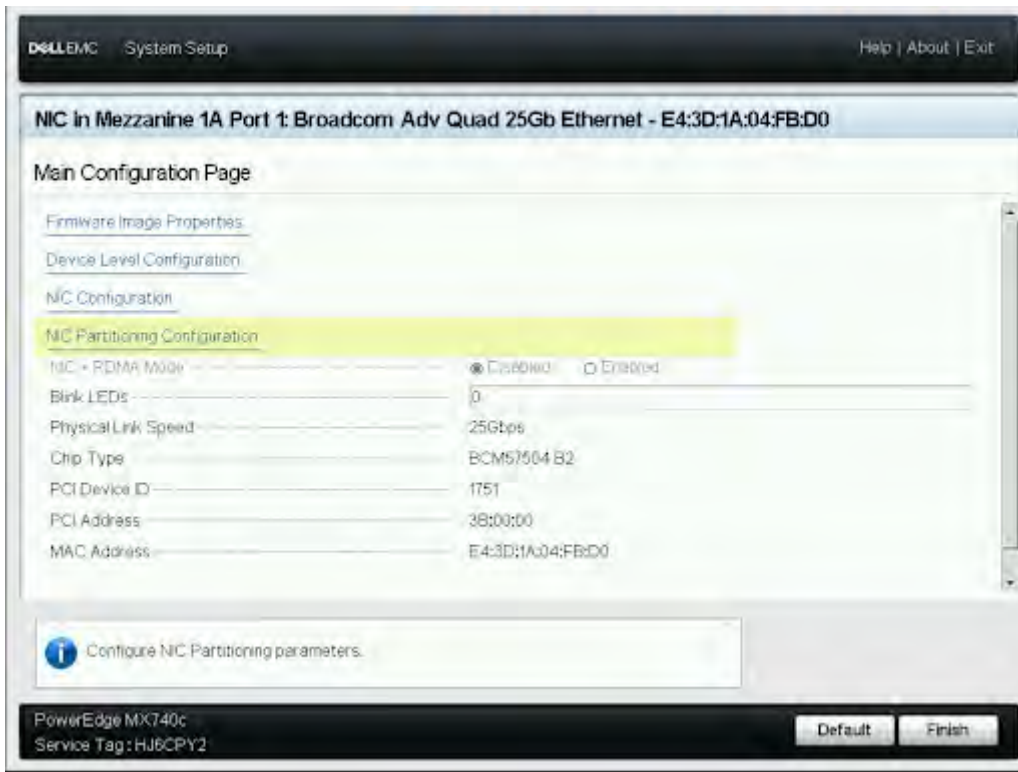
To enable NPAR:

1. Select the target NIC from the BIOS HII Menu and set the Multi-Function Mode or Virtualization Mode option. The choice of options affects the whole NIC instead of the individual port.

Broadcom Adv. Dual 25Gb Ethernet - B0:26:28:C4:06:7A	
Virtualization Mode .....	NPar
NParEP Mode .....	None
Number of MSI-X Vectors per VF .....	NPar
Maximum Number of PF MSI-X Vectors .....	SR-IOV
Link FEC .....	NPar + SR-IOV
Operational Link Speed .....	Disabled
Firmware Link Speed .....	Auto Negotiated
DCBX Mode .....	Auto Negotiated
LLDP nearest bridge .....	Disabled
LLDP nearest non-TPMR bridge .....	<input type="radio"/> Disabled <input checked="" type="radio"/> Enabled
Auto-negotiation Protocol .....	<input type="radio"/> Disabled <input checked="" type="radio"/> Enabled
	IEEE and Consortium

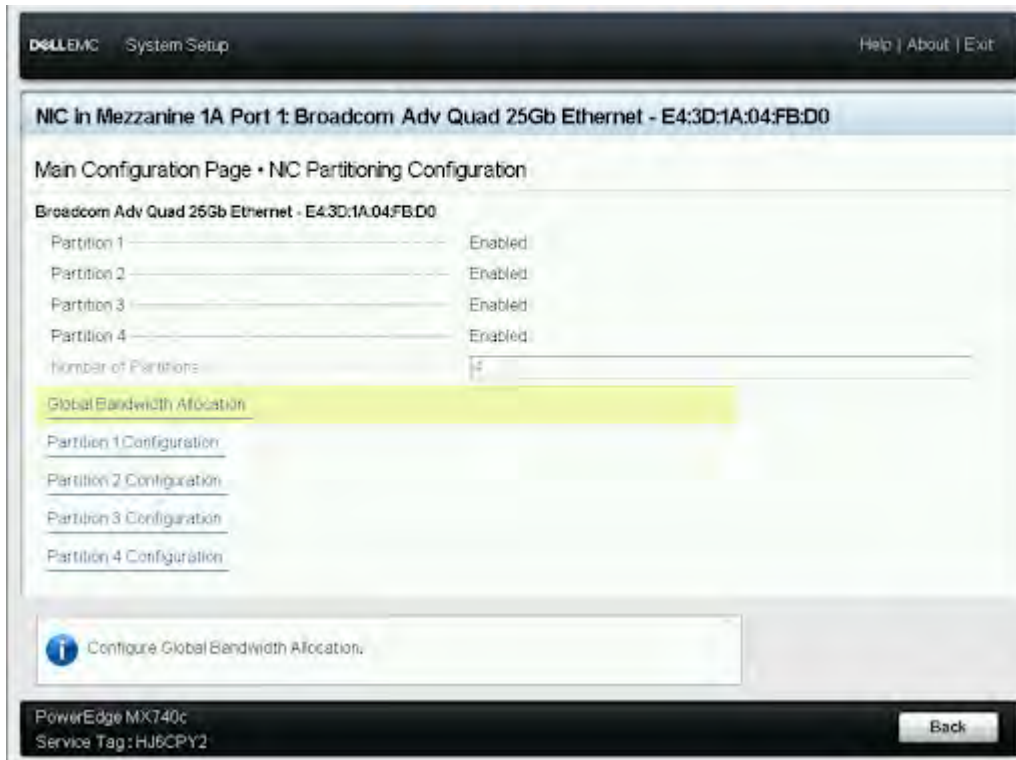
**NOTE:** For some ARI capable OEM systems, the **NParEP** button is available to explicitly allow the Broadcom Ethernet Network adapter to support up to 16 partitions. Switching from single function mode to multifunction mode, the device needs to be re-enumerated, therefore changes do not take effect until a system reboot occurs.

2. Once NPAR is enabled, the **NIC Partitioning Main Configuration** menu option is available from the main NIC Configuration Menu associated with each physical port.

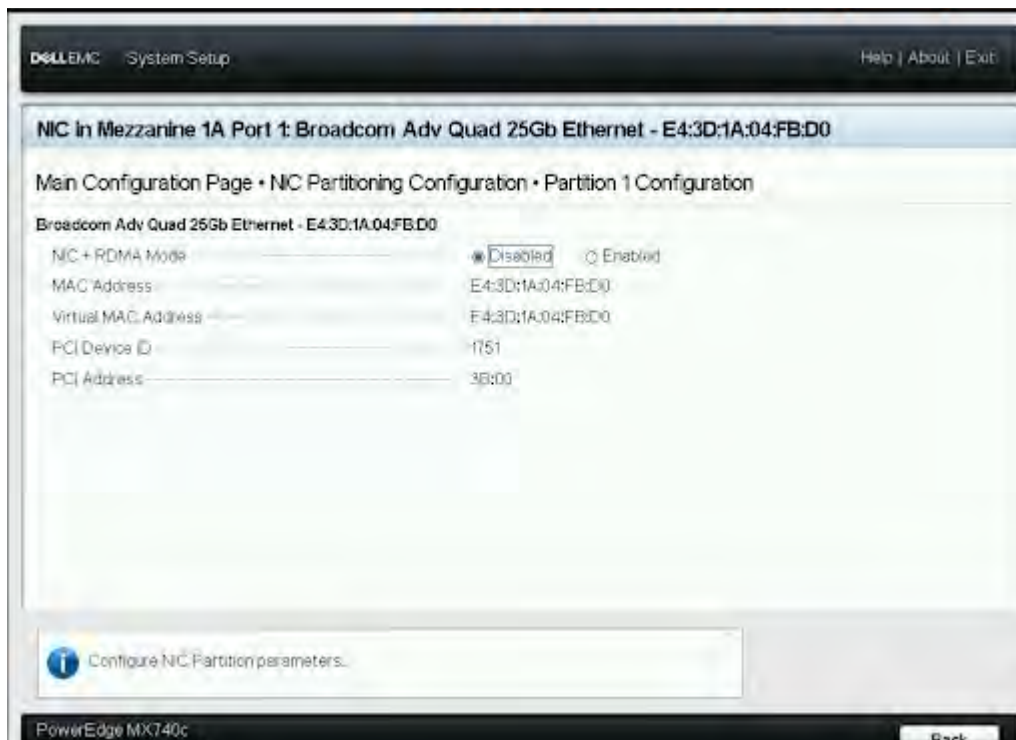


3. The NIC Partition Configuration Menu (shown below) allows the user to choose the number of partitions that should be allocated from the selected physical port. Each Broadcom Ethernet Network adapter can support a maximum of 16 partitions on an ARI capable server. By default, dual-port adapters are configured for eight partitions per physical port.

Configuration options for each partition are also accessible from this menu. For some OEM systems, the HII menu also includes a Global Bandwidth Allocation page where the minimum (reserved) and maximum (limit) TX bandwidth for all partitions can be configured.



4. Set the NIC Partition Configuration parameters (see [Table 34](#)).



**Table 34: NPAR Parameters**

Parameter	Description	Valid Options
BW Limit	Maximum percentage of available bandwidth this partition is allowed.	Value 0 to 100
BW Limit Valid	Functions as an on/off switch for the BW Limit setting.	True/False
RDMA Support	Functions as an on/off switch for RDMA support on this partition. <b>NOTE:</b> Only two partitions per physical port can support RDMA. For a dual-port device, up to 4 NPAR partitions can support RDMA.	Enabled/Disabled

**NOTE:** The NPAR minimum bandwidth parameter cannot be changed on the BCM575XX. BCM5741X devices are permitted to change the NPAR minimum value.

## 12.4 Reducing NIC Memory Consumption with NPAR

The default value of receive buffers was selected to work well for typical configurations. If you have many NICs in a system, have enabled NPAR on multiple NICs, or if you have only a small amount of RAM, you may see a Code 12 yellow bang in the Device Manager for some of the NICs. Code 12 means that the driver failed to load because there were not enough resources. In this case, the resource is a specific type of kernel memory called Non-Paged Pool (NPP) memory.

If you are getting a Code 12, or for other reasons wish to reduce the amount of NPP memory consumed by the NIC:

- Reduce the number of RSS queues from the default of 8 to 4 or 2. Each RSS queue has its own set of receive buffers allocated, so reducing the number of RSS queues reduces the allocated NPP memory. There can be performance implications from reducing the number of RSS queues, as fewer cores participate in processing receive packets from that NIC. Per processor CPU utilization should be monitored to ensure that there are no “hot” processors after this change.
- Reduce memory allocation by reducing the number of receive buffers allocated. The default value of 0 means the driver should automatically determine the number of receive buffers. For typical configurations, a setting of 0 (=auto) maps to XXXX receive buffers per queue. You can choose a smaller value such as 1500, 1000, or 500. (The value needs to be in multiples of 500 and between the range of 500 and 15000.) As mentioned above, a smaller number of receive buffers increases the risk of packet drop and a corresponding impact to packet retransmissions and decreased throughput.

The parameters “Maximum Number of RSS Queues” and “Receive Buffers (0=Auto)” can be modified using the **Advanced** properties tab for each NIC in the **Device Manager**. If you want to modify multiple NICs at the same time, it is faster to use the *Set-NetAdapterAdvancedProperty PowerShell cmdlet*. For example, to assign two RSS queues for all NICs in a system whose NIC name starts with “SI”, run the following command:

```
Set-NetAdapterAdvancedProperty Sl* -RegistryKeyword *NumRSSQueues -RegistryValue 2
```

Similarly, to set the number of Receive buffers to 1500, run the following command:

```
Set-NetAdapterAdvancedProperty Sl* -RegistryKeyword *ReceiveBuffers -RegistryValue 1500
```

For an overview of how to use PowerShell to modify NIC properties, see [Microsoft.com](https://www.microsoft.com).

## 12.5 Advanced NPAR

Advanced NPAR is known as switch dependent partitioning or Virtual Ethernet Port Aggregator (VEPA) Multi-Channel QinQ. The switch performs all switching between partitions and replication of broadcast and multicast packets.

**Figure 42: S-Tag Frame**

Destination MAC	Source MAC	802.1 Q Header (S-TAG)	802.1 Q Header (C-TAG)	IP Header	TCP Header	PayLoad
-----------------	------------	------------------------	------------------------	-----------	------------	---------

**Figure 43: S-Tag Header**

Tag Protocol ID (TPID) 2 bytes		PCP 3 bits	DEI 1 bit	C-VLAN ID 12 bits
0x88/0x91	0xa8/0x00			

**Figure 44: C-Tag Header**

Tag Protocol ID (TPID) 2 bytes		PCP 3 bits	DEI 1 bit	C-VLAN ID 12 bits
0x81	0x00			

**Table 35: Advanced NPAR Terms**

Term	Definition
TPID	Tag Protocol ID
S-VLAN	Stacked VLAN. Double VLAN tag structure allows service providers to add VLAN tags (S-Tag) to forward frame traffic across the network.
S-Tag	Service VLAN tag.
C-VLAN	The VLAN that a customer uses in double tag frames.
C-Tag	Customer VLAN tag. The inner VLAN tag in a double tag frame.
Intra-Partition Traffic	The traffic between the PF and VFs of a partition.
Inter-Partition Traffic	The traffic between PCIe functions on different partitions.
PF	NIC physical function.
VF	NIC virtual function.

### 12.5.1 Supported Broadcom Devices

- BCM57508-NGM2100D
- BCM57504-NGM425D

### 12.5.2 Supported Operating Systems

- VMware ESXi 7 u3 and higher, ESXi 8.0 and higher

### 12.5.3 Supported Features and Limitations

Supported Features:

- The OS driver via in-band utility or uEFI menu (Dell) can set features. Advanced NPAR related configuration is set through the BMC and UEFI drivers.
- RoCE v2 traffic inherits the S-VID from VF or PF.

- Stateless offloads on PF and VF levels are supported and independently of the NPAR mode.
- QoS is supported for the egress traffic on partition.

#### Limitations:

- OS-BMC is unsupported when Advanced NPAR is enabled.
- The options can only be set in HII menu for certain release (2.26).
- User tool, bnxtnvm, does not show the S-Tag options.
- When operating in Advanced NPAR mode, only the Ethernet adapter firmware can see the LLDP packets.
- NIC Teaming: Switch dependent teaming is not supported in the Advanced NPAR mode.

## 12.5.4 Supported Hardware Configurations

#### PF/VF (SR-IOV) Configuration:

- Number PFs/Adapter: Max 16 PFs.
- Number PFs/Port: 16/(number of ports). Example: 16/2 ports = 8 per port.
- Number VFs/Adapter: Max 256 VFs
- Number VFs/PF: Max 64 VFs. The number of VFs/PF shall be in multiples of 8.

## 12.5.5 Required Hardware Settings

#### Required hardware settings:

- Advanced NPAR mode enabled.
- DCBX = disabled
- Default EVB mode = VEPA (enabled)

**NOTE:** Ensure that EVB mode is set to VEPA (shown in [Figure 46](#)). No other mode is supported.

**Figure 45: UEFI Menu for Advanced NPAR Mode**

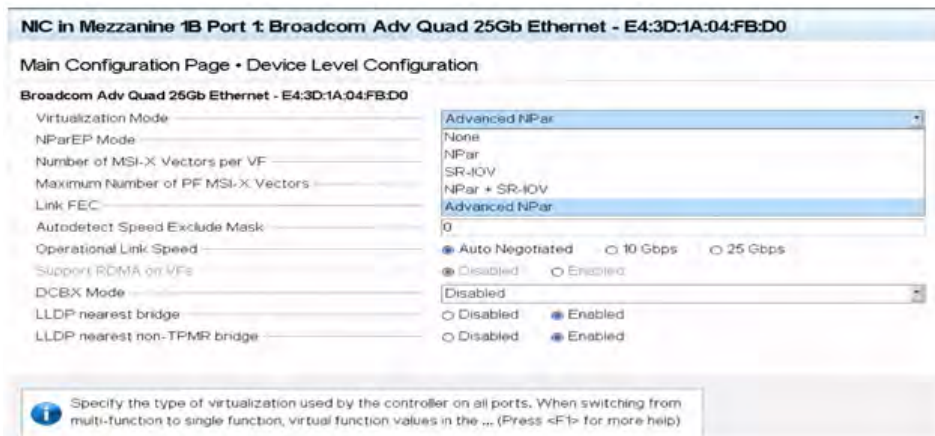


Figure 46: UEFI Menu for DCBX and EVB Modes

**NIC in Mezzanine 1B Port 1: Broadcom Adv Quad 25Gb Ethernet - E4:3D:1A:04:FB:D0**

Main Configuration Page • Device Level Configuration

Autodetect Speed Exclude Mask	0
Operational Link Speed	<input checked="" type="radio"/> Auto Negotiated <input type="radio"/> 10 Gbps <input type="radio"/> 25 Gbps
Support RDMA on VFs	<input checked="" type="radio"/> Disabled <input type="radio"/> Enabled
DCBX Mode	Disabled
LLDP nearest bridge	<input type="radio"/> Disabled <input checked="" type="radio"/> Enabled
LLDP nearest non-TPMR bridge	<input type="radio"/> Disabled <input checked="" type="radio"/> Enabled
Auto-negotiation Protocol	IEEE 802.3by & Consortium
Media Auto Detect	<input type="radio"/> Disabled <input checked="" type="radio"/> Enabled
Default EVB Mode	<input type="radio"/> VEB <input checked="" type="radio"/> VEPA <input type="radio"/> None
Flow Offload	<input checked="" type="radio"/> Disabled <input type="radio"/> Enabled
Port Link Training	<input checked="" type="radio"/> Disabled <input type="radio"/> Enabled
Adapter Error Recovery	<input type="radio"/> Disabled <input checked="" type="radio"/> Enabled

Figure 47: UEFI Menu for S-Tag Ether Type

**NIC in Mezzanine 1B Port 2: Broadcom Adv Quad 25Gb Ethernet - E4:3D:1A:04:FB:D1**

Main Configuration Page • NIC Partitioning Configuration

**Broadcom Adv Quad 25Gb Ethernet - E4:3D:1A:04:FB:D1**

Partition 1	Enabled
Partition 2	Disabled
Partition 3	Disabled
Partition 4	Disabled
Number of Partitions	4
S-Tag Ether Type	<input checked="" type="radio"/> 0x88a8 <input type="radio"/> 0x9100

Global Bandwidth Allocation

Partition 1 Configuration

Partition 2 Configuration

Partition 3 Configuration

Creating an S-VLAN requires the use of a second encapsulation tag, called S-Tag. The TPID of a S-Tag is set to 0x88a8 or 0x9100.

Figure 48: UEFI Menu for S-VLAN ID

**NIC in Mezzanine 1B Port 2: Broadcom Adv Quad 25Gb Ethernet - E4:3D:1A:04:FB:D1**

Main Configuration Page • NIC Partitioning Configuration • Partition 1 Configuration

**Broadcom Adv Quad 25Gb Ethernet - E4:3D:1A:04:FB:D1**

NIC > RDMA Mode	<input checked="" type="radio"/> Disabled <input type="radio"/> Enabled
S-VLAN ID	3
MAC Address	E4:3D:1A:04:FB:D1
Virtual MAC Address	E4:3D:1A:04:FB:D1
PCI Device ID	1751
PCI Address	88:00:01

Specify the s-VLAN ID (s-Tag) to be used by the partition in Advanced NPar mode. The s-VLAN ID must be unique per partition and in the range from ... (Press <F1> for more help)

All partition S-VLAN IDs in the same physical port are unique in the range of 3-4094. The default S-VLAN ID for a partition numbered X is X+2.

## 12.6 Ethernet Adapter Operations

### Ingress Traffic:

Determines the destination partition of each inbound frame by examining the S-VID in the S-tag of the frame and strips the frame of the S-tag prior to delivering the frame to the partition.

### Egress Traffic:

Apply an S-tag, as described in IEEE 802.1Q, to each frame that is outbound from a partition as the frame traverses the Port-mapping S-VLAN component.

### Intra-Partition and Inter-Partition Traffic:

The inter-partition switching is handled by the adjacent switch while the intra-partition switching is handled by the Ethernet adapter. The Ethernet adapter embedded switch supports switching of the traffic between the PF and its children VFs on the same partition. The traffic switched by the Ethernet adapter embedded switch can not be S-Tagged. Traffic is originated on a PF or VF of a partition that is going to a PF or VF the adjacent (external) switch shall use S-Tag to forward the frame traffic.



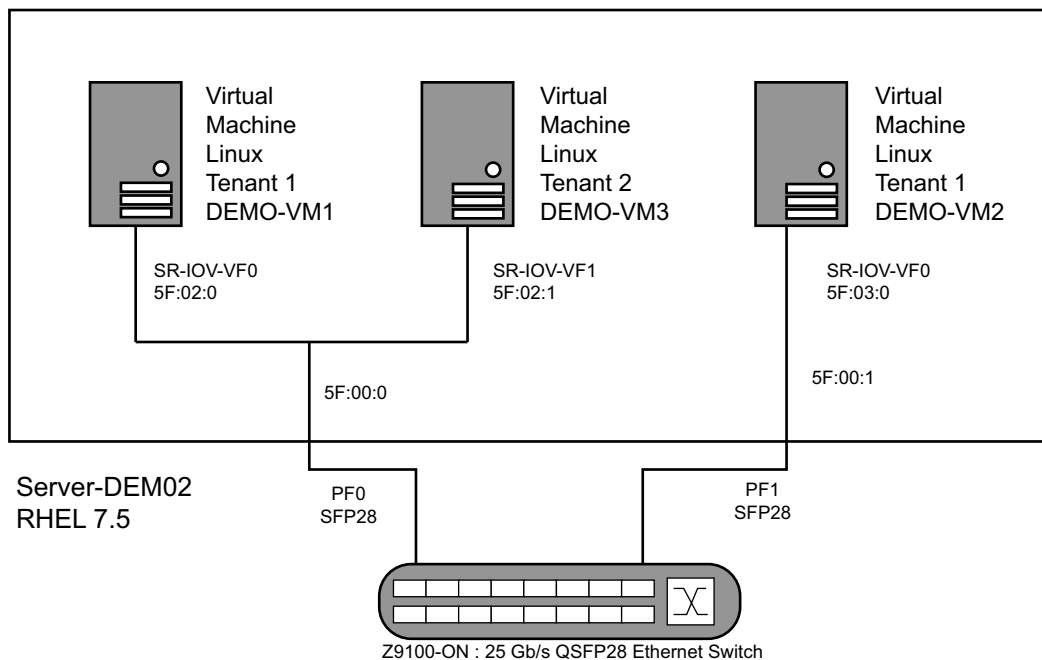
## 13 Tunneling Configuration Examples

Broadcom BCM5741X and BCM5751X devices support VXLAN, GRE, and IP-in-IP tunneling offloads. This section provides tunnelling configuration examples.

### 13.1 Network Diagram

The test network shown in [Table 49](#) uses one Linux server with one two port Ethernet adapter. SR-IOV is enabled in the adapter and two VFs are instantiated on the first port. A VF from the second port is exposed to the third VM (see [SR-IOV – Configuration and Use Case Examples](#) for information on SR-IOV bring up).

Figure 49: Network Diagram



### 13.2 VEB and VEPA Modes

VEB (Virtual Ethernet Bridging) mode generates an internal bridge within the NIC for VM-to-VM communication. The Ethernet frames are traverses through the internal bridge. VEPA (Virtual Ethernet Port Aggregator) mode transports the frames to the external switch. The switch handles the frame transport between the ports. VEB and VEPA can be configured through UEFI HII.

#### 13.2.1 VLAN Configuration

A VLAN can be configured on the Linux VF using:

**NOTE:** In this example physical function Port 1 (for example, eth2) VF 0 is configured to VLAN 2..

```
ip link set eth2 vf 0 vlan 2
```

When the VLAN is configured at the VF level, the Ethernet switch is configured as a VLAN.

## 13.3 VLAN Double Tagging

A VLAN can be configured at the PF level and another VLAN can be configured at the VF level inside the VM. Once the VF is exposed inside the VM, the Linux L2 driver can be installed to activate the interface. The following commands can be used to enable the VLAN inside the VM:

```
modprobe 802.1q;
ip link add link <IntName> name <Vlan.2> type vlan id <vlan num>
ip link set <IntName> up;
ip addr add <IPAddr>/mask broadcast <Gateway Addr> dev <IntName>
```

## 13.4 GRE Tunnelling

An IP GRE is an IP inside an IP tunnel which can carry private network traffic between two heterogeneous networks. On the VM, use the following commands:

```
modprobe ip_gre;
ip tunnel add gre45 mode gre local <public IP> remote <Private IP>
ip link set dev gre45 up;
ip addr add <private IP>/mask broadcast <broadcast ip> dev gre45
```

In this example, gre45 is the interface name.

## 13.5 IP-in-IP Tunnelling

Similar to GRE, IP-in-IP is another encapsulation that carries a private IP onto a public IP. Use the following commands:

```
modprobe ipip
ip tunnel add ipip45 mode ipip remote <Peer IP> local <private ip> ttl 255 dev <VM Interface Name>
ip link set dev ipip45 up
ip addr add dev ipip45 <IP addr> peer <Peer IP>/mask
ip link set dev ipip45 up
```

**NOTE:** In this example, ipip45 is the name of the newly created tunnel device.

## 13.6 VXLAN – Configuration and Use Case Examples

VXLAN encapsulation permits multiple virtual machines or containers residing on one server to be isolated from each other in virtual tunnels by encapsulating traffic with VXLAN headers. Broadcom Ethernet Network adapters accelerate this encapsulation and de-encapsulation in hardware.

This example discusses basic VXLAN connectivity between two Linux servers. Each server has one physical NIC enabled with outer IP address set to 1.1.1.4 and 1.1.1.2, respectively.

A VXLAN interface with ID 10 is created with multicast group 239.0.0.10 and is associated with physical network port eth1 on each server.

An IP address for the host is created on each server and associated to that VXLAN interface. Once the VXLAN interface is brought up, the VM present in system 1 can communicate with the VM present in system 2 using the VXLAN interfaces. The VLXAN format is shown in [Table 36](#).

**Table 36: VXLAN Frame Format**

MAC header	Outer IP header with proto = UDP	UDP header with Destination port= VXLAN	VXLAN header (Flags, VNI)	Original L2 Frame	FCS
------------	----------------------------------	---	---------------------------	-------------------	-----

Table 37 provides VXLAN command and configuration examples.

**Table 37: VXLAN Command and Configuration Examples**

System 1	System 2
ifconfig eth1 1.1.1.4/24	ifconfig eth1 1.1.1.2/24
ip link add vxlan10 type vxlan id 10 group 239.0.0.10 dev eth1 dstport 4789	ip link add vxlan10 type vxlan id 10 group 239.0.0.10 dev eth1 dstport 4789
ifconfig vxlan10 192.168.1.5 mtu 1450	ifconfig vxlan10 192.168.1.10 mtu 1450
ip --d link show vxlan10	–
ping 192.168.1.10	–

## 14 RoCE – Configuration and Use Case Examples

This section provides configuration and use case examples for Remote Direct Memory Access over Converged Ethernet (RoCE).

**NOTE:** If using NPAR + SR-IOV mode, only two VFs from each parent physical port can enable RDMA support, or a total of four VFs + RDMA per physical device.

**NOTE:** RoCE over VF is not supported on BCM575XX devices.

**NOTE:** RoCE SRIOV + NPAR is not a supported combination.

**NOTE:** RoCE SRIOV is not supported on BCM5741X.

**NOTE:** RoCE SRIOV is not supported on BCM5755X in 218.x release. Starting with the 2.19 release, RoCE SRIOV is supported on upto 128 VFs.

### 14.1 Enabling RoCE

To enable RoCE for PFs or VFs, the enable the RDMA selection in the HII menu in the BIOS before the RDMA option takes effect in the host or guest OS.

To enable RDMA in single function mode (if Virtualization Mode is None or SR-IOV):

1. During the system boot, access the **System Setup** → **NetXtreme-E NIC** → **Main Configuration Page** and set **NIC+ RMDA Mode** to **Enabled**.

To enable RDMA if Virtualization Mode is NPAR or NPAR + SR-IOV:

1. During the system boot, access the **System Setup** → **NetXtreme-E NIC** → **NIC Partitioning Configuration** → **Partition 1 (or 2) Configuration** and set **NIC+ RMDA Mode** to **Enabled**.

## 14.2 Linux Configuration and Use Case Examples

This section describes how to install the `bnxt_en` Linux L2 and/or `bnxt_re` RoCE driver and user space library for the Broadcom Ethernet Network adapter BCM9574XX and BCM9575XX 10/20/25/40/50/100/200 Gb/s Ethernet Network Controllers (ENCs). RoCE is supported on the BCM9575XX and the BCM95741X Ethernet network controllers. See the [NetExtreme-E Linux RoCE Configuration Guide](#) for information on configuring RoCE for these network controllers.

### 14.2.1 Requirements

To configure RoCE in Linux, the following items are required:

- Linux L2 driver `bnxt_en`
- Linux RDMA RoCE driver `bnxt_re`
- Linux RoCE user space library `libbnxt_re`

#### 14.2.1.1 BNXT\_EN Driver Dependencies

If kernels older than 4.7 are used or if the `CONFIG_VLAN_MODULE` kernel option is set as a module option, the `vxlan.ko` module must be loaded before the `bnxt_en.ko` module.

#### 14.2.1.2 BNXT\_RE Driver Dependencies

The RoCE driver (`bnxt_re`) depends on the L2 Linux driver (`bnxt_en`) networking counterpart. The `bnxt_re` driver depends on the IB stack available with the Linux kernel or the OFED IB stack.

#### 14.2.1.3 LIBBNXT\_RE User Library Dependencies

The user space RoCE driver depends on following:

- Ethernet driver for Broadcom Ethernet Network adapters (`bnxt_en`)
- RoCE driver for Broadcom Ethernet Network adapters (`bnxt_re`)
- uVerbs device interface, it is an IB-stack component (`ib_uverbs`)
- User space RDMA-CM, it is an IB-stack component (`rdma_ucm`)

### 14.2.2 Installing Drivers and the RoCE Library

The following sections provide information on how to install the RDMA stack, drivers, and RoCE library.

**NOTE:** Only one of either Native IB Stack or OFED must be installed. Unless a specific need for OFED is required, using the Linux native IB stack is recommended.

#### 14.2.2.1 Installing the Native IB Stack

To install the native IB stack:

1. Use one of the following options:

```
- yum -y install libibverbs* infiniband-diags perftest qperf librdmacm-utils
- yum -y groupinstall "InfiniBand Support"
```

Using the yum commands requires a RedHat subscription. If a RedHat subscription is not available, locate the following RPM packages from the RedHat installation CD or ISO:

```
- infiniband-diag.x86_64
- libibverbs-devel.x86_64
```

```
- libibverbs-utils.x86_64
- libibverbs.x86_64
- librdmacm-utils.x86_64
- iperf3-devel.x86_64
- iperf3.x86_64
```

2. Install all of the RPM packages using the following yum command:

```
# yum -y install infiniband-diag.x86_64 libibverbs-devel.x86_64 libibverbs-utils.x86_64
libibverbs.x86_64 librdmacm-utils.x86_64 iperf3-devel.x86_64 iperf3.x86_64
```

**NOTE:** If Ubuntu 14.04 and above is used, use the following command:

```
sudo apt-get install perftest libibverbs* ibverbs-utils
```

**NOTE:** If SLES 12 SP2/SP3 and above is used, use the following command:

```
zypper install ofed* libibverbs* librdmacm* perftest
```

3. If yast2 must be used, the following command is used to install RoCE.

```
# yast2 --install libibverbs libipathverbs compat-dapl dapl libamso-rdmav2 libcxgb3-rdmav2 libcxgb4
-rdmav2 libmlx4-rdmav2 libmthca-rdmav2 libnes-rdmav2 librdmacm ib-bonding ibsim ibutils
ibvexdmttools infiniband-diags libibcm libibcommon1 libibmad5 libibumad3 libsdp ofed ofed-doc ofed-
kmp-default ofed-kmp-pae ofed-kmp-trace opensm mstflint rds-tools ibutils libibcommon libibcommon1
srptools
```

### 14.2.2.2 Installing the OFED IB Stack

To install the OFED IB stack:

1. See the OFED release notes provided in the following links and install OFED before compiling bnx2\_re driver:

- [http://downloads.openfabrics.org/downloads/OFED/release\\_notes/OFED\\_3.18-2\\_release\\_notes](http://downloads.openfabrics.org/downloads/OFED/release_notes/OFED_3.18-2_release_notes)
- <http://www.openfabrics.org/downloads/OFED/ofed-3.18-2/OFED-3.18-2.tgz>

OFED requires several development packages to compile and install. Install the following packages using the SDK CD:

- zlib-devel
- libnl-devel
- tcl-devel
- glib2-devel
- libudev-devel

2. Retrieve the OFED file from the following link:

<http://www.openfabrics.org/downloads/OFED/ofed-3.18-2/OFED-3.18-2.tgz>

3. Install the file using the following commands:

```
tar xzf OFED-x.y.z3.18-2.tgz
cd OFED-x.y.z3.18-2
./install.pl
```

4. Select Option 2.

5. Select Option 3.

After the **OFED is installed completely** message is displayed, run the `ofed_info -s` command to verify that the correct version of OFED is installed. Then proceed to the L2 + ROCE driver installation.

**NOTE:** The OFED library installs only once on the system regardless of the number of kernels in the OS. It always overrides with latest. Ensure that OFED is compiling and installing for the correct kernel to install the RoCE driver.

### 14.2.2.3 Installing the Broadcom Ethernet Network adapter Drivers and RDMA Library

Install the `bnxt_en`, `bnxt_re` drivers, and `libbnxt_re` RDMA library as shown in [Installing the Linux Driver](#).

### 14.2.3 Verifying RoCE Functionality

To verify RoCE functionality:

1. Verify that the Broadcom NIC driver is loaded using the following command:

```
# modinfo bnxt_en
```

2. Verify that the Broadcom RoCE driver is loaded using the following command:

```
# modinfo bnxt_re
```

3. Verify that the RoCE kernel is loaded using the following commands:

```
# lsmod | grep ib_
ib_isert                50818  0
iscsi_target_mod       291713  1 ib_isert
ib_iser                 47861  0
libiscsi                57233  1 ib_iser
scsi_transport_iscsi   99909  2 ib_iser,libiscsi
ib_srpt                 43624  0
target_core_mod        340809  3 iscsi_target_mod,ib_srpt,ib_isert
ib_srp                  52589  0
scsi_transport_srp     20993  1 ib_srp
ib_ipoib                115042  0
ib_ucm                  22636  0
ib_uverbs               78543  2 ib_ucm,rdma_ucm
ib_umad                 22119  0
rdma_cm                 59529  4 rprdma,ib_iser,rdma_ucm,ib_isert
ib_cm                   51741  5 rdma_cm,ib_srp,ib_ucm,ib_srpt,ib_ipoib
ib_core                 236827  14
rdma_cm,ib_cm,iw_cm,rprdma,ib_srp,ib_ucm,ib_iser,ib_srpt,ib_umad,ib_uverbs,bnxt_re,rdma_ucm,ib_i
poib,ib_isert
```

4. Verify that the Broadcom RoCE devices are available using the following commands:

```
# ibv_devices

[root@dhcp-10-13-107-210 ~]# ibv_devices
device                node GUID
-----
bnxt_re1              021918ffffead11fa1
bnxt_re0              021018ffffead1fa0
```

5. Verify that the Broadcom RoCE device is enabled and can be accessed using the following commands:

```
# ibv_devinfo -d bnxt_re0

[root@dhcp-10-13-107-210 ~]# ibv_devinfo -d bnxt_re0
hca_id: bnxt_re0
transport:                               Infini Band (0)
```

```

node_guid:                0210:18ff: fead:1fa0
sys_image_guid:           0210:18ff: fead:1fa0
vendor_id:                0x14e4
vendor_part_id:           5847
hw_ver:                   0x1405
phys_port_cnt:            1
    port:                  1
        state:              PORT_ACTIVE (4)
        max_mtu:             4096 (5)
        active_mtu:         1024 (3)
        sm_lid:              0
        port_lid:           0
        port_lmc:            0x00
        link_layer:          Ethernet

# ibv_devinfo -d bnxt_re1

```

```

[root@dhcp-10-13-107-210 ~]# ibv_devinfo -d bnxt_re1
hca_id: bnxt_re0
    transport:              Infini Band (0)
    node_guid:              0210:18ff: fead:1fa1
    sys_image_guid:         0210:18ff: fead:1fa1
    vendor_id:              0x14e4
    vendor_part_id:         5847
    hw_ver:                 0x1405
    phys_port_cnt:          1
        port:                1
            state:            PORT_ACTIVE (4)
            max_mtu:          4096 (5)
            active_mtu:       1024 (3)
            sm_lid:            0
            port_lid:         0
            port_lmc:         0x00
            link_layer:        Ethernet

```

## 14.2.4 RoCE Connectivity Tests

This section provides information on how to test basic RoCE connectivity using rping. The test utilities use a server and client model during RoCE testing.

### 1. Server side: rping -s -a <ServerIP> -v

```

[root@localhost ~]# rping -s -a 10.0.0.5 -v
server ping data: rdma-ping-1052: IJKLMNOPQRSTUVWXYZ[\]^_'abcdefghijklmnopqrstuvw
server ping data: rdma-ping-1053: IJKLMNOPQRSTUVWXYZ[\]^_'abcdefghijklmnopqrstuvw
server ping data: rdma-ping-1054: IJKLMNOPQRSTUVWXYZ[\]^_'abcdefghijklmnopqrstuvw
server ping data: rdma-ping-1055: IJKLMNOPQRSTUVWXYZ[\]^_'abcdefghijklmnopqrstuvw
server ping data: rdma-ping-1056: IJKLMNOPQRSTUVWXYZ[\]^_'abcdefghijklmnopqrstuvw
server ping data: rdma-ping-1057: IJKLMNOPQRSTUVWXYZ[\]^_'abcdefghijklmnopqrstuvw
server ping data: rdma-ping-1058: IJKLMNOPQRSTUVWXYZ[\]^_'abcdefghijklmnopqrstuvw
server ping data: rdma-ping-1059: IJKLMNOPQRSTUVWXYZ[\]^_'abcdefghijklmnopqrstuvw
server ping data: rdma-ping-1060: IJKLMNOPQRSTUVWXYZ[\]^_'abcdefghijklmnopqrstuvw
server ping data: rdma-ping-1061: IJKLMNOPQRSTUVWXYZ[\]^_'abcdefghijklmnopqrstuvw
server ping data: rdma-ping-1062: IJKLMNOPQRSTUVWXYZ[\]^_'abcdefghijklmnopqrstuvw
server ping data: rdma-ping-1063: IJKLMNOPQRSTUVWXYZ[\]^_'abcdefghijklmnopqrstuvw

```

### 2. Client side: rping -c -a <ServerIP> -v

```
[root@localhost ~]# rping -c -a 10.0.0.5 -v
server ping data: rdma-ping-1035: FGHIJKLMNOPQRSTUVWXYZ[\]^_'abcdefghijklmnopqr
server ping data: rdma-ping-1036: GHIJKLMNOPQRSTUVWXYZ[\]^_'abcdefghijklmnopqrs
server ping data: rdma-ping-1037: HIJKLMNOPQRSTUVWXYZ[\]^_'abcdefghijklmnopqrst
server ping data: rdma-ping-1038: IJKLMNOPQRSTUVWXYZ[\]^_'abcdefghijklmnopqrstu
server ping data: rdma-ping-1039: JKLMNOPQRSTUVWXYZ[\]^_'abcdefghijklmnopqrstuv
server ping data: rdma-ping-1040: KLMNOPQRSTUVWXYZ[\]^_'abcdefghijklmnopqrstuvw
server ping data: rdma-ping-1041: LMNOPQRSTUVWXYZ[\]^_'abcdefghijklmnopqrstuvwx
server ping data: rdma-ping-1042: MNOPQRSTUVWXYZ[\]^_'abcdefghijklmnopqrstuvwxy
server ping data: rdma-ping-1043: NOPQRSTUVWXYZ[\]^_'abcdefghijklmnopqrstuvwxyz
server ping data: rdma-ping-1044: OPQRSTUVWXYZ[\]^_'abcdefghijklmnopqrstuvwxyza
server ping data: rdma-ping-1045: PQRSTUVWXYZ[\]^_'abcdefghijklmnopqrstuvwxyzAB
server ping data: rdma-ping-1046: QQRSTUVWXYZ[\]^_'abcdefghijklmnopqrstuvwxyzABC
server ping data: rdma-ping-1047: RSTUVWXYZ[\]^_'abcdefghijklmnopqrstuvwxyzABCD
server ping data: rdma-ping-1048: STUVWXYZ[\]^_'abcdefghijklmnopqrstuvwxyzABCDE
server ping data: rdma-ping-1049: TUVWXYZ[\]^_'abcdefghijklmnopqrstuvwxyzABCDEF
server ping data: rdma-ping-1050: UVWXYZ[\]^_'abcdefghijklmnopqrstuvwxyzABCDEFG
server ping data: rdma-ping-1050: VWXYZ[\]^_'abcdefghijklmnopqrstuvwxyzABCDEFGH
server ping data: rdma-ping-1050: WXYZ[\]^_'abcdefghijklmnopqrstuvwxyzABCDEFGHI
server ping data: rdma-ping-1050: XYZ[\]^_'abcdefghijklmnopqrstuvwxyzABCDEFGHIJ
server ping data: rdma-ping-1051: XYZ[\]^_'abcdefghijklmnopqrstuvwxyzABCDEFGHIK
server ping data: rdma-ping-1051: YZ[\]^_'abcdefghijklmnopqrstuvwxyzABCDEFGHIKM
server ping data: rdma-ping-1051: Z[\]^_'
```

### 3. Basic ping pong test:

a. Server side: `ibv_rc_pingpong -g 0 -d bnxt_re<x> -i 1`

```
[root@dhcp-10-13-107-210 ~]# ibv_rc_pingpong -g0 -d bnxt_re0 -il
local address: LID 0x0000, QPN 0x000094, PSN 0x390112, GID fe80::210:18ff:fead:1fa0
remote address: LID 0x0000, QPN 0x000095, PSN 0xe00732, GID fe80::210:19ff:fead:1fa0
8192000 bytes in 0.01 seconds = 4410.53 Mbit/sec
1000 iters in 0.01 seconds = 14.86 usec/iter
```

b. Client side: `ibv_rc_pingpong -g 0 -d bnxt_re<x> -i 1 <ServerIP>`

```
[root@dhcp-10-13-107-210 ~]# ibv_rc_pingpong -g0 -d bnxt_re0 -il 10.0.0.50
local address: LID 0x0000, QPN 0x000095, PSN 0xe00732, GID fe80::210:18ff:fead:1fa0
remote address: LID 0x0000, QPN 0x000094, PSN 0x390112, GID fe80::210:19ff:fead:1fa0
8192000 bytes in 0.01 seconds = 4432.60 Mbit/sec
1000 iters in 0.01 seconds = 14.78 usec/iter
```

#### 14.2.4.1 Performance Test (perftest) Package

The perftest package is a collection of tests written over uverbs intended for use as a performance micro-benchmark. The tests are used for tuning as well as for functional testing. The server-side IB server and the client-side IB client must be run.

The perftest package contains the following bandwidth and latency benchmarks:

- `ib_send_bw`
- `ib_send_lat`
- `ib_write_bw`
- `ib_write_lat`
- `ib_read_bw`
- `ib_read_lat`

1. The IB test tool help commands are viewed using the following `-h` parameter in the command:



```
# ib_send_bw -h
```

2. Verify that the IP and RoCE node is up on both sides with the following command:

```
ifconfig InterfaceName ; ibv_devices;
```

The Interface Name is the name of the Ethernet interface.

```
[root@localhost ~]# ifconfig ens2f8 : ibv_devices
ens2f8: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
    inet 192.168.1.2 netmask 255.255.255.0 broadcast 192.168.1.255
    inet6 fe80::9dc:71ff:fc0:c10 prefixlen 64 scopeid 0x20<link>
    ether 9c:dc:71:c0:0c:10 txqueuelen 1000 (Ethernet)
    RX packets 249 bytes 41586 (40.6 KiB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 439 bytes 51150 (49.9 KiB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

device            node GUID
-----
bnxt_re1          9dc71ffffc00c10
bnxt_re0          9dc71ffffc00c10
```

3. Run `ib_send_bw` on both sides with the following commands:

- a. Server Side: `ib_send_bw -d <RoCE Node> --report_gbits`

```
[root@localhost ~]# ib_send_bw -d bnxt_re0 --report_gbits
*****
* Waiting for client to connect... *
*****
-----
Send BW Test
Dual-port      : OFF          Device      : bnxt_re0
Number of qps  : 1           Transport type : IB
Connection type : RC         Using SRQ    : OFF
RX depth       : 512
CQ Moderation  : 100
MTU            : 1024[IB]
Link type      : Ethernet
GUID index     : 2
Max inline data : 8[IB]
rdma_cm QPs    : OFF
Data ex. method : Ethernet
-----
local address: LID 0000 QPN 0x0000 PSN 0x6ec690
GUID: 00:00:00:00:00:00:00:00:00:00:255:255:192:168:01:01
remote address: LID 0000 QPN 0x0000 PSN 0x24bda2
GUID: 00:00:00:00:00:00:00:00:00:00:255:255:192:168:01:02
-----
#bytes      #iterations  BW peak[Gb/sec]  BW average[Gb/sec]  AvgRate[Mpps]
65536       1000          0.00             22.98                0.043825
```

b. Client Side: `ib_send_bw -d <RoCE Node> --report_gbits <Server IP>`

```

root@localhost ~# ib_send_bw -d bnxt_re8 --report_gbits 192.168.1.1
-----
Send BW Test
Dual-port      : OFF          Device      : bnxt_re8
Number of qps  : 1           Transport type : IB
Connection type : RC         Using SRQ    : OFF
TX depth      : 128
CQ Moderation  : 1000
MTU           : 1024100
Link type     : Ethernet
GID index     : 2
Max inline data : 8100
rdma_cm QPs   : OFF
Data ex. method : Ethernet
-----
local address: LID 0000 QPN 0x0000 PSM 0x2dhd2
GID: 00:00:00:00:00:00:00:00:00:00:255:255:192:168:01:02
remote address: LID 0000 QPN 0x0000 PSM 0x60c690
GID: 00:00:00:00:00:00:00:00:00:00:255:255:192:168:01:01
-----
#bytes      #iterations  BW peak[Gb/sec]  BW average[Gb/sec]  MsgRate[Mpps]
65536       1000         22.91            22.91                0.843689
-----

```

**NOTE:** With dual-port NICs, if both ports are on the same subnet, RDMA perfest commands may fail. This is caused by an arp flux issue in Linux. Use multiple subnets for testing or bringing the other interface down.

## 14.2.5 RoCE Congestion Control

Broadcom Ethernet Network adapter Congestion Control (NCC) is a standards-based congestion control mechanism for RDMA over Ethernet v2 (RoCEv2), which utilizes Explicit Congestion Notification (RFC 3168) to signal to RoCE transmitting stations the congestion status of the network, which is used by Broadcom Ethernet Network adapters to control transmit rate, thereby reducing congestion, reducing packet drops, and minimizing network latency by keeping switch transmit queues minimally loaded.

NCC can work in conjunction with Flow Control including pause frames and Priority Flow Control to guarantee a lossless network, or NCC can operate alone.

### 14.2.5.1 Using the Section

This section is intended to provide information on the process of configuring a new server for RoCE use with RoCEv2 enabled, Congestion Control enabled, and optionally, PFC enabled and configured.

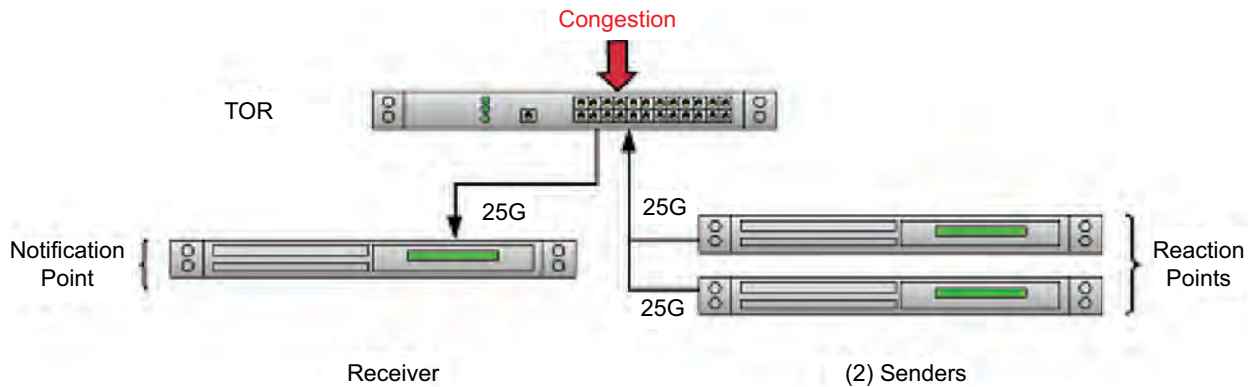
Any installation of other vendor's proprietary Infiniband software stack should be removed before proceeding with this section.

### 14.2.5.2 Understanding Congestion Control

#### 14.2.5.2.1 Basic Topology

Congestion control topology provides a basic topology of how to test congestion control. In this topology, there are two senders transmitting packets at 25G to a receiver that can only receive at 25G.

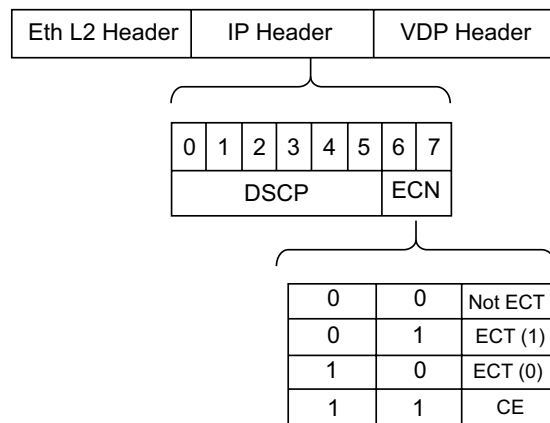
Figure 50: Congestion Control Topology



#### 14.2.5.2.2 Background

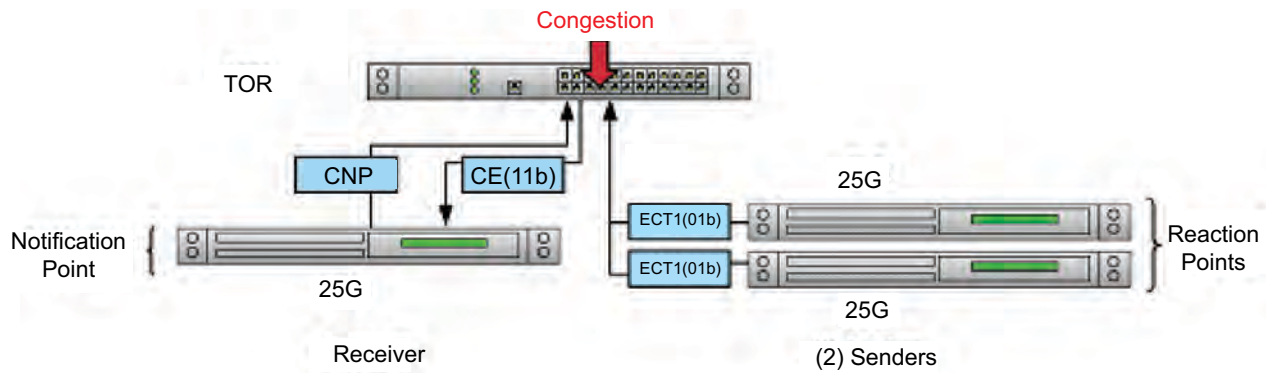
Congestion control flow provides a simple overview of ECN congestion management.

1. After ECN has been configured and enabled, the ECN bits in the IP header of RoCEv2 packets are either ECT1(01b) or ECT0 (10b) based on the user configuration.



2. As the Egress Queue of the switch reaches the congestion marking threshold as defined by the user, the switch starts marking the ECN bits to CE(11b) to notify to the notification point that congestion has been detected for that port.
3. Once the CE bit is detected, the notification point transmits Congestion Notification Packet (CNP) packet(s) to the reaction points(s) to signify congestion.
4. Upon the reception of the CNP(s), the reaction point reduces its transmit rate based on a certain algorithm so that the congestion can be mitigated.
5. The reaction point continues to react to rate limit requests until Congestion Notification Packet (CNP) packets are no longer received.
6. At times when CNPs are not received by the transmitter, the transmitter attempt to gradually increase its transmission rate.

**NOTE:** Both reduction in rate and increase in rate are be accomplished per QP

**Figure 51: Congestion Control Flow**

### 14.2.5.3 NIC Configuration

In order to configure the NIC to be lossless for RoCE, the correct traffic class is required to be used as the default traffic class is currently set to lossy. There are two available methods to configure the traffic classes and other QoS settings:

- LLDP Agent
- BnxtQoS

### 14.2.5.4 LLDP Agent

The LLDP agent is used to set QoS mappings (user priority, Linux TC, or NIC CoS queue) and configure Enhanced Transmission Selection (ETS) to configure bandwidth allocation if required. It is required to install the LLDP agent in order to properly configure QoS, which is necessary for maximum performance.

#### 14.2.5.4.1 Debian/Ubuntu

To install the LLDP agent using Debian/Ubuntu, use the following commands:

```
sudo apt install libconfig9 libnl-3-200
wget http://ftp.us.debian.org/debian/pool/main/l/lldpad/lldpad_0.9.46-3.1_amd64.deb
sudo dpkg -i lldpad_0.9.46-3.1_amd64.deb
sudo systemctl enable lldpad
sudo systemctl start lldpad
```

#### 14.2.5.4.2 Red Hat/CentOS

To install the LLDP agent using RedHat/CentOS, use the following commands:

```
sudo yum install lldpad
sudo systemctl enable lldpad
sudo systemctl start lldpad
```

### 14.2.5.5 BnxtQoS

Similar to the LLDP Agent, the Broadcom QoS configuration utility is used to set QoS mappings, priority flow control, and to configure Enhanced Transmission Selection (ETS). BnxtQoS only configures settings on the NIC side and requires the following:

- Switch configuration to match that of the NIC QoS settings.
- Disabling of the LLDPAD on the host and in the firmware (if enabled).

**Table 38: Usage: bnxtqos -dev=<interface name> <command> [-options [...]]**

Commands	
version	Display program version details.
set_ets	This command is used to configure the priority to tc and Bandwidths. This command also used to configure TSA and Bandwidths. <b>Syntax:</b> bnxtqos -dev=<interface name> set_ets tsa=<tc[0-7]:tc_type[ets/strict]> [priority2tc=<priority[0-7]:tc>] tcbw=<bandwidths> with comma separated and in units of percentage (%). <b>Example:</b> bnxtqos -dev=p7p1 set_ets tsa=0:ets,1:ets,2:strict,3:strict,4:strict,5:strict,6:strict,7:strict priority2tc=0:0,1:0,2:0,3:0,4:0,5:1,6:0,7:0 tcbw=70,30
set_pfc	To enable PFC on given priority. Valid values are from 0 to 7. <b>Syntax:</b> bnxtqos -dev=<interface name> set_pfc enabled=<0-7> the values should be with comma separated <b>Example:</b> bnxtqos -dev=p7p1 set_pfc enabled=5,6
set_apptlv	This parameter is used to configure the priority of the APPTLV. <b>Syntax:</b> bnxtqos -dev=<interfac name> set_apptlv app=<priority,selector,protocol> <b>Syntax:</b> bnxtqos -dev=<interfac name> set_apptlv -d app=<priority,selector,protocol>  <b>Example:</b> bnxtqos -dev=p7p1 set_apptlv app=5,1,35093 <b>Example:</b> bnxtqos -dev=p7p1 set_apptlv -d app=5,1,35093
get_qos	This command is used to get the configured priorities and bandwidth parameters.
ratelimit	This command is used to set the rate limit for each TC in units of percentage (%). <b>Example:</b> bnxtqos -dev=p7p1 ratelimit 80,60,70
dump	This command is used to dump the supported dump strings. The supported dump string are pri2cos. <b>Syntax:</b> bnxtqos -dev=<device name> dump <dump string> <b>Example:</b> bnxtqos -dev=p7p1 dump pri2cos
Options	
-v[v][v]	Enable extra console output (increase verbosity).
-V	Enable maximum verbosity of console output.
-d	Removes the application TLV.

### 14.2.5.6 DSCP vs. VLAN Mode

NCC has two operational modes: VLAN and DSCP. If your network uses VLAN tagging for traffic, the VLAN header is used to track traffic priority values, necessary to separate RoCE traffic from other traffic for congestion control. If your network uses DSCP tagging for traffic, the proper DSCP mapping is required to be set. In either case, the corresponding switch must be configured correctly to ensure the priority (VLAN or DSCP) map to a specific traffic class priority on the switch that has the ECN enabled and configured for the specified marking thresholds.

Determine the interface name of the NIC with the following command:

```
ifconfig -a
```

Create a boot-time configuration file for the interface.

#### 14.2.5.6.1 Debian/Ubuntu

Add to /etc/network/interfaces:

```
auto <iface>
iface <iface> inet static
```

```

address w.x.y.z
netmask a.b.c.d
gateway e.f.g.h
<iface> must be in the form of <iface>.<vlan id> to use VLANs

```

### 14.2.5.6.2 RedHat/CentOS

#### 1. Add to /etc/sysconfig/network-scripts/ifcfg-<iface>:

```

DEVICE=<iface>
ONBOOT=yes
BOOTPROTO=static
NM_CONTROLLED=no
NETMASK=a.b.c.d
IPADDR=w.x.y.z
# Uncomment below for VLAN
#VLAN=yes
#VLAN_EGRESS_PRIORITY_MAP=0:0,1:1,2:1,3:1,4:1,5:1,6:1,7:1
*<iface> must be in the form of <iface>.<vlan id> to use VLANs

```

#### 2. Create an /etc/ifup-roce-<IFACE> script to allow boot-time RoCE configuration:

```

#!/bin/sh
sleep 2

/sbin/modprobe bnxt_re
sleep 5

/sbin/ethtool -A $IFACE autoneg off rx off tx off
sleep 2

#Using bnxtqos
bnxtqos -dev=$IFACE set_apptlv app=1,5,34
bnxtqos -dev=$IFACE set_apptlv app=1,1,35093
bnxtqos -dev=$IFACE set_apptlv app=1,3,4791
bnxtqos -dev=$IFACE set_apptlv set_ets
tsa=0:ets,1:ets,2:strict,3:strict,4:strict,5:strict,6:strict,7:strict
up2tc=0:0,1:1,2:0,3:0,4:0,5:0,6:0,7:0 tcbw=10,90
bnxtqos -dev=$IFACE set_pfc enabled=1

#Uncomment if using LLDP Agent
#/usr/sbin/lldptool -T -i $IFACE -V APP app=1,5,34*
#/usr/sbin/lldptool -T -i $IFACE -V APP app=1,1,35093
#/usr/sbin/lldptool -T -i $IFACE -V APP app=1,3,4791
#/usr/sbin/lldptool -T -i $IFACE -V ETS-CFG
tsa=0:ets,1:ets,2:strict,3:strict,4:strict,5:strict,6:st
#rict,7:strict up2tc=0:0,1:1,2:0,3:0,4:0,5:0,6:0,7:0 tcbw=10,90
#/usr/sbin/lldptool -T -i $IFACE -V PFC enabled=1
#systemctl restart lldpad
sleep 10

/sbin/modprobe bnxt_re
sleep 5

mkdir -p /sys/kernel/config/rdma_cm/bnxt_re0
echo RoCE v2 > /sys/kernel/config/rdma_cm/bnxt_re0/ports/1/default_roce_mode

mkdir -p /sys/kernel/config/bnxt_re/bnxt_re0

```

```
cd /sys/kernel/config/bnxt_re/bnxt_re0/ports/1/cc/

# Uncomment below for DSCP Mode
#echo -n 0x22 > roce_dscp           # DSCP RoCE packets
#echo -n 0x6 > cnp_prio             # Priority for CNP Packets
#echo -n 0x23 > cnp_dscp           # DSCP for CNP packets
#echo -n 0x1 > vlan_tx_disable     # Set DSCP mode

# Uncomment below for VLAN mode
#echo -n 0x0 > vlan_tx_disable     # Set VLAN mode
#echo -n 0x6 > cnp_prio           # Priority for CNP Packets
```

The support for DSCP tagging through lldptool is supported on a newer kernel.

3. Set the script to be executable with the following command:

```
sudo chmod 755 /etc/ifup-roce-<IFACE>
```

#### 14.2.5.6.3 Debian

To configure a boot-time RoCE configuration, create the following `/etc/network/if-up.d/roce` script with the following commands:

```
#!/bin/sh

if [ -x /etc/ifup-roce- $\{IFACE\}$  ]; then
  /etc/ifup-roce- $\{IFACE\}$ 
fi

Set the script to be executable with the following command:

sudo chmod 755 /etc/network/if-up.d/roce
```

#### 14.2.5.6.4 Red Hat

1. Create an `/sbin/ifup-local` script to allow boot-time RoCE configuration with the following commands:

```
#!/bin/sh

export IFACE=$1

if [ -x /etc/ifup-roce- $\{IFACE\}$  ]; then
  . /etc/ifup-roce- $\{IFACE\}$ 
fi
```

2. Set the script to be executable with the following commands:

```
sudo chmod 755 /sbin/ifup-local
```

3. Reconfigure the interface with the following commands:

```
sudo ifdown <IFACE>
sudo ifup <IFACE>
```

#### 14.2.5.7 Congestion Control Settings

Different topologies may require different tunings. See [Performance](#) for additional information.

### 14.2.5.7.1 Congestion Control Parameters

**Table 39: Congestion Control Parameters**

CC Parameter <sup>a</sup>	Description
inact_cp	Inactivity time after which the CC parameters of a QP are re-initialized.
nph_per_state	Number of phases per state. 0 to 7. How many RTTs in Fast Recovery state.
cnp_dscp	DSCP value for RoCE Congestion Notification Packets.
roce_dscp	DSCP value for RoCE Packets.
cc_mode	0 for DCTCP algorithm, 1 for TCP. Default 0.
cnp_prio	Priority for RoCE Congestion Notification Packets.
ecn_enable	Enable congestion control.
ecn_marking	Specifies which ECN bit to set. 0x1 for 01b or 0x2 for 10b.
g	Running average weight in computing cp (Congestion Probability). The weight is $2^g$ , g between 1 and 7, default 4.
init_cr	Current rate after on QP creation and after QP has not transmitted for more than inctivity_th. Value between 0 and 1023 as fraction of full link BW.
int_tr	Target rate after on QP creation and after QP has not transmitted for more than inctivity_th. Value between 0 and 1023 as fraction of full link BW.
roce_prio	Priority for RoCE Packets.
rtt	Time period [us] over which cnp and transmitted packets counts accumulate. At the end of Rtt ratio between CNPs and TxPkts computed and cp is update.
tcp_cp	The reduction level on reception of CNP when cc_mode is 1 (TCP mode). Value between 0 and 1023, representing 0-1.0.
vlan_tx_disable	Disables the VLAN headers (required if using DSCP).
apply	Applies the settings.

a. Releases prior to 20.8 have slightly different naming conventions.

The following is an example of setting congestion control parameters:

```
mkdir -p /sys/kernel/config/rdma_cm/bnxt_re0
echo RoCE v2 > /sys/kernel/config/rdma_cm/bnxt_re0/ports/1/default_roce_mode

mkdir -p /sys/kernel/config/bnxt_re/bnxt_re0
cd /sys/kernel/config/bnxt_re/bnxt_re0/ports/1/cc/

# DSCP configuration
echo -n 0x1 > roce_dscp
echo -n 0x1 > roce_prio
echo -n 0x1 > vlan_tx_disable

# Set CC parameters for and enable
echo -n 0x1 > ecn_enable
echo -n 0x1 > ecn_marking
echo -n 0xc8 > tcp_cp
echo -n 0x3 > nph_per_state
echo -n 0x28 > rtt
echo -n 0x4 > g
echo -n 0x1f4 > init_cr
echo -n 0x3ff > init_tr
echo -n 0x1f40 > inact_cp
echo -n 0x1 > apply
```



### 14.2.5.7.2 Setup Script

A setup script is provided as part of the release to configure the priority, DSCP, and congestion control parameters. The script does not load the drivers, bring up the interface, or setup the addresses.

**Usage:** `bnxt_setupcc.sh [OPTION]...`

```
-d - RoCE Device Name (e.g. bnxt_re0, bnxt_re_bond)
-i - Ethernet Interface Name (example: plp1 or for bond, specify slave interfaces like -i p6p1 -i p6p2)
-r [0-7] - RoCE Packet Priority
-s VALUE - RoCE Packet DSCP Value
-c [0-7] - RoCE CNP Packet Priority
-p VALUE - RoCE CNP Packet DSCP Value
-h - Display help
```

**NOTE:** In SIT release 212.1.x, the `cosq` used for CNPs has been removed from the user scope. Only two traffic classes are available to the user. Previous versions of the script were still configuring the 3rd traffic to be used for CNPs which results in unexpected behaviors (ensure that traffic class 2 is not set in the `prio2tc`).

The following is an example of running the setup script:

```
cd $REL_DIR/Linux/netxtreme-bnxt_en-x.y.z/bnxt_re
./bnxt_setupcc.sh -d bnxt_re0 -i plp1-r 5 -s 0x22 -c 6 -p 0x23
```

### 14.2.5.8 Switch Configuration

NCC uses ECN to react to marked packets within the network switch infrastructure during times of congestion. The correct ECN threshold value is specific to each switch port, and is dependent on the link speed of that port. The following table shows which values are to be used for deterministic marking (`cc_mode=0`).

Link Speed	ECN Min./Max. Threshold (kilobytes)
10 Gb/s	12/12
25 Gb/s	16/16
50 Gb/s	22/22
100 Gb/s	33/33

Each switch port participating in NCC must be configured with the above ECN threshold for the class of service associated with RoCEv2 traffic. Different vendors have different naming conventions that specifies the minimum and maximum marking threshold as well as the marking percentage. The following sections provide examples of setting such parameters aforementioned.

#### 14.2.5.8.1 Operation

With the switches, servers, and NICs configured, RDMA applications can be deployed to run over the RoCE network. Any `libverbs`-linked application can be used with appropriate changes for using IP addresses rather than GUIDs for end-point addressing.

When using NCC, it is important to specify a RoCEv2 GUID. RoCEv1 does not support NCC.

1. To see a list of GUIDs supported by the device, use the following `ibv_devinfo -vvvv`:

```
[root@Host1 ~]# ibv_devinfo -vvvv
hca_id:bnxt_rel
```

```

transport:InfiniBand (0)
fw_ver:      212.1.93.0
node_guid:020a:f7ff:fea6:9cf1
sys_image_guid:020a:f7ff:fea6:9cf1
vendor_id:0x14e4
vendor_part_id:5847
hw_ver:      0x1406
phys_port_cnt:1
max_mr_size:0x8000000000
page_size_cap:0x201000
...
    active_width:1X (1)
    active_speed:25.0 Gb/s (32)
    phys_state:DISABLED (3)

GID[ 0]:fe80:0000:0000:0000:020a:f7ff:fea6:9cf0, RoCE V1 (IPV6)
GID[ 1]:fe80:0000:0000:0000:020a:f7ff:fea6:9cf0, RoCE V2 (IPV6)
GID[ 2]:0000:0000:0000:0000:0000:ffff:ac10:0102, RoCE V1 (IPV4)
GID[ 3]:0000:0000:0000:0000:0000:ffff:ac10:0102, RoCE V2 (IPV4)

```

## 2. Odd-numbered GIDs are used for RoCEv2 and this can be verified with the following command:

```
# cat /sys/class/infiniband/bnxt_re0/ports/1/gid_attrs/types/1
```

**NOTE:** If the above directory does not exist, this indicates the installed kernel/OFED version does not support RoCE version 2.

A typical application of exercising the RoCE interface would be to use rping or perftest. Perftest can be obtained from github or via most Linux package repositories such as:

```
RHEL/CentOS: yum install perftest
Ubuntu: apt install perftest
```

For example, the perftest tools can be run as:

```
# ib_write_bw -d bnxt_re0 -x 3 -F --report_gbits -p 1800 -s <message_size> -q <num of qps> -D <duration>
192.168.30.3
```

```
# ib_write_lat -d bnxt_re0 -x 3 -F --report_gbits -p 1800 -s 2 192.168.30.3
```

For a complete explanation of the parameters used by `ib_write_bw/lat` and other IB utilities, see github as previously described.

### 14.2.5.8.2 Analytics

RoCE and congestion control statistics can be viewed from the sysfs files using the following commands:

```
# cat /sys/kernel/debug/<device name>/info (for example, /sys/kernel/debug/bnxt_re0/info)
bnxt_re debug info:
Adapter count: 1
=====[ IBDEV bnxt_re_bond0 ]=====
    link state: UP
    Max QP: 0xff80
    Max SRQ: 0x7fff
    Max CQ: 0xffff
    Max MR: 0x3ffff
    Max MW: 0x8000c
```

```

Active QP: 2
Active SRQ: 0
Active CQ: 1
Active MR: 0
Active MW: 1
Recoverable Errors: 0
Active QPs P0: 0
Active QPs P1: 2
Rx Pkts: 42907336148
Rx Bytes: 2664310290076
Tx Pkts: 229562600627
Tx Bytes: 248444124528782
CNP Tx Pkts: 0
CNP Tx Bytes: 0
CNP Rx Pkts: 338012327
CNP Rx Bytes: 25012912198

```

### 14.2.5.9 Counter Definitions

The following tables provide counter definitions.

**Table 40: Device Resource Limits**

Counters	Description
Max QP	Max number of QP limit
Max SRQ	Max number of SRQ limit
Max CQ	Max number of CQs limit
Max MR	Max number of memory region limit
Max MW	Max number of memory window limit

**Table 41: Active Resources**

Counters	Description
Active QP	Number of active QPs.
Active SRQ	Number of active SRQs.
Active CQ	Number of active CQs.
Active MR	Number of active Memory Regions.
Active MW	Number of active Memory Windows.

**Table 42: Bytes and Packet Counters**

Counters	Description
RX Pkts	Number of RoCE (v1/v2) packets received.
RX Bytes	Number of RoCE (v1/v2) bytes received.
TX Pkts	Number of RoCE (v1/v2) packets transmitted.
TX Bytes	Number of RoCE (v1/v2) bytes transmitted.

**Table 43: Congestion Notification Counters**

Counters	Description
CNP TX Pkts	Number of RoCE CNP packets received.
CNP TX Bytes	Number of RoCE CNP bytes received.
CNP RX Pkts	Number of RoCE CNP packets transmitted.
CNP RX Bytes	Number of RoCE CNP bytes transmitted.

**Table 44: Recoverable Errors**

Counters	Description
Recoverable Errors	Number of recoverable errors detected. Recoverable errors are detected by the HW. HW instructs FW to initiate the recovery process. RC connection does not teardown as a result of these errors.
to_retransmits	Number of retransmission requests.
rnr_naks_rcvd	Number of RNR (Receiver-Not-Ready) NAKs received.
dup_req	Number of duplicated requests detected.
missing_resp	Number of responses missing.

**Table 45: Fatal Errors**

Counters	Description
seq_err_naks_rcvd	Number of PSN sequencing error NAKs received.
max_retry_exceeded	Number of retransmission requests exceeded the maximum.
unrecoverable_err	Number of unrecoverable errors detected.
bad_resp_err	Number of bad response errors detected.
local_qp_op_err	Number of QP local operation errors detected.
local_protection_err	Number of local protection errors detected.
mem_mgmt_op_err	Number of times HW detected an error because of illegal bind/fast register/invalidate attempted by the driver.
remote_invalid_req_err	Number of invalid request received from the remote rdma initiator.
remote_access_err	Number of times H/W received a REMOTE ACCESS ERROR NAK from the peer.
remote_op_err	Number of times HW received a REMOTE OPERATIONAL ERROR NAK from the peer.

**Table 46: Responder Errors**

Counters	Description
res_exceed_max	Number of times HW detected incoming Send, RDMA write or RDMA read messages which exceed the maximum transfer length.
res_length_mismatch	Number of times HW detected incoming RDMA write message payload size does not match write length in the RETH.
res_exceeds_wqe	Number of times HW detected Send payload exceeds RQ/SRQ RQE buffer capacity.
res_opcode_err	Number of times HW detected First, Only, Middle, Last packets for incoming requests are improperly ordered with respect to the previous packet.
res_rx_invalid_rkey	Number of times HW detected a incoming request with an R_KEY that did not reference a valid MR/MW.
res_rx_domain_err	Number of times HW detected a incoming request with an R_KEY that referenced a MR/MW that was not in the same PD as the QP on which the request arrived.

**Table 46: Responder Errors (Continued)**

Counters	Description
res_rx_no_perm	Number of times HW detected an incoming RDMA write request with an R_KEY that referenced a MR/MW which did not have the access permission needed for the operation.
res_rx_range_err	Number of times HW detected an incoming RDMA write request that had a combination of R_KEY, VA and length that was out of bounds of the associated MR/MW.
res_tx_invalid_rkey	Number of times HW detected a R_KEY that did not reference a valid MR/MW while processing incoming read request.
res_tx_domain_err	Number of times HW detected an incoming request with an R_KEY that referenced a MR/MW that was not in the same PD as the QP on which the RDMA read request is received.
res_tx_no_perm	Number of times HW detected an incoming RDMA read request with an R_KEY that referenced a MR/MW which did not have the access permission needed for the operation.
res_tx_range_err	Number of times HW detected an incoming RDMA read request that had a combination of R_KEY, VA and length that was out of bounds of the associated MR/MW.
res_irrq_oflow	Number of times HW detected that peer sent us more RDMA read or atomic requests that the negotiated maximum.
res_unsup_opcode	Number of times HW detected that peer sent us a request with an opcode for a request type that is not supported on this QP.
res_unaligned_atomic	Number of times HW detected that VA of an atomic request is on a memory boundary that prevents atomic execution.
res_rem_inv_err	Number of times HW detected an incoming send with invalidate request in which the R_KEY to invalidate did not MR/MW which could be invalidated.
res_mem_error64	Number of times HW detected a RQ/SRQ SGE which points to an inaccessible memory.
res_srq_err	Number of times HW detected a QP moving to error state because the associated SRQ is in error.
res_cmp_err	Number of time HW detected that there is no CQE space available on CQ or CQ is not in valid state.
res_invalid_dup_rkey	Number of times HW detected invalid R_KEY while re-sending responses to duplicate read requests.
res_wqe_format_err	Number of times HW detected error in the format of the WQE in the RQ/SRQ.
res_cq_load_err	Number of times HW detected error while attempting to load the CQ context.
res_srq_load_err	Number of times HW detected error while attempting to load the SRQ context.

**Table 47: PCI Errors**

Counters	Description
res_tx_pci_err	Number of responder transmit PCI errors detected.
res_rx_pci_err	Number of responder receive PCI errors detected.

## 14.2.5.10 Applications

### 14.2.5.10.1 OpenMPI

If OpenMPI is used, the OpenMPI source and configuration must be patched using the following commands:

```
$ cd openmpi-x.y.z
$ cat > openmpi.patch << EOF
diff --git a/opal/mca/common/verbs/common_verbs_port.c b/opal/mca/common/verbs/common_verbs_port.c
index 831ba3f..7eb30 100644
--- a/opal/mca/common/verbs/common_verbs_port.c
+++ b/opal/mca/common/verbs/common_verbs_port.c
@@ -68,6 +68,10 @@ int opal_common_verbs_port_bw(struct ibv_port_attr *port_attr,
```

```

        /* EDR: 25.78125 Gb/s * 64/66, in megabits */
        *bandwidth = 25000;
        break;
+   case 64:
+       /* ODR: 50 Gb/s * 64/66, in megabits */
+       *bandwidth = 50000;
+       break;
+   case 128:
+       /* ODR: 100 Gb/s * 64/66, in megabits */
+       *bandwidth = 100000;
+       break;
    default:
        /* Who knows? */
        return OPAL_ERR_NOT_FOUND;
EOF
$ patch -p1 < openmpi.patch
$ make
$ sudo make install

```

A device configuration section must also be added using the following commands:

```

$ sudo cat >> /usr/local/share/openmpi/mca-btl-openib-device-params.ini << EOF
[Broadcom Netxtreme]
vendor_id = 0x14e4
vendor_part_id =
5637,5638,5652,5824,5825,5838,5839,5846,5847,5848,5849,5855,5858,5859,5861,5867,5869,5871,5872,5873
use_eager_rdma = 1
mtu = 1024
receive_queues = P,128,256,192,128:S,65536,256,192,128
max_inline_data = 96
EOF

```

### 14.2.5.11 Performance

NCC is a comprehensive solution used for congestion management that can help reduce packet losses and congestion spreading as well as improve latency by keeping switch queue levels low.

Congestion control performance is measured by these network traffic performance metrics during periods of congestion:

- Fairness of bandwidth allocation between qp
- Link utilization
- Latency

Under heavy congestion, NCC can enforce fairness at a per-QP level, with low variation between streams. Streams not crossing the point of network contention are not affected.

Performance may vary depending on the topology, number of flows, traffic types, and the application used. Careful consideration and understanding should be used when adjusting the different NCC parameters. As a default, deterministic marking with DCTCP is recommended and can be configured using the setup script provided.

#### 14.2.5.11.1 Priority Flow Control

Although Priority Flow Control (PFC) is not required, it is recommended to enable PFC to ensure that packets are not dropped in bursty or transient scenarios. In cases where PFC is not being utilized, adjustments of the cc tunables are required to ensure the ramp up and ramp down is appropriate based on the user's scenario.

### 14.2.5.11.2 CNP Traffic Classes

It is recommended that CNPs are classified onto a different traffic class on both the NIC and the switch for optimal performance to ensure CNP are not utilizing the same buffer as the RoCE packets.

### 14.2.5.11.3 Disable CPU Power Saving

Intel CPU power saving technology can cause packet RoCE drops with bursty traffic. It is highly recommended to set BIOS power saving mode to maximum performance, and disable c-states and p-states with kernel options by editing the `/etc/default/grub` file, and appending the following to `GRUB_CMDLINE_LINUX_DEFAULT`:

```
intel_pstate=disable processor.max_cstate=1 intel_idle.max_cstate=0
```

Rebuild the grub configuration using the following command:

```
sudo /usr/sbin/grub-mkconfig* -o /boot/grub/grub.cfg
```

### 14.2.5.12 Limitations

In dual-port NICs, if both ports are on same subnet, RDMA perfest commands may fail. The possible cause is due to an arp flux issue in the Linux OS. To work around this limitation, use multiple subnets for testing or bring the second port/interface down.

### 14.2.5.13 Known Issues

`Bnxt_en` and `Bnxt_re` are designed to function as a pair. Older `Bnxt_en` drivers prior to version 1.7.x do not support RDMA and cannot be loaded at the same time as the `Bnxt_re` (RDMA) driver. The user may experience a system crash and reboot if `Bnxt_re` is loaded with older `Bnxt_en` drivers. It is recommend that the user load the `Bnxt_en` and `Bnxt_re` module from the same `netxtreme-bnxt_en-<1.7.x>.tar.gz` bundle.

To prevent mismatching a combination of `bnxt_en` and `bnxt_re` from being loaded, the following is required:

- If RedHat/CentOS 7.2 OS was installed to the target system using PXEboot with `bnxt_en` DUD or a kernel module RPM, delete the file `bnxt_en.ko` found in `/lib/modules/$(uname -r)/extra/bnxt_en/bnxt_en.ko` or edit `/etc/depmod.d/`.
- `bnxt_en.conf` to override to use updated version. Users can also erase the current Broadcom Ethernet Network adapter Linux kernel driver using the `rpm -e kmod-bnxt_en` command. RHEL 7.3/SLES 12 Sp2 has `bnxt_en` inbox driver (older than v1.7.x). This driver must be removed and the latest `bnxt_en` be added before applying the `bnxt_re` (RoCE drivers).

## 14.3 Windows and Use Case Examples

### 14.3.1 SMB Direct

Windows Server 2012 includes a new feature called SMB Multichannel, part of the SMB 3.0 protocol, which increases the network performance and availability for file servers.

SMB Multichannel allows file servers to use multiple network connections simultaneously and provides the following capabilities:

- Increased Throughput – The file server can simultaneously transmit more data using multiple connections for high speed network adapters or multiple network adapters.
- Network Fault Tolerance – When using multiple network connections at the same time, the clients can continue to work uninterrupted despite the loss of a network connection.
- Automatic Configuration – SMB Multichannel automatically discovers the existence of multiple available network paths and dynamically adds connections as required.

SMB Multichannel is the feature responsible for detecting the RDMA capabilities of NICs to enable the SMB Direct feature (SMB over RDMA). Without SMB Multichannel, SMB uses regular TCP/IP with these RDMA-capable NICs ( all provide a TCP/IP stack side-by-side with the new RDMA stack).

With SMB Multichannel, SMB detects the RDMA capability and create multiple RDMA connections for that single session (two per interface). This allows SMB to use the high throughput, low latency, and low CPU utilization offered by these RDMA NICs. It also offers fault tolerance if multiple RDMA interfaces are used.

### 14.3.1.1 Kernel Mode

Windows Server 2012 and beyond invokes the RDMA capability in the NIC for SMB file traffic if both ends are enabled for RDMA. Broadcom NDIS miniport bnxnd.sys v20.6.2 and beyond support RoCEv1 and RoCEv2 via the NDKPI interface. The default setting is RoCEv1.

To enable RDMA:

1. The goal is to exercise as many RDMA connections per network interface as possible using a simple server to client relationship. In order to achieve that, following registry key needs to be added and set its value to 16:
  - a. HKLM\System\CurrentControlSet\Services\LanmanWorkstation\Parameters\ConnectionCountPerRdmaNetworkInterface
  - b. The default is 2; with a valid range from 1 to 16.
  - c. This allows 16 RDMA connections per Network share. In order to scale up from here with a single interface, additional alias IP address must be created with different subnets.
2. Upgrade the NIC NVRAM using the appropriate board packages. In the UEFI HII, enable support for RDMA.
3. Go to the adapter Advanced Properties page and set NetworkDirect Functionality to Enabled for each Broadcom Ethernet Network adapter miniport, or using PowerShell window, run the following command:

```
Set-NetAdapterAdvancedProperty -RegistryKeyword *NetworkDirect -RegistryValue 1
```

4. The following Powershell commands return true if NetworkDirect is enabled.

```
Get-NetOffloadGlobalSetting  
Get-NetAdapterRDMA
```

### 14.3.1.2 User Mode

Before running a user mode application written to NDSPI, copy and install the bxndspi.dll user mode driver. To copy and install the user mode driver:

1. Copy bxndspi.dll to C:\Windows\System32.
2. Install the driver by running the following command:

```
rundll32.exe .\bxndspi.dll,Config install|more
```

### 14.3.1.3 Verifying that NIC is Enabled for RDMA

To verify that the NIC is enabled for RDMA:

1. Verify that NetworkDirect Functionality is enabled for each interface under the **Advance Tab**.
2. Using PowerShell, issue the following commands and ensure it returns True:



```
Get-NetAdapterRDMA
```

#### 14.3.1.4 Verifying SMB Multichannel is Enabled

To verify that SMB multichannel is enabled:

1. Issue the following command and ensure Client RDMA Capable is True:

```
Get-SmbMultiChannelConnection
```

2. SMB multichannel is enabled by defaults. To enable or disable SMB multichannel, use the following commands:

##### Server Side:

- Enable – `Set-SmbServerConfiguration -EnableMultiChannel $true`
- Disable – `Set-SmbServerConfiguration -EnableMultiChannel $false`

##### Client Side:

- Enable – `Set-SmbClientConfiguration -EnableMultiChannel $true`
- Disable – `Set-SmbClientConfiguration -EnableMultiChannel $false`

#### 14.3.1.5 Setting Up the System

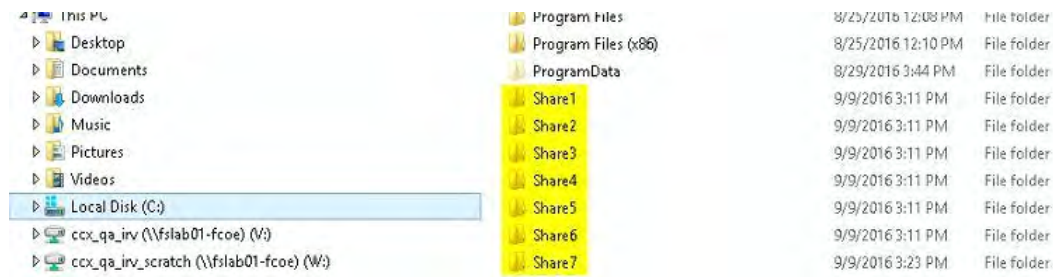
To setup the system:

1. Configure the server by creating seven or more IP address with different subnets. For example:
  - 172.1.55.1
  - 172.2.55.2
  - 172.3.55.3
  - 172.4.55.4
  - 172.5.55.5
  - 172.6.55.6
  - 172.7.55.7

Figure 52: Server/Target Configuration

```
Ethernet adapter Port1:
Connection-specific DNS Suffix . :
Link-local IPv6 Address . . . . . : fe80::25f8:305c:563f:acf3%21
IPv4 Address. . . . . : 172.1.55.1
Subnet Mask . . . . . : 255.255.0.0
IPv4 Address. . . . . : 172.2.55.2
Subnet Mask . . . . . : 255.255.0.0
IPv4 Address. . . . . : 172.3.55.3
Subnet Mask . . . . . : 255.255.0.0
IPv4 Address. . . . . : 172.4.55.4
Subnet Mask . . . . . : 255.255.0.0
IPv4 Address. . . . . : 172.5.55.5
Subnet Mask . . . . . : 255.255.0.0
IPv4 Address. . . . . : 172.6.55.6
Subnet Mask . . . . . : 255.255.0.0
IPv4 Address. . . . . : 172.7.55.7
Subnet Mask . . . . . : 255.255.0.0
IPv4 Address. . . . . : 172.8.55.8
Subnet Mask . . . . . : 255.255.0.0
```

2. Create seven folders named Share1 to Share7 and individually share them.

**Figure 53: Shared Folders**

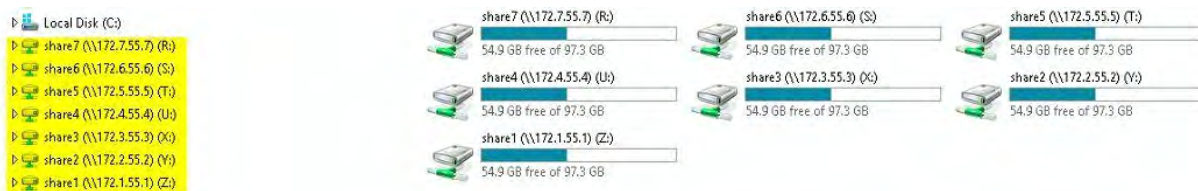
3. Configure the client by creating seven or more IP Address with different subnets that are accessible by the server system.

For example:

- 172.1.54.1
- 172.2.54.2
- 172.3.54.3
- 172.4.54.4
- 172.5.54.5
- 172.6.54.6
- 172.7.54.7

4. Map a network drive to the server as follows:

- Drive letter: \\172.1.55.1\Share1
- Drive letter: \\172.2.55.2\Share2
- Drive letter: \\172.3.55.3\Share3
- Drive letter: \\172.4.55.4\Share4
- Drive letter: \\172.5.55.5\Share5
- Drive letter: \\172.6.55.6\Share6
- Drive letter: \\172.7.55.7\Share7

**Figure 54: Drive Mapping**

Each of the network drives generate 16 RDMA connections for a total of 112 with seven drives.

### 14.3.1.6 Traffic Generation

To exercise all 16 RDMA connections simultaneously using a different traffic generation tool:

- If MLTT (License Software) is being used:
  - Change the number of thread counts from 1 to 16. This ensures that all 16 channels have traffic running.
- If Diskspd (freeware and successor to SQLIO) is being used:
  - Use a combination of the following two parameters which produce a product of 16, for example:

-o2 -t8, -o4 -t4, or -o1 -t16

-o: Outstanding I/O request. This is your queue depth.  
-t: Number of threads per file.

### 14.3.1.7 Verifying RDMA

To verify RDMA:

Create a file share on the remote system and open that share using Windows Explorer. To avoid a hard disk read/write speed bottleneck, a RAM disk is recommended as the network share under test.

1. From PowerShell, run the following commands:

```
Get-SmbMultichannelConnection | fl *RDMA*  
ClientRdmaCapable : True  
ServerRdmaCapable : True
```

2. If both client and server show True, then any file transfers over this SMB connection use SMB.
3. The following commands can be used to enable/disable SMB Multichannel:

**Server Side:**

- Enable – Set-SmbServerConfiguration -EnableMultiChannel \$true
- Disable – Set-SmbServerConfiguration -EnableMultiChannel \$false

**Client Side:**

- Enable – Set-SmbClientConfiguration -EnableMultiChannel \$true
- Disable – Set-SmbClientConfiguration -EnableMultiChannel \$false

By default, the driver sets up two RDMA connections for each network share per IP address (on a unique subnet). Scale up the number of RDMA connections by adding multiple IP addresses, each with different a subnet, for the same physical port under test. Multiple network shares can be created and mapped to each link partner using the unique IP addresses created.

**Example:**

- On Server 1, create the following IP addresses for Network Port1:
  - 172.1.10.1
  - 172.2.10.2
  - 172.3.10.3
- On the same Server 1, create three shares:
  - Share1
  - Share2
  - Share3
- On the network link partners:
  - Connect to \\172.1.10.1\share1
  - Connect to \\172.2.10.2\share2
  - Connect to \\172.3.10.3\share3
  - And so forth.

### 14.3.1.8 Verification

#### 14.3.1.8.1 How to Verify RDMA Connections

Verification can be performed using either Performance Monitor or Power Shell.

#### 14.3.1.8.2 Perfmon

Using Perfmon, the total number of active RDMA connections can be verified using the RDMAActivity counter. Verification for which of the 16 channels are being exercised can also be done by using the SMB direct connection counters.

**NOTE:** Ensure that traffic is running in order to see the counters.

Figure 55: Perfmon for Verification (Part 1)

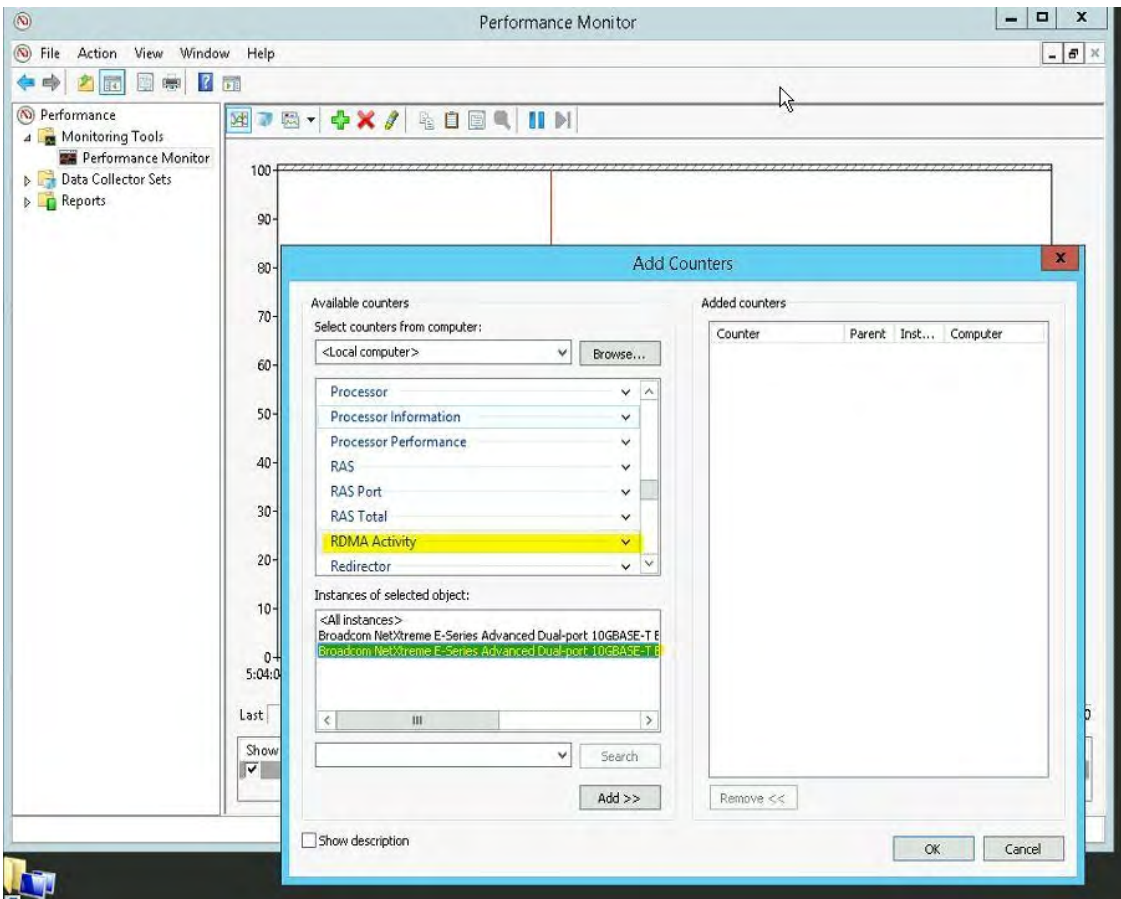


Figure 56: Perfmon for Verification (Part 2)

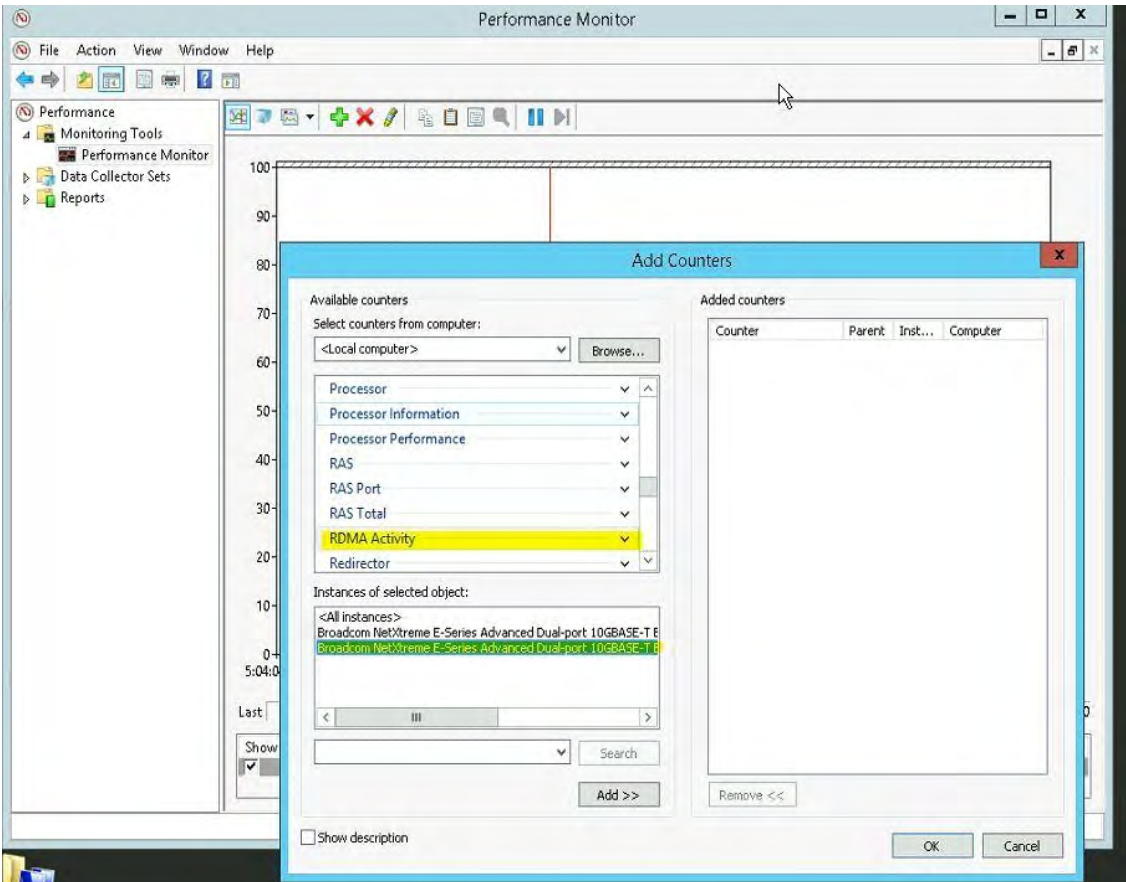
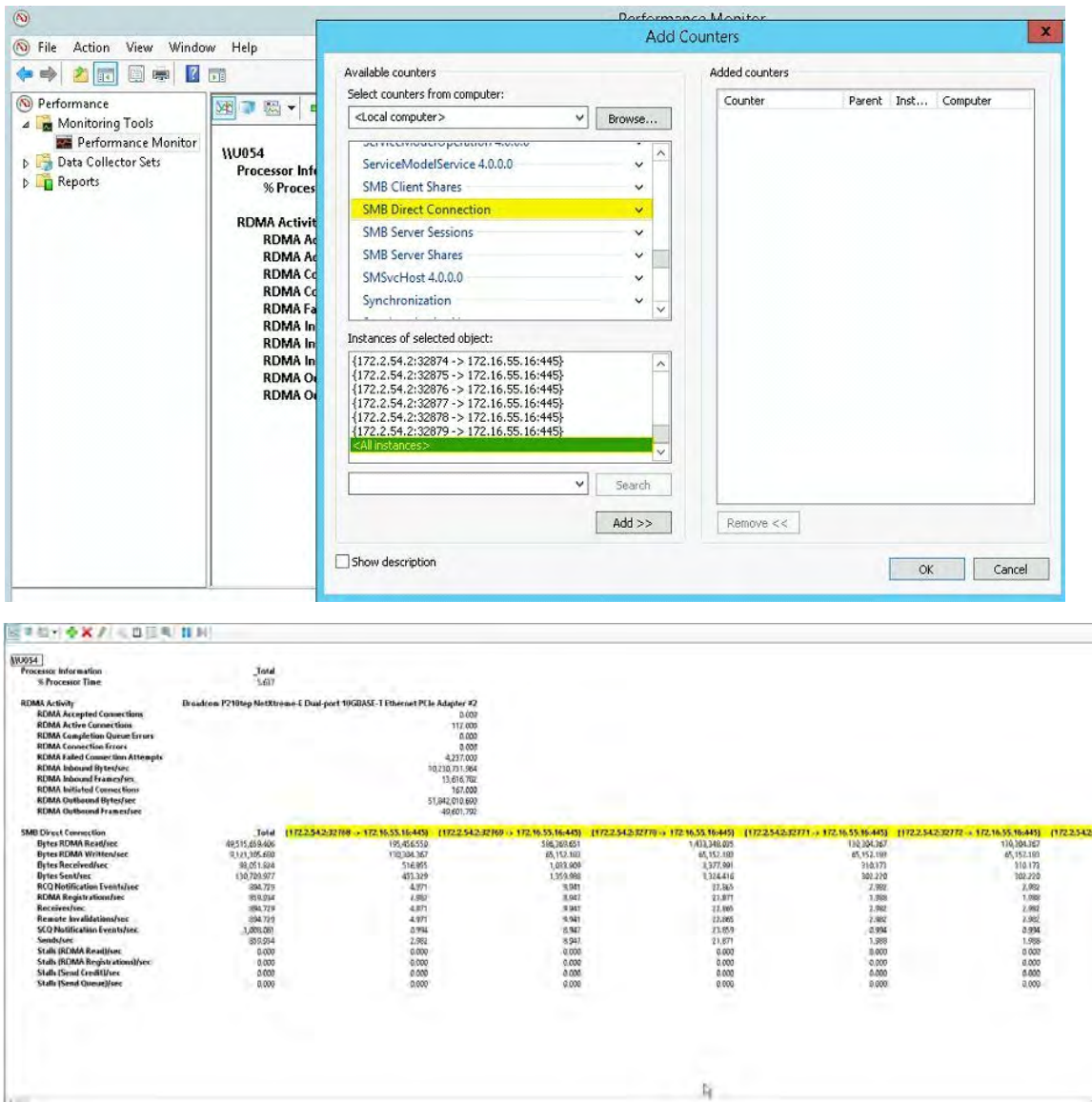


Figure 57: Perfmon for Verification (Part 3)

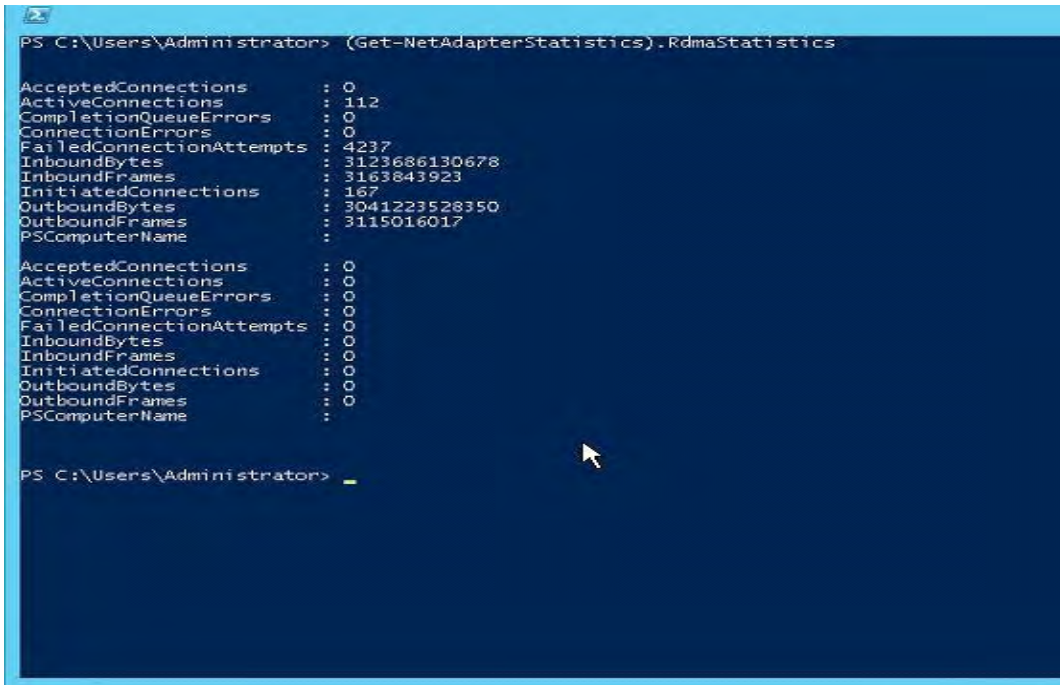


### 14.3.1.8.3 Powershell

PowerShell can be used for verification. Use the following commands at the Powershell prompt. There are connections listed if using RDMA. Otherwise, it defaults to TCP.

- Netstat -xan – Shows RDMA Connections
- Netstat -an – Shows TCP Connections
- (Get-NetAdapterStatistics). RDMAStatistics – Lists RDMA Activity Connections

Figure 58: Powershell for Verification



```

PS C:\Users\Administrator> (Get-NetAdapterStatistics).RdmaStatistics
AcceptedConnections      : 0
ActiveConnections       : 112
CompletionQueueErrors    : 0
ConnectionErrors        : 0
FailedConnectionAttempts : 4237
InboundBytes             : 3123686130678
InboundFrames           : 3163843923
InitiatedConnections    : 167
OutboundBytes           : 3041223528350
OutboundFrames          : 3115016017
PSComputerName          :
AcceptedConnections      : 0
ActiveConnections       : 0
CompletionQueueErrors    : 0
ConnectionErrors        : 0
FailedConnectionAttempts : 0
InboundBytes            : 0
InboundFrames           : 0
InitiatedConnections    : 0
OutboundBytes           : 0
OutboundFrames          : 0
PSComputerName          :
PS C:\Users\Administrator> _

```

## 14.4 VMware ESX and Use Case Examples

### 14.4.1 Limitations

The current version of the RoCE supported driver requires ESXi-6.5.0 GA build 4564106 or above.

### 14.4.2 BNXT RoCE Driver Requirements

The BNXTNET L2 driver must be installed with the `disable_roce=0` module parameter before installing the driver.

To set the module parameter, use the following command:

```
esxcfg-module -s "disable_roce=0" bnxtnet
```

Use ESX6.5 L2 driver version 20.6.9.0 (RoCE supported L2 driver) or above.

### 14.4.3 Installation

To install the RoCE driver:

1. Copy the `<bnxtroce>-<driver version>.vib` file in `/var/log/vmware` using the following commands:

```
$ cd /var/log/vmware
$ esxcli software vib install --no-sig-check -v <bnxtroce>-<driver version>.vib
```

2. Reboot the machine.
3. Verify that the drivers are correctly installed using the following command:

```
esxcli software vib list | grep bnxtroce
```



To disable ECN (enabled by default) for RoCE traffic use the `tos_ecn=0` module parameter for `bnxtroce`.

## 14.4.4 Configuring Paravirtualized RDMA Network Adapters

See [Vmware.com](http://Vmware.com) for additional information on setting up and using Paravirtualized RDMA (PVRDMA) network adapters.

### 14.4.4.1 Configuring a Virtual Center for PVRDMA

To configure a Virtual Center for PVRDMA:

1. Create DVS (requires a Distributed Virtual Switch for PVRDMA).
2. Add the host to the DVS.

### 14.4.4.2 Tagging vmknic for PVRDMA on ESX Hosts

To tag a vmknic for PVRDMA to use on ESX hosts:

1. Select the host and right-click on **Settings** to switch to the settings page of the **Manage** tabs.
2. In the **Settings** page, expand **System** and click **Advanced System Settings** to show the **Advanced System Settings** key-pair value and its summary.
3. Click **Edit** to bring up the **Edit Advanced System Settings**.  
Filter on **PVRDMA** to narrow all the settings to just `Net.PVRDMAvmknic`.
4. Set the `Net.PVRDMAvmknic` value to `vmknic`, as in example `vmk0`.

### 14.4.4.3 Setting the Firewall Rule for PVRDMA

To set the firewall rule for PVRDMA:

1. Select the host and right-click on **Settings** to switch to the settings page of the **Manage** tabs.
2. In the **Settings** page, expand **System** and click **Security Profile** to show the firewall summary.
3. Click **Edit** to bring up the **Edit Security Profile**.
4. Scroll down to find `pvrDMA` and check the box to set the firewall.

### 14.4.4.4 Adding a PVRDMA Device to the VM

To add a PVRDMA device to the VM:

1. Select the VM and right-click on **Edit Settings**.
2. Add a new Network Adapter.
3. Select the network as a **Distributed Virtual Switch** and **Port Group**.
4. For the **Adapter Type**, select **PVRDMA** and click **OK**.

## 14.4.5 Configuring the VM on Linux Guest OS

**NOTE:** The user must install the appropriate development tools including git before proceeding with the configuration steps below.

1. Download the PVRDMA driver and library using the following commands:

```
git clone git://git.openfabrics.org/~aditr/pvrdma_driver.git
git clone git://git.openfabrics.org/~aditr/libpvrdma.git
```

2. Compile and install the PVRDMA guest driver and library.
3. To install the driver, execute `make && sudo insmod pvrdma.ko` in the directory of the driver.  
The driver must be loaded after the paired vmxnet3 driver is loaded.

**NOTE:** The installed RDMA kernel modules may not be compatible with the PVRDMA driver. If so, remove the current installation and restart. Then follow the installation instructions. See the README in the driver's directory for more information about the different RDMA stacks.

4. To install the library, execute `./autogen.sh && ./configure --sysconfdir=/etc && make && sudo make install` in the directory of the library.

**NOTE:** The installation path of the library needs to be in the shared library cache. Follow the instructions in the INSTALL file in the library's directory.

**NOTE:** The firewall settings may need to be modified to allow RDMA traffic. Ensure the proper firewall settings are in place.

5. Add the `/usr/lib` in the `/etc/ld.so.conf` file and reload the `ldconf` by running `ldconfig`
6. Load IB modules using `modprobe rdma_ucm`.
7. Load the PVRDMA kernel module using `insmod pvrdma.ko`.
8. Assign an IP address to the PVRDMA interface.
9. Verify whether the IB device is created by running the `ibv_devinfo -v` command.

## 15 DCBX – Data Center Bridging

Broadcom Ethernet Network adapters support IEEE 802.1Qaz DCBX as well as the older CEE DCBX specification. DCB configuration is obtained by exchanging the locally configured settings with the link peer. Since the two ends of a link may be configured differently, DCBX uses a concept of *willing* to indicate which end of the link is ready to accept parameters from the other end. This is indicated in the DCBX protocol using a single bit in the ETS Configuration and PFC TLV, this bit is not used with ETS Recommendation and Application Priority TLV. By Default, the Broadcom Ethernet Network adapter is in *willing* mode while the link partner network switch is in *non-willing* mode. This ensures the same DCBX setting on the Switch propagates to the entire network.

Users can manually set Broadcom Ethernet Network adapters to *non-willing* mode and perform various PFC, Strict Priority, ETS, and APP configurations from the host side. See the driver readme.txt for additional details on available configurations. This section provides an example of how such a setting can be done in Windows with Windows PowerShell. Additional information on DCBX, QoS, and associated use cases are described in additional details in a separate white paper, beyond the scope of this user manual.

The following settings in the UEFI HII menu are required to enable DCBX support:

**System Setup**→**Device Settings**→**NetXtreme-E NIC**→**Device Configuration Menu**

### 15.1 QoS Profile – Default QoS Queue Profile

Quality of Server (QoS) resources configuration is necessary to support various PFC and ETS requirements where finer tuning beyond bandwidth allocation is needed. Broadcom Ethernet Network adapters allow the administrator to select between devoting NIC hardware resources to support Jumbo Frames and/or combinations of lossy and lossless Class of Service queues (CoS queues). Many combinations of configuration are possible and therefore can be complicated to compute. This option allows a user to select from a list of precomputed QoS Queue Profiles. These precomputed profiles are design to optimize support for PFC and ETS requirements in typical customer deployments.

The following is a summary description for each QoS Profile.

**Table 48: QoS Profiles**

Profile No.	Jumbo Frame Support	No. of Lossy CoS Queues/Port	No. of Lossless CoS Queues/Port	Support for 2-Port SKU
<b>Profile #1</b>	Yes	0	1 (PFC Supported)	Yes (25 Gb/s)
<b>Profile #2</b>	Yes	4	2 (PFC Supported)	No
<b>Profile #3</b>	No. (MTU <= 2 KB)	6	2 (PFC Supported)	Yes (25 Gb/s)
<b>Profile #4</b>	Yes	1	2 (PFC Supported)	Yes (25 Gb/s)
<b>Profile #5</b>	Yes	1	0 (No PFC Support)	Yes (25 Gb/s)
<b>Profile #6</b>	Yes	8	0 (No PFC Support)	Yes (25 Gb/s)
<b>Profile #7</b>	This configuration maximizes packet-buffer allocations to two lossless CoS Queues to maximize RoCE performance while trading off flexibility.			
	Yes	0	2	Yes (25 Gb/s)
<b>Default</b>	Yes	Same as Profile #4		Yes

## 15.2 DCBX Mode – Enable (IEEE only)

This option allows a user to enable/disable DCBX with the indicated specification. IEEE only indicates that IEEE 802.1Qaz DCBX is selected.

Windows Driver setting:

After enabling the indicated options in the UEFI HII menu to set firmware level settings, perform the follow selection in the Windows driver advanced properties.

Open **Windows Manager** → **Broadcom NetXtreme E Series adapter** → **Advanced Properties** → **Advanced tab**

Quality of Service = Enabled

Priority & VLAN = Priority& VLAN enabled

VLAN = <ID>

Set desired VLAN id

To exercise the DCB related command in Windows PowerShell, install the appropriate DCB Windows feature.

1. In the **Task Bar**, right-click the Windows PowerShell icon and then click **Run as Administrator**. Windows PowerShell opens in elevated mode.
2. In the Windows PowerShell console, type:

```
Install-WindowsFeature "data-center-bridging"
```

## 15.3 DCBX Willing Bit

The DCBX willing bit is specified in the DCB specification. If the willing bit on a device is true, the device is willing to accept configurations from a remote device through DCBX. If the willing bit on a device is false, the device rejects any configuration from a remote device and enforces only the local configurations.

Use the following to set the willing bit to True or False. 1 for enabled, 0 for disabled.

**Example** `set-netQoSdcbxSetting -Willing 1`

Use the following to create a Traffic Class.

```
C:\> New-NetQoSTrafficClass -name "SMB class" -priority 4 -bandwidthPercentage 30 -Algorithm ETS
```

**NOTE:** By default, all IEEE 802.1p values are mapped to a default traffic class, which has 100% of the bandwidth of the physical link. The command shown above creates a new traffic class to which any packet tagged with eight IEEE 802.1p value 4 is mapped, and its Transmission Selection Algorithm (TSA) is ETS and has 30% of the bandwidth. It is possible to create up to seven new traffic classes. In addition to the default traffic class, there is at most eight traffic classes in the system.

Use the following in displaying the created Traffic Class:

```
C:\> Get-NetQoSSTrafficClass
Name           Algorithm Bandwidth(%) Priority
-----
[Default]      ETS       70          0-3,5-7
SMB class      ETS       30          4
```

Use the following in modifying the Traffic Class:

```
PS C:\> Set-NetQoSTrafficClass -Name "SMB class" -BandwidthPercentage 40
PS C:\> get-NetQoSTrafficClass
Name Algorithm Bandwidth(%) Priority
-----
[Default] ETS          60 0-3,5-7
SMB class ETS          40          4
```

Use the following to remove the Traffic Class:

```
PS C:\> Remove-NetQoSTrafficClass -Name "SMB class"
PS C:\> Get-NetQoSTrafficClass
Name Algorithm Bandwidth(%) Priority
-----
[Default] ETS          100          0-7
```

Use the following to create Traffic Class (Strict Priority):

```
C:\> New-NetQoSTrafficClass -name "SMB class" -priority 4 -bandwidthPercentage 30-Algorithm Strict
```

Enabling PFC:

```
PS C:\> Enable-NetQoSFlowControl -priority 4
PS C:\> Get-NetQoSFlowControl -priority 4
Priority Enabled
-----
4 True

PS C:\> Get-NetQoSFlowControl
```

Disabling PFC:

```
PS C:\> disable-NetQoSflowControl -priority 4
PS C:\> get-NetQoSFlowControl -priority 4
Priority Enabled
-----
4 False
```

Use the following to create QoS Policy:

```
PS C:\> New-NetQoSPolicy -Name "SMB policy" -SMB -PriorityValue8021Action 4

Name : SMB policy
Owner : Group Policy (Machine)
NetworkProfile : All
Precedence : 127
```

**NOTE:** The previous command creates a new policy for SMB. SMB is an inbox filter that matches TCP port 445 (reserved for SMB). If a packet is sent to TCP port 445 it is tagged by the operating system with IEEE 802.1p value of 4 before the packet is passed to a network miniport driver. In addition to SMB, other default filters include iSCSI (matching TCP port 3260), NFS (matching TCP port 2049), LiveMigration (matching TCP port 6600), FCOE (matching EtherType 0x8906) and NetworkDirect. NetworkDirect is an abstract layer created on top of any RDMA implementation on a network adapter. NetworkDirect must be followed by a Network Direct port. In addition to the default filters, a user can classify traffic by application's executable name (as in the first example below), or by IP address, port, or protocol.

Use the following to create QoS Policy based on the Source/Destination Address:

```
PS C:\> New-NetQosPolicy "Network Management" -IPDstPrefixMatchCondition 10.240.1.0/24 -
IPProtocolMatchCondition both -NetworkProfile all -PriorityValue8021Action 7
Name : Network Management
Owner : Group Policy (Machine)
Network Profile : All
Precedence : 127
IPProtocol : Both
IPDstPrefix : 10.240.1.0/24
PriorityValue : 7
```

Use the following to display QoS Policy:

```
PS C:\> Get-NetQosPolicy
Name : Network Management
Owner : (382ACFAD-1E73-46BD-A0A-6-4EE0E587B95)
NetworkProfile : All
Precedence : 127
IPProtocol : Both
IPDstPrefix : 10.240.1.0/24
PriorityValue : 7
Name : SMB policy
Owner : (382AFAD-1E73-46BD-A0A-6-4EE0E587B95)
NetworkProfile : All
Precedence : 127
Template : SMB
PriorityValue : 4
```

Use the following to modify the QoS Policy:

```
PS C:\> Set-NetqosPolicy -Name "Network Management" -IPSrcPrefixMatchCondition 10.235.2.0/24 -
IPProtocolMatchCondition both -PriorityValue802.1Action 7
PS C:\> Get-NetQosPolicy -name "network management"
Name : Network Management
Owner : {382ACFD-1E73-46BD-A0A0-4EE0E587B95}
NetworkProfile : All
Precedence : 127
IPProtocol : Both
IPSrcPrefix : 10.235.2.0/24
IPDstPrefix : 10.240.1.0/24
PriorityValue : 7
```

Use the following to remove QoS Policy:

```
PS C:\> Remove-NetQosPolicy -Name "Network Management"
```

## 16 DPDK – Configuration and Use Case Examples

The testpmd application can be used to test the DPDK in a packet forwarding mode and also to access NIC hardware features such as Flow Director. It also serves as an example of how to build a more fully-featured application using the DPDK SDK. The below chapters shows how to build and run the testpmd application and how to configure the application from the command line and the run-time environment.

### 16.1 Compiling the Application

To compile the application:

1. Set the required environmental variables and go to the source code with the following commands:

```
cd /Linux/Linux_DPDK
tar zxvf dpdk-*.gz
cd dpdk-*
make config T=x86_64-native-linuxapp-gcc && make
```

2. Allocate system resources and attach the UIO module with the following commands:

```
mkdir -p /mnt/huge
mount -t hugetlbfs nodev /mnt/huge
echo 2048 > /sys/devices/system/node/node0/hugepages/hugepages-2048kB/nr_hugepages
echo 2048 > /sys/devices/system/node/node1/hugepages/hugepages-2048kB/nr_hugepages
modprobe uio
insmod ./build/kmod/igb_uio.ko
```

3. Bind the device with the following command:

```
usertools/dpdk-devbind.py -b igb_uio <PCI Device ID>
```

**Example:** ./usertools/dpdk-devbind.py -b igb\_uio 3b:00.0

4. The PCI device information is displayed by lspci with the following commands:

```
[root@localhost /]# lspci |grep Eth
3b:00.0 Ethernet controller: Broadcom Limited BCM57414 NetXtreme-E 10Gb/25Gb RDMA Ethernet
Controller (rev 01)
3b:00.1 Ethernet controller: Broadcom Limited BCM57414 NetXtreme-E 10Gb/25Gb RDMA Ethernet
Controller (rev 01)
```

### 16.2 Running the Application

To run the application, execute the following command:

```
build/app/testpmd -l 0,1,2,3,4,5,6,7,8 -- -i --nb-ports=1 --nb-cores=8 --txq=2 --rxq=2
```

Options for this example:

- -l – CORELIST List of cores to run.
- -i – Interactive Run testpmd in interactive mode.
- --nb-cores=N – Sets the number of forwarding cores.
- --nb-ports=N – Sets the number of forwarding ports.
- --rxq=N – Sets the number of RX queues per port to N.
- --txq=N – Sets the number of TX queues per port to N.

## 16.3 Testpmd Runtime Functions

When the testpmd application is started in interactive mode, (-i|--interactive), it displays a prompt that can be used to start and stop forwarding, configure the application, display statistics (including the extended NIC statistics aka xstats), set the Flow Director, and other tasks.

```
testpmd>
```

## 16.4 Control Functions

This section contains the control functions.

### start

Starts packet forwarding with current configuration:

```
testpmd> start
```

### start tx\_first

Starts packet forwarding with current configuration after sending specified number of bursts of packets:

```
testpmd> start tx_first (""|burst_num)
```

The default burst number is 1 when burst\_num not presented.

### stop

Stops packet forwarding and display accumulated statistics:

```
testpmd> stop
```

### quit

Quits to prompt:

```
testpmd> quit
```

## 16.5 Display Functions

The functions in this section are used to display information about the testpmd configuration or the NIC status.

- show port – Displays information for a given port or all ports.
- show port rss reta – Displays the RSS redirection table entry indicated by masks on port X.
- show port rss-hash – Displays the RSS hash functions and RSS hash key of a port.
- show (rxq|txq) – Displays information for a given port's RX/TX queue.
- show config – Displays the configuration of the application. The configuration comes from the command-line.
- set fwd – Sets the packet forwarding mode.
- read rxd – Displays an RX descriptor for a port RX queue.
- read txd – Displays a TX descriptor for a port TX queue.



## 16.6 Configuration Functions

The testpmd application can be configured from the runtime as well as from the command line. This section describes the available configuration functions that are available.

**csum set** – Selects the hardware or software calculation of the checksum when transmitting a packet using the csum forwarding engine: `testpmd> csum set (ip|udp|tcp|sctp|outer-ip) (hw|sw) (port_id).`

## Revision History

### NetXtreme-E-UG312-2CS; June 22, 2023

**Updated:**

- [Table 18, LED Functions](#) – added 200 Gb/s support.
- [Table 21, Host Interface Features](#) – updated PTP description.
- [Table 26, Validated Cables and Modules](#)
- [SFP28](#) – Updated description.

**Added:**

- [Configuring 200G Link Speeds](#)

### NetXtreme-E-UG311-2CS; June 1, 2023

**Updated:**

- Removed CCM support.

**Added:**

- Advanced NPAR

### NetXtreme-E-UG310-2CS; November 15, 2022

**Updated:**

- Limitations – Added note on NPAR configuration support.

### NetXtreme-E-UG309-2CS; August 31, 2022

**Updated:**

- Limitations – Added note on NPAR configuration support.

### NetXtreme-E-UG308-2CS; February 1, 2022

**Updated:**

- Device Configuration Menu – Updated number of MS-X Vectors per PF section.

### NetXtreme-E-UG307-2CS; November 2, 2021

**Updated:**

- Device Configuration Menu

### NetXtreme-E-UG306-2CS; July 30, 2021

**Updated:**

- [Table 29, Validated Cables and Modules](#)

## **NetXtreme-E-UG305-2CS; June 10, 2021**

### **Updated:**

- Supported Operating Systems
- Device Configuration Menu

## **NetXtreme-E-UG304-2CS; March 23, 2021**

### **Updated:**

- Figure 45, Device Configuration Menu

## **NetXtreme-E-UG303-2CS; February 10, 2021**

### **Updated:**

- Supported Operating Systems
- Added BCM957504-NGM250 to the functional description table.
- System-Level Configuration
- Auto-Negotiation Configuration

### **Added:**

- VMWare Enhanced Networking Stack (ENS)

## **NetXtreme-E-UG302-2CS; July 28, 2020**

### **Updated:**

- Supported Operating Systems

## **NetXtreme-E-UG301-2CS; June 8, 2020**

### **Updated:**

- Supported Operating Systems
- Host Interface Features
- Hardware Requirements
- Installing the VMware Driver
- Limitations

## **NetXtreme-E-UG300-2CS; February 20, 2020**

Initial release

