

# NVIDIA Mellanox NICs Performance Report with DPDK 22.03

Rev 1.2

#### Notice

This document is provided for information purposes only and shall not be regarded as a warranty of a certain functionality, **o**ndition, or quality of a product. NVIDIA Corporation ("NVIDIA") makes no representations or warranties, expressed or implied, as to the accuracy or completeness of the information contained in this document and assumes no responsibility for any errors contained herein. NVIDIA shall have no liability for the consequences or use of such information or for any infringement of patents or other rights of third parties that may result from its use. This document is not a commitment develop, release, or deliver any Material (defined below), code, or functionality.

NVIDIA reserves the right to make corrections, modifications, enhancements, improvements, and any other changes to this document, at any time without notice.

Customer should obtain the latest relevant information before placing orders and should verify that such information is current and complete.

NVIDIA products are sold subject to the NVIDIA standard terms and conditions of sale supplied at the time of order acknowledgement, unless otherwise agreed in an individual sales agreement signed by authorized representatives of NVIDIA and customer ("Terms of Sale"). NVIDIA hereby expressly objects to applying any customer general terms and conditions with regards to the purchase of the NVIDIA product referenced in this document. No contactual obligations are formed either directly or indirectly by this document.

NVIDIA products are not designed, authorized, or warranted to be suitable for use in medical, military, aircraft, space, or Ife support equipment, nor in applications where failure or malfunction of the NVIDIA product can reasonably be expected to result in personal injury, deah, or property or environmental damage. NVIDIA accepts no liability for inclusion and/or use of NVIDIA products in such equipment or applications and therefore such inclusion and/or use is at customer's own risk.

NVIDIA makes no representation or warranty that products based on this document will be suitable for any specified use. Testing of all parameters of each product is not necessarily performed by NVIDIA. It is customer's sole responsibility to evaluate and determine the applicability of any information contained in this document, ensure the product is suitable and fit for the application planned by customer, and perform the necessary testing for the application in order to avoid a default of the application or the product. Weaknesses in customer's product designs may affect the quality and reliability of the NVIDIA product and may result in additional or different conditions and/or requirements beyond those contained in this document. NVIDIA accepts no liability elated to any default, damage, costs, or problem which may be based on or attributable to: (i) the use of the NVIDIA productin any manner that is contrary to this document or (ii) customer product designs.

No license, either expressed or implied, is granted under any NVIDIA patent right, copyright, or other NVIDIA intellectual poperty right under this document. Information published by NVIDIA regarding third-party products or services does not constitute a license from NVIDIA to use such products or services or a warranty or endorsement thereof. Use of such information may require a license from a third party under the patents or other intellectual property rights of the third party, or a license from NVIDIA under the patents or other intellectual property rights of NVIDIA.

Reproduction of information in this document is permissible only if approved in advance by NVIDIA in writing, reproduced without alteration and in full compliance with all applicable export laws and regulations, and accompanied by all associated conditions, limitations, and notices.

#### Trademarks

NVIDIA, the NVIDIA logo, and Mellanox are trademarks and/or registered trademarks of NVIDIA Corporation in the U.S. and othercountries. Other company and product names may be trademarks of the respective companies with which they are associated.

For the complete and most updated list of Mellanox trademarks, visithttp://www.mellanox.com/page/trademarks.

Copyright

© 2021 NVIDIA Corporation. All rights reserved.





# **Table of Contents**

1	Abo	ut this Report	8
	1.1	Target Audience	8
	1.2	Terms and Conventions	8
2	Test	Description	9
	2.1	Hardware Components	9
	2.2	Zero Packet Loss Test	9
	2.3	Zero Packet Loss over SR-IOV Test	9
	2.4	Single Core Performance Test	9
3	Test	#1 Mellanox ConnectX-4 Lx 25GbE Throughput at Zero Packet Loss (2x 25GbE)	. 10
	3.1	Test Settings	11
	3.2	Test Results	12
4	Test	#2 Mellanox ConnectX-5 25GbE Throughput at Zero Packet Loss (2x 25GbE)	. 13
	4.1	Test Settings	14
	4.2	Test Results	15
5	Test	#3 Mellanox ConnectX-5 Ex 100GbE Throughput at Zero Packet Loss (1x 100GbE)	. 16
	5.1	Test Settings	17
	5.2	Test Results	18
6	Tost	#4 Mellanov ConnectY-5 Ex 100GhE Single Core Performance (2x 100GhE)	10
U	61	Tect Settings	20
	6.2	Test Results	20
-	Teet	#E Mallanay CompactV E 25065 Single Core Performance (2) 25065)	
1	Test	Toot Softings	. 22
	7.1 7.2		23 24
	7.2		24
8 FS	1est Xi 7 0113	#6 Mellanox ConnectX-5 25GbE Throughput at Zero Packet Loss (2x 25GbE) using SR-IOV over VMW	are 25
LJ	R 1	Tect Settings	27
	8.2	Test Results	28
0	Tost	#7 Mallanov ConnectV F Ex 100GhE Throughput at Zoro Backet Loss (1x 100GhE) using SP IOV over	
э кv	M Hype	rvisor	. 29
	9.1	Test Settings	30
	9.2	Test Results	33
10	Tost	#8 Mellanov ConnectX-6Dv 25GbF Throughput at Zero Packet Loss (2v 25GbF)	2/
10	10.1	Test Settings	35
	10.2	Test Results	36
11	Test	#0 Mallaney ConnectV C Dy 100ChE DCla Cond Throughput at Zara Daskat Loss (1y 100ChE)	27
11	11.1	Toot Softings	. 37
	11.1		02
42	11.2		55
12	lest	#10 Mellanox ConnectX-6Dx 100GbE PCIe Gen4 Single Core Performance (2x 100GbE)	. 40
	12.1		41
	12.2		42
13	Test	#11 Mellanox ConnectX-6 Dx 100GbE PCle Gen4 Throughput at Zero Packet Loss (2x 100GbE)	. 43
	13.1	Test Settings	44
	13.2	I EST KESUITS	45
14	Test	#12 Mellanox ConnectX-6 Dx 100GbE Throughput at Zero Packet Loss (1x 100GbE) using SR-IOV ove	r
KV	ии нуре	rvisor	. 46

	14.1	Test Settings	48
	14.2	Test Results	50
15	Test	#13 Mellanox ConnectX-6 Dx 200GbE PCIe Gen4 Throughput at Zero Packet Loss (1x 200GbE)	. 51
	15.1	Test Settings	52
	15.2	Test Results	53
16	Test	#14 BlueField-2 25GbE Throughput at Zero Packet Loss (2x 25GbE)	. 54
	16.1	Test Settings	55
	16.2	Test Results	56
17	Test	<pre>#15 Mellanox ConnectX-6 Lx 25GbE Throughput at Zero Packet Loss (2x 25GbE)</pre>	. 57
	17.1	Test Settings	58
	17.2	Test Results	59

# List of Figures

Figure 1: Test #1 Setup – Mellanox ConnectX-4 Lx 25GbE Dual-Port connected to IXIA	10
Figure 2: Test #1 Results – Mellanox ConnectX-4 Lx 25GbE Dual-Port Throughput at Zero Packet Loss	12
Figure 3: Test #2 Setup – Mellanox ConnectX-5 25GbE Dual-Port connected to IXIA	13
Figure 4: Test #2 Results – Mellanox ConnectX-5 25GbE Dual-Port Throughput at Zero Packet Loss	15
Figure 5: Test #3 Setup – Mellanox ConnectX-5 Ex 100GbE connected to IXIA	16
Figure 6: Test #3 Results – Mellanox ConnectX-5 Ex 100GbE Throughput at Zero Packet Loss	18
Figure 7: Test #4 Setup – Two Mellanox ConnectX-5 Ex 100GbE connected to IXIA	19
Figure 8: Test #4 Results – Mellanox ConnectX-5 Ex 100GbE Single Core Performance	21
Figure 9: Test #5 Setup – Two Mellanox ConnectX-5 25GbE connected to IXIA	22
Figure 10: Test #5 Results – Mellanox ConnectX-5 25GbE Single Core Performance	24
Figure 11: Test #6 Setup – Mellanox ConnectX-5 25GbE connected to IXIA using ESXi SR-IOV	26
Figure 12: Test#6 Results – Mellanox ConnectX-5 25GbE Throughput at Zero Packet Loss using ESXi SR-IOV	28
Figure 13: Test #7 Setup – Mellanox ConnectX-5 Ex 100GbE connected to IXIA using KVM SR-IOV	30
Figure 14: Test #7 Results – Mellanox ConnectX-5 Ex 100GbE Throughput at Zero Packet Loss using KVM SR-IOV	33
Figure 15: Test #8 Setup – Mellanox ConnectX-6 Dx 25GbE Dual-Port connected to IXIA	34
Figure 16: Test #8 Results – Mellanox ConnectX-6Dx 25GbE Dual-Port Throughput at Zero Packet Loss	36
Figure 17: Test #9 Setup – Mellanox ConnectX-6 Dx 100GbE connected to IXIA	37
Figure 18: Test #9 Results – Mellanox ConnectX-5 Ex 100GbE Throughput at Zero Packet Loss	39
Figure 19: Test #10 Setup – Two Mellanox ConnectX-6 Dx 100GbE connected to IXIA	40
Figure 20: Test #10 Results – Mellanox ConnectX-6Dx 100GbE Single Core Performance	42
Figure 21: Test #11 Setup – Mellanox ConnectX-6 Dx 100GbE connected to IXIA	43
Figure 22: Test #11 Results – Mellanox ConnectX-6 Dx 100GbE dual port PCIe Gen4 Throughput at Zero Packet Loss	45
Figure 23 - Test #12 Setup – Mellanox ConnectX-6 Dx 100GbE connected to IXIA using KVM SR-IOV	47
Figure 24 - Test #12 Results – Mellanox ConnectX-6 Dx 100GbE Throughput at Zero Packet Loss using KVM SR-IOV	50
Figure 25 - Test #13 Setup – Mellanox ConnectX-6 Dx 200GbE connected to IXIA	51
Figure 26 - Test #13 Results – Mellanox ConnectX-6 Dx 200GbE dual port PCIe Gen4 Throughput at Zero Packet Loss	53
Figure 27 -Test #14 Setup – BlueField-2 25GbE Dual-Port connected to IXIA	54
Figure 28 - Test #14 Results – BlueField-2 25GbE Dual-Port Throughput at Zero Packet Loss	56
Figure 29 - Test #15 Setup – Mellanox ConnectX-6 Lx 25GbE Dual-Port connected to IXIA	57
Figure 30 - Test #15 Results – Mellanox ConnectX-6 Lx 25GbE Dual-Port Throughput at Zero Packet Loss	59

# List of Tables

Table 1 - Document History	7
Table 2 - Terms, Abbreviations and Acronyms	8
Table 3: Test #1 Setup	10
Table 4: Test #1 Settings	11
Table 5: Test #1 Results – Mellanox ConnectX-4 Lx 25GbE Dual-Port Throughput at Zero Packet Loss	12
Table 6: Test #2 Setup	13
Table 7: Test #2 Settings	14
Table 8: Test #2 Results – Mellanox ConnectX-5 25GbE Dual-Port Throughput at Zero Packet Loss	15
Table 9: Test #3 Setup	16
Table 10: Test #3 Settings	17
Table 11: Test #3 Results – Mellanox ConnectX-5 Ex 100GbE Throughput at Zero Packet Loss	18
Table 12: Test #4 Setup	19
Table 13: Test #4 Settings	20
Table 14: Test #4 Results – Mellanox ConnectX-5 Ex 100GbE Single Core Performance	21
Table 15: Test #5 Setup	22
Table 16: Test #5 Settings	23
Table 17: Test #5 Results – Mellanox ConnectX-5 25GbE Single Core Performance	24
Table 18: Test #6 Setup	25
Table 19: Test#6 Settings	27
Table 20: Test#6 Results – Mellanox ConnectX-5 25GbE Throughput at Zero Packet Loss using ESXi SR-IOV	28
Table 21: Test #7 Setup	29
Table 22: Test #7 Settings	30
Table 23: Test #7 Results – Mellanox ConnectX-5 Ex 100GbE Throughput at Zero Packet Loss using KVM SR-IOV	33
Table 24: Test #8 Setup	34
Table 25: Test #8 Settings	35
Table 26: Test #8 Results – Mellanox ConnectX-6Dx 25GbE Dual-Port Throughput at Zero Packet Loss	36
Table 27: Test #9 Setup	37
Table 28: Test #9 Settings	38
Table 29: Test #9 Results – Mellanox ConnectX-6 Dx 100GbE Throughput at Zero Packet Loss	39
Table 30: Test #10 Setup	40
Table 31: Test #10 Settings	41
Table 32: Test #10 Results – Mellanox ConnectX-6 Dx 100GbE Single Core Performance	42
Table 33: Test #11 Setup	43
Table 34: Test #11 Settings	44
Table 35: Test #11 Results – Mellanox ConnectX-6 Dx 100GbE dual port PCIe Gen4 Zero Packet Loss Throughput	45
Table 36 - Test #12 Setup	46
Table 37 - Test #12 Settings	48
Table 38 - Test #12 Results – Mellanox ConnectX-6 Dx 100GbE Throughput at Zero Packet Loss using KVM SR-IOV	50
Table 39 - Test #13 Setup	51
Table 40 - Test #13 Settings	52
Table 41 - Test #13 Results – Mellanox ConnectX-6 Dx 200GbE single port PCIe Gen4 Throughput at Zero Packet Loss	53
Table 42 - Test #14 Setup	54
Table 43 - Test #14 Settings	55
Table 44 - Test #14 Results – BlueField-2 25GbE Dual-Port Throughput at Zero Packet Loss	56
Table 45 - Test #15 Setup	57
Table 46 - Test #15 Settings	58
Table 47 - Test #15 Results – Mellanox ConnectX-6 Lx 25GbE Dual-Port Throughput at Zero Packet Loss	59

# **Document History**

#### Table 1 - Document History

Version	Date	Description of Change	
1.0	07-JUL-2022	Initial report release	
1.1	17-JAN-2023	Adjust the number of cores used in command line for Test#11	
1.2	21-Feb-2023	Fix a typo in Test#13 results	

# 1 About this Report

The purpose of this document is to provide packet rate performance data for NVIDIA<sup>®</sup> Mellanox<sup>®</sup> Network Interface Cards (NICs - ConnectX<sup>®</sup>-4 Lx, ConnectX<sup>®</sup>-5, ConnectX<sup>®</sup>-5 Ex, ConnectX<sup>®</sup>-6 Lx, ConnectX<sup>®</sup>-6 Dx) and Data Processing Unit (BlueField-2 DPU) (that has been achieved with the specified Data Plane Development Kit (DPDK) release. The report provides the measured packet rate performance as well as the hardware layout, procedures, and configurations for replicating these tests.

The document does not cover all network speeds available with the ConnectX<sup>®</sup> or BlueField<sup>®</sup> family of NICs / DPUs and is intended as a general reference of achievable performance for the specified DPDK release.

### 1.1 Target Audience

This document is intended for engineers implementing applications with DPDK to guide and help achieving optimal performance.

### **1.2** Terms and Conventions

The following terms, abbreviations, and acronyms are used in this document.

Term	Description			
DPU	Data Processing Unit			
DUT	Device Under Test			
MPPS	Million Packets Per Seconds			
PPS	Packets Per Second			
OFED	OpenFabrics Enterprise Distribution; An open-source software for RDMA & kernel bypass.			
	Read more on Mellanox OFED <u>here</u> .			
SR-IOV	Single Root IO Virtualization			
ZPL	Zero Packet Loss			

#### Table 2 - Terms, Abbreviations and Acronyms

# 2 Test Description

### 2.1 Hardware Components

The following hardware components are used in the test setup:

- One of the following servers:
  - o HPE® ProLiant DL380 Gen10 Server
  - o HPE® ProLiant DL380 Gen10 Plus Server
- One of the followings NICs, SmartNICs or DPUs:
  - Mellanox ConnectX-4 Lx, ConnectX-5, ConnectX-5 Ex, ConnectX-6 Lx, ConnectX-6 Dx Network Interface Cards (NICs) and BlueField-2 Data Processing Unit (DPU)
- IXIA<sup>®</sup> XM12 packet generator

### 2.2 Zero Packet Loss Test

Zero Packet Loss tests utilize **I3fwd** (<u>http://www.dpdk.org/doc/guides/sample\_app\_ug/I3\_forward.html</u>) as the test application for testing maximum throughput with zero packet loss at various frame sizes based on RFC2544 <u>https://tools.ietf.org/html/rfc2544</u>.

The packet generator transmits a specified frame rate towards the Device Under Test (DUT) and counts the received frame rate sent back from the DUT. Throughput is determined with the maximum achievable transmit frame rate and is equal to the received frame rate i.e. zero packet loss.

- Duration for each test is 60 seconds.
- Traffic of 8192 IP flows is generated per port.
- ▶ IxNetwork (Version 9.20EA) is used with the IXIA packet generator.

## 2.3 Zero Packet Loss over SR-IOV Test

The test is conducted similarly to the bare-metal zero packet loss test with the distinction of having the DPDK application running in a Guest OS inside a VM utilizing SR-IOV virtual function.

## 2.4 Single Core Performance Test

Single Core performance tests utilize **testpmd** (<u>http://www.dpdk.org/doc/guides/testpmd\_app\_ug</u>), for testing the max throughput while using a single CPU core. The duration of the test is 60 seconds and the average throughput that is recorded during that time is used as the result of the test.

- Duration for each test is 60 seconds.
- Traffic of 8192 UDP flows is generated per port.
- ▶ IxNetwork (Version 9.20EA) is used with the IXIA packet generator.

# 3 Test#1 Mellanox ConnectX-4 Lx 25GbE Throughput at Zero Packet Loss (2x 25GbE)

#### Table 3: Test #1 Setup

Item	Description
Test #1	Mellanox ConnectX-4 Lx 25GbE Dual-Port Throughput at zero packet loss
Server	HPE ProLiant DL380 Gen10
CPU	Intel(R) Xeon(R) Platinum 8168 CPU @ 2.70GHz 24 CPU cores * 2 NUMA nodes
RAM	384GB: 6 * 32GB DIMMs * 2 NUMA nodes @ 2666MHz
BIOS	U30 rev. 1.36 (02/15/2018)
NIC	One MCX4121A-ACAT - ConnectX-4 Lx network interface card 25GbE dual-port SFP28; PCIe3.0 x8; ROHS R6
Operating System	Ubuntu 20.04.2 LTS (Focal Fossa)
Kernel Version	5.4.0-65-generic.x86_64
GCC version	gcc (Ubuntu 9.3.0-17ubuntu1~20.04) 9.3.0
Mellanox NIC firmware version	14.32.1010
Mellanox OFED driver version	MLNX_OFED_LINUX-5.5-1.0.3.2
DPDK version	22.03
Test Configuration	1 NIC, 2 ports used on the NIC. Each port receives a stream of 8192 IP flows from the IXIA Each port has 4 queues assigned for a total of 8 queues 1 queue assigned per logical core with a total of 8 logical cores

The Device Under Test (DUT) is made up of the HPE server and the Mellanox ConnectX-4 Lx Dual-Port NIC. The DUT is connected to the IXIA packet generator which generates traffic towards the ConnectX-4 Lx NIC. The ConnectX-4 Lx data traffic is passed through DPDK to the test application **I3fwd** and is redirected to the opposite direction on the opposing port. IXIA measures throughput and packet loss.



#### Table 4: Test #1 Settings

Item	Description
BIOS	<ol> <li>Workload Profile = "Low Latency";</li> <li>Jitter Control = Manual, 3400. (Setting turbo boost frequency to 3.4 GHz)</li> <li>See "Configuring and tuning HPE ProLiant Servers for low-latency applications": hpe.com &gt; Search "DL380 gen10 low latency"</li> </ol>
BOOT Settings	isolcpus=24-47 intel_idle.max_cstate=0 processor.max_cstate=0 intel_pstate=disable nohz_full=24-47 rcu_nocbs=24-47 rcu_nocb_poll default_hugepagesz=1G hugepagesz=1G hugepages=64 audit=0 nosoftlockup
DPDK Settings	Compile DPDK using: meson <build> -Dexamples=I3fwd ; ninja -C <build> During testing, I3fwd was given real-time scheduling priority.</build></build>
L3fwd settings	Updated values /l3fwd/l3fwd.h: #define RTE_TEST_RX_DESC_DEFAULT 4096 #define RTE_TEST_TX_DESC_DEFAULT 4096 #define MAX_PKT_BURST 64
Command Line	./build/examples/dpdk-l3fwd -c 0xff0000000000 -n 4 -a d8:00.0,txq_inline=200,txq_mpw_en=1 -a d8:00.1,txq_inline=200,txq_mpw_en=1socket-mem=0,8192p 0x3 -P config='(0,0,47),(0,1,46),(0,2,45),(0,3,44),(1,0,43),(1,1,42),(1,2,41),(1,3,40)'eth-dest=0,00:52:11:22:33:10eth-dest=1,00:52:11:22:33:20
Other optimizations	<ul> <li>a) Flow Control OFF: "ethtool -A \$netdev rx off tx off"</li> <li>b) Memory optimizations: "sysctl -w vm.zone_reclaim_mode=0"; "sysctl -w vm.swappiness=0"</li> <li>c) Move all IRQs to far NUMA node: "IRQBALANCE_BANNED_CPUS=\$LOCAL_NUMA_CPUMAP irqbalanceoneshot"</li> <li>d) Disable irqbalance: "systemctl stop irqbalance"</li> <li>e) Change PCI MaxReadReq to 1024B for each port of each NIC: Run "setpci -s \$PORT_PCI_ADDRESS 68.w", it will return 4 digits ABCD&gt; Run "setpci -s \$PORT_PCI_ADDRESS 68.w=3BCD"</li> <li>f) Set CQE COMPRESSION to "AGGRESSIVE": mlxconfig -d \$PORT_PCI_ADDRESS set CQE_COMPRESSION=1</li> <li>G) Disable Linux realtime throttling: echo_1 &gt; (proc/sys/kernel/sched_rt_runtime_us)</li> </ul>

Frame Size (Bytes)	Frame Rate (Mpps)	Line Rate [50G] (Mpps)	% Line Rate
64	70.80	74.4	95.17
128	39.22	42.23	92.88
256	22.64	22.64	100
512	11.75	11.75	100
1024	5.99	5.99	100
1280	4.81	4.81	100
1518	4.06	4.06	100

Table 5: Test #1 Results – Mellanox ConnectX-4 Lx 25GbE Dual-Port Throughput at Zero Packet Loss





# 4 Test#2 Mellanox ConnectX-5 25GbE Throughput at Zero Packet Loss (2x 25GbE)

#### Table 6: Test #2 Setup

Item	Description
Test #2	Mellanox ConnectX-5 25GbE Dual-Port Throughput at zero packet loss
Server	HPE ProLiant DL380 Gen10
CPU	Intel(R) Xeon(R) Platinum 8168 CPU @ 2.70GHz 24 CPU cores * 2 NUMA nodes
RAM	384GB: 6 * 32GB DIMMs * 2 NUMA nodes @ 2666MHz
BIOS	U30 rev. 1.36 (02/15/2018)
NIC	One MCX512A-ACAT ConnectX-5 EN network interface card; 10/25GbE dual-port SFP28; PCIe3.0 x8; tall bracket; ROHS R6
Operating System	Ubuntu 20.04.2 LTS (Focal Fossa)
Kernel Version	5.4.0-65-generic.x86_64
GCC version	gcc (Ubuntu 9.3.0-17ubuntu1~20.04) 9.3.0
Mellanox NIC firmware version	16.32.1010
Mellanox OFED driver version	MLNX_OFED_LINUX-5.5-1.0.3.2
DPDK version	22.03
Test	1 NIC, 2 ports;
Configuration	Each port receives a stream of 8192 IP flows from the IXIA
	Each port has 4 queues assigned for a total of 8 queues
	1 queue assigned per logical core with a total of 8 logical cores

The Device Under Test (DUT) is made up of the HPE server and the Mellanox ConnectX-5 Dual-Port NIC. The DUT is connected to the IXIA packet generator which generates traffic towards the ConnectX-5 NIC.

The ConnectX-5 data traffic is passed through DPDK to the test application **I3fwd** and is redirected to the opposite direction on the same port. IXIA measures throughput and packet loss.



#### Table 7: Test #2 Settings

Item	Description
BIOS	<ol> <li>Workload Profile = "Low Latency";</li> <li>Jitter Control = Manual, 3400. (Setting turbo boost frequency to 3.4 GHz)</li> <li>See "Configuring and tuning HPE ProLiant Servers for low-latency applications": hpe.com &gt; Search "DL380 gen10 low latency"</li> </ol>
BOOT Settings	isolcpus=24-47 intel_idle.max_cstate=0 processor.max_cstate=0 intel_pstate=disable nohz_full=24- 47 rcu_nocbs=24-47 rcu_nocb_poll default_hugepagesz=1G hugepagesz=1G hugepages=64 audit=0 nosoftlockup
DPDK Settings	Compile DPDK using: meson <build> -Dexamples=I3fwd ; ninja -C <build> During testing, I3fwd was given real-time scheduling priority.</build></build>
L3fwd settings	Updated values /l3fwd/l3fwd.h: #define RTE_TEST_RX_DESC_DEFAULT 4096 #define RTE_TEST_TX_DESC_DEFAULT 4096 #define MAX_PKT_BURST 64
Command Line	./build/examples/dpdk-l3fwd -c 0xff000000000 -n 4 -a d8:00.0,mprq_en=1,rxqs_min_mprq=1 -a d8:00.1,mprq_en=1,rxqs_min_mprq=1socket-mem=0,8192p 0x3 -P config='(0,0,47),(0,1,46),(0,2,45),(0,3,44),(1,0,43),(1,1,42),(1,2,41),(1,3,40)'eth-dest=0,00:52:11:22:33:10eth-dest=1,00:52:11:22:33:20
Other optimizations	<ul> <li>a) Flow Control OFF: "ethtool -A \$netdev rx off tx off"</li> <li>b) Memory optimizations: "sysctl -w vm.zone_reclaim_mode=0"; "sysctl -w vm.swappiness=0"</li> <li>c) Move all IRQs to far NUMA node: "IRQBALANCE_BANNED_CPUS=\$LOCAL_NUMA_CPUMAP irqbalanceoneshot"</li> <li>d) Disable irqbalance: "systemctl stop irqbalance"</li> <li>e) Change PCI MaxReadReq to 1024B for each port of each NIC: Run "setpci -s \$PORT_PCI_ADDRESS 68.w", it will return 4 digits ABCD&gt; Run "setpci -s \$PORT_PCI_ADDRESS 68.w=3936"</li> <li>f) Set CQE COMPRESSION to "AGGRESSIVE": mlxconfig -d \$PORT_PCI_ADDRESS set CQE_COMPRESSION=1</li> <li>g) Disable Linux realtime throttling: echo -1 &gt; /proc/sys/kernel/sched rt runtime us</li> </ul>

Frame Size (Bytes)	Frame Rate (Mpps)	Line Rate [50G] (Mpps)	% Line Rate
64	74.40	74.40	100.00
128	42.23	42.23	100.00
256	22.64	22.64	100.00
512	11.75	11.75	100.00
1024	5.99	5.99	100.00
1280	4.81	4.81	100.00
1518	4.06	4.06	100.00

Table 8: Test #2 Results – Mellanox ConnectX-5 25GbE Dual-Port Throughput at Zero Packet Loss

#### Figure 4: Test #2 Results – Mellanox ConnectX-5 25GbE Dual-Port Throughput at Zero Packet Loss



# 5 Test#3 Mellanox ConnectX-5 Ex 100GbE Throughput at Zero Packet Loss (1x 100GbE)

#### Table 9: Test #3 Setup

Item	Description
Test #3	Mellanox ConnectX-5 Ex 100GbE Throughput at zero packet loss
Server	HPE ProLiant DL380 Gen10
СРИ	Intel(R) Xeon(R) Platinum 8168 CPU @ 2.70GHz 24 CPU cores * 2 NUMA nodes
RAM	384GB: 6 * 32GB DIMMs * 2 NUMA nodes @ 2666MHz
BIOS	U30 rev. 1.36 (02/15/2018)
NIC	One MCX516A-CDAT ConnectX-5 Ex network interface card
	100GbE dual-port QSFP28; PCle3.0/PCle4 x16; ROHS R6
Operating System	Ubuntu 20.04.2 LTS (Focal Fossa)
Kernel Version	5.4.0-65-generic.x86_64
GCC version	gcc (Ubuntu 9.3.0-17ubuntu1~20.04) 9.3.0
Mellanox NIC firmware version	16.32.1010
Mellanox OFED driver version	MLNX_OFED_LINUX-5.5-1.0.3.2
DPDK version	22.03
Test Configuration	1 NIC, 1 port used on NIC; Port has 12 queues assigned to it, 1 queue per logical core for a total of 12 logical cores. Each port receives a stream of 8192 IP flows from the IXIA

The Device Under Test (DUT) is made up of the HPE server and the Mellanox ConnectX-5 Ex Dual-Port NIC (only the first port is used in this test). The DUT is connected to the IXIA packet generator which generates traffic towards the ConnectX-5 Ex NIC.

The ConnectX-5 Ex data traffic is passed through DPDK to the test application **I3fwd** and is redirected to the opposite direction on the same port. IXIA measures throughput and packet loss.



NVIDIA Mellanox NICs Performance Report with DPDK 22.03

## Table 10: Test #3 Settings

Item	Description
BIOS	<ol> <li>Workload Profile = "Low Latency";</li> <li>Jitter Control = Manual, 3400. (Setting turbo boost frequency to 3.4 GHz)</li> </ol>
	See "Configuring and tuning HPE ProLiant Servers for low-latency applications":
	hpe.com > Search "DL380 gen10 low latency"
BOOT Settings	isolcpus=24-47 intel_idle.max_cstate=0 processor.max_cstate=0 intel_pstate=disable nohz_full=24-47 rcu_nocbs=24-47 rcu_nocb_poll default_hugepagesz=1G hugepagesz=1G hugepages=64 audit=0 nosoflockup
DPDK Settings	Compile DPDK using:
	meson <build> -Dexamples=l3fwd ; ninja -C <build></build></build>
	During testing, I3fwd was given real-time scheduling priority.
L3fwd settings	Updated values /I3fwd/I3fwd.h:
	#define RTE_TEST_RX_DESC_DEFAULT 4096
	#define RTE_TEST_TX_DESC_DEFAULT 4096
	#define MAX_PKT_BURST 64
Command Line	./build/examples/dpdk-l3fwd -c 0xfff000000000 -n 4 -a 0000:af:00.0,mprq_en=1,rxqs_min_mprq=1 socket-mem=0,8192p 0x1 -P copfig='(0,0,47) (0,1,46) (0,2,45) (0,2,44) (0,4,42) (0,5,42) (0,6,41) (0,7,40) (0,8,29) (0,9,28) (0,10,27,40)
	),(0,11,36)'eth-dest=0,00:52:11:22:33:10
Other optimizations	a) Flow Control OFF: "ethtool -A \$netdev rx off tx off"
	b) Memory optimizations: "sysctl -w vm.zone_reclaim_mode=0"; "sysctl -w vm.swappiness=0"
	c) Move all IRQs to far NUMA node: "IRQBALANCE_BANNED_CPUS=\$LOCAL_NUMA_CPUMAP irqbalanceoneshot"
	d) Disable irqbalance: "systemctl stop irqbalance"
	e) Change PCI MaxReadReq to 1024B for each port of each NIC:
	Run "setpci -s \$PORT_PCI_ADDRESS 68.w", it will return 4 digits ABCD>
	Run "setpci -s \$PORT_PCI_ADDRESS 68.w=3BCD"
	f) Set CQE COMPRESSION to "AGGRESSIVE": mlxconfig -d \$PORT_PCI_ADDRESS set CQE_COMPRESSION=1
	g) Disable Linux realtime throttling: echo -1 > /proc/sys/kernel/sched_rt_runtime_us

Frame Size (Bytes)	Frame Rate (Mpps)	Line Rate [100G] (Mpps)	% Line Rate
64	148.66	148.81	99.91
128	84.46	84.46	100.00
256	45.29	45.29	100.00
512	23.50	23.50	100.00
1024	11.97	11.97	100.00
1280	9.62	9.62	100.00
1518	8.13	8.13	100.00

Table 11: Test #3 Results – Mellanox ConnectX-5 Ex 100GbE Throughput at Zero Packet Loss





# 6 Test#4 Mellanox ConnectX-5 Ex 100GbE Single Core Performance (2x 100GbE)

#### Table 12: Test #4 Setup

Item	Description
Test #4	Mellanox ConnectX-5 Ex 100GbE Single Core Performance
Server	HPE ProLiant DL380 Gen10
СРИ	Intel(R) Xeon(R) Platinum 8168 CPU @ 2.70GHz 24 CPU cores * 2 NUMA nodes
RAM	384GB: 6 * 32GB DIMMs * 2 NUMA nodes @ 2666MHz
BIOS	U30 rev. 1.36 (02/15/2018)
NIC	Two MCX516A-CDAT- ConnectX-5 Ex network interface cards
	100GbE dual-port QSFP28; PCle3.0/PCle4 x16; ROHS R6
Operating System	Ubuntu 20.04.2 LTS (Focal Fossa)
Kernel Version	5.4.0-65-generic.x86_64
GCC version	gcc (Ubuntu 9.3.0-17ubuntu1~20.04) 9.3.0
Mellanox NIC firmware version	16.32.1010
Mellanox OFED driver version	MLNX_OFED_LINUX-5.5-1.0.3.2
DPDK version	22.03
Test Configuration	2 NICs, each using 1 port
	Each port receives a stream of 8192 UDP flows from the IXIA
	Each port has 1 queue assigned, a total of two queues for two ports and both queues are assigned to the same single logical core.

The Device Under Test (DUT) is made up of the HPE server and two Mellanox ConnectX-5 Ex NICs utilizing one port each. The DUT is connected to the IXIA packet generator which generates traffic towards the first port of both ConnectX-5 Ex NICs.

The ConnectX-5 Ex data traffic is passed through DPDK to the test application **testpmd** and is redirected to the opposite direction on the opposing NIC's port. IXIA measures throughput and packet loss.



## Table 13: Test #4 Settings

Item	Description
BIOS	<ol> <li>Workload Profile = "Low Latency";</li> <li>Jitter Control = Manual, 3400. (Setting turbo boost frequency to 3.4 GHz)</li> <li>See "Configuring and tuning HPE ProLiant Servers for low-latency applications": hpe.com &gt; Search "DL380 gen10 low latency"</li> </ol>
BOOT Settings	isolcpus=24-47 intel_idle.max_cstate=0 processor.max_cstate=0 intel_pstate=disable nohz_full=24- 47 rcu_nocbs=24-47 rcu_nocb_poll default_hugepagesz=1G hugepagesz=1G hugepages=64 audit=0
DPDK Settings	Compile DPDK using: meson <build> ; ninja -C <build> During testing, testpmd was given real-time scheduling priority.</build></build>
Command Line	./build/app/dpdk-testpmd -c 0x110000000000 -n 4 -a 86:00.0 -a af:00.0socket-mem=0,8192 port-numa-config=0,1,1,1socket-num=1burst=64txd=1024rxd=1024mbcache=512rxq=1 txq=1nb-cores=1 -i -arss-udpdisable-crc-striprecord-core-cyclesrecord-burst-stats
Other optimizations	<ul> <li>a) Flow Control OFF: "ethtool -A \$netdev rx off tx off"</li> <li>b) Memory optimizations: "sysctl -w vm.zone_reclaim_mode=0"; "sysctl -w vm.swappiness=0"</li> <li>c) Move all IRQs to far NUMA node: "IRQBALANCE_BANNED_CPUS=\$LOCAL_NUMA_CPUMAP irqbalanceoneshot"</li> <li>d) Disable irqbalance: "systemctl stop irqbalance"</li> <li>e) Change PCI MaxReadReq to 1024B for each port of each NIC: Run "setpci -s \$PORT_PCI_ADDRESS 68.w", it will return 4 digits ABCD&gt; Run "setpci -s \$PORT_PCI_ADDRESS 68.w=3BCD"</li> <li>f) Set CQE COMPRESSION to "AGGRESSIVE": mlxconfig -d \$PORT_PCI_ADDRESS set CQE_COMPRESSION=1</li> <li>g) Disable Linux real-time throttling echo -1 &gt; /proc/sys/kernel/sched_rt_runtime_us</li> </ul>

Frame Size (Bytes)	Frame Rate (Mpps)	Line Rate [200G] (Mpps)	Line Rate [100G] (Mpps)	Throughput (Gbps)	CPU Cycles per packet NOTE: Lower is Better
64	78.70	297.62	148.81	40.291	30
128	76.43	168.92	84.46	78.247	29
256	64.80	90.58	45.29	132.749	31
512	46.84	46.99	23.50	191.845	32
1024	23.95	23.95	11.97	196.161	30
1280	19.23	19.23	9.62	196.880	32
1518	16.25	16.25	8.13	197.376	30

Table 14: Test #4 Results – Mellanox ConnectX-5 Ex 100GbE Single Core Performance

Figure 8: Test #4 Results – Mellanox ConnectX-5 Ex 100GbE Single Core Performance



# 7 Test#5 Mellanox ConnectX-5 25GbE Single Core Performance (2x 25GbE)

#### Table 15: Test #5 Setup

Item	Description
Test #5	Mellanox ConnectX-5 25GbE Single Core Performance
Server	HPE ProLiant DL380 Gen10
CPU	Intel(R) Xeon(R) Platinum 8168 CPU @ 2.70GHz 24 CPU cores * 2 NUMA nodes
RAM	384GB: 6 * 32GB DIMMs * 2 NUMA nodes @ 2666MHz
BIOS	U30 rev. 1.36 (02/15/2018)
NIC	Two MCX512A-ACA ConnectX-5 EN network interface cards; 10/25GbE dual-port SFP28; PCIe3.0 x8; tall bracket; ROHS R6
Operating System	Ubuntu 20.04.2 LTS (Focal Fossa)
Kernel Version	5.4.0-65-generic.x86_64
GCC version	gcc (Ubuntu 9.3.0-17ubuntu1~20.04) 9.3.0
Mellanox NIC firmware version	16.32.1010
Mellanox OFED driver version	MLNX_OFED_LINUX-5.5-1.0.3.2
DPDK version	22.03
Test Configuration	2 NICs; 1 port used on each.
	Each port receives a stream of 8192 UDP flows from the IXIA
	Each port has 1 queue assigned, a total of two queues for two ports, and both queues are assigned to the same single logical core.

The Device Under Test (DUT) is made up of the HPE server and two Mellanox ConnectX-5 25GbE NICs utilizing one port each. The DUT is connected to the IXIA packet generator which generates traffic towards the first port of both ConnectX-5 25GbE NICs.

The ConnectX-5 25GbE data traffic is passed through DPDK to the test application **testpmd** and is redirected to the opposite direction on the opposing NIC's port. IXIA measures throughput and packet loss.



## Table 16: Test #5 Settings

Item	Description
BIOS	<ol> <li>Workload Profile = "Low Latency"</li> <li>Jitter Control = Manual, 3400. (Setting turbo boost frequency to 3.4 GHz)</li> <li>See "Configuring and tuning HPE ProLiant Servers for low-latency applications": hpe.com &gt; Search "DL380 gen10 low latency"</li> </ol>
BOOT Settings	isolcpus=24-47 intel_idle.max_cstate=0 processor.max_cstate=0 intel_pstate=disable nohz_full=24-47 rcu_nocbs=24-47 rcu_nocb_poll default_hugepagesz=1G hugepagesz=1G hugepages=64 audit=0 nosoftlockup
DPDK Settings	Compile DPDK using: meson <build> ; ninja -C <build> During testing, testpmd was given real-time scheduling priority.</build></build>
Command Line	./build/app/dpdk-testpmd -c 0x3000000000 -n 4 -a d8:00.0 -a d9:00.0socket-mem=0,8192 port-numa-config=0,1,1,1socket-num=1burst=64txd=1024rxd=1024mbcache=512 rxq=1txq=1nb-cores=1 -i -arss-udpdisable-crc-striprecord-core-cyclesrecord-burst- stats
Other optimizations	<ul> <li>a) Flow Control OFF: "ethtool -A \$netdev rx off tx off"</li> <li>b) Memory optimizations: "sysctl -w vm.zone_reclaim_mode=0"; "sysctl -w vm.swappiness=0"</li> <li>c) Move all IRQs to far NUMA node: "IRQBALANCE_BANNED_CPUS=\$LOCAL_NUMA_CPUMAP irqbalanceoneshot"</li> <li>d) Disable irqbalance: "systemctl stop irqbalance"</li> <li>e) Change PCI MaxReadReq to 1024B for each port of each NIC: Run "setpci -s \$PORT_PCI_ADDRESS 68.w", it will return 4 digits ABCD&gt; Run "setpci -s \$PORT_PCI_ADDRESS 68.w=3BCD"</li> <li>f) Set CQE COMPRESSION to "AGGRESSIVE": mlxconfig -d \$PORT_PCI_ADDRESS set CQE_COMPRESSION=1</li> <li>g) Disable Linux roaltime throttling: esho. 1 &gt; (pres/cus/kernel/cshod_st_suptime_us)</li> </ul>

Frame Size (Bytes)	Frame Rate (Mpps)	Line Rate [25G] (Mpps)	Line Rate [50G] (Mpps)	Throughput (Gbps)	CPU Cycles per packet NOTE: Lower is Better
64	69.00	37.2	74.4	25.328	27
128	42.15	21.11	42.23	43.157	25
256	22.64	11.32	22.64	46.371	27
512	11.75	5.87	11.75	48.114	27
1024	5.99	2.99	5.99	49.037	32
1280	4.81	2.4	4.81	49.224	29
1518	4.06	2.03	4.06	49.342	32

Table 17: Test #5 Results – Mellanox ConnectX-5 25GbE Single Core Performance

Figure 10: Test #5 Results – Mellanox ConnectX-5 25GbE Single Core Performance



# 8 Test#6 Mellanox ConnectX-5 25GbE Throughput at Zero Packet Loss (2x 25GbE) using SR-IOV over VMware ESXi 7.0U3

#### Table 18: Test #6 Setup

Item	Description
Test #6	Mellanox ConnectX-5 25GbE Dual-Port Throughput at zero packet loss SRIOV over VMware ESXi 7.0U3
Server	HPE ProLiant DL380 Gen10
CPU	Intel(R) Xeon(R) Platinum 8168 CPU @ 2.70GHz 24 CPU cores * 2 NUMA nodes
RAM	384GB: 6 * 32GB DIMMs * 2 NUMA nodes @ 2666MHz
BIOS	U30 rev. 1.36 (02/15/2018)
NIC	One MCX512A-ACAT ConnectX-5 EN network interface card; 10/25GbE dual-port SFP28; PCIe3.0 x8; tall bracket; ROHS R6
Hypervisor	VMware ESXi 7.0U3
Hypervisor Build	VMware-VMvisor-Installer-7.0-18945352.x86_64.iso
Hypervisor Mellanox Driver	Mellanox-nmlx5_4.23.73.0007-10EM.703.1.0.18806049
Guest Operating System	Red Hat Enterprise Linux Server release 7.7 (Maipo)
Guest Kernel Version	3.10.0-1062.el7.x86_64
Guest GCC version	4.8.5 20150623 (Red Hat 4.8.5-28) (GCC)
Guest Mellanox OFED driver version	MLNX_OFED_LINUX-5.5-1.0.3.2
Mellanox NIC firmware version	16.32.1010
DPDK version	22.03
Test Configuration	<ol> <li>NIC, 2 ports with 1 VF per port (SR-IOV);</li> <li>Each port receives a stream of 8192 IP flows from the IXIA</li> <li>Each VF (SR-IOV) has 4 queues assigned for a total of 8 queues</li> <li>1 queue assigned per logical core with a total of 8 logical cores.</li> </ol>

The Device Under Test (DUT) is made up of the HPE server and the Mellanox ConnectX-5 NIC with dual-port. The DUT is connected to the IXIA packet generator which generates traffic towards the ConnectX-5 NIC. The ConnectX-5 data traffic is passed to VF1 (SR-IOV assigned to Port1) and VF2 (SR-IOV assigned to Port2) to VM running over ESXi 6.5 hypervisor. VM runs **I3fwd** over DPDK and is redirects traffic to the opposite direction on the same VF/port. IXIA measures throughput and packet loss.





#### Table 19: Test#6 Settings

Item	Description
BIOS	1) Workload Profile = "Low Latency";
	2) Jitter Control = Manual, 3400. (Setting turbo boost frequency to 3.4 GHz)
	3) Change "Workload Profile" to "Custom"
	4) Change VT-x, VT-d and SR-IOV from "Disabled" to "Enabled".
	See "Configuring and tuning HPE ProLiant Servers for low-latency applications":
	hpe.com > Search "DL380 gen10 low latency"
BOOT Settings Guest OS	isolcpus=0-22 intel_idle.max_cstate=0 processor.max_cstate=0 intel_pstate=disable idle=poll nohz_full=0-22 rcu_nocbs=0-22 rcu_nocb_poll default_hugepagesz=1G hugepagesz=1G hugepages=16 nosoftlockup
Hypervisor settings	1) Enable SRIOV via NIC configuration tool: (requires installation of mft-tools)
	/opt/mellanox/bin/mlxconfig -d <pci id=""> set NUM_OF_VFS=2 SRIOV_EN=1 CQE_COMPRESSION=1</pci>
	reboot
	2) Install Driver
	esxcli software vib install -d Mellanox-nmlx5_4.21.71.1-10EM.702.0.0.17473468.zip
	reboot
	esxcfg-module -s 'max_vfs=1,1,1,1,1,1,1,1 supported_num_ports=8' nmlx5_core
	reboot
	3) Virtual Hardware Configuration:
	"CPU": 23
	"Cores per Socket" : 1 (Sockets = 23) or 23 (Socket = 1)
	"Hardware virtualization": enabled
	"Scheduling Affinity": 25-47
	"CPU/MMU Virtualization": "Hardware CPU and MMU"
	"RAM": 32768 MB
	"Reservation": 32768 MB
	"Reserve all guest memory (All locked)": enabled
	VM options > Advanced > "Configuration Parameters" > "Edit Configuration" : Add parameter: numa.nodeAffinity = 1
	4) Create virtual switch:
	Networking>Virtual Switches>Add standard virtual switch>Switch_SRIOV_1> Uplink :
	select vmnicXXXX matching the card under test
	5) Add port group to Switch_SRIOV_XX (VLAN=0):
	Networking>Port groups>Add port group>SRIOV_PG1>Switch_SRIOV_XX
	6) Add 2xSRIOV network adapters to VM (same settings for both ports):
	Select correct port group created previously (SRIOV_PG1)
	Adapter Type: SR-IOV passthrough
	Physical function: select pci for the portX of the card under the test
DPDK Settings on Guest OS	Compile DPDK using:
	meson <build> -Dexamples=l3fwd ; ninja -C <build></build></build>
	During testing, I3fwd was given real-time scheduling priority.
L3fwd settings on Guest OS	Updated values /I3fwd/I3fwd.h:
	#define RTE_TEST_RX_DESC_DEFAULT 2048

Item	Description
	#define RTE_TEST_TX_DESC_DEFAULT 2048 #define MAX_PKT_BURST 64
Command Line on Guest OS	./build/examples/dpdk-l3fwd -c 0x7f8000 -n 4 -a 13:00.0,mprq_en=1,rxqs_min_mprq=1 -a 1b:00.0,mprq_en=1,rxqs_min_mprq=1socket-mem=8192p 0x3 -P config='(0,0,22),(0,1,21),(0,2,20),(0,3,19),(1,0,18),(1,1,17),(1,2,16),(1,3,15)'eth- dest=0,00:52:11:22:33:10eth-dest=1,00:52:11:22:33:20
Other optimizations on Guest OS	<ul> <li>a) Flow Control OFF: "ethtool -A \$netdev rx off tx off"</li> <li>b) Memory optimizations: "sysctl -w vm.zone_reclaim_mode=0"; "sysctl -w vm.swappiness=0"</li> <li>c) Move all IRQs to far NUMA node: "IRQBALANCE_BANNED_CPUS=\$LOCAL_NUMA_CPUMAP irqbalanceoneshot"</li> <li>d) Disable irqbalance: "systemctl stop irqbalance"</li> <li>e) Disable Linux realtime throttling: echo -1 &gt; /proc/sys/kernel/sched rt runtime us</li> </ul>

		0 1	0
Frame Size (Bytes)	Frame Rate (Mpps)	Line Rate [50G] (Mpps)	% Line Rate
64	74.4	74.4	100.00
128	42.23	42.23	100.00
256	22.62	22.64	100.00
512	11.75	11.75	100.00
1024	5.99	5.99	100.00
1280	4.81	4.81	100.00
1518	4.06	4.06	100.00

Table 20: Test#6 Results – Mellanox ConnectX-5 25GbE Throughput at Zero Packet Loss using ESXi SR-IOV





# 9 Test#7 Mellanox ConnectX-5 Ex 100GbE Throughput at Zero Packet Loss (1x 100GbE) using SR-IOV over KVM Hypervisor

#### Table 21: Test #7 Setup

Item	Description
Test #7	Mellanox ConnectX-5 Ex 100GbE Throughput at zero packet loss using SR-IOV over KVM Hypervisor
Server	HPE ProLiant DL380 Gen10
СРИ	Intel(R) Xeon(R) Platinum 8168 CPU @ 2.70GHz 24 CPU cores * 2 NUMA nodes
RAM	384GB: 6 * 32GB DIMMs * 2 NUMA nodes @ 2666MHz
BIOS	U30 rev. 1.36 (02/15/2018)
NIC	One MCX516A-CDAT ConnectX-5 Ex network interface card 100GbE dual-port QSFP28; PCIe3.0/PCIe4 x16; ROHS R6
Hypervisor	Ubuntu 20.04.2 LTS (Focal Fossa) QEMU emulator version 4.2.1 (Debian 1:4.2-3ubuntu6.11)
Hypervisor Kernel Version	5.4.0-65-generic.x86_64
Hypervisor Mellanox Driver	MLNX_OFED_LINUX-5.5-1.0.3.2
Guest Operating System	Red Hat Enterprise Linux Server release 7.7 (Maipo)
Guest Kernel Version	3.10.0-1062.el7.x86_64
Guest GCC version	4.8.5 20150623 (Red Hat 4.8.5-28) (GCC)
Guest Mellanox OFED driver version	MLNX_OFED_LINUX-5.5-1.0.3.2
Mellanox NIC firmware version	16.32.1010
DPDK version	22.03
Test Configuration	<ul> <li>1 NIC, 1 port over 1 VF (SR-IOV); VF has 12 queues assigned to it, 1 queue per logical core for a total of 12 logical cores.</li> <li>Each physical port receives a stream of 8192 IP flows from the IXIA directed to VF</li> </ul>
Mellanox NIC firmware version DPDK version Test Configuration	<ul> <li>16.32.1010</li> <li>22.03</li> <li>1 NIC, 1 port over 1 VF (SR-IOV); VF has 12 queues assigned to it, 1 queue per logic core for a total of 12 logical cores.</li> <li>Each physical port receives a stream of 8192 IP flows from the IXIA directed to VF assigned to Guest OS.</li> </ul>

The Device Under Test (DUT) is made up of the HPE server and the Mellanox ConnectX-5 Ex NIC with a dual- port (only first port used in this test) running Red Hat Enterprise Linux Server with qemu-KVM managed via libvirt, Guest OS running DPDK is based on Red Hat Enterprise Linux Server as well. The DUT is connected to the IXIA packet generator which generates traffic towards the ConnectX-5 Ex NIC. The ConnectX-5 Ex data traffic is passed through a virtual function (VF/SR-IOV) to DPDK running on the Guest OS, to the test application **I3fwd** and is redirected to the opposite direction on the same port. IXIA measures throughput and packet loss. Figure 13: Test #7 Setup – Mellanox ConnectX-5 Ex 100GbE connected to IXIA using KVM SR-IOV



## 9.1 Test Settings

Table 22: Test #7 Settings

Item	Description
BIOS	<ol> <li>Workload Profile = "Low Latency";</li> <li>Jitter Control = Manual, 3400. (Setting turbo boost frequency to 3.4 GHz)</li> <li>Change "Workload Profile" to "Custom"</li> <li>Change VT-x, VT-d and SR-IOV from "Disabled" to "Enabled".</li> <li>See "Configuring and tuning HPE ProLiant Servers for low-latency applications":</li> </ol>
	hpe.com > Search "DL380 gen10 low latency"
Hypervisor BOOT Settings	isolcpus=24-47 intel_idle.max_cstate=0 processor.max_cstate=0 nohz_full=24-47 rcu_nocbs=24-47 intel_pstate=disable default_hugepagesz=1G hugepagesz=1G hugepages=70 audit=0 nosoftlockup intel_iommu=on iommu=pt rcu_nocb_poll
Hypervisor settings	<ol> <li>1) Enable SRIOV via NIC configuration tool: (requires installation of mft-tools) mlxconfig -d /dev/mst/mt4121_pciconf1 set NUM_OF_VFS=1 SRIOV_EN=1 CQE_COMPRESSION=1 echo 1 &gt; /sys/class/net/<i>ens6f0</i>/device/sriov_numvfs</li> <li>2) Assign VF</li> </ol>
	HCA_netintf= <i>ens6f0</i> #assign a VF to the DUT device
	VF_PCI_address="0000:at:00.2" #VF PCI address
	echo \$VF_PCI_address > /sys/bus/pci/drivers/mlx5_core/unbind
	modprobe vfio-pci
	echo "\$(cat /sys/bus/pci/devices/\$VF_PCI_address/vendor) \$(cat /sys/bus/pci/devices/\$VF_PCI_address/device)" > /sys/bus/pci/drivers/vfio-pci/new_id

Item	Description
	# Now the VF may be assigned to Guest (passthrough) with libvirt virt-manager.
	3) Setting VF MAC - use the command below (find out the vf-index from "ip link show"), ip link set < <pf interface="" nic="">&gt; <vf index=""> mac <mac address=""> : (mac is random)</mac></vf></pf>
	ip link set \$HCA_netintf vf 0 mac 00:52:11:22:33:42
	4) VM tuning: vcpupin and memory backing from hugepages:
	To persistently configure vcpu pinning and memory backing, add the below config to the VM's XML config before starting the VM. Add the following two elements to the XML: <cputune> and <memorybacking> and also increase the number of cpus and memory: virsh edit <vmid> (to get vmID use - virsh listall)</vmid></memorybacking></cputune>
	Example xml configuration: (change "nodeset" and "cpuset" attributes to suit the local NUMA node in your setup)
	<domain id="1" type="kvm"></domain>
	<name>perf-dpdk-01-005-RH-7.4</name>
	<uuid>06f283fc-fd76-4411-8b6a-72fe94f50376</uuid>
	<memory unit="KiB">33554432</memory>
	<currentmemory unit="KiB">33554432</currentmemory>
	<memorybacking></memorybacking>
	<hugepages></hugepages>
	<page nodeset="0" size="1048576" unit="KiB"></page>
	<nosharepages></nosharepages>
	<locked></locked>
	<vcpu placement="static">23</vcpu>
	<cputune></cputune>
	<vcpupin cpuset="24" vcpu="0"></vcpupin>
	<vcpupin cpuset="25" vcpu="1"></vcpupin>
	<vcpupin cpuset="26" vcpu="2"></vcpupin>
	<vcpupin cpuset="27" vcpu="3"></vcpupin>
	<vcpupin cpuset="28" vcpu="4"></vcpupin>
	<vcpupin cpuset="29" vcpu="5"></vcpupin>
	<vcpupin cpuset="30" vcpu="6"></vcpupin>
	<vcpupin cpuset="31/" vcpu="/"></vcpupin>
	<vcpupin cpuset="32" vcpu="8"></vcpupin>
	<pre><vcpupin cpuset="30" vcpu="12"></vcpupin> <vcpupin cpuset="37" vcpu="13"></vcpupin></pre>
	<pre><vcpupin cpuset="38" vcpu="14"></vcpupin></pre>
	<vcpupin cpuset="39" vcpu="15"></vcpupin>
	<vcpupin cpuset="40" vcpu="16"></vcpupin>
	<vcpupin cpuset="41" vcpu="17"></vcpupin>
	<vcpupin cpuset="42" vcpu="18"></vcpupin>

Item	Description
	<vcpupin cpuset="43" vcpu="19"></vcpupin> <vcpupin cpuset="44" vcpu="20"></vcpupin> <vcpupin cpuset="45" vcpu="21"></vcpupin> <vcpupin cpuset="46" vcpu="22"></vcpupin> 
Other optimizations on Hypervisor	<ul> <li>a) Flow Control OFF: "ethtool -A \$netdev rx off tx off"</li> <li>b) Memory optimizations: "sysctl -w vm.zone_reclaim_mode=0"; "sysctl -w vm.swappiness=0"</li> <li>c) Move all IRQs to far NUMA node: "IRQBALANCE_BANNED_CPUS=\$LOCAL_NUMA_CPUMAP irqbalanceoneshot"</li> <li>d) Disable irqbalance: "systemctl stop irqbalance"</li> <li>e) Change PCI MaxReadReq to 1024B for each port of each NIC: Run "setpci -s \$PORT_PCI_ADDRESS 68.w", it will return 4 digits ABCD&gt; Run "setpci -s \$PORT_PCI_ADDRESS 68.w=3BCD"</li> <li>f) Disable Linux realtime throttling: echo -1 &gt; /proc/sys/kernel/sched_rt_runtime_us</li> </ul>
Guest BOOT Settings	isolcpus=0-22 intel_idle.max_cstate=0 processor.max_cstate=0 intel_pstate=disable idle=poll nohz_full=0-22 rcu_nocbs=0-22 rcu_nocb_poll default_hugepagesz=1G hugepagesz=1G hugepages=16 nosoftlockup
Other optimizations on Guest OS	<ul> <li>a) Flow Control OFF: "ethtool -A \$netdev rx off tx off"</li> <li>b) Memory optimizations: "sysctl -w vm.zone_reclaim_mode=0"; "sysctl -w vm.swappiness=0"</li> <li>c) Move all IRQs to far NUMA node: "IRQBALANCE_BANNED_CPUS=\$LOCAL_NUMA_CPUMAP irqbalanceoneshot"</li> <li>d) Disable irqbalance: "systemctl stop irqbalance"</li> <li>e) Disable Linux realtime throttling: echo -1 &gt; /proc/sys/kernel/sched_rt_runtime_us</li> </ul>
DPDK Settings on Guest OS	Compile DPDK using: meson <build> -Dexamples=l3fwd ; ninja -C <build> During testing, l3fwd was given real-time scheduling priority.</build></build>
L3fwd settings on Guest OS	Updated values /l3fwd/l3fwd.h: #define RTE_TEST_RX_DESC_DEFAULT 2048 #define RTE_TEST_TX_DESC_DEFAULT 2048 #define MAX_PKT_BURST 64
Command Line on Guest OS	<pre>./build/examples/dpdk-l3fwd -c 0x3ffc00 -n 4 -a 00:07.0,mprq_en=1,rxqs_min_mprq=1,mprq_log_stride_num=8socket-mem=8192p 0x1 -P config='(0,0,21),(0,1,20),(0,2,19),(0,3,18),(0,4,17),(0,5,16),(0,6,15),(0,7,14),(0,8,13),(0,9,12) ),(0,10,11),(0,11,10)'eth-dest=0,00:52:11:22:33:10</pre>

Frame Size (Bytes)	Frame Rate (Mpps)	Line Rate [100G] (Mpps)	% Line Rate
64	148.66	148.81	99.91
128	84.46	84.46	100
256	45.29	45.29	100
512	23.50	23.50	100
1024	11.97	11.97	100
1280	9.62	9.62	100
1518	8.13	8.13	100

Table 23: Test #7 Results – Mellanox ConnectX-5 Ex 100GbE Throughput at Zero Packet Loss using KVM SR-IOV





# 10 Test#8 Mellanox ConnectX-6Dx 25GbE Throughput at Zero Packet Loss (2x 25GbE)

#### Table 24: Test #8 Setup

Item	Description
Test #8	Mellanox ConnectX-6Dx 25GbE Dual-Port Throughput at zero packet loss
Server	HPE ProLiant DL380 Gen10
CPU	Intel(R) Xeon(R) Platinum 8168 CPU @ 2.70GHz 24 CPU cores * 2 NUMA nodes
RAM	384GB: 6 * 32GB DIMMs * 2 NUMA nodes @ 2666MHz
BIOS	U30 rev. 1.36 (02/15/2018)
NIC	One MCX623102AN-ADAT ConnectX-6 Dx EN adapter card; 25GbE; Dual-port SFP28; PCIe 4.0/3.0 x16
Operating System	Ubuntu 20.04.2 LTS (Focal Fossa)
Kernel Version	5.4.0-65-generic.x86_64
GCC version	gcc (Ubuntu 9.3.0-17ubuntu1~20.04) 9.3.0
Mellanox NIC firmware version	22.32.1010
Mellanox OFED driver version	MLNX_OFED_LINUX-5.5-1.0.3.2
DPDK version	22.03
Test Configuration	1 NIC, 2 ports; Each port receives a stream of 8192 IP flows from the IXIA
	1 queue assigned per logical core with a total of 8 logical cores

The Device Under Test (DUT) is made up of the HPE server and the Mellanox ConnectX-5 Dual-Port NIC. The DUT is connected to the IXIA packet generator which generates traffic towards the ConnectX-5 NIC. The ConnectX-5 data traffic is passed through DPDK to the test application **I3fwd** and is redirected to the opposite

direction on the same port. IXIA measures throughput and packet loss.

Figure 15: Test #8 Setup – Mellanox ConnectX-6 Dx 25GbE Dual-Port connected to IXIA



#### Table 25: Test #8 Settings

Item	Description
BIOS	<ol> <li>Workload Profile = "Low Latency";</li> <li>Jitter Control = Manual, 3400. (Setting turbo boost frequency to 3.4 GHz)</li> <li>See "Configuring and tuning HPE ProLiant Servers for low-latency applications": hpe.com &gt; Search "DL380 gen10 low latency"</li> </ol>
BOOT Settings	isolcpus=0-23 intel_idle.max_cstate=0 processor.max_cstate=0 intel_pstate=disable nohz_full=0-23 rcu_nocbs=0-23 rcu_nocb_poll default_hugepagesz=1G hugepagesz=1G hugepages=64 audit=0 nosoftlockup idle=poll
DPDK Settings	Compile DPDK using: meson <build> -Dexamples=I3fwd ; ninja -C <build> During testing, I3fwd was given real-time scheduling priority.</build></build>
L3fwd settings	Updated values /l3fwd/l3fwd.h: #define RTE_TEST_RX_DESC_DEFAULT 4096 #define RTE_TEST_TX_DESC_DEFAULT 4096 #define MAX_PKT_BURST 64
Command Line	./build/examples/dpdk-l3fwd -c 0xff0000 -n 4 -a 37:00.0,mprq_en=1,rxqs_min_mprq=1 -a 37:00.1,mprq_en=1,rxqs_min_mprq=1socket-mem=8192p 0x3 -P config='(0,0,23),(0,1,22),(0,2,21),(0,3,20),(1,0,19),(1,1,18),(1,2,17),(1,3,16)'eth- dest=0,00:52:11:22:33:10eth-dest=1,00:52:11:22:33:20
Other optimizations	<ul> <li>a) Flow Control OFF: "ethtool -A \$netdev rx off tx off"</li> <li>b) Memory optimizations: "sysctl -w vm.zone_reclaim_mode=0"; "sysctl -w vm.swappiness=0"</li> <li>c) Move all IRQs to far NUMA node: "IRQBALANCE_BANNED_CPUS=\$LOCAL_NUMA_CPUMAP irqbalanceoneshot"</li> <li>d) Disable irqbalance: "systemctl stop irqbalance"</li> <li>e) Change PCI MaxReadReq to 1024B for each port of each NIC: Run "setpci -s \$PORT_PCI_ADDRESS 68.w", it will return 4 digits ABCD&gt; Run "setpci -s \$PORT_PCI_ADDRESS 68.w=3936"</li> <li>f) Set CQE COMPRESSION to "AGGRESSIVE": mlxconfig -d \$PORT_PCI_ADDRESS set CQE_COMPRESSION=1</li> <li>g) Disable Linux realtime throttling: echo -1 &gt; /proc/sys/kernel/sched_rt_runtime_us</li> </ul>

Frame Size (Bytes)	Frame Rate (Mpps)	Line Rate [50G] (Mpps)	% Line Rate
64	74.40	74.40	100.00
128	42.23	42.23	100.00
256	22.64	22.64	100.00
512	11.75	11.75	100.00
1024	5.99	5.99	100.00
1280	4.81	4.81	100.00
1518	4.06	4.06	100.00

Table 26: Test #8 Results – Mellanox ConnectX-6Dx 25GbE Dual-Port Throughput at Zero Packet Loss





# 11 Test#9 Mellanox ConnectX-6 Dx 100GbE PCIe Gen4 Throughput at Zero Packet Loss (1x 100GbE)

Table 27: Test #9 Setup

Item	Description
Test #9	Mellanox ConnectX-6 Dx 100GbE Dual-Port PCIe Gen 4 Throughput at zero packet loss
Server	HPE ProLiant DL380 Gen10 Plus
CPU	Intel(R) Xeon(R) Platinum 8380 CPU @ 2.30GHz 40 CPU cores * 2 NUMA nodes
RAM	512GB: 32 * 16GB DIMMs * 2 NUMA nodes @ 3200MHz
BIOS	BIOS Revision: 1.42
NIC	One MCX623106AN-CDAT ConnectX-6 Dx EN adapter card; 100GbE; Dual-port QSFP56; PCIe 4.0/3.0 x16;
Operating System	Ubuntu 20.04.2 LTS (Focal Fossa)
Kernel Version	5.4.0-90-generic.x86_64
GCC version	gcc (Ubuntu 9.3.0-17ubuntu1~20.04) 9.3.0
Mellanox NIC firmware version	22.32.1010
Mellanox OFED driver version	MLNX_OFED_LINUX-5.5-1.0.3.2
DPDK version	22.03
Test Configuration	1 NIC, 1 port used on NIC; Port has 12 queues assigned to it, 1 queue per logical core for a total of 12 logical cores.
	Each port receives a stream of 8192 IP flows from the IXIA

The Device Under Test (DUT) is made up of the HPE server and the Mellanox ConnectX-6 Dx Dual-Port NIC (only the first port is used in this test). The DUT is connected to the IXIA packet generator which generates traffic towards the ConnectX-6Dx NIC.

The ConnectX-6Dx data traffic is passed through DPDK to the test application **I3fwd** and is redirected to the opposite direction on the same port. IXIA measures throughput and packet loss.



#### Table 28: Test #9 Settings

Item	Description
BIOS	Select Workload Profile = "Low Latency";
	hpe.com > Search "DL380 gen10 plus low latency"
BOOT Settings	ro isolcpus=0-39 nohz_full=0-39 rcu_nocbs=0-39 intel_iommu=on iommu=pt default_hugepagesz=1G hugepagesz=1G hugepages=80 intel_idle.max_cstate=0 processor.max_cstate=0 intel_pstate=disable rcu_nocb_poll audit=0
DPDK Settings	Compile DPDK using: meson <build> -Dexamples=I3fwd ; ninja -C <build> During testing, I3fwd was given real-time scheduling priority.</build></build>
L3fwd settings	Updated values /I3fwd/I3fwd.h: #define RTE_TEST_RX_DESC_DEFAULT 4096 #define RTE_TEST_TX_DESC_DEFAULT 4096 #define MAX_PKT_BURST 64
Command Line	./build/examples/dpdk-l3fwd -c 0xfff000000000 -n 4 -a 0000:af:00.0,mprq_en=1,mprq_log_stride_num=8socket-mem=0,8192p 0x1 -P config='(0,0,47),(0,1,46),(0,2,45),(0,3,44),(0,4,43),(0,5,42),(0,6,41),(0,7,40),(0,8,39),(0,9,38),(0,10,37) ,(0,11,36)'eth-dest=0,00:52:11:22:33:10
Other optimizations	a) Flow Control OFF: "ethtool -A \$netdev rx off tx off"
	b) Memory optimizations: "sysctl -w vm.zone_reclaim_mode=0"; "sysctl -w vm.swappiness=0"
	c) Move all IRQs to far NUMA node: "IRQBALANCE_BANNED_CPUS=\$LOCAL_NUMA_CPUMAP irqbalanceoneshot"
	d) Disable irqbalance: "systemctl stop irqbalance"
	e) Change PCI MaxReadReq to 1024B for each port of each NIC:
	Run "setpci -s \$PORT_PCI_ADDRESS 68.w", it will return 4 digits ABCD>
	Run "setpci -s \$PORT_PCI_ADDRESS 68.w=3BCD"
	f) Set CQE COMPRESSION to "AGGRESSIVE": mlxconfig -d \$PORT_PCI_ADDRESS set CQE_COMPRESSION=1
	g) Disable Linux realtime throttling: echo -1 > /proc/sys/kernel/sched_rt_runtime_us

Frame Size (Bytes)	Frame Rate (Mpps)	Line Rate [100G] (Mpps)	% Line Rate
64	148.81	148.81	100.00
128	84.46	84.46	100.00
256	45.29	45.29	100.00
512	23.50	23.50	100.00
1024	11.97	11.97	100.00
1280	9.62	9.62	100.00
1518	8.13	8.13	100.00

Table 29: Test #9 Results – Mellanox ConnectX-6 Dx 100GbE Throughput at Zero Packet Loss





# 12 Test#10 Mellanox ConnectX-6Dx 100GbE PCIe Gen4 Single Core Performance (2x 100GbE)

#### Table 30: Test #10 Setup

Item	Description
Test #10	Mellanox ConnectX-6Dx 100GbE PCI Gen4 Single Core Performance
Server	HPE ProLiant DL380 Gen10 Plus
CPU	Intel(R) Xeon(R) Platinum 8380 CPU @ 2.30GHz 40 CPU cores * 2 NUMA nodes
RAM	512GB: 32 * 16GB DIMMs * 2 NUMA nodes @ 3200MHz
BIOS	BIOS Revision: 1.42
NIC	Two MCX623106AN-CDAT ConnectX-6 Dx EN adapter cards; 100GbE; Dual-port QSFP56; PCIe 4.0/3.0 x16;
Operating System	Ubuntu 20.04.2 LTS (Focal Fossa)
Kernel Version	5.4.0-90-generic.x86_64
GCC version	gcc (Ubuntu 9.3.0-17ubuntu1~20.04) 9.3.0
Mellanox NIC firmware version	22.32.1010
Mellanox OFED driver version	MLNX_OFED_LINUX-5.5-1.0.3.2
DPDK version	21.11
Test Configuration	2 NICs; 1 port used on each.
	Each port receives a stream of 8192 UDP flows from the IXIA
	Each port has 1 queue assigned, a total of two queues for two ports, and both queues are assigned to the same single logical core.

The Device Under Test (DUT) is made up of the HPE server and two Mellanox ConnectX-6 Dx 100GbE NICs utilizing one port each. The DUT is connected to the IXIA packet generator which generates traffic towards the first port of both ConnectX-6 Dx 100GbE NICs.

The ConnectX-6 Dx 100GbE data traffic is passed through DPDK to the test application **testpmd** and is redirected to the opposite direction on the opposing NIC's port. IXIA measures throughput and packet loss.



## Table 31: Test #10 Settings

Item	Description
BIOS	Select Workload Profile = "Low Latency"; See "Configuring and tuning HPE ProLiant Servers for low-latency applications": hpe.com > Search "DL380 gen10 plus low latency"
BOOT Settings	ro isolcpus=0-39 nohz_full=0-39 rcu_nocbs=0-39 intel_iommu=on iommu=pt default_hugepagesz=1G hugepagesz=1G hugepages=80 intel_idle.max_cstate=0 processor.max_cstate=0 intel_pstate=disable rcu_nocb_poll audit=0
DPDK Settings	Compile DPDK using: meson <build> ; ninja -C <build> During testing, testpmd was given real-time scheduling priority.</build></build>
Command Line	./build/app/dpdk-testpmd -c 0xc000000000 -n 4 -a 0000:2b:00.1 -a 0000:0f:00.1socket- mem=8192,0port-numa-config=0,0,1,0socket-num=0burst=64txd=1024rxd=1024 mbcache=512rxq=1txq=1nb-cores=1 -i -arss-udprecord-core-cyclesrecord-burst-stats
Other optimizations	<ul> <li>a) Flow Control OFF: "ethtool -A \$netdev rx off tx off"</li> <li>b) Memory optimizations: "sysctl -w vm.zone_reclaim_mode=0"; "sysctl -w vm.swappiness=0"</li> <li>c) Move all IRQs to far NUMA node: "IRQBALANCE_BANNED_CPUS=\$LOCAL_NUMA_CPUMAP irqbalanceoneshot"</li> <li>d) Disable irqbalance: "systemctl stop irqbalance"</li> <li>e) Change PCI MaxReadReq to 1024B for each port of each NIC: Run "setpci -s \$PORT_PCI_ADDRESS 68.w", it will return 4 digits ABCD&gt; Run "setpci -s \$PORT_PCI_ADDRESS 68.w=3BCD"</li> <li>f) Set CQE COMPRESSION to "AGGRESSIVE": mlxconfig -d \$PORT_PCI_ADDRESS set CQE_COMPRESSION=1</li> <li>g) Disable Linux realtime throttling: echo -1 &gt; /proc/sys/kernel/sched_rt_runtime_us</li> </ul>

Frame Size (Bytes)	Frame Rate (Mpps)	Line Rate [200G] (Mpps)	Line Rate [100G] (Mpps)	Throughput (Gbps)	CPU Cycles per packet NOTE: Lower is Better
64	79.27	297.62	148.81	40.585	26
128	77.83	168.92	84.46	79.697	25
256	74.44	90.58	45.29	152.463	23
512	46.99	46.99	23.50	192.472	23
1024	23.95	23.95	11.97	196.164	23
1280	19.23	19.23	9.62	196.918	23
1518	16.25	16.25	8.13	197.395	24

Table 32: Test #10 Results – Mellanox ConnectX-6 Dx 100GbE Single Core Performance

Figure 20: Test #10 Results – Mellanox ConnectX-6Dx 100GbE Single Core Performance



# 13 Test#11 Mellanox ConnectX-6 Dx 100GbE PCIe Gen4 Throughput at Zero Packet Loss (2x 100GbE)

Table 33: Test #11 Setup

Item	Description
Test #11	Mellanox ConnectX-6 Dx 100GbE Dual-Port PCIe Gen 4 Throughput at zero packet loss
Server	HPE ProLiant DL380 Gen10 Plus
CPU	Intel(R) Xeon(R) Platinum 8380 CPU @ 2.30GHz 40 CPU cores * 2 NUMA nodes
RAM	512GB: 32 * 16GB DIMMs * 2 NUMA nodes @ 3200MHz
BIOS	BIOS Revision: 1.42
NIC	One MCX623106AN-CDAT ConnectX-6 Dx EN adapter card; 100GbE; Dual-port QSFP56; PCIe 4.0 x16 ; No Crypto
Operating System	Ubuntu 20.04.2 LTS (Focal Fossa)
Kernel Version	5.4.0-90-generic.x86_64
GCC version	gcc (Ubuntu 9.3.0-17ubuntu1~20.04) 9.3.0
Mellanox NIC firmware version	22.32.1010
Mellanox OFED driver version	MLNX_OFED_LINUX-5.5-1.0.3.2
DPDK version	22.03
Test Configuration	1 NIC, 2 port used on NIC; each port has 8 queues assigned to it, 1 queue per logical core for a total of 16 logical cores. Each port receives a stream of 8192 IP flows from the IXIA

The Device Under Test (DUT) is made up of the Dell server and the Mellanox ConnectX-6 Dx Dual-Port NIC (both ports are used in this test). The DUT is connected to the IXIA packet generator which generates traffic towards the ConnectX-6 Dx NIC ports.

The ConnectX-6 Dx data traffic is passed via PCIe Gen 4 bus through DPDK to the test application **I3fwd** and is redirected to the opposite direction on the same port. IXIA measures throughput and packet loss.



#### Table 34: Test #11 Settings

Item	Description
BIOS	Select Workload Profile = "Low Latency";
	See "Configuring and tuning HPE ProLiant Servers for low-latency applications":
	hpe.com > Search "DL380 gen10 plus low latency"
BOOT Settings	ro isolcpus=0-39 nohz_full=0-39 rcu_nocbs=0-39 intel_iommu=on iommu=pt default_hugepagesz=1G hugepagesz=1G hugepages=80 intel_idle.max_cstate=0 processor.max_cstate=0 intel_pstate=disable rcu_nocb_poll audit=0
DPDK Settings	Compile DPDK using: meson <build> -Dexamples=l3fwd &amp;&amp; ninja -C <build></build></build>
	During testing, I3fwd was given real-time scheduling priority.
L3fwd settings	Updated values /I3fwd/I3fwd.h:
	#define RTE_TEST_RX_DESC_DEFAULT 4096
	#define RTE_TEST_TX_DESC_DEFAULT 4096
	#define MAX_PKT_BURST 64
Command Line	./build/examples/dpdk-l3fwd -c 0xfff0000000 -n 4socket-mem=4096 -a
	0000:2b:00.0,mprq_en=1,rxqs_min_mprq=1,mprq_log_stride_num=9,txq_inline_mpw=128,rxq_pkt_ pad_en=1 -a
	pad_en=1p 0x3 -P
	config='(0,0,39),(0,1,38),(0,2,37),(0,3,36),(0,4,35),(0,5,34),(0,6,33),(0,7,32),(1,0,31),(1,1,30),(1,2,29),(1,3,28),(1,4,27),(1,5,26),(1,6,25),(1,7,24)'eth-dest=0,00:52:11:22:33:10eth-dest=1,00:52:11:22:33:20
Other optimizations	a) Flow Control OFF: "ethtool -A \$netdev rx off tx off" (for both ports)
	b) Memory optimizations: "sysctl -w vm.zone_reclaim_mode=0"; "sysctl -w vm.swappiness=0"
	c) Move all IRQs to far NUMA node: "IRQBALANCE_BANNED_CPUS=\$LOCAL_NUMA_CPUMAP irqbalanceoneshot"
	d) Disable irqbalance: "systemctl stop irqbalance"
	e) Change PCI MaxReadReq to 4096B for each port of each NIC:
	Run "setpci -s \$PORT_PCI_ADDRESS 68.w", it will return 4 digits ABCD>
	Run "setpci -s \$PORT_PCI_ADDRESS 68.w= 5BCD "
	f) Set CQE COMPRESSION to "AGGRESSIVE": mlxconfig_d \$PORT_PCL_ADDRESS_cet_COE_COMPRESSION=1
	g) Set PCI write ordering: mlxconfig -d \$PORT_PCI_ADDRESS set PCI_WR_ORDERING=1
	<ul> <li>h) Disable Linux real-time throttling: echo -1 &gt; /proc/sys/kernel/sched_rt_runtime_us</li> <li>i) Disable auto neg for both ports: ethtool -s \$PORT_PCI_ADDRESS autoneg off speed 100000</li> </ul>

Table 35: Test #11 Results -	- Mellanox ConnectX-6 Dx	100GbE Dual-Port PCIe Gen4	4 Zero Packet Loss Throughput
------------------------------	--------------------------	----------------------------	-------------------------------

Frame Size (Bytes)	Frame Rate (Mpps)	Line Rate [200G] (Mpps)	Line Rate [100G] (Mpps)	% Line Rate
64	213.98	297.62	148.81	71.90
128	142.87	168.92	84.46	84.59
256	71.22	90.58	45.29	78.64
512	46.99	46.99	23.50	100.00
1024	23.95	23.95	11.97	100.00
1280	19.23	19.23	9.62	100.00
1518	16.25	16.25	8.13	100.00

Figure 22: Test #11 Results – Mellanox ConnectX-6 Dx 100GbE Dual-Port PCIe Gen4 Throughput at Zero Packet Loss



# 14 Test#12 Mellanox ConnectX-6 Dx 100GbE Throughput at Zero Packet Loss (1x 100GbE) using SR-IOV over KVM Hypervisor

#### Table 36 - Test #12 Setup

Item	Description
Test #12	Mellanox ConnectX-6 Dx 100GbE Throughput at zero packet loss using SR-IOV over KVM Hypervisor
Server	HPE ProLiant DL380 Gen10
СРИ	Intel(R) Xeon(R) Platinum 8168 CPU @ 2.70GHz 24 CPU cores * 2 NUMA nodes
RAM	384GB: 6 * 32GB DIMMs * 2 NUMA nodes @ 2666MHz
BIOS	U30 rev. 1.36 (02/15/2018)
NIC	One MCX623106AN-CDAT ConnectX-6 Dx EN adapter card; 100GbE; Dual-port QSFP56; PCIe 4.0/3.0 x16;
Hypervisor	Ubuntu 20.04.2 LTS (Focal Fossa) QEMU emulator version 4.2.1 (Debian 1:4.2-3ubuntu6.11)
Hypervisor Kernel Version	5.4.0-65-generic.x86_64
Hypervisor Mellanox Driver	MLNX_OFED_LINUX-5.5-1.0.3.2
Guest Operating System	Red Hat Enterprise Linux Server release 7.7 (Maipo)
Guest Kernel Version	3.10.0-1062.el7.x86_64
Guest GCC version	4.8.5 20150623 (Red Hat 4.8.5-28) (GCC)
Guest Mellanox OFED driver version	MLNX_OFED_LINUX-5.5-1.0.3.2
Mellanox NIC firmware version	22.32.1010
DPDK version	22.03
Test Configuration	1 NIC, 1 port over 1 VF (SR-IOV); VF has 12 queues assigned to it, 1 queue per logical core for a total of 12 logical cores.
	Each physical port receives a stream of 8192 IP flows from the IXIA directed to VF assigned to Guest OS.

The Device Under Test (DUT) is made up of the HPE server and the Mellanox ConnectX-6 Dx NIC with a dual- port (only first port used in this test) running Red Hat Enterprise Linux Server with qemu-KVM managed via libvirt, Guest OS running DPDK is based on Red Hat Enterprise Linux Server as well. The DUT is connected to the IXIA packet generator which generates traffic towards the ConnectX-6 Dx NIC. The ConnectX-6 Dx data traffic is passed through a virtual function (VF/SR-IOV) to DPDK running on the Guest OS, to the test application **I3fwd** and is redirected to the opposite direction on the same port. IXIA measures throughput and packet loss.



#### Table 37 - Test #12 Settings

Item	Description
BIOS	1) Workload Profile = "Low Latency";
	2) Jitter Control = Manual, 3400. (Setting turbo boost frequency to 3.4 GHz)
	3) Change "Workload Profile" to "Custom"
	4) Change VT-x, VT-d and SR-IOV from "Disabled" to "Enabled".
	See "Configuring and tuning HPE ProLiant Servers for low-latency applications":
	hpe.com > Search "DL380 gen10 low latency"
Hypervisor BOOT Settings	isolcpus=24-47 intel_idle.max_cstate=0 processor.max_cstate=0 nohz_full=24-47 rcu_nocbs=24- 47 intel_pstate=disable default_hugepagesz=1G hugepagesz=1G hugepages=70 audit=0 nosoftlockup intel_iommu=on iommu=pt rcu_nocb_poll
Hypervisor settings	1) Enable SRIOV via NIC configuration tool: (requires installation of mft-tools)
	mlxconfig -d /dev/mst/mt4121_pciconf1 set NUM_OF_VFS=1 SRIOV_EN=1 CQE_COMPRESSION=1
	echo 1 > /sys/class/net/ <i>ens5f0</i> /device/sriov_numvfs
	2) Assign VF
	HCA netintf= <i>ens5f0</i> #assign a VF to the DUT device
	VF_PCI_address="0000:af:00.2" #VF PCI address
	echo \$VF_PCI_address > /sys/bus/pci/drivers/mlx5_core/unbind
	modprobe vfio-pci
	echo "\$(cat /sys/bus/pci/devices/\$VF_PCI_address/vendor) \$(cat /sys/bus/pci/devices/\$VF_PCI_address/device)" > /sys/bus/pci/drivers/vfio-pci/new_id
	# Now the VF may be assigned to Guest (passthrough) with libvirt virt-manager.
	3) Setting VF MAC - use the command below (find out the vf-index from "ip link show"), ip link set < <pf interface="" nic="">&gt; <vf index=""> mac <mac address=""> : (mac is random)</mac></vf></pf>
	ip link set \$HCA_netintf vf 0 mac 00:52:11:22:33:42
	4) VM tuning: vcpupin and memory backing from hugepages:
	To persistently configure vcpu pinning and memory backing, add the below config to the VM's XML config before starting the VM. Add the following two elements to the XML: <cputune> and <memorybacking> and also increase the number of cpus and memory: virsh edit <vmid> (to get vmID use - virsh listall)</vmid></memorybacking></cputune>
	Example xml configuration: (change "nodeset" and "cpuset" attributes to suit the local NUMA node in your setup)
	<domain id="1" type="kvm"></domain>
	<name>perf-dpdk-01-005-RH-7.4</name>
	<uuid>06f283fc-fd76-4411-8b6a-72fe94f50376</uuid>
	<memory unit="KiB">33554432</memory>
	<currentmemory unit="KiB">33554432</currentmemory>
	<memorybacking></memorybacking>
	<hugepages></hugepages>
	<page nodeset="0" size="1048576" unit="KiB"></page>
	<nosharepages></nosharepages>

Item	Description
	<locked></locked>
	<vcpu placement="static">23</vcpu>
	<cputune></cputune>
	<vcpupin cpuset="24" vcpu="0"></vcpupin>
	<vcpupin cpuset="25" vcpu="1"></vcpupin>
	<vcpupin cpuset="26" vcpu="2"></vcpupin>
	<vcpupin cpuset="27" vcpu="3"></vcpupin>
	<vcpupin cpuset="28" vcpu="4"></vcpupin>
	<vcpupin cpuset="29" vcpu="5"></vcpupin>
	<vcpupin cpuset="30" vcpu="6"></vcpupin>
	<vcpupin cpuset="31" vcpu="7"></vcpupin>
	<vcpupin cpuset="32" vcpu="8"></vcpupin>
	<vcpupin cpuset="33" vcpu="9"></vcpupin>
	<vcpupin cpuset="34" vcpu="10"></vcpupin>
	<vcpupin cpuset="35" vcpu="11"></vcpupin>
	<vcpupin cpuset="36" vcpu="12"></vcpupin>
	<vcpupin cpuset="37" vcpu="13"></vcpupin>
	<vcpupin cpuset="38" vcpu="14"></vcpupin>
	<vcpupin cpuset="39" vcpu="15"></vcpupin>
	<vcpupin cpuset="40" vcpu="16"></vcpupin>
	<vcpupin cpuset="41" vcpu="17"></vcpupin>
	<vcpupin cpuset="42" vcpu="18"></vcpupin>
	<vcpupin cpuset="43" vcpu="19"></vcpupin>
	<vcpupin cpuset="44" vcpu="20"></vcpupin>
	<vcpupin cpuset="45" vcpu="21"></vcpupin>
	<vcpupin cpuset="46" vcpu="22"></vcpupin>
Other optimizations on	a) Flow Control OFF: "ethtool -A \$netdev rx off tx off"
Hypervisor	b) Memory optimizations: "sysctl -w vm.zone_reclaim_mode=0"; "sysctl -w vm.swappiness=0"
	c) Move all IRQs to far NUMA node: "IRQBALANCE_BANNED_CPUS=\$LOCAL_NUMA_CPUMAP irqbalanceoneshot"
	d) Disable irqbalance: "systemctl stop irqbalance"
	e) Change PCI MaxReadReq to 1024B for each port of each NIC:
	Run "setpci -s \$PORT_PCI_ADDRESS 68.w", it will return 4 digits ABCD>
	Run "setpci -s \$PORT_PCI_ADDRESS 68.w=3BCD"
	f) Disable Linux realtime throttling: echo -1 > /proc/sys/kernel/sched_rt_runtime_us
Guest BOOT Settings	isolcpus=0-22 intel_idle.max_cstate=0 processor.max_cstate=0 intel_pstate=disable idle=poll nohz_full=0-22 rcu_nocbs=0-22 rcu_nocb_poll default_hugepagesz=1G hugepagesz=1G hugepages=16 nosoftlockup
Other optimizations on	a) Flow Control OFF: "ethtool -A \$netdev rx off tx off"
Guest OS	b) Memory optimizations: "sysctl -w vm.zone_reclaim_mode=0"; "sysctl -w vm.swappiness=0"
	c) Move all IRQs to far NUMA node: "IRQBALANCE_BANNED_CPUS=\$LOCAL_NUMA_CPUMAP irqbalanceoneshot"
	d) Disable irqbalance: "systemctl stop irqbalance"
	e) Disable Linux realtime throttling: echo -1 > /proc/sys/kernel/sched_rt_runtime_us

Item	Description
DPDK Settings on Guest OS	Compile DPDK using: meson <build> -Dexamples=I3fwd ; ninja -C <build> During testing, I3fwd was given real-time scheduling priority.</build></build>
L3fwd settings on Guest OS	Updated values /I3fwd/I3fwd.h: #define RTE_TEST_RX_DESC_DEFAULT 2048 #define RTE_TEST_TX_DESC_DEFAULT 2048 #define MAX_PKT_BURST 64
Command Line on Guest OS	./build/examples/dpdk-l3fwd -c 0x3ffc00 -n 4 -a 00:07.0,mprq_en=1,rxqs_min_mprq=1,mprq_log_stride_num=8socket-mem=8192p 0x1 -P  config='(0,0,21),(0,1,20),(0,2,19),(0,3,18),(0,4,17),(0,5,16),(0,6,15),(0,7,14),(0,8,13),(0,9,12),(0,10, 11),(0,11,10)'eth-dest=0,00:52:11:22:33:10

Table 38 - Test #12 Results – Mellanox ConnectX-6 Dx 100GbE Throughput at Zero Packet Loss using KVM SR-IOV

Frame Size (Bytes)	Frame Rate (Mpps)	Line Rate [100G] (Mpps)	% Line Rate
64	148.81	148.81	100.00
128	84.46	84.46	100.00
256	45.29	45.29	100.00
512	23.50	23.50	100.00
1024	11.97	11.97	100.00
1280	9.62	9.62	100.00
1518	8.13	8.13	100.00





# 15 Test#13 Mellanox ConnectX-6 Dx 200GbE PCIe Gen4 Throughput at Zero Packet Loss (1x 200GbE)

#### Table 39 - Test #13 Setup

Item	Description		
Test #13	Mellanox ConnectX-6 Dx 200GbE single-port PCIe Gen4 throughput at zero packet loss		
Server	HPE ProLiant DL380 Gen10 Plus		
CPU	Intel(R) Xeon(R) Platinum 8380 CPU @ 2.30GHz 40 CPU cores * 2 NUMA nodes		
RAM	512GB: 16 * 32GB DIMMs @ 3200MHz		
BIOS	BIOS Revision: 1.42		
NIC	One MCX623105AN-VDAT ConnectX-6 Dx EN adapter card, 200GbE, Single-port QSFP56, PCIe 4.0 x16, No Crypto		
Operating System	Ubuntu 20.04.2 LTS (Focal Fossa)		
Kernel Version	5.4.0-90-generic.x86_64		
GCC version	gcc (Ubuntu 9.3.0-17ubuntu1~20.04) 9.3.0		
Mellanox NIC firmware version	22.32.1010		
Mellanox OFED driver version	MLNX_OFED_LINUX-5.5-1.0.3.2		
DPDK version	22.03		
Test Configuration	1 NIC, 1 port used on NIC; Port has 16 queues assigned to it, 1 queue per logical core for a total of 16 logical cores. Fach port receives a stream of 8192 IP flows from the IXIA		

The Device Under Test (DUT) is made up of the Dell server and the Mellanox ConnectX-6 Dx Single-Port NIC . The DUT is connected to the IXIA packet generator which generates traffic towards the ConnectX-6 Dx NIC port. The ConnectX-6 Dx data traffic is passed via PCIe Gen 4 bus through DPDK to the test application **I3fwd** and is redirected to the opposite direction on the same port. IXIA measures throughput and packet loss.



#### Table 40 - Test #13 Settings

Item	Description
BIOS	Select Workload Profile = "Low Latency";
	See "Configuring and tuning HPE ProLiant Servers for low-latency applications":
	hpe.com > Search "DL380 gen10 plus low latency"
BOOT Settings	ro isolcpus=40-79 nohz_full=40-79 rcu_nocbs=40-79 intel_iommu=on iommu=pt
	processor.max_cstate=0 intel_pstate=disable rcu_nocb_poll audit=0
DPDK Settings	Compile DPDK using:
	meson <build> ; ninja -C <build></build></build>
	During testing, I3fwd was given real-time scheduling priority.
L3fwd settings	Updated values /I3fwd/I3fwd.h:
	#define RTE_TEST_RX_DESC_DEFAULT 4096
	#define RTE_TEST_TX_DESC_DEFAULT 4096
	#define MAX_PKT_BURST 64
Command Line	/build/examples//dpdk-l3fwd -c 0xffff0000000000000000 -n 4socket-mem=0,4096 -a
	0000:a2:00.0,mprq_en=1,rxqs_min_mprq=1,mprq_log_stride_num=9,txq_inline_mpw=128,rxq_pkt
	config='(0,0,79),(0,1,78),(0,2,77),(0,3,76),(0,4,75),(0,5,74),(0,6,73),(0,7,72),(0,8,71),(0,9,70),(0,10,69)
	,(0,11,68),(0,12,67),(0,13,66),(0,14,65),(0,15,64)'eth-dest=0,00:52:11:22:33:10
Other optimizations	a) Flow Control OFF: "ethtool -A \$netdev rx off tx off" (for both ports)
	b) Memory optimizations: "sysctl -w vm.zone_reclaim_mode=0"; "sysctl -w vm.swappiness=0"
	c) Move all IRQs to far NUMA node: "IRQBALANCE_BANNED_CPUS=\$LOCAL_NUMA_CPUMAP irgbalanceoneshot"
	d) Disable irqbalance: "systemctl stop irqbalance"
	e) Change PCI MaxReadReq to 4096B for each port of each NIC:
	Run "setpci -s \$PORT_PCI_ADDRESS 68.w", it will return 4 digits ABCD>
	Run "setpci -s \$PORT_PCI_ADDRESS 68.w= 5BCD "
	f) Set CQE COMPRESSION to "AGGRESSIVE":
	mlxconfig -d \$PORT_PCI_ADDRESS set CQE_COMPRESSION=1
	g) Set PCI write ordering: mixconfig -d \$PORI_PCI_ADDRESS set PCI_WR_ORDERING=1
	i) Disable Linux real-time throttling: echo -1 > /proc/sys/kernel/sched_rt_runtime_us i) Disable auto neg for both ports: ethtool -s \$PORT_PCI_ADDRESS autoneg off speed 200000

Table 41 - Test #13 Results – Mellanox ConnectX-6 Dx 200GbE single port PCIe Gen4 Throughput at Zero Packet Loss

Frame Size (Bytes)	Frame Rate (Mpps)	Line Rate [200G] (Mpps)	Line Rate [100G] (Mpps)	% Line Rate
64	214.18	297.62	148.81	71.96
128	143.23	168.92	84.46	84.79
256	71.23	90.58	45.29	78.64
512	46.95	46.99	23.50	99.9
1024	23.95	23.95	11.97	100
1280	19.23	19.23	9.62	100
1518	16.25	16.25	8.13	100

Figure 26 - Test #13 Results – Mellanox ConnectX-6 Dx 200GbE dual port PCIe Gen4 Throughput at Zero Packet Loss



# 16 Test#14 BlueField-2 25GbE Throughput at Zero Packet Loss (2x 25GbE)

#### Table 42 - Test #14 Setup

Item	Description
Test #14	BlueField-2 25GbE Dual-Port Throughput at zero packet loss
Server	HPE ProLiant DL380 Gen10
Data Processing Unit (DPU)	One MBF2H332A-AEEOT_A1 BlueField-2 P-Series SmartNIC; 25GbE; Dual-port SFP56; PCle Gen3/4 x8
DPU hosted CPUs	BlueField-2 A1 A72 @2.5GHz , 8 Cores-Processor
DPU RAM	DDR On-board Memory 16GB
DPU BIOS	U30 rev. 1.36 (02/15/2018)
Operating System	BlueField-2,DOCA_v1.2.0_BlueField_OS_Ubuntu_20.04-5.4.0-1022-bluefield-5.5-1.0.3.2-3.8.0.11969-1-aarch64
DPU Kernel Version	5.4.0-1022-bluefield, aarch64
DPU GCC version	gcc (Ubuntu 9.3.0-17ubuntu1~20.04) 9.3.0
Mellanox NIC/DPU firmware version	24.32.1010
Mellanox OFED driver version	MLNX_OFED_LINUX-5.5-1.0.3.2
DPDK version	22.03
Test Configuration	1 NIC/DPU, 2 ports;
	Each port receives a stream of 7500 UDP flows from the IXIA
	1 queue assigned per logical core with a total of 2,4 and 8 logical cores

The Device Under Test (DUT) is made up of the HPE server and one BlueField-2 25GbE DPU utilizing two ports. It is connected to the IXIA packet generator which generates traffic towards both ports of the BlueField-2 25GbE DPU. BlueField-2 25GbE data traffic is passed through DPDK to the test application **testpmd** that is running on the ARM cores (**embedded in the DPU**) and is redirected to the opposite direction using the second port. IXIA measures throughput and packet loss. The test measured the results while using 1,2,4,6 or 7 ARM cores.



#### Table 43 - Test #14 Settings

Item	Description
BIOS	<ol> <li>Workload Profile = "Low Latency";</li> <li>Jitter Control = Manual, 3400. (Setting turbo boost frequency to 3.4 GHz)</li> </ol>
	See "Configuring and tuning HPE ProLiant Servers for low-latency applications": hpe.com > Search "DL380 gen10 low latency"
DPU BOOT Settings	ro crashkernel=auto console=ttyAMA1 console=hvc0 console=ttyAMA0 earlycon=pl011,0x01000000 earlycon=pl011,0x01800000 modprobe.blacklist=mlx5_core,mlx5_ib isolcpus=1-7 nohz_full=1-7 rcu_nocbs=1-7
DPDK Settings	Compile DPDK using: meson <build> ; ninja -C <build></build></build>
Command Lines	1 Core: /build/app/dpdk-testpmd -c 0x5master-lcore=0 -n 4 -w 03:00.0 -w 03:00.1socket-mem=1024 burst=64txq=1rxq=1rxd=1024txd=1024mbcache=512nb-cores=1 -i -arss-udpport- topology=loop
	2 Cores: /build/app/dpdk-testpmd -c 0x15master-lcore=0 -n 4 -w 03:00.0 -w 03:00.1socket-mem=1024 -burst=64txq=2rxq=2rxd=1024txd=1024mbcache=512nb-cores=2 -i -arss-udpport- topology=loop
	4 Cores: /build/app/dpdk-testpmd -c 0xabmaster-lcore=0 -n 4 -w 03:00.0 -w 03:00.1socket-mem=1024 -burst=64txq=4rxq=4rxd=1024txd=1024mbcache=512nb-cores=4 -i -arss-udpport- topology=loop
	6 Cores: /build/app/dpdk-testpmd -c 0x7fmaster-lcore=0 -n 4 -w 03:00.0 -w 03:00.1socket-mem=1024 -burst=64txq=6rxq=6rxd=1024txd=1024mbcache=512nb-cores=6 -i -arss-udpport- topology=loop
Other optimizations	a) Flow Control OFF: "ethtool -A \$netdev rx off tx off"
	b) Memory optimizations: "sysctl -w vm.zone_reclaim_mode=0"; "sysctl -w vm.swappiness=0"
	c) Move all IRQs to far NUMA node: "IRQBALANCE_BANNED_CPUS=\$LOCAL_NUMA_CPUMAP irqbalanceoneshot"
	d) Disable irqbalance: "systemctl stop irqbalance"
	e) Change PCI MaxReadReq to 1024B for each port of each NIC:
	Run "setpci -s \$PORT_PCI_ADDRESS 68.w", it will return 4 digits ABCD>
	Run "setpci -s \$PORT_PCI_ADDRESS 68.w=3900"
	f) Set CQE COMPRESSION to "AGGRESSIVE": mlxconfig -d \$PORT_PCI_ADDRESS set CQE_COMPRESSION=1
	g) Disable Linux realtime throttling: echo -1 > /proc/sys/kernel/sched_rt_runtime_us

Frame Size Line Rate [500 (Bytes) (Mpps)	Line Rate [50G]	Frame Rate (Mpps)				Line rate %
	(Mpps)	1 Core	2 Cores	4 Cores	6 Cores	(6 Cores)
64	74.40	24.57	46.22	73.94	74.40	100.00
128	42.23	23.87	42.18	42.18	42.23	100.00
256	22.64	22.53	22.62	22.64	22.64	100.00
512	11.75	11.75	11.75	11.75	11.75	100.00
1024	5.99	5.99	5.99	5.99	5.99	100.00
1280	4.81	4.81	4.81	4.81	4.81	100.00
1518	4.06	4.06	4.06	4.06	4.06	100.00

Table 44 - Test #14 Results – BlueField-2 25GbE Dual-Port Throughput at Zero Packet Loss

Figure 28 - Test #14 Results – BlueField-2 25GbE Dual-Port Throughput at Zero Packet Loss



# 17 Test#15 Mellanox ConnectX-6 Lx 25GbE Throughput at Zero Packet Loss (2x 25GbE)

#### Table 45 - Test #15 Setup

Item	Description
Test #8	Mellanox ConnectX-6 Lx 25GbE Dual-Port Throughput at zero packet loss
Server	HPE ProLiant DL380 Gen10
CPU	Intel(R) Xeon(R) Platinum 8168 CPU @ 2.70GHz 24 CPU cores * 2 NUMA nodes
RAM	384GB: 6 * 32GB DIMMs * 2 NUMA nodes @ 2666MHz
BIOS	U30 rev. 1.36 (02/15/2018)
NIC	One MCX631102AN-ADAT, ConnectX-6 Lx EN adapter card, 25GbE, Dual-port SFP28, PCIe 4.0 x8, No Crypto
Operating System	Ubuntu 20.04.2 LTS (Focal Fossa)
Kernel Version	5.4.0-65-generic.x86_64
GCC version	gcc (Ubuntu 9.3.0-17ubuntu1~20.04) 9.3.0
Mellanox NIC firmware version	22.32.1010
Mellanox OFED driver version	MLNX_OFED_LINUX-5.5-1.0.3.2
DPDK version	22.03
Test Configuration	1 NIC, 2 ports; Each port receives a stream of 8192 IP flows from the IXIA Each port has 4 queues assigned for a total of 8 queues 1 queue assigned per logical core with a total of 8 logical cores

The Device Under Test (DUT) is made up of the HPE server and the Mellanox ConnectX-6 Lx Dual-Port NIC. The DUT is connected to the IXIA packet generator which generates traffic towards the ConnectX-6 Lx NIC. The ConnectX-6 Lx data traffic is passed through DPDK to the test application I3fwd and is redirected to the opposite direction on the same port. IXIA measures throughput and packet loss.



25GbE

Port 1

Port 2

ConnectX.6LX

Port 1

Port 2

Packet Generator

የየዖ

CPU

DPDK

#### Table 46 - Test #15 Settings

Item	Description
BIOS	<ol> <li>Workload Profile = "Low Latency";</li> <li>Jitter Control = Manual, 3400. (Setting turbo boost frequency to 3.4 GHz)</li> <li>See "Configuring and tuning HPE ProLiant Servers for low-latency applications": hpe.com &gt; Search "DL380 gen10 low latency"</li> </ol>
BOOT Settings	isolcpus=0-23 intel_idle.max_cstate=0 processor.max_cstate=0 intel_pstate=disable nohz_full=0-23 rcu_nocbs=0-23 rcu_nocb_poll default_hugepagesz=1G hugepagesz=1G hugepages=64 audit=0 nosoftlockup idle=poll
DPDK Settings	Compile DPDK using: meson <build> -Dexamples=I3fwd ; ninja -C <build> During testing, I3fwd was given real-time scheduling priority.</build></build>
L3fwd settings	Updated values /l3fwd/l3fwd.h: #define RTE_TEST_RX_DESC_DEFAULT 4096 #define RTE_TEST_TX_DESC_DEFAULT 4096 #define MAX_PKT_BURST 64
Command Line	./build/examples/dpdk-l3fwd -c 0xff0000 -n 4 -a 12:00.0,mprq_en=1,rxqs_min_mprq=1 -a 12:00.1,mprq_en=1,rxqs_min_mprq=1socket-mem=8192p 0x3 -P config='(0,0,23),(0,1,22),(0,2,21),(0,3,20),(1,0,19),(1,1,18),(1,2,17),(1,3,16)'eth- dest=0,00:52:11:22:33:10eth-dest=1,00:52:11:22:33:20
Other optimizations	<ul> <li>a) Flow Control OFF: "ethtool -A \$netdev rx off tx off"</li> <li>b) Memory optimizations: "sysctl -w vm.zone_reclaim_mode=0"; "sysctl -w vm.swappiness=0"</li> <li>c) Move all IRQs to far NUMA node: "IRQBALANCE_BANNED_CPUS=\$LOCAL_NUMA_CPUMAP irqbalanceoneshot"</li> <li>d) Disable irqbalance: "systemctl stop irqbalance"</li> <li>e) Change PCI MaxReadReq to 1024B for each port of each NIC: Run "setpci -s \$PORT_PCI_ADDRESS 68.w", it will return 4 digits ABCD&gt; Run "setpci -s \$PORT_PCI_ADDRESS 68.w=3936"</li> <li>f) Set CQE COMPRESSION to "AGGRESSIVE": mlxconfig -d \$PORT_PCI_ADDRESS set CQE_COMPRESSION=1</li> <li>g) Disable Linux realtime throttling: echo -1 &gt; /proc/sys/kernel/sched_rt_runtime_us</li> </ul>

Frame Size (Bytes)	Frame Rate (Mpps)	Line Rate [50G] (Mpps)	% Line Rate
64	74.40	74.40	100.00
128	42.23	42.23	100.00
256	22.64	22.64	100.00
512	11.75	11.75	100.00
1024	5.99	5.99	100.00
1280	4.81	4.81	100.00
1518	4.06	4.06	100.00

Table 47 - Test #15 Results – Mellanox ConnectX-6 Lx 25GbE Dual-Port Throughput at Zero Packet Loss



