Mellanox NIC's Performance Report with DPDK 20.02

Rev 1.1

Table of Contents

List of Figures

List of Tables

About this Report

The purpose of this report is to provide packet rate performance data for Mellanox ConnectX-4 Lx, ConnectX-5 and ConnectX-5 Ex Network Interface Cards (NICs) 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 family of NICs and is intended as a general reference of achievable performance for the specified DPDK release.

Target Audience

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

Document Revision History

Table 1: Document Revision History

Revision	Date	Description
1.0	05-Apr-2020	Initial report release
1.1	23-Apr-2020	Fixes typo in test results; Updated test description layout and graphics

1 Test Description

1.1 Hardware Components

The following hardware components are used in the test setup:

1. HPE® ProLiant DL380 Gen10 Server
2. Mellanox ConnectX® NIC
3. IXIA® XM12 packet generator

1.2 Zero Packet Loss Test

Zero Packet Loss tests utilize **l3fwd** (http://www.dpdk.org/doc/guides/sample_app_ug/l3_forward.html) as the test application for testing maximum throughput with zero packet loss at various frame sizes based on RFC2544 https://tools.ietf.org/html/rfc2544.

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 9.00EA) is used with the IXIA packet generator.

1.3 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.

1.4 Single Core Performance Test

Single Core performance tests utilize **testpmd** (http://www.dpdk.org/doc/guides/testpmd_app_ug), 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 9.00EA) is used with the IXIA packet generator.

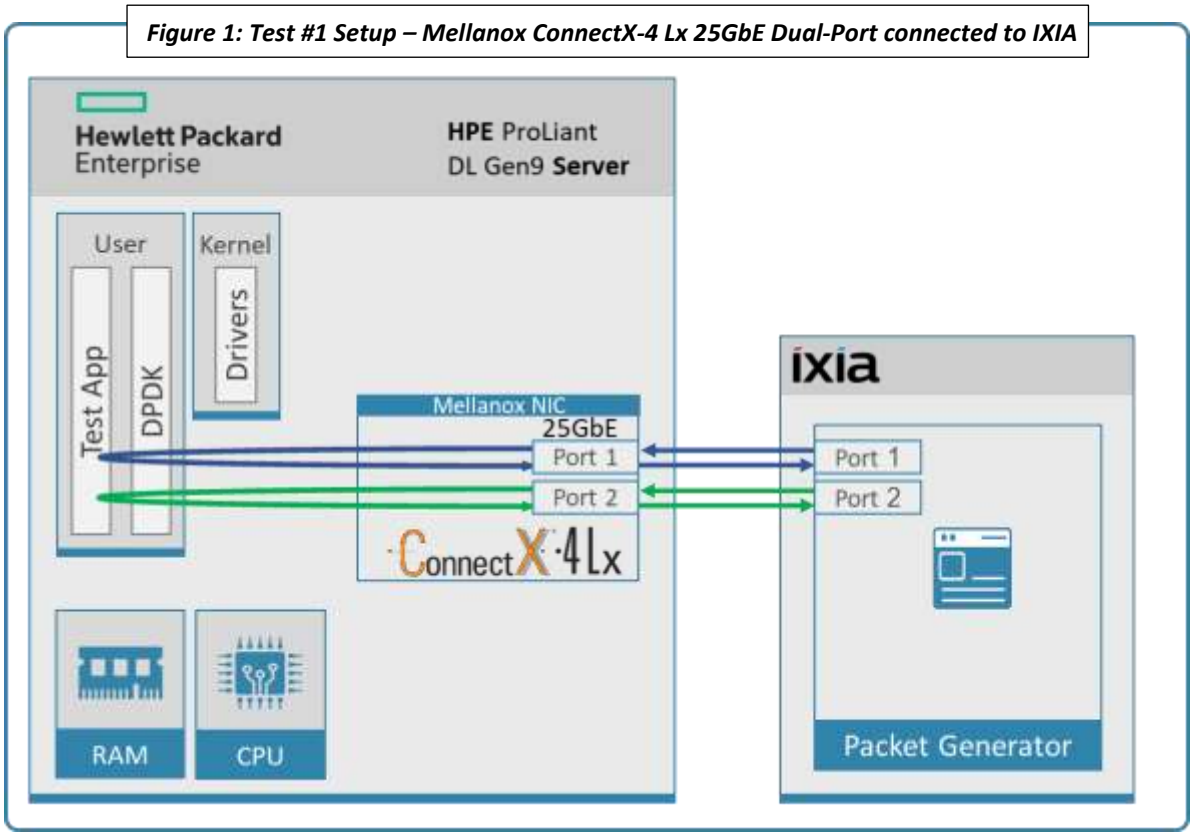
2 Test#1 Mellanox ConnectX-4 Lx 25GbE Throughput at Zero Packet Loss (2x 25GbE)

Table 2: Test #1 Setup

Item	Description
Test	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	Red Hat Enterprise Linux Server release 7.5 (Maipo)
Kernel Version	3.10.0-862.el7.x86_64
GCC version	4.8.5 20150623 (Red Hat 4.8.5-28) (GCC)
Mellanox NIC firmware version	14.27.1016
Mellanox OFED driver version	MLNX_OFED_LINUX-5.0-1.0.0.0
DPDK version	20.02
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 **l3fwd** and is redirected to the opposite direction on the opposing port. IXIA measures throughput and packet loss.

Figure 1: Test #1 Setup – Mellanox ConnectX-4 Lx 25GbE Dual-Port connected to IXIA



2.1 Test Settings

Table 3: Test #1 Settings

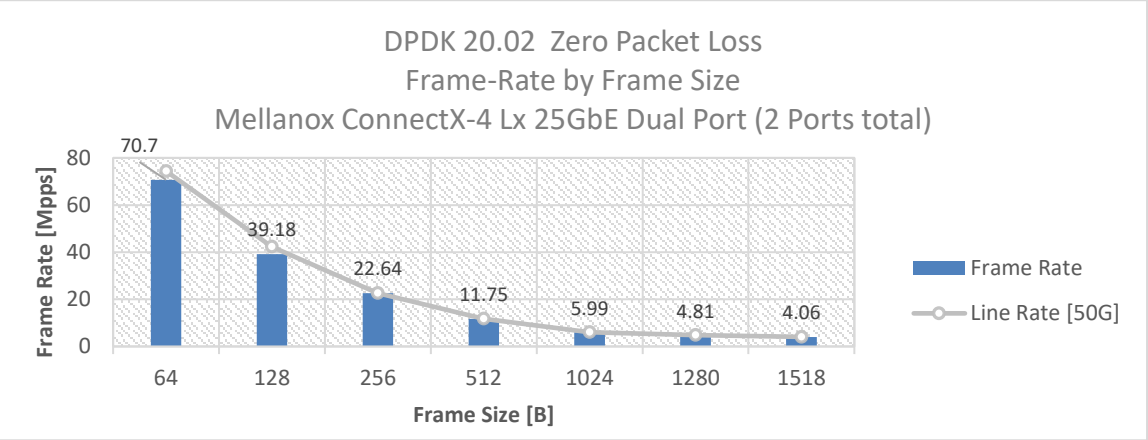
Item	Description
BIOS	1) Workload Profile = “Low Latency”; 2) Jitter Control = Manual, 3400. (Setting turbo boost frequency to 3.4 GHz) See “Configuring and tuning HPE ProLiant Servers for low-latency applications”: hpe.com > Search “DL380 gen10 low latency”
BOOT Settings	isolcpus=24-47 intel_idle.max_cstate=0 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 nosoftlockup
DPDK Settings	Enable mlx5 PMD before compiling DPDK: In .config file generated by "make config", set: "CONFIG_RTE_LIBRTE_MLX5_PMD=y" During testing, l3fwd was given real-time scheduling priority.
L3fwd settings	Added /l3fwd/main.c:85: #define RTE_TEST_RX_DESC_DEFAULT 4096 #define RTE_TEST_TX_DESC_DEFAULT 4096 Added /l3fwd/l3fwd.h:47: #define MAX_PKT_BURST 64
Command Line	/root/dpdk/examples/l3fwd/build/app/l3fwd -c 0xff0000000000 -n 4 -w d9:00.0,txq_inline=200,txq_mpw_en=1 -w d9:00.1,txq_inline=200,txq_mpw_en=1 --socket-mem=0,8192 -- -p 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=0,00:52:11:22:33:10 --eth-dest=1,00:52:11:22:33:20
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 irqbalance --oneshot" 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

2.2 Test Results

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

Frame Size (Bytes)	Frame Rate (Mpps)	Line Rate [50G] (Mpps)	% Line Rate
64	70.7	74.4	95.025
128	39.18	42.23	92.78
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

Figure 2: Test #1 Results – Mellanox ConnectX-4 Lx 25GbE Dual-Port Throughput at Zero Packet Loss



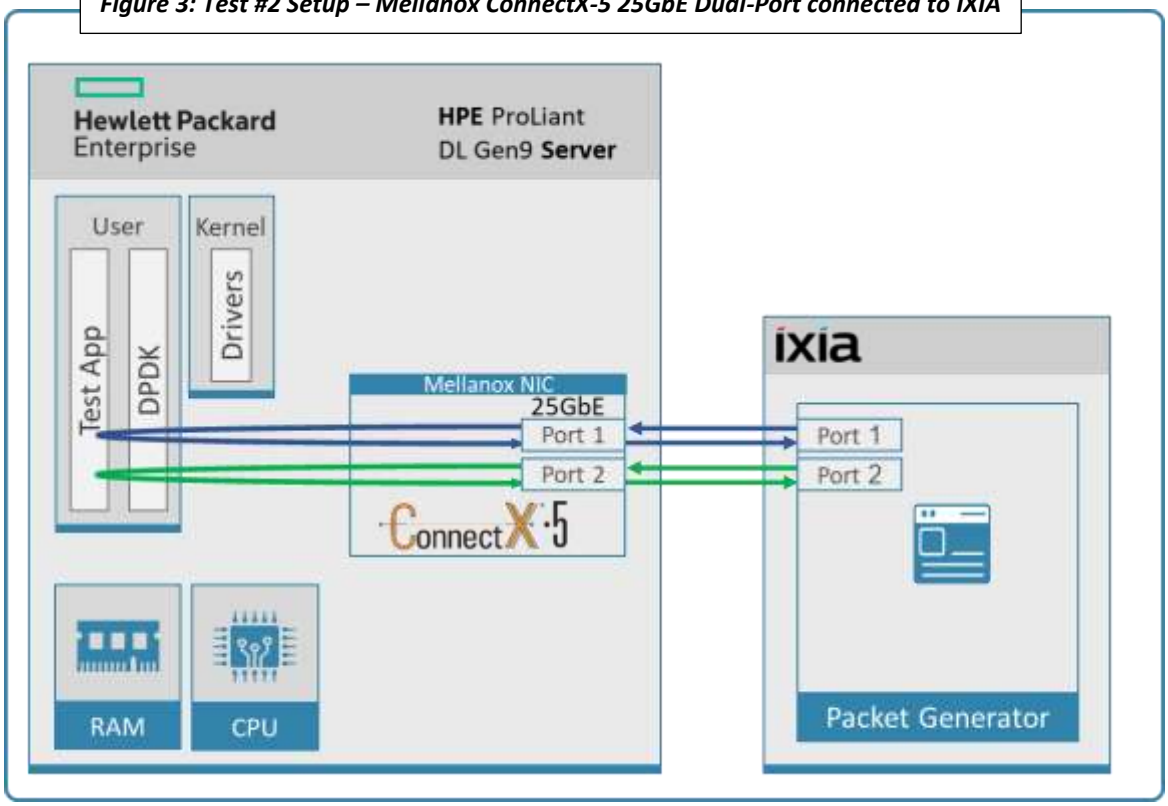
3 Test#2 Mellanox ConnectX-5 25GbE Throughput at Zero Packet Loss (2x 25GbE)

Table 5: Test #2 Setup

Item	Description
Test	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	Red Hat Enterprise Linux Server release 7.5 (Maipo)
Kernel Version	3.10.0-862.el7.x86_64
GCC version	4.8.5 20150623 (Red Hat 4.8.5-28) (GCC)
Mellanox NIC firmware version	16.27.1016
Mellanox OFED driver version	MLNX_OFED_LINUX-5.0-1.0.0.0
DPDK version	20.02
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-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 **l3fwd** and is redirected to the opposite direction on the same port. IXIA measures throughput and packet loss.

Figure 3: Test #2 Setup – Mellanox ConnectX-5 25GbE Dual-Port connected to IXIA



3.1 Test Settings

Table 6: Test #2 Settings

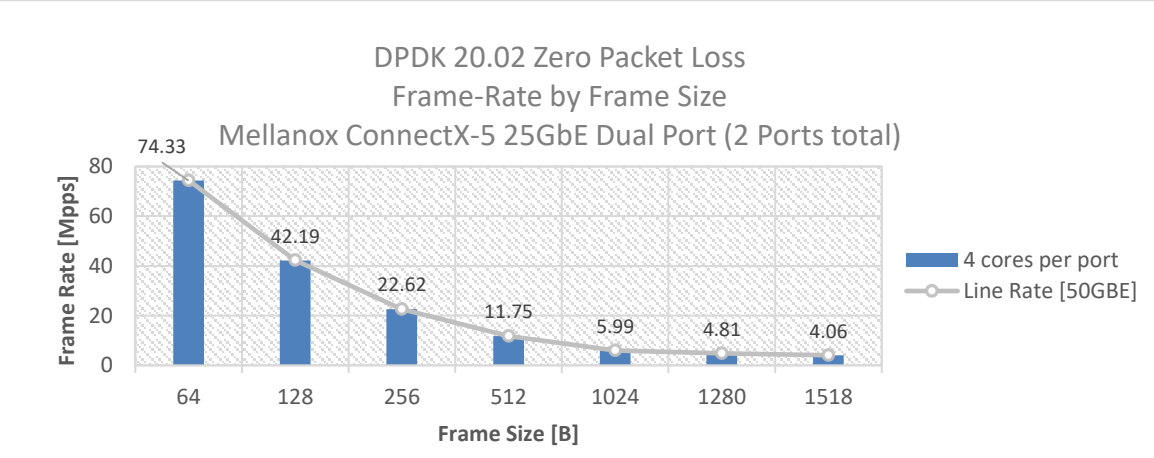
Item	Description
BIOS	1) Workload Profile = “Low Latency”; 2) Jitter Control = Manual, 3400. (Setting turbo boost frequency to 3.4 GHz) See “Configuring and tuning HPE ProLiant Servers for low-latency applications”: hpe.com > Search “DL380 gen10 low latency”
BOOT Settings	isolcpus=24-47 intel_idle.max_cstate=0 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 nosoftlockup
DPDK Settings	Enable mlx5 PMD before compiling DPDK: In .config file generated by "make config", set: "CONFIG_RTE_LIBRTE_MLX5_PMD=y" During testing, l3fwd was given real-time scheduling priority.
L3fwd settings	Added /l3fwd/main.c:85: #define RTE_TEST_RX_DESC_DEFAULT 4096 #define RTE_TEST_TX_DESC_DEFAULT 4096 Added /l3fwd/l3fwd.h:47: #define MAX_PKT_BURST 64
Command Line	./examples/l3fwd/build/app/l3fwd -c 0xff0000000000 -n 4 -w d8:00.0,mprq_en=1 -w d8:00.1,mprq_en=1 --socket-mem=0,8192 -- -p 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=0,00:52:11:22:33:10 --eth-dest=1,00:52:11:22:33:20
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 irqbalance --oneshot" 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=3936" 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

3.2 Test Results

Table 7: Test #2 Results – Mellanox ConnectX-5 25GbE Dual-Port Throughput at Zero Packet Loss

Frame Size (Bytes)	Frame Rate (Mpps)	Line Rate [50G] (Mpps)	% Line Rate
64	74.33	74.40	99.9
128	42.19	42.23	99.9
256	22.62	22.64	99.9
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

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



4 Test#3 Mellanox ConnectX-4 Lx 40GbE Throughput at Zero Packet Loss (2x 40GbE)

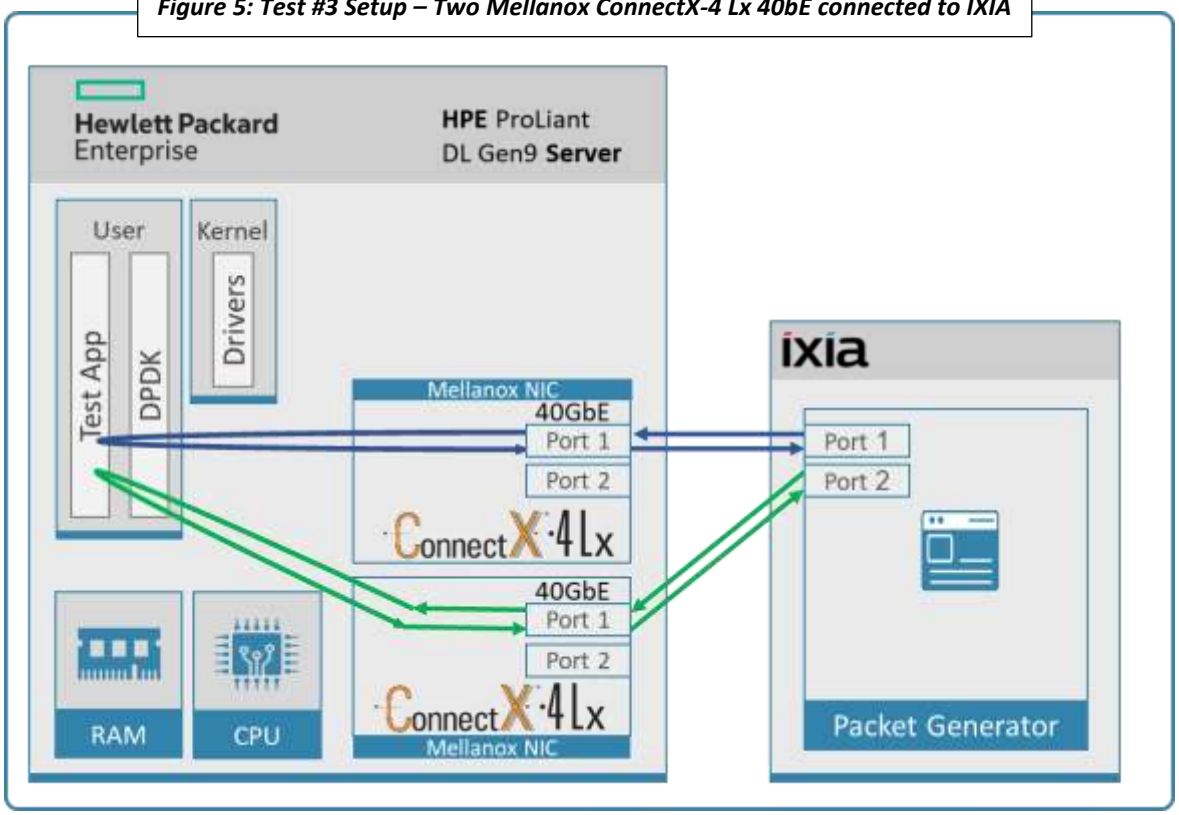
Table 8: Test #3 Setup

Item	Description
Test	Test #3 - Mellanox ConnectX-4 Lx 40GbE Throughput at Zero Packet Loss (2x 40GbE)
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 MCX4131A-GCAT ConnectX®-4 Lx EN network interface card, 50GbE single-port QSFP28, PCIe3.0 x8, tall bracket, ROHS R6 (Connected to 40GbE Ixia ports)
Operating System	Red Hat Enterprise Linux Server release 7.5 (Maipo)
Kernel Version	3.10.0-862.el7.x86_64
GCC version	4.8.5 20150623 (Red Hat 4.8.5-28) (GCC)
Mellanox NIC firmware version	14.27.1016
Mellanox OFED driver version	MLNX_OFED_LINUX-5.0-1.0.0.0
DPDK version	20.02
Test Configuration	2 NICs, 1 port used on each NIC, each port has 4 queues assigned to it, 1 queue per logical core for a total of 8 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 two Mellanox ConnectX-4 Lx Single-Port NICs on the same NUMA node. The DUT is connected to the IXIA packet generator which generates traffic towards both ConnectX-4 Lx NICs.

The traffic is passed through DPDK to the test application **l3fwd** and is redirected to the opposite direction on the same port. IXIA measures throughput and packet loss.

Figure 5: Test #3 Setup – Two Mellanox ConnectX-4 Lx 40bE connected to IXIA



4.1 Test Settings

Table 9: Test #3 Settings

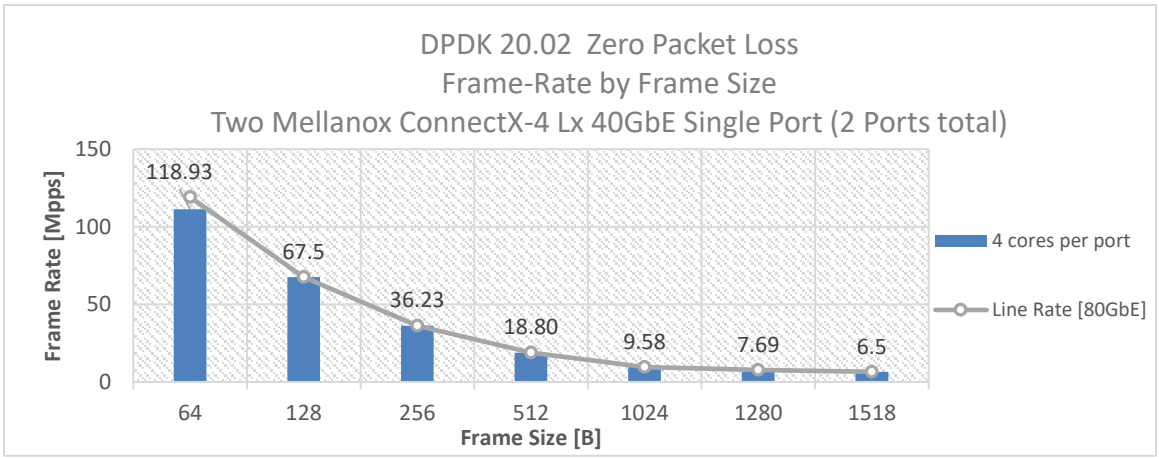
Item	Description
BIOS	1) Workload Profile = “Low Latency”; 2) Jitter Control = Manual, 3400. (Setting turbo boost frequency to 3.4 GHz) See “Configuring and tuning HPE ProLiant Servers for low-latency applications”: hpe.com > Search “DL380 gen10 low latency”
BOOT Settings	isolcpus=0-23 intel_idle.max_cstate=0 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
DPDK Settings	Enable mlx5 PMD before compiling DPDK: In .config file generated by "make config", set: "CONFIG_RTE_LIBRTE_MLX5_PMD=y" During testing, l3fwd was given real-time scheduling priority.
L3fwd settings	Added /l3fwd/main.c:85: #define RTE_TEST_RX_DESC_DEFAULT 4096 #define RTE_TEST_TX_DESC_DEFAULT 4096 Added /l3fwd/l3fwd.h:47: #define MAX_PKT_BURST 64
Command Line	./examples/l3fwd/build/app/l3fwd -c 0xff0000 -n 4 -w 12:00.0,txq_inline=200,txq_mpw_en=1 -w 37:00.0,txq_inline=200,txq_mpw_en=1 --socket-mem=8192 -- -p 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=0,00:52:11:22:33:10 --eth-dest=1,00:52:11:22:33:20
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 irqbalance --oneshot" 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

4.2 Test Results

Table 10: Test #3 Results – Mellanox ConnectX-4 Lx 40GbE Throughput at Zero Packet Loss

Frame Size (Bytes)	Frame Rate (Mpps)	Line Rate [2*40G] (Mpps)	% Line Rate
64	118.93	119.05	99.90
128	67.50	67.57	99.90
256	36.23	36.23	100.00
512	18.80	18.80	100.00
1024	9.58	9.58	100.00
1280	7.69	7.69	100.00
1518	6.50	6.50	100.00

Figure 6: Test #3 Results – Mellanox ConnectX-4 Lx 40GbE Throughput at Zero Packet Loss



5 Test#4 Mellanox ConnectX-5 Ex 100GbE Throughput at Zero Packet Loss (1x 100GbE)

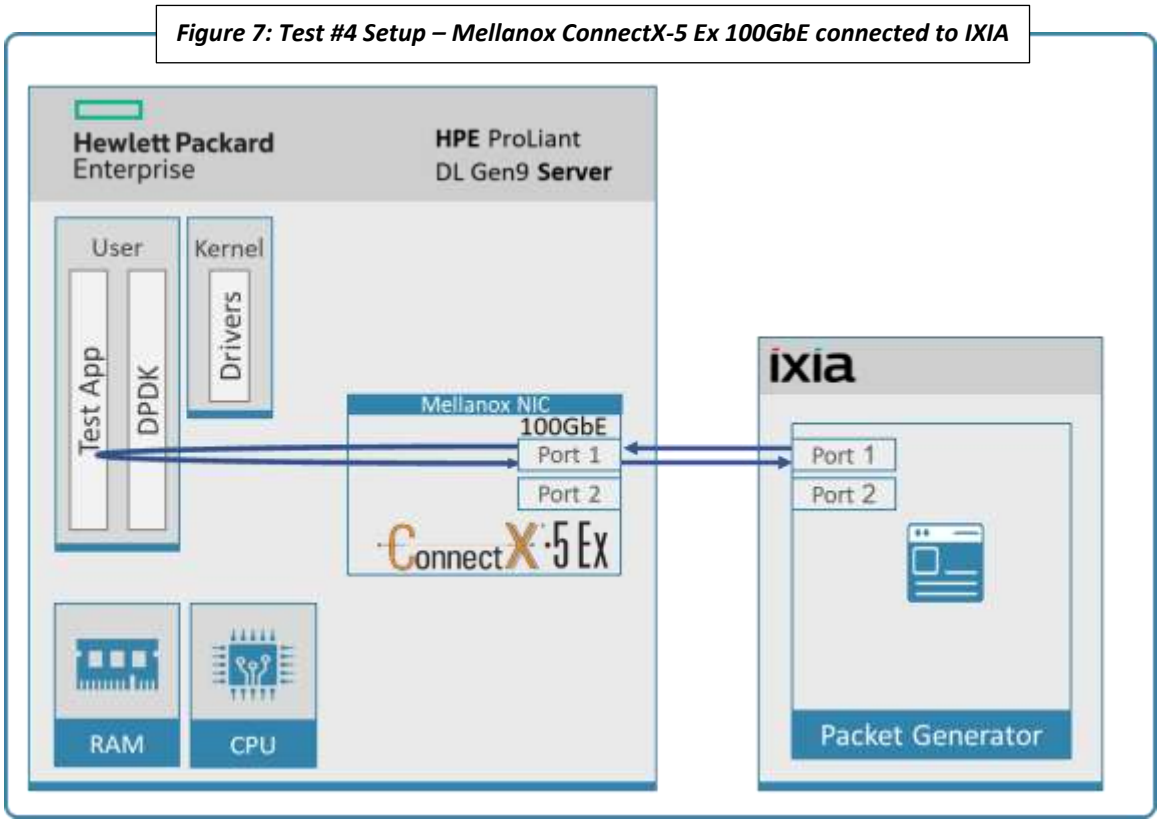
Table 11: Test #4 Setup

Item	Description
Test	Test #4 – Mellanox ConnectX-5 Ex 100GbE 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 MCX516A-CDAT- ConnectX-5 Ex network interface card 100GbE dual-port QSFP28; PCIe3.0/PCIe4 x16; ROHS R6
Operating System	Red Hat Enterprise Linux Server release 7.5 (Maipo)
Kernel Version	3.10.0-862.el7.x86_64
GCC version	4.8.5 20150623 (Red Hat 4.8.5-28) (GCC)
Mellanox NIC firmware version	16.27.1016
Mellanox OFED driver version	MLNX_OFED_LINUX-5.0-1.0.0.0
DPDK version	20.02
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 **l3fwd** and is redirected to the opposite direction on the same port. IXIA measures throughput and packet loss.

Figure 7: Test #4 Setup – Mellanox ConnectX-5 Ex 100GbE connected to IXIA



5.1 Test Settings

Table 12: Test #4 Settings

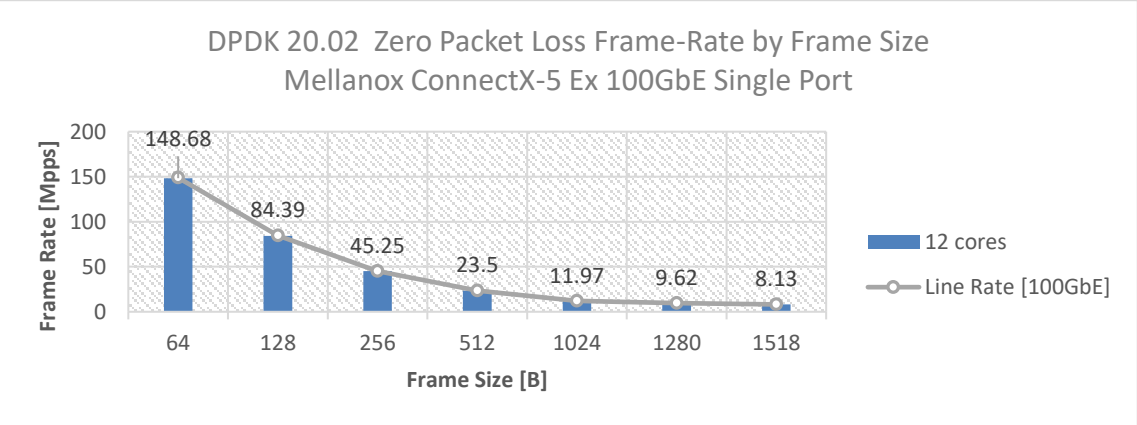
Item	Description
BIOS	1) Workload Profile = “Low Latency”; 2) Jitter Control = Manual, 3400. (Setting turbo boost frequency to 3.4 GHz) See “Configuring and tuning HPE ProLiant Servers for low-latency applications”: hpe.com > Search “DL380 gen10 low latency”
BOOT Settings	isolcpus=24-47 intel_idle.max_cstate=0 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	Enable mlx5 PMD before compiling DPDK: In .config file generated by "make config", set: "CONFIG_RTE_LIBRTE_MLX5_PMD=y" During testing, l3fwd was given real-time scheduling priority.
L3fwd settings	Added /l3fwd/main.c:85: #define RTE_TEST_RX_DESC_DEFAULT 2048 #define RTE_TEST_TX_DESC_DEFAULT 2048 Added /l3fwd/l3fwd.h:47: #define MAX_PKT_BURST 64
Command Line	/examples/l3fwd/build/app/l3fwd -c 0xfff000000000 -n 4 -w 0000:af:00.0,mprq_en=1 --socket-mem=0,8192 -- -p 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 irqbalance --oneshot" 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

5.2 Test Results

Table 13: Test #4 Results – Mellanox ConnectX-5 Ex 100GbE Throughput at Zero Packet Loss

Frame Size (Bytes)	Frame Rate (Mpps)	Line Rate [100G] (Mpps)	% Line Rate
64	148.68	148.81	99.91
128	84.39	84.46	99.91
256	45.25	45.29	99.91
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

Figure 8: Test #4 Results – Mellanox ConnectX-5 Ex 100GbE Throughput at Zero Packet Loss



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

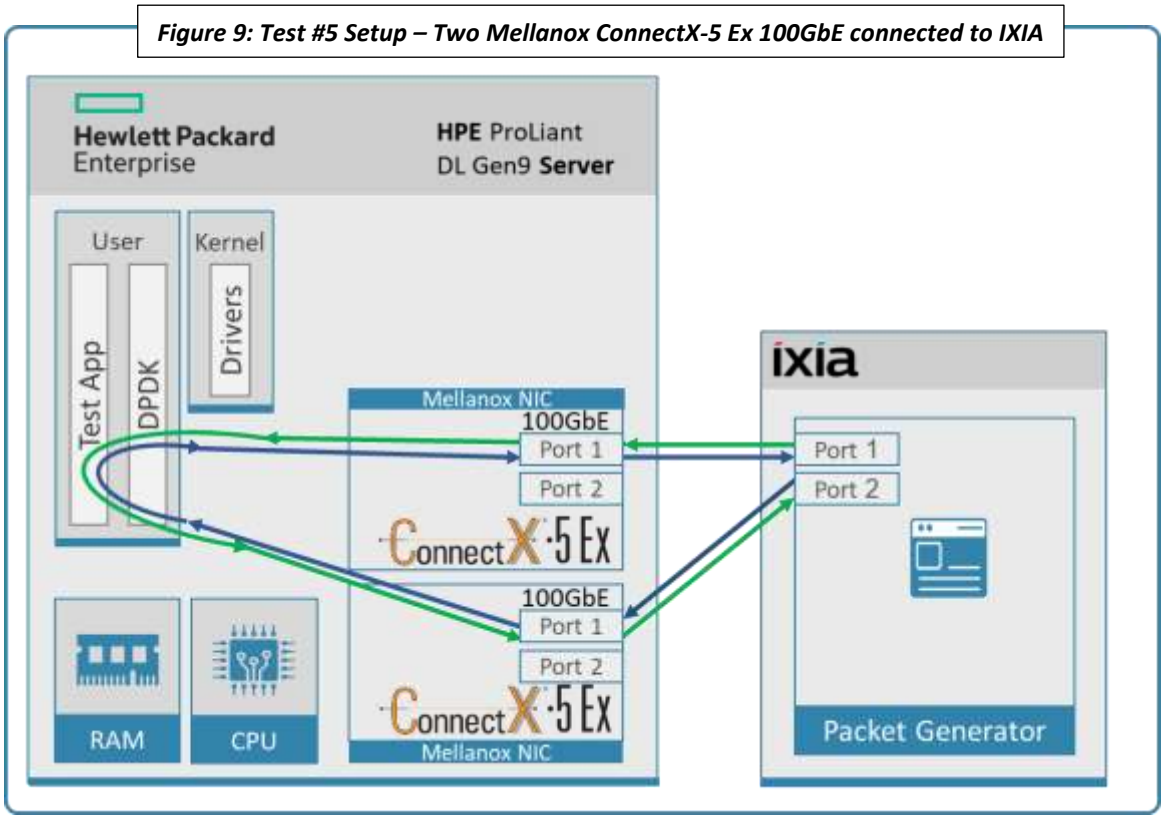
Table 14: Test #5 Setup

Item	Description
Test	Test #5 – Mellanox ConnectX-5 Ex 100GbE 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 MCX516A-CDAT- ConnectX-5 Ex network interface card 100GbE dual-port QSFP28; PCIe3.0/PCIe4 x16; ROHS R6
Operating System	Red Hat Enterprise Linux Server release 7.5 (Maipo)
Kernel Version	3.10.0-862.el7.x86_64
GCC version	4.8.5 20150623 (Red Hat 4.8.5-28) (GCC)
Mellanox NIC firmware version	16.27.1016
Mellanox OFED driver version	MLNX_OFED_LINUX-5.0-1.0.0.0
DPDK version	20.02
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.

Figure 9: Test #5 Setup – Two Mellanox ConnectX-5 Ex 100GbE connected to IXIA



6.1 Test Settings

Table 15: Test #5 Settings

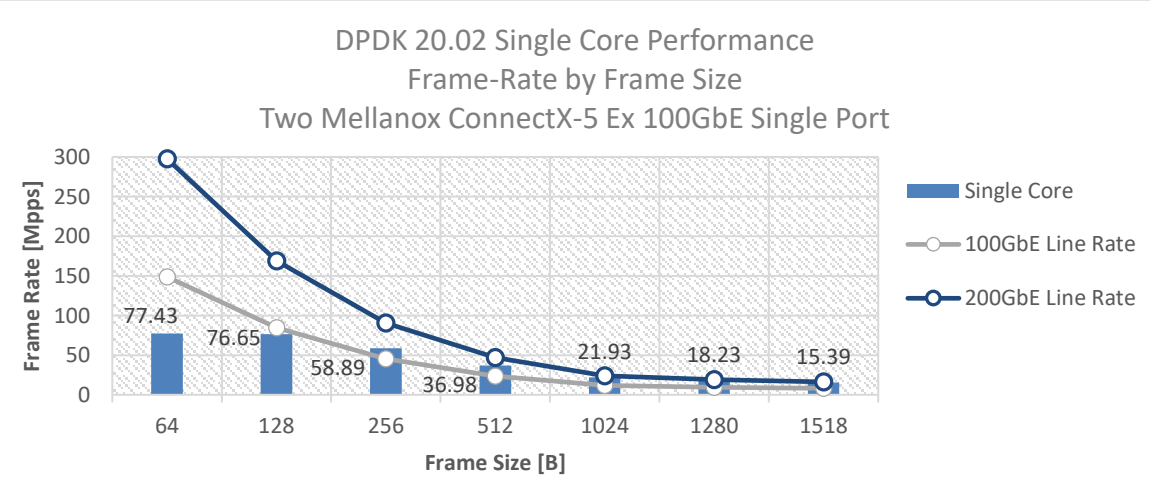
Item	Description
BIOS	1) Workload Profile = “Low Latency”; 2) Jitter Control = Manual, 3400. (Setting turbo boost frequency to 3.4 GHz) See “Configuring and tuning HPE ProLiant Servers for low-latency applications”: hpe.com > Search “DL380 gen10 low latency”
BOOT Settings	isolcpus=24-47 intel_idle.max_cstate=0 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
DPDK Settings	Enable mlx5 PMD before compiling DPDK: In .config file generated by "make config", set: "CONFIG_RTE_LIBRTE_MLX5_PMD=y" set: "CONFIG_RTE_TEST_PMD_RECORD_CORE_CYCLES=y" During testing, testpmd was given real-time scheduling priority.
Command Line	./build/app/testpmd -c 0x110000000000 -n 4 -w 86:00.0 -w af:00.0 --socket-mem=0,8192 -- --port-numa-config=0,1,1,1 --socket-num=1 --burst=64 --txd=1024 --rxd=1024 --mbcache=512 --rxq=1 --txq=1 --nb-cores=1 -i -a --rss-udp --no-numa --disable-crc-strip
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 irqbalance --oneshot" 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

6.2 Test Results

Table 16: Test #5 Results – Mellanox ConnectX-5 Ex 100GbE Single Core Performance

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	77.65	297.62	148.81	39.756	33
128	76.65	168.92	84.46	78.493	34
256	58.93	90.58	45.29	120.683	32
512	37.03	46.99	23.50	151.658	34
1024	21.93	23.95	11.97	179.621	34
1280	18.23	19.23	9.62	186.657	35
1518	15.39	16.25	8.13	186.913	36

Figure 10: Test #5 Results – Mellanox ConnectX-5 Ex 100GbE Single Core Performance



7 Test#6 Mellanox ConnectX-5 25GbE Single Core Performance (2x 25GbE)

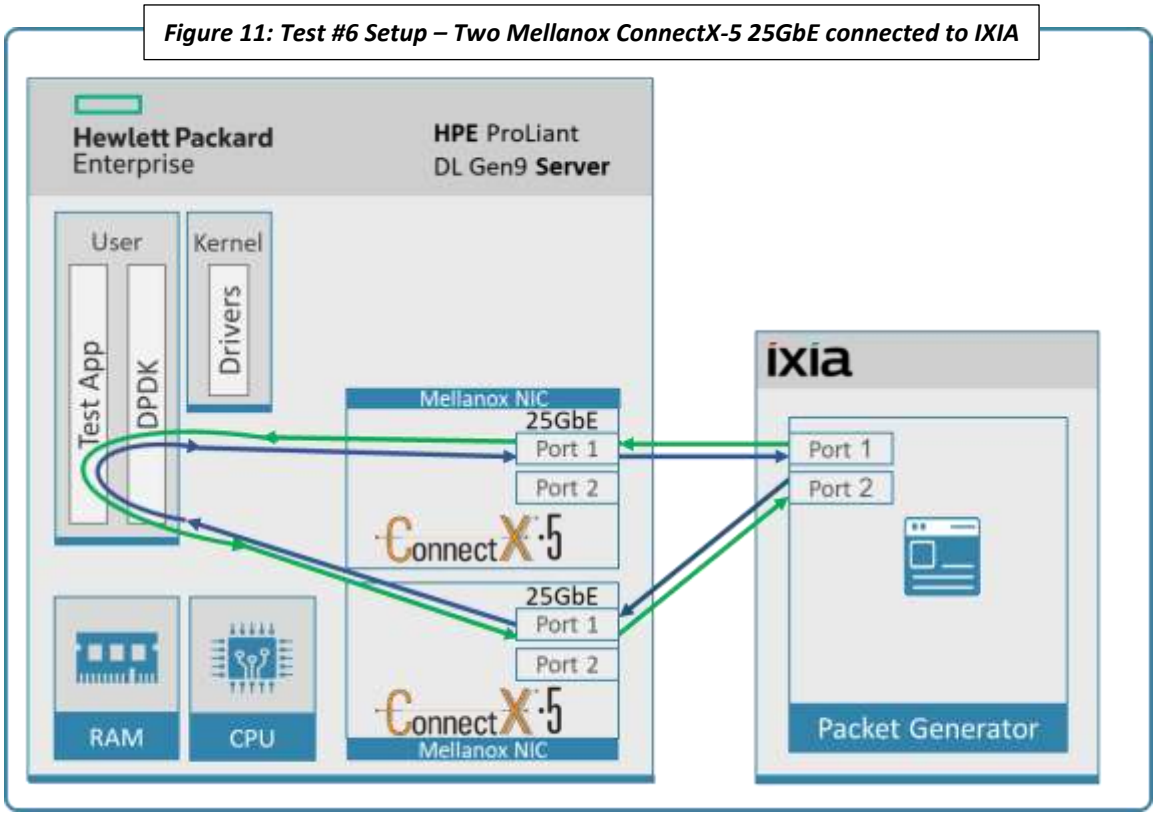
Table 17: Test #6 Setup

Item	Description
Test	Test #6 – 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 card; 10/25GbE dual-port SFP28; PCIe3.0 x8; tall bracket; ROHS R6
Operating System	Red Hat Enterprise Linux Server release 7.5 (Maipo)
Kernel Version	3.10.0-862.el7.x86_64
GCC version	4.8.5 20150623 (Red Hat 4.8.5-28) (GCC)
Mellanox NIC firmware version	16.27.1016
Mellanox OFED driver version	MLNX_OFED_LINUX-5.0-1.0.0.0
DPDK version	20.02
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 redirected to the opposite direction on the opposing NIC's port. IXIA measures throughput and packet loss.

Figure 11: Test #6 Setup – Two Mellanox ConnectX-5 25GbE connected to IXIA



7.1 Test Settings

Table 18: Test #6 Settings

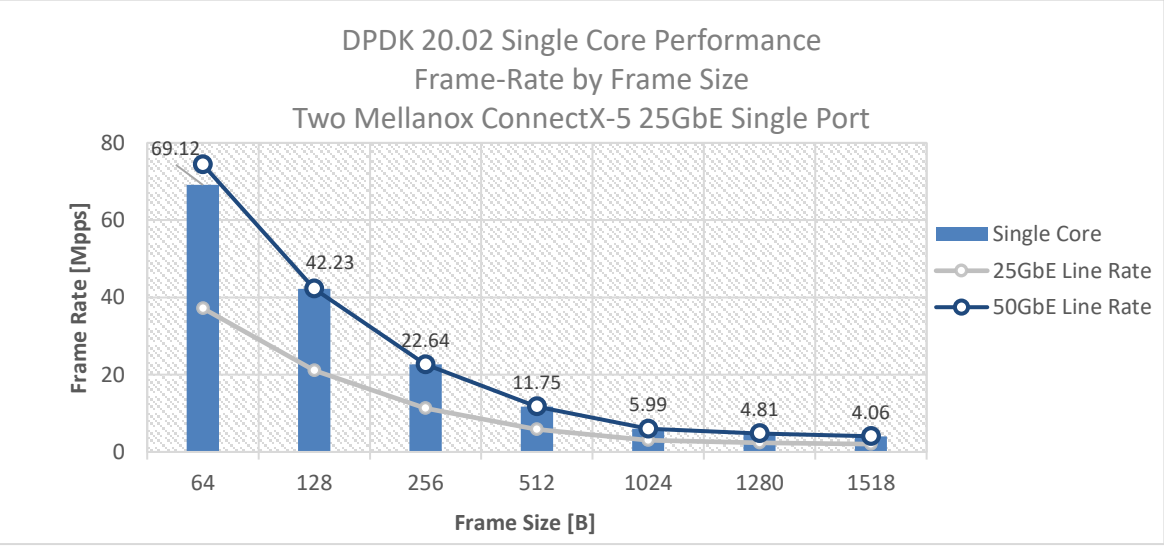
Item	Description
BIOS	1) Workload Profile = "Low Latency" 2) Jitter Control = Manual, 3400. (Setting turbo boost frequency to 3.4 GHz) See "Configuring and tuning HPE ProLiant Servers for low-latency applications": hpe.com > Search "DL380 gen10 low latency"
BOOT Settings	isolcpus=24-47 intel_idle.max_cstate=0 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 nosoftlockup
DPDK Settings	Enable mlx5 PMD before compiling DPDK: In .config file generated by "make config", set: "CONFIG_RTE_LIBRTE_MLX5_PMD=y" set: "CONFIG_RTE_TEST_PMD_RECORD_CORE_CYCLES=y" During testing, testpmd was given real-time scheduling priority.
Command Line	./build/app/testpmd -c 0x30000000000 -n 4 -w d8:00.0 -w d9:00.0 --socket-mem=0,8192 -- --port-numa-config=0,1,1,1 --socket-num=1 --burst=64 --txd=1024 --rxd=1024 --mbcache=512 --rxq=1 --txq=1 --nb-cores=1 -i -a --rss-udp --no-numa --disable-crc-strip
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 irqbalance --oneshot" 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

7.2 Test Results

Table 19: Test #6 Results – Mellanox ConnectX-5 25GbE Single Core Performance

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	69.13	37.2	74.4	35.397	33
128	42.23	21.11	42.23	43.242	31
256	22.64	11.32	22.64	46.377	30
512	11.75	5.87	11.75	48.12	31
1024	5.99	2.99	5.99	49.042	31
1280	4.81	2.4	4.81	49.231	31
1518	4.06	2.03	4.06	49.35	33

Figure 12: Test #6 Results – Mellanox ConnectX-5 25GbE Single Core Performance



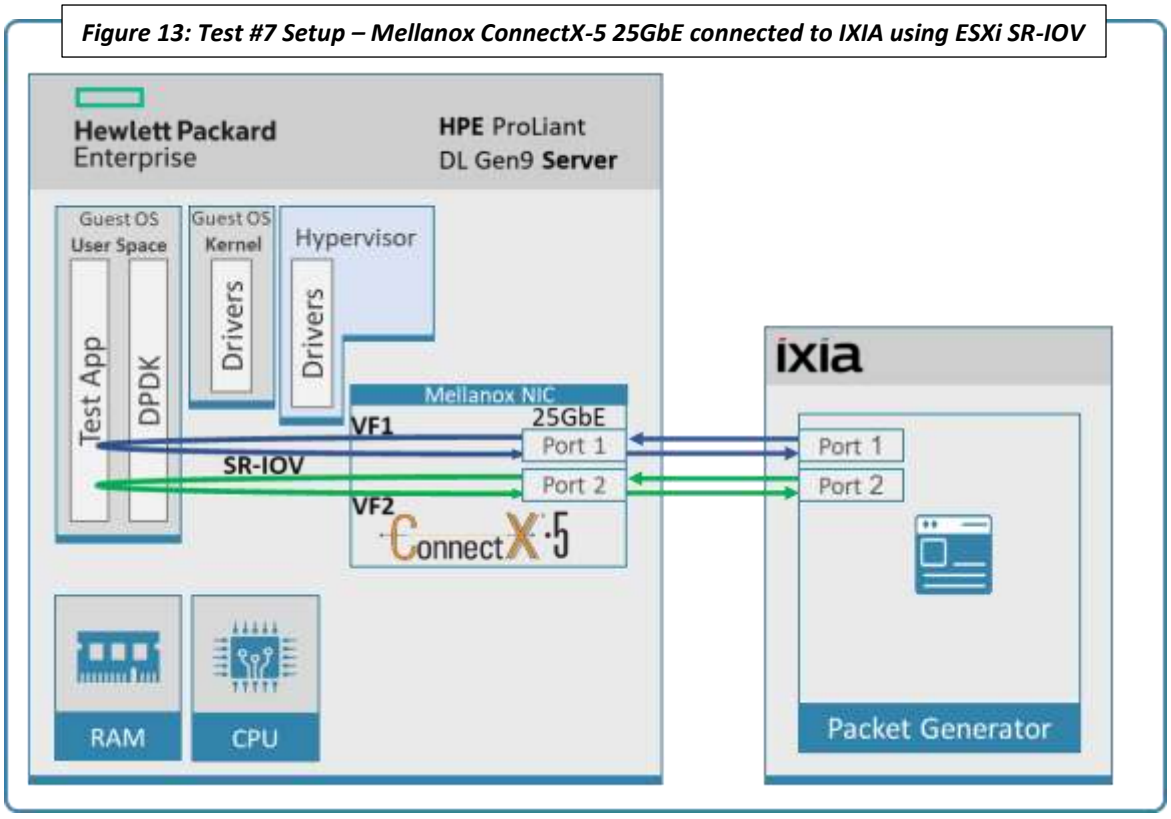
8 Test#7 Mellanox ConnectX-5 25GbE Throughput at Zero Packet Loss (2x 25GbE) using SR-IOV over VMware ESXi 6.5

Table 20: Test #7 Setup

Item	Description
Test	Test #7 – Mellanox ConnectX-5 25GbE Dual-Port Throughput at zero packet loss SRIOV over VMware ESXi 6.5U2
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 6.5U2
Hypervisor Build	VMware-ESXi-6.5.0-Update2-9298722-HPE-Gen9plus-650.U2.10.3.5.5-Sep2018.iso
Hypervisor Mellanox Driver	MLNX-NATIVE-ESX-ConnectX-4-5_4.16.14.2
Guest Operating System	Red Hat Enterprise Linux Server release 7.5 (Maipo)
Guest Kernel Version	3.10.0-862.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-5.0-1.0.0.0
Mellanox NIC firmware version	16.27.1016
DPDK version	20.02
Test Configuration	1 NIC, 2 ports with 1 VF per port (SR-IOV); Each port receives a stream of 8192 IP flows from the IXIA Each VF (SR-IOV) 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 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 **l3fwd** over DPDK and is redirects traffic to the opposite direction on the same VF/port. IXIA measures throughput and packet loss.

Figure 13: Test #7 Setup – Mellanox ConnectX-5 25GbE connected to IXIA using ESXi SR-IOV



8.1 Test Settings

Table 21: Test#7 Settings

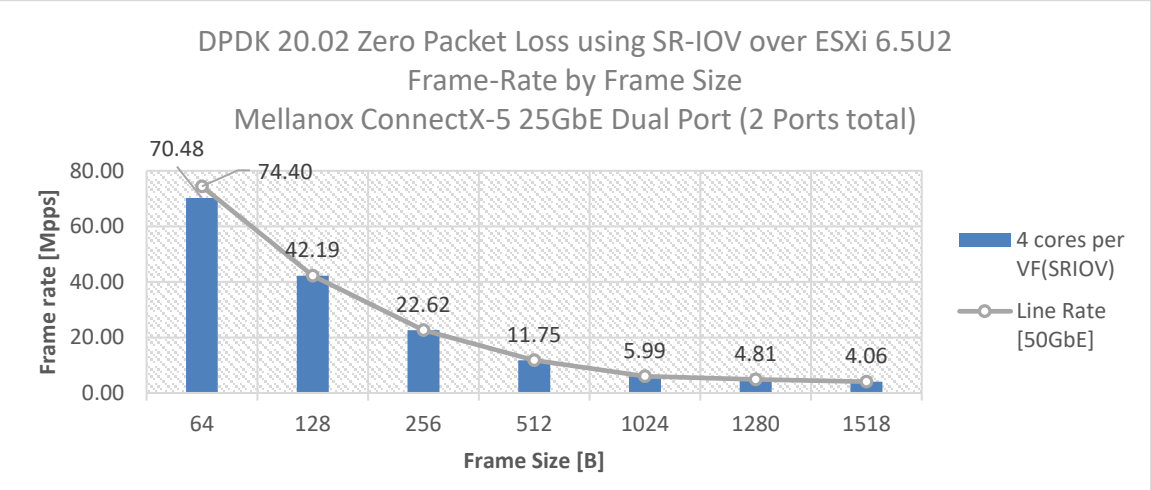
Item	Description
BIOS	1) Workload Profile = “Low Latency”; 2) Jitter Control = Manual, 3400. (Setting turbo boost frequency to 3.4 GHz) 3) Change "Workload Profile" to "Custom" 4) Change VT-x, VT-d and SR-IOV from "Disabled" to "Enabled". See “Configuring and tuning HPE ProLiant Servers for low-latency 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 reboot 2) Install Driver esxcli software vib install -d MLNX-NATIVE-ESX-ConnectX-4-5_4.16.14.2-10EM-650.0.0.4598673.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	Enable mlx5 PMD before compiling DPDK: In .config file generated by "make config", set: "CONFIG_RTE_LIBRTE_MLX5_PMD=y" During testing, l3fwd was given real-time scheduling priority.
L3fwd settings on Guest OS	Added /l3fwd/main.c:85: #define RTE_TEST_RX_DESC_DEFAULT 2048 #define RTE_TEST_TX_DESC_DEFAULT 2048 Added /l3fwd/l3fwd.h:47: #define MAX_PKT_BURST 64
Command Line on Guest OS	./examples/l3fwd/build/app/l3fwd -c 0x7f8000 -n 4 -w 13:00.0,mprq_en=1 -w 1b:00.0,mprq_en=1 --socket-mem=8192 -- -p 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=0,00:52:11:22:33:10 --eth-dest=1,00:52:11:22:33:20
Other optimizations on Guest OS	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 irqbalance --oneshot" d) Disable irqbalance: "systemctl stop irqbalance" e) Disable Linux realtime throttling: echo -1 > /proc/sys/kernel/sched_rt_runtime_us

8.2 Test Results

Table 22: Test#7 Results – Mellanox ConnectX-5 25GbE Throughput at Zero Packet Loss using ESXi SR-IOV

Frame Size (Bytes)	Frame Rate (Mpps)	Line Rate [50G] (Mpps)	% Line Rate
64	70.48	74.4	94.73
128	42.19	42.23	99.90
256	22.62	22.64	99.90
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

Figure 14: Test#7 Results – Mellanox ConnectX-5 25GbE Throughput at Zero Packet Loss using ESXi SR-IOV



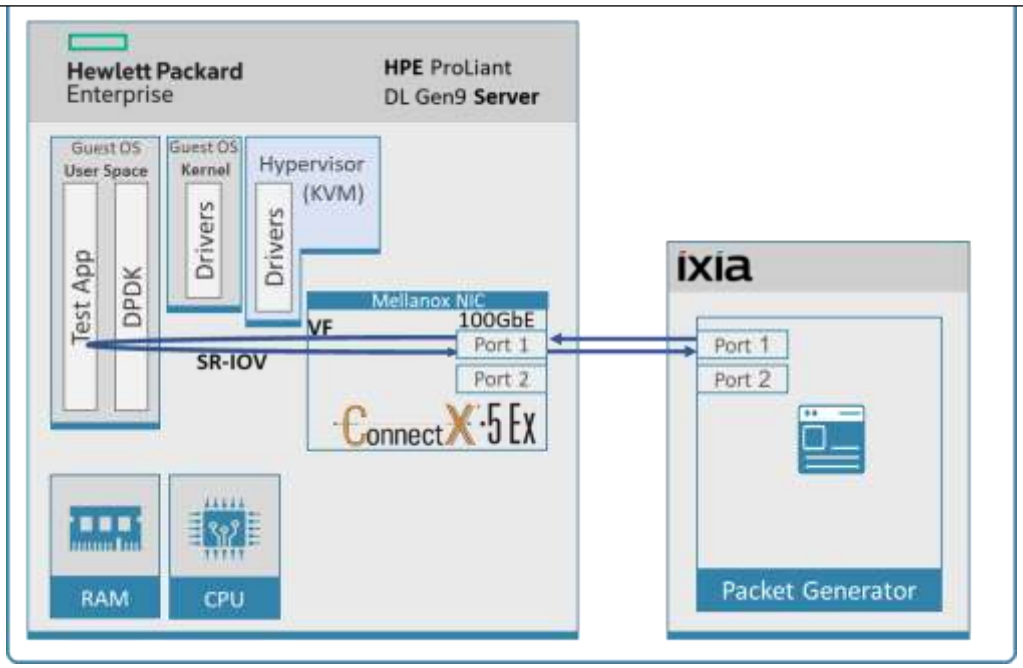
9 Test#8 Mellanox ConnectX-5 Ex 100GbE Throughput at Zero Packet Loss (1x 100GbE) using SR-IOV over KVM Hypervisor

Table 23: Test #8 Setup

Item	Description
Test	Test #8– Mellanox ConnectX-5 Ex 100GbE Throughput at zero packet loss using SR-IOV over KVM
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 MCX516A-CDAT- ConnectX-5 Ex network interface card 100GbE dual-port QSFP28; PCIe3.0/PCIe4 x16; ROHS R6
Hypervisor	Red Hat Enterprise Linux Server release 7.5 (Maipo) QEMU emulator version 1.5.3 (qemu-kvm-1.5.3-156.el7)
Hypervisor Kernel Version	3.10.0-862.el7.x86_64
Hypervisor Mellanox Driver	MLNX_OFED_LINUX-5.0-1.0.0.0
Guest Operating System	Red Hat Enterprise Linux Server release 7.5 (Maipo)
Guest Kernel Version	3.10.0-862.el7.x86_64
Guest GCC version	4.8.5 20150623 (Red Hat 4.8.5-28) (GCC)
Guest Mellanox OFED driver Ver	MLNX_OFED_LINUX-5.0-1.0.0.0
Mellanox NIC firmware version	16.27.1016
DPDK version	20.02
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 dual- port ConnectX-5 Ex NIC (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 **l3fwd** and is redirected to the opposite direction on the same port. IXIA measures throughput and packet loss.

Figure 15: Test #8 Setup – Mellanox ConnectX-5 Ex 100GbE connected to IXIA using KVM SR-IOV



9.1 Test Settings

Table 24: Test #8 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/***ens6f0***/device/sriov_numvfs
	
	2) Assign VF
	HCA_netintf=***ens6f0*** #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 NIC interface>> <vf index> mac <MAC Address> : (mac is random)
	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 list --all)
	Example xml configuration: (change "nodeset" and "cpuset" attributes to suit the local NUMA node in your setup)
	<domain type='kvm' id='1'>
	<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>
	<hugepages>
	<page size='1048576' unit='KiB' nodeset='0'/>
	</hugepages>
	<nosharepages/>
	<locked/>
	</memoryBacking>
	<vcpu placement='static'>23</vcpu>
	<cputune>
	<vcpupin vcpu='0' cpuset='24'/>
	<vcpupin vcpu='1' cpuset='25'/>
	<vcpupin vcpu='2' cpuset='26'/>
	<vcpupin vcpu='3' cpuset='27'/>
	<vcpupin vcpu='4' cpuset='28'/>
	<vcpupin vcpu='5' cpuset='29'/>
	<vcpupin vcpu='6' cpuset='30'/>
	<vcpupin vcpu='7' cpuset='31'/>
	<vcpupin vcpu='8' cpuset='32'/>
	<vcpupin vcpu='9' cpuset='33'/>
	<vcpupin vcpu='10' cpuset='34'/>
	<vcpupin vcpu='11' cpuset='35'/>
	<vcpupin vcpu='12' cpuset='36'/>
	<vcpupin vcpu='13' cpuset='37'/>
	<vcpupin vcpu='14' cpuset='38'/>
	<vcpupin vcpu='15' cpuset='39'/>
	<vcpupin vcpu='16' cpuset='40'/>
	<vcpupin vcpu='17' cpuset='41'/>
	<vcpupin vcpu='18' cpuset='42'/>
	<vcpupin vcpu='19' cpuset='43'/>
	<vcpupin vcpu='20' cpuset='44'/>
	<vcpupin vcpu='21' cpuset='45'/>
	<vcpupin vcpu='22' cpuset='46'/>
	</cputune>

Item	Description
Other optimizations on Hypervisor	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 irqbalance --oneshot" 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 Guest OS	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 irqbalance --oneshot" d) Disable irqbalance: "systemctl stop irqbalance" e) Disable Linux realtime throttling: echo -1 > /proc/sys/kernel/sched_rt_runtime_us
DPDK Settings on Guest OS	Enable mlx5 PMD before compiling DPDK: In .config file generated by "make config", set: "CONFIG_RTE_LIBRTE_MLX5_PMD=y" During testing, l3fwd was given real-time scheduling priority.
L3fwd settings on Guest OS	Added /l3fwd/main.c:85: #define RTE_TEST_RX_DESC_DEFAULT 2048 #define RTE_TEST_TX_DESC_DEFAULT 2048 Added /l3fwd/l3fwd.h:47: #define MAX_PKT_BURST 64
Command Line on Guest OS	./examples/l3fwd/build/app/l3fwd -c 0x3ffc00 -n 4 -w 00:06.0,mprq_en=1 --socket-mem=8192 -- -p 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

9.2 Test Results

Table 25:Test #8 Results – Mellanox ConnectX-5 Ex 100GbE Throughput at Zero Packet Loss using KVM SR-IOV

Frame Size (Bytes)	Frame Rate (Mpps)	Line Rate [100G] (Mpps)	% Line Rate
64	148.68	148.81	99.91
128	84.31	84.46	99.82
256	45.25	45.29	99.67
512	23.5	23.50	100
1024	11.97	11.97	100
1280	9.62	9.62	100
1518	8.13	8.13	100

Figure 16:Test #8 Results – Mellanox ConnectX-5 Ex 100GbE Throughput at Zero Packet Loss using KVM SR-IOV