Demonstration of Full-Duplex Packet Transfers over PCI Express with Sustained 200 Gbps Throughput

Lukáš Kekely, Martin Špinler, Štepán Friedl, Jiří Sikora, Jan Kořenek
CESNET a. l. e.
Zikova 4, 160 00 Prague, Czech Republic
{kekely, spinler, friedl, jiri.sikora, korenek}@cesnet.cz

Viktor Puš
Netcope Technologies a.s.
Sochorova 34, 616 00 Brno, Czech Republic
pus@netcope.com

***Abstract*—CESNET (Czech NREN) and Netcope Technologies have a long research history in the area of high-speed network monitoring using FPGA accelerated cards (i.e. SmartNICs). Now, we are ready to demonstrate a new NFB-200G2QL accelerator specifically designed to push the achievable traffic processing throughput to 200 Gbps in a single card. The card is equipped with two 100 Gbps Ethernet interfaces (QSFP28+ standard), powerful Virtex UltraScale+ FPGA, and two PCIe Gen3 $\times 16$ interfaces. Unique high-speed DMA engines in the FPGA together with highly optimized Linux drivers enable to achieve 200 Gbps data transfer throughput through the PCIe interfaces with minimal CPU overhead. Captured network traffic can be independently distributed among individual cores of two physical CPUs (NUMA nodes) without utilization of QPI. As a result, wire-speed packet capture to the host memory from both fully saturated 100 Gbps Ethernet interfaces is achieved and various network monitoring applications can utilize the power of the latest FPGAs and CPUs for data processing. This is especially useful when traffic of both directions of a single 100GbE link needs to be processed.**

The proposed demonstration shows how packets of arbitrary length can be received from two 100 Gbps Ethernet links at wire-speed and captured to the host memory at sustained 200 Gbps without any loss. The opposite direction of communication is also shown, i.e. how packets can be transmitted from the host memory and fully saturate two 100 Gbps Ethernet network interfaces. The reception and the transmission of data can be even shown operating simultaneously (full-duplex) without any degradation of performance in either direction. Achieved throughputs are demonstrated by counters and graphs showing live statistics of generated, received/transmitted and captured packets. We can also show detailed statistics of CPU load during the transfers of data for different packet lengths.

I. Introduction

Computer networks have shifted from being just a cheap and fast way of communication to become a platform for provisioning of a wide variety of other services (trade, advertisement, games, social media, multimedia . . .). A constant growth is visible in the number of networking capable devices, the number of provided services, the number of active users and time they spend online every day. Apart from increasing user base, the sheer amount of transferred data during each communication is also rising. These trends lead to clearly visible exponential growth in network traffic volume – every 5 years the volume of exchanged network data grows 12 times [1]. The main consequence of the described growth is a need for more powerful and faster network devices. Therefore, an apparent shift towards 100 Gbps and faster technologies (e.g. 200/400 Gbps Ethernet standardization) is currently ongoing in high-speed networks.

A common response to the need for faster devices is the utilization of hardware accelerated techniques. The main advantage of hardware acceleration is in the ability to specialize the computational architectures according to specific needs of the task at hand. This enables for tailored utilization of various parallel and pipelined data processing approaches that can lead to notable increases in overall performance of designed devices. We provide means for hardware acceleration utilization in high-speed networking through our extensive COMBO card family. The COMBO family includes many development boards focused on network data processing with industry standard PCI form factor. The heart of each card is a field-programmable gate array (FPGA) chip supplemented by memories, PCI Express connectors, power supplies, etc. Due to the flexibility of FPGA chips, the functionality of COMBO cards can be adjusted to accelerate many different use cases. In the proposed demonstration, we want to showcase the unique features and performance of our lastest (fastest) COMBO card.

The following text starts with a general discussion of the card's technology background that deals with the main features of the card itself, key FPGA firmware parameters, available IP cores, and sufficient PCI Express connections. Then a specific description of the proposed demonstration scenario and its architecture is presented. Finally, the text is concluded by assessment of our future work towards creation of an even faster FPGA accelerated card.

II. Technology Background

100 Gigabit Ethernet (100GbE) was first defined by the IEEE 802.3ba standard [2] from 2010 and currently is the fastest deployed standard of Ethernet for computer networks. It enables for transmitting data at a rate of 100 Gbps, which translates up to nearly 150 millions packets per second for the shortest ones. Please note that packet rate this high means that a new packet arrives and needs to be processed every 6.7 ns. The 100GbE standard encompasses a number of different physical layer specifications, most notable are 100GBASE-LR4 and 100GBASE-SR4 both working with four independent lanes (wavelengths of light) transmitting at 25 Gbps in a single-mode or multi-mode fiber optic cable. We demonstrated our

978-1-7281-0214-6/18/$31.00 ©2018 IEEE
DOI 10.1109/FPT.2018.00079

Fig. 1. Front view photo of NFB-200G2QL acceleration card.

COMBO-100G card back in 2014 [3] as the first FPGA accelerated PCI Express adapter card to support 100 Gbps Ethernet technology worldwide.

Since the introduction of the COMBO-100G card, we have been able to scale up the data transfers throughput from 100 Gbps to 200 Gbps on our latest NFB-200G2QL low-profile card. The card is shown in Figure 1 and it is world's first PCI Express adapter equipped with two 100GbE ports that is designed to enable wire-speed processing of traffic at full speed of both of them. This hardware-accelerated SmartNIC with FPGA utilizes unique high-speed DMA modules that enable to achieve 200 Gbps throughput of data transfers over PCI Express between the card and memory of the host computer. This high throughput makes the card ideal for deployment in the fastest backbone networks and in high-throughput data centers. Main unique features of the NFB-200G2QL include:

- two 100GbE QSFP28+ transceiver interfaces (cages) that apart from native 2×100G also supports 4×50G, 2×40G, 8×25G or 8×10G modes
- powerful Xilinx Virtex UltraScale+ VU7P FPGA chip,
- three static QDR-IIIe memories (up to 288 Mb each),
- two PCI Express Gen3 interfaces with 16 lanes each,
- PCI Express half-length and low-profile form factor,
- total power consumption of less than 65 W,
- passive cooling with NACA/NASA-style air scoop shape of fins [4] to maximize airflow and heat exchange,
- connector of external PPS input for precise synchronization of timestamps.

The card is currently in commercial production and it is available from Netcope Technologies including basic drivers and firmware IP cores [5].

The Xilinx Virtex UltraScale+ VU7P FPGA chip [6] is the heart of the card. Compared to other computing devices, such as fixed ASICs or programmable CPUs, FPGAs allow changing their internal structure by programming their firmware. A typical arrangement of the FPGA firmware for high-speed applications is a pipelined processing, which takes advantage of FPGA's inherent massive parallelism to achieve the required throughput and performance. In the case of 2×100 Gbps traffic processing, a reasonable configuration is two 512 bits wide pipelines or one 1024 bits wide clocked at 200 MHz (5 ns). In order to sustain wire-speed processing on such wide buses for every packet length (even 64 B or 65 B), special care must be taken during the design and implementation of all parts of an FPGA firmware. Therefore, we have to come up with unique approaches and novel architectures to tackle this issue in various processing engines (e.g. [7], [8], [9]).

As a base of the card's FPGA firmware, we have developed a platform for rapid development of hardware-accelerated applications. The platform includes a set of firmware IP cores, especially blocks for network interfaces (from 1GbE up to 100GbE) and a unique high-performance programmable DMA bus-master connection to the software layer via PCIe bus. The software layer consists of Linux device drivers, various tools for card management, and libraries for high-speed data transfers between the card and the host memory (proprietary SZE2 or Intel DPDK). The framework also specifies generic interfaces towards optional traffic processing pipeline in FPGA firmware. Our HaNIC design is one of the examples of such processing pipeline. It extends the functionality of a basic NIC by the support of packet parsing, filtering, and configurable hash-based distribution among multiple CPU cores.

General packet processing pipelines can be described on a rather high level of abstraction using P4 language [10], [11] and its relatively simple syntax. P4 (Programming Protocol-independent Packet Processors) is novel and open source language that evades the typical problem of conservative network approaches, such as a fixed set of supported protocols, fixed set of actions (functionality), and so on. The main purpose is to provide a way to define the packet processing functionality of network devices, paying attention to reconfigurability in the field, protocol independence and target (platform) independence. Usage of P4 language to directly describe the functionality of firmware processing architectures is enabled thanks to our unique P4-to-VHDL compiler [12]. It can transform P4 source code directly to synthesizable VHDL implementations of high-speed pipelines specifically tailored for FPGAs. The internal structure of the individual stages of the generated pipeline is inspired by hand-written modules which were developed by a skilled HDL programmer. This way the FPGA on the NFB-200G2QL card (and other similar SmartNICs) can be easily adjusted to perform different operations on passing network data based on the requirements of specific use cases.

Two PCI Express endpoints are needed as a workaround of the missing Gen3 ×32 PCIe endpoint support in current FPGAs, motherboards, and CPUs. This is required because the effective throughput of PCIe Gen3 ×16 is only slightly above 100 Gbps. Therefore, our card utilizes two ×16 slot in order to achieve required 200 Gbps throughput into the host memory. A possible alternative would be the utilization of

new and faster PCI Express Gen4 standard. This way, only one $\times 16$ interface should be sufficient for 200 Gbps effective data throughput. But again, current FPGAs, motherboards and CPUs do not support the Gen4 $\times 16$ interface yet. Furthermore, using two PCIe endpoints enables direct data transfers between the card and two physical CPUs (NUMA nodes) without the QPI bottleneck.

III. Demo Description

The goal of the proposed demonstration is to present the unique performance of the low-profile NFB-200G2QL card and its high-speed DMA bus-master module. A similar (simpler) demonstration has already been performed at [13], [14]. We want to especially stress out the ability of the card to:

- process data at wire-speed for both fully saturated 100 GbE interfaces without any packet loss,
- capture received traffic via PCIe into the host memory at sustained 200 Gbps regardless of the frame length,
- replay generated traffic from the host memory via PCIe at sustained 200 Gbps regardless of the frame length,
- and even perform the capture and the replay operations simultaneously without any degradation of performance.

In order to demonstrate these features, we prepared demo architecture as illustrated in the Figure 2.

The NFB-200G2QL card is connected into two PCIe slots of standard server motherboard with two (or one) relatively fast multicore CPUs and fully filled memory banks (for maximal memory throughput). Inside the card's FPGA there is our HaNIC firmware configured to capture all of the incoming traffic and distribute it through PCIe among available CPU cores. It can also simultaneously send all of the traffic generated at CPU cores towards network ports. On the software side, packet capture and generation (replay) are performed by simple multicore tools implemented in C. They utilize our proprietary SZE2 kernel bypass API that stores packets in ring buffers inside the host memory and provides direct (zero-copy) access to them. Alternatively, standard Intel DPDK API can be used instead, but it is not as optimal as our SZE2 (higher CPU and/or PCIe overhead). Finally, live statistics gathered from software tools and selected FPGA firmware modules are displayed in GUI at the screen. Also, parameters like operation mode and packet lengths can be changed from the GUI.

Both Ethernet ports of the card should be connected to a tester device that can generate and receive (analyze) 100GbE network traffic at wire-speed. But since conventional hardware testers supporting 100 GbE ports (e.g. in our labs we use Spirent TestCenter) are too large and heavy to transport, we can instead implement the required traffic generation and capture capabilities inside the FPGA firmware and connect the optic cables in a loopback. This further shows the versatility of the FPGA firmware.

Described demo architecture can operate in three basic modes: packet capture, packet replay, and simultaneous capture and replay.

In packet capture mode, packets of configurable length are generated at the maximum allowed rate and sent over the fiber into both 100 Gbps Ethernet ports. There, the packets are received by on-card PMA, PCS and MAC engines, distributed into multiple DMA channels and transferred via 2 PCIe endpoints at 200 Gbps into the ring buffers inside server's main memory. In the memory, the packets are accessed and counted. Processing has the form of only a simple accounting because we want to demonstrate that the card is capable of delivering the 200 Gbps of data into the software and not the performance of some specific advanced packet processing in the CPU cores. Finally, live packet capture performance statistics are shown in the GUI on the screen. This mode corresponds to typical network monitoring or security scenarios, where traffic of both directions of a tapped 100GbE link needs to be processed.

In packet replay mode, packets of configurable length are prepared by CPU cores in the host memory and copied into multiple DMA ring buffers. From there, they are picked up by the DMA controllers in the card's FPGA and transferred via PCIe into its local buffers. Then, they are transferred using standard Ethernet layers and transceivers onto two optical 100 GbE lanes. Finally, live sending performance statistics are shown in the GUI on the screen. This mode corresponds to data center deployment, where large amounts of data need to be transferred.

During simultaneous packet capture and replay mode, packets are prepared by the CPU cores and sent through PCIe onto the optical Ethernet links in the same fashion as in the packet replay mode. Then, instead of just counting and dropping them in the FPGA firmware right after the reception, they are all forwarded to DMA controller and back into the host memory in the same fashion as in the packet capture mode. So the DMA module and the PCIe bus are constantly transferring data at full speed in both directions (full-duplex operation).

IV. Future Work

Moving from the first FPGA card with 100 Gbps throughput to the current NFB-200G2QL as the first 200 Gbps card, we want to keep continuing a similar trend and be the first to reach 400 Gbps. The effort to design practical 400 Gbps FPGA card brings some interesting challenges. Although 400 Gigabit Ethernet (400GbE) was defined recently, by the IEEE 802.3bs standard [15] from the end of 2017, it most notably introduced PAM4 encoding on the physical layer. PAM4 doubles the number of bits in serial data transmissions by increasing the number of available modulation levels from 2 to 4 but does so at the cost of higher noise susceptibility. To counteract this, nontrivial RS-FEC computation was also added into the standard as a mandatory part of the basic Ethernet encoding and decoding process. Furthermore, PCIe Gen3 is not practical for transferring 400 Gbps of data between the card and the host memory as something like one $\times 64$ or four $\times 16$ interfaces would be needed. Utilization of newer and two times faster PCIe Gen4 is the more reasonable choice. Major FPGA vendors currently do not offer any production chips with both PAM4 and PCIe Gen4 support that would be sufficient for the 400 Gbps card. But, we expect that some interesting new FPGAs should become available in the following year or two.

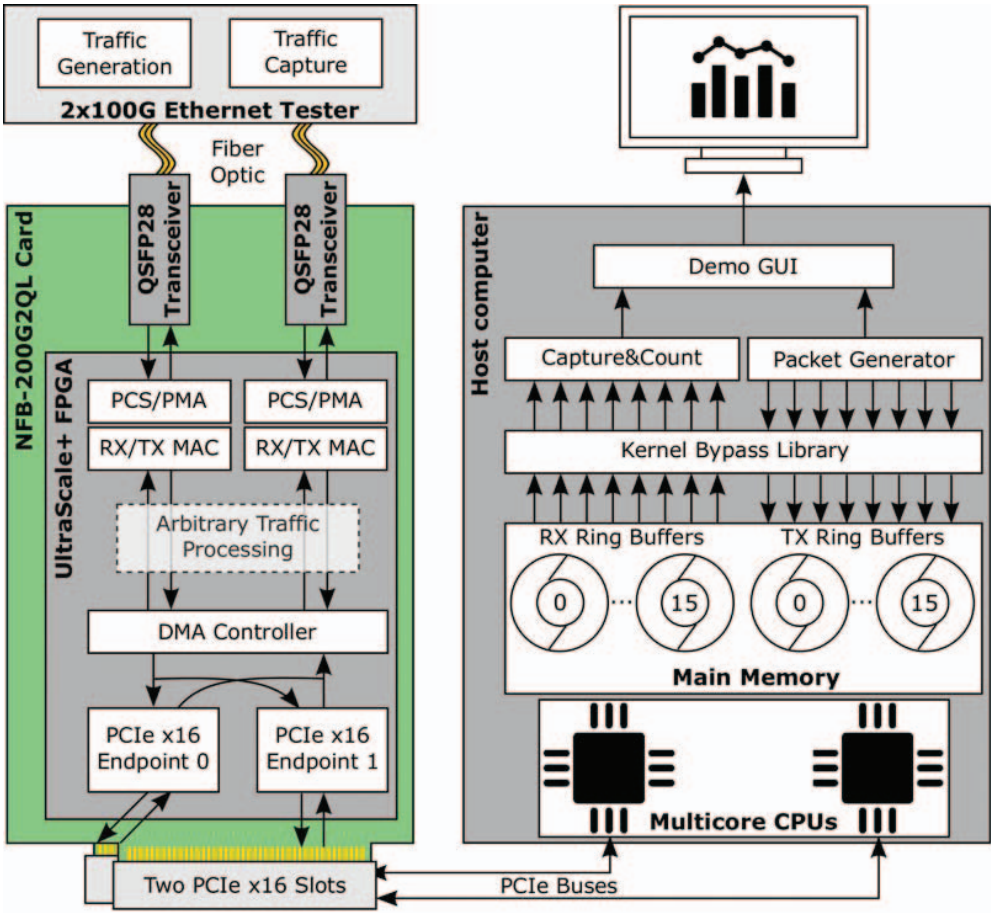

Fig. 2. Logical scheme illustrating key components of NFB-200G2QL demonstration system architecture.

ACKNOWLEDGMENT

This material is based upon research work supported by the project Reg. No. CZ.02.1.01/0.0/0.0/16_013/0001797 co-funded by the Ministry of Education, Youth and Sports of the Czech Republic, and by the Technology Agency of the Czech Republic project TH02010214.

REFERENCES

[1] IEEE 802.3 Ethernet Working Group, "IEEE 802.3 industry connections Ethernet bandwidth assessment," IEEE, San Diego, CA, USA, Tech. Rep., July 2012. [Online]. Available: http://www.ieee802.org/3/ad_hoc/bwa/BWA_Report.pdf

[2] IEEE Computer Society, "Amendment 4: Media access control parameters, physical layers, and management parameters for 40 Gb/s and 100 Gb/s operation," *IEEE Standard 802.3ba-2010*, pp. 1–457, June 2010.

[3] L. Kekely, J. Kořenek, V. Puš, and M. Dvořák, "100G packet capture live demo," *24th International Conference on Field Programmable Logic and Applications (Demo Night)*, September 2014.

[4] C. W. Frick, W. F. Davis, L. Randall, and E. A. Mossman, "An experimental investigation of NACA submerged-duct entrances," *NACA ACR No. 5120*, November 1945.

[5] Netcope Technologies, "NFB-200G2QL FPGA-based hardware," *White Paper: Product Brief*, September 2018. [Online]. Available: https://www.netcope.com/en/resources/nfb-200g2ql-product-brief

[6] Xilinx, "UltraScale architecture and product data sheet: Overview," *Preliminary Product Specification DS890 (v3.2)*, pp. 1–44, January 2018.

[7] J. Cabal, P. Benáček, L. Kekely, M. Kekely, V. Puš, and J. Kořenek, "Configurable FPGA packet parser for terabit networks with guaranteed wire-speed throughput," in *Proceedings of the 2018 ACM/SIGDA International Symposium on Field-Programmable Gate Arrays*. Association for Computing Machinery, 2018, pp. 249–258.

[8] M. Kekely, L. Kekely, and J. Kořenek, "Memory aware packet matching architecture for high-speed networks," in *Proceedings of the 21st Euromicro Conference on Digital Systems Design*. IEEE, 2018.

[9] J. Cabal, L. Kekely, and J. Kořenek, "High-speed computation of CRC codes for FPGAs," in *2018 International Conference on Field Programmable Technology (ICFPT)*. IEEE, 2018, (Accepted).

[10] P. Bosshart, D. Daly, G. Gibb, M. Izzard, N. McKeown, J. Rexford, C. Schlesinger, D. Talayco, A. Vahdat, G. Varghese, and D. Walker, "P4: Programming protocol-independent packet processors," *SIGCOMM Computer Communication Review*, vol. 44, no. 3, pp. 87–95, Jul. 2014.

[11] P4 Language Consortium, "P4," September 2018. [Online]. Available: http://p4.org/

[12] P. Benáček, V. Puš, H. Kubátová, and T. Čejka, "P4-To-VHDL: Automatic generation of high-speed input and output network blocks," *Microprocessors and Microsystems*, vol. 56, pp. 22–33, 2018.

[13] L. Kekely, M. Špinler, Š. Friedl, J. Sikora, and J. Kořenek, "Live demonstration of FPGA based networking accelerator for 200 Gbps data transfers," *The 30th IEEE/IFIP Network Operations and Management Symposium (NOMS)*, April 2018.

[14] ——, "Accelerated wire-speed packet capture at 200 gbps," *28th International Conference on Field Programmable Logic and Applications (Demo Night)*, August 2018.

[15] IEEE Computer Society, "Amendment 10: Media access control parameters, physical layers and management parameters for 200 Gb/s and 400 Gb/s operation," *IEEE Standard 802.3bs-2017*, pp. 1–372, 2017.