IEEE INFOCOM 2023 - IEEE Conference on Computer Communications Workshops (INFOCOM WKSHPS) | 978-1-6654-9427-4/23/$31.00 ©2023 IEEE | DOI: 10.1109/INFOCOMWKSHPS57453.2023.10225750

400G Ethernet Packet Capture Demo Based on Network Development Kit for FPGAs

Jakub Cabal, Vladislav Válek, Martin Špinler, Daniel Kondys, Jan Kořenek
CESNET, z.s.p.o.
Zikova 4, 160 00 Prague, Czech Republic
Email: cabal,vladislav.valek,spinler,kondys,korenek@cesnet.cz

I. Abstract

CESNET is ready to present a packet capture demo at 400G networks. As part of its research activities, it has developed a new system for FPGA cards for fast packet reception to DPDK. To simplify the hardware development, we created an open-source Network Development Kit (NDK) framework. Although the demo is presented only on the 400G card, which we have developed in collaboration with Reflex CES, the open-source NDK framework and fast packet capture are available on many other FPGA cards. The open-source framework is ready to use, and we would be happy if you not only use it but also contribute to its further development.

II. Related Works

Currently, there are several open-source frameworks for developing network applications on FPGA cards. NetFPGA PLUS [1] and OpenNIC [2] target only cards based on Xilinx FPGAs and support only 100G Ethernet. Corundum [3] brings support for cards with FPGA from multiple vendors. In addition to the FPGA multi-vendor (Xilinx/AMD and Intel) support, the NDK is the first to provide support for modern technologies such as 400G Ethernet and PCIe Gen5. Another key feature of the NDK is novel bus architecture that enables processing of multiple packets per clock cycle.

III. Hardware Configuration

For the demo, we have selected the Reflex CES XpressSX AGI-FH400G with Agilex I-Series FPGA [4] as the target FPGA card. This card contains a QSFP-DD slot for connecting a 400G Ethernet (400GBASE) module and up to two PCIe Gen5 interfaces (edge connector + expansion PCIe board). The card is plugged into the server with AMD Ryzen 3960X CPU and 64 GB of DDR4 RAM overclocked to 3600 MHz. To achieve 400 Gbps throughput, the FPGA card is connected to the PC using two PCIe slots, each in a Gen4 x8x8 (bifurcated) configuration. The interconnection is shown in Figure 1.

IV. NDK Framework for FPGA Cards

The 400G demo firmware utilizes the NDK Minimal application, which is available as open-source on GitHub [5]. The demo firmware also includes a very fast DMA Module, also known as the DMA Medusa IP [6]. This DMA IP supports multiple PCIe endpoints and multiple independent DMA queues.

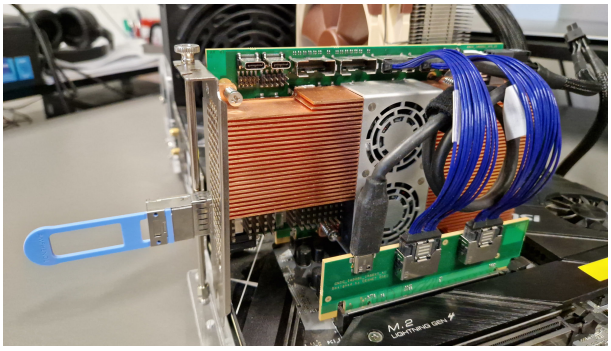

Fig. 1. Reflex CES XpressSX AGI-FH400G card in host PC.

The NDK framework dedicates the Application Core to a custom application (user logic). The NDK is designed for network applications processing packets in a deep pipeline. Depending on the specific FPGA card, there can be multiple streams for receiving and sending Ethernet packets, as well as multiple streams for sending packets to the host CPU through the DMA Module. There is also a bus for the configuration of the user application from software. Ethernet and DMA streams use a combination of MFB (Multi Frame Bus) for packet data and MVB (Multi Value Bus) for packet headers and metadata. The Multi Buses support multiple packets/values in a single word, which allows to achieve high throughput on short packets. The principles of Multi Buses were presented in [7]. The NDK also includes many open-source IPs that can be used in user applications.

The demo firmware implements direct connections between the DMA streams and the Ethernet without any packet processing in the application core. A round-robin packet distribution to the DMA queues is implemented in the direction from FPGA to host PC. The firmware also includes a debug module that contains optional loopback paths, a packet generator, and throughput meters.

V. NDK Software Stack

The NDK framework is provided with a complete software stack including Linux kernel and DPDK driver. The kernel-level driver handles all communication with the PCIe device. In the NDP mode, the kernel driver also provides full control of the DMA transfers. In the DPDK mode, the kernel-level driver only provides a low-level API that allows DPDK drivers

978-1-6654-9427-4/23/$31.00 ©2023 IEEE

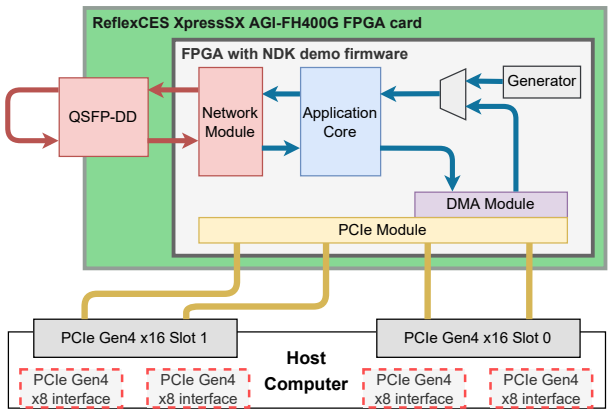

Fig. 2. Block diagram of the 400G Packet Capture Demo.

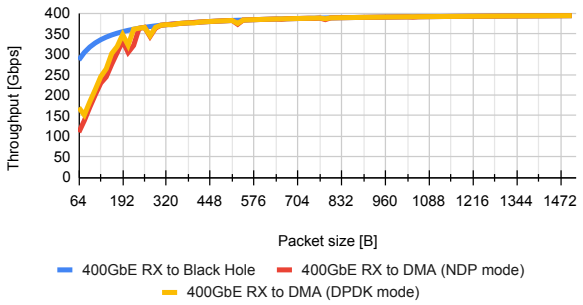

Fig. 3. Performance chart of the 400G Packet Capture Demo.

to fully control the DMA transfers. The NDK software library is available for userspace-level tools and provides a simple API for accessing PCIe devices and the DMA transfers.

A set of base software tools for working with an NDK firmware is also available. Some tools is used to configure the NDK firmware and its components, such as the network interfaces. The other tools allow packet transfers via the DMA using PCAP files or empty debug packets.

VI. Demo Description

The principle of the demo is shown in Figure 2. Ethernet packets with a software-configurable length are generated inside the FPGA at full speed. The packets are sent out of the card through the Network Module and are received back in the FPGA after passing through an external loopback. In the Application Core, packets are distributed to individual DMA queues using a round-robin method to ensure even CPU core utilization. The DMA Medusa IP then ensures transfers of these packets to the RAM memory of the host PC. The DMA IP provides 32 DMA queues in the demo firmware. The transfers are done via two PCIe Gen4 x16 slots bifurcated into four x8 interfaces, which ensure sufficient throughput. The Ethernet packet generator is controlled by a Python script, which simultaneously shows throughput measured using the throughput meters in the firmware. This script also runs the ndp-read tool, which controls DMA transfers in the RX (FPGA to host RAM) direction.

The achieved throughput is shown in Figure 3. With the DMA disabled (blue line, packets are discarded at the DMA input), the theoretical maximum throughput of 400 Gigabit Ethernet is achieved. With the DMA enabled (red line), throughput is lower on packet lengths up to 256 bytes due to higher overhead of DMA transfers and PCIe bus. A critical point for the DMA transfers is also the performance of the host PC memory controller; therefore, it is necessary to overclock the RAM memory of the host PC to achieve the 400 Gbps throughput. The ndp-read tool runs the DMA transfers in the NDP mode, using the driver loaded in the kernel. The demo can also be run in the DPDK mode with comparable performance (yellow line). In this mode a DPDK driver controls the DMA transfers directly.

VII. Conclusions and Future Work

The demo showed a 400G packet capture based on an open-source NDK framework and an FPGA card. The demo firmware achieved wire-speed throughput for packet lengths above 256 bytes. The NDK is open-source, provides a standard DPDK interface, and can integrate custom hardware acceleration to speed up the target application. The NDK is thus also a suitable platform for the easy implementation of academic experiments using modern technologies.

Acknowledgment

This work was supported by Security Research grant VJ02010024 in the project Flow-based Encrypted Traffic Analysis granted by the Ministry of Interior of the Czech Republic and by the project E-INFRA CZ (LM2023054) granted by the MEYS of the Czech Republic.

References

[1] University of Cambridge. NetFPGA PLUS. Retrieved January 9, 2023, from https://github.com/NetFPGA/NetFPGA-PLUS
[2] AMD (Xilinx). AMD OpenNIC Project. Retrieved January 9, 2023, from https://github.com/Xilinx/open-nic
[3] A. Forencich, A. C. Snoeren, G. Porter and G. Papen, "Corundum: An Open-Source 100-Gbps NIC," 2020 IEEE 28th Annual International Symposium on Field-Programmable Custom Computing Machines (FCCM), 2020, pp. 38-46, doi: 10.1109/FCCM48280.2020.00015.
[4] Reflex CES. XpressSX AGI-FH400G. Retrieved January 9, 2023, from https://www.reflexces.com/pcie-boards/intel-agilex-i-series-soc/xpresssx-agi-fh400g
[5] CESNET. Minimal application based on Network Development Kit (NDK) for FPGA cards. GitHub. Retrieved January 9, 2023, from https://github.com/CESNET/ndk-app-minimal
[6] J. Kubálek, J. Cabal, M. Špinler and R. Iša, "DMA Medusa: A Vendor-Independent FPGA-Based Architecture for 400 Gbps DMA Transfers," 2021 IEEE 29th Annual International Symposium on Field-Programmable Custom Computing Machines (FCCM), 2021, pp. 258-258, doi: 10.1109/FCCM51124.2021.00045.
[7] L. Kekely, J. Cabal, V. Puš and J. Kořenek, "Multi Buses: Theory and Practical Considerations of Data Bus Width Scaling in FPGAs," 2020 23rd Euromicro Conference on Digital System Design (DSD), 2020, pp. 49-56, doi: 10.1109/DSD51259.2020.00020.
[8] CESNET. The NDK software framework. GitHub. Retrieved January 9, 2023, from https://github.com/CESNET/ndk-sw