Case Study: Usage of High Level Synthesis in HPC Networking

[Extended Abstract]

Pavel Benáček
CESNET, a.l.e.
Zikova 4, 160 00
Prague, Czech Republic
benacek@cesnet.cz

Lukáš Richter
Netcope Technologies, a.s.
Sochorova 3232, 616 00
Brno, Czech Republic
richter@netcope.com

Michal Kekely
Netcope Technologies, a.s.
Sochorova 3232, 616 00
Brno, Czech Republic
kekely@netcope.com

Viktor Puš
Netcope Technologies, a.s.
Sochorova 3232, 616 00
Brno, Czech Republic
pus@netcope.com

ABSTRACT

Networking is indeed one of the critical components of any HPC system, affecting significantly its overall cost, performance and ease of use. And while there is a massive shift towards using high level tools and languages (such as OpenCL) for the computation part of HPC, the networking has traditionally been rather fixed-function monolith. Taking an inspiration from Software Defined Networking and Network Functions Virtualization, also the HPC world has the opportunity to switch to much more flexible network programming paradigm. The P4 language [2, 1] was originally designed for a description of networking switches' dataplane. However, we examine using it in a related domain: SmartNICs. We see the P4 as an interface for application designers to formally express their requirements for the functionality of the network. This achievement can be further extended by embedding hardware accelerators (such as modules for pattern matching, cryptography, etc.) into the FPGA based SmartNIC. Network-related parts of the compute load can be then moved into the networking hardware, effectively blurring the edge between the HPC network and compute components. The work of FPGA firmware developers is by no means replaced by the P4 to FPGA compiler. Rather, firmware developers can spend less time over small changes and other mundane tasks and will be able to focus on more complex tasks, such as effective architectures and algorithms for network-compute accelerators. In order to demonstrate the feasibility of P4 for high-speed packet processing in FPGA, we use two existing and mature FPGA-based projects: Netcope Packet Capture is an FPGA firmware capable of 100 Gbps line rate traffic filtering and forwarding to output network ports or host RAM via PCI Express. Netcope Session Filter is a large capacity exact match packet filtering firmware with flow state memory, which is able to keep NetFlow/IPFIX records and to export flow aggregates to RAM periodically. Both firmwares have been hand-written in VHDL and maintained for several years, making full advantage of expert low-level code optimizations for FPGA resources and frequency. The clear disadvantage of hand-based maintenance is the low flexibility when adding new functionality, such as protocol support. The talk will present how individual existing firmware features map to P4 language constructs, to be then automatically compiled into FPGA SmartNIC. We will cover challenges and outcomes associated with converting these two pre-existing firmwares to P4-generated pipelines.

H2RC'17 November 17, 2017, Denver, CO, USA

DOI: 10.1145/1235

CCS Concepts

•Networks → Network components; •Hardware → Networking hardware; Hardware description languages and compilation;

Keywords

FPGA, SmartNIC, P4, HPC, 100 Gbps

1. ACKNOWLEDGMENTS

This research has been partially supported by the the "CESNET E-infrastructure" project LM2015042 funded by the Ministry of Education, Youth and Sports of the Czech Republic and by the "NFV200" project TH02010214 funded by the Technology Agency of the Czech Republic.

2. REFERENCES

[1] The P4 Language Consortium. $P4_{16}$ language specification, May 2017.

[2] The P4 Language Consortium. The P4 Language Specification, May 2017.