OCT: The Open Cloud FPGA Testbed

Suranga Handagala
Northeastern University
Boston, MA
Email: s.handagala@northeastern.edu

Martin Herbordt
Boston University
Boston, MA
Email: herbordt@bu.edu

Miriam Leeser
Northeastern University
Boston, MA
Email: mel@coe.neu.edu

***Abstract*—The Open Cloud Testbed is a US National Science Foundation (NSF) funded project to promote and support cloud research. The full title is "Developing a Testbed for the Research Community Exploring Next-Generation Cloud Platforms." The project is funded under the NSF CISE Community Research Infrastructure program at the Grand scale. OCT builds on the Massachusetts Open Cloud (MOC) and CloudLab. OCT is unique among FPGAs in the cloud in the extent that it reaches out to the cloud research community.**

1. Introduction

The aim of the Open Cloud Testbed (OCT) [1] is to construct and support a testbed for research and experimentation into new cloud platforms including both hardware and software. The OCT combines proven software technologies with a real production cloud enhanced with Field Programmable Gate Arrays (FPGAs). The combination of a testbed and production cloud allows a) larger scale compared to isolated testbeds, b) reproducible experimentation based on realistic user behavior and applications, and c) a model for transitioning successful research results to practice.

The OCT differs from many existing FPGAs in cluster projects in that it aims to bring together cloud researchers and hardware innovators. It builds on previous research from CloudLab and the Mass Open Cloud (MOC).

2. Current Status: Xilinx FPGAs in the OCT

At the time of FPL 2021, there are eight Xilinx Alveo U280 FPGA accelerator cards available to any US researcher. The topology is shown in Figure 1. FPGAs are connected to a host via PCIe and also directly connected to the network and each other. Each FPGA is programmed with a shell that provides core infrastructure for running applications. The PCIe connection allows users to program the FPGA, and also to configure its compute kernels in the user partition and perform host-to-FPGA/FPGA-to-host memory transfers. The U280 has two QSFP28 ports for high speed network access, providing a bandwidth up to 100 Gbps, and one of them on each U280 is connected to a Dell Z9100-ON data center switch using 100G passive direct attach copper (DAC) cables. By using such a topology, the

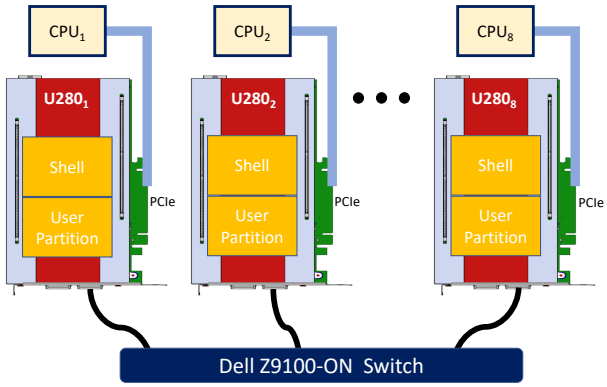

Figure 1. Eight Alveos in the OCT.

FPGAs can communicate with one another using network and ethernet layer infrastructure implemented on the user partition. For the ethernet layer, we use Xilinx UltraScale+ Integrated 100G Ethernet Subsystem available with a no charge license. We support two network layer implementations that are publicly available; the UDP stack developed by the Xilinx University Program [2], and the TCP stack developed by ETH Zurich [3].

2.1. Tools and Access for Users

The tool flow to target the FPGAs is shown in Fig. 2. We have two platforms; a development platform and a target platform. The development platform is a compute instance in MOC which has the necessary development tools such as Vitis pre-installed. Users can install other tools required for application development depending on their requirements. The target platform is any of the eight Cloudlab servers equipped with a U280 accelerator card. Both MOC and cloudlab instances are based on OpenStack [4].

An MOC user develops FPGA applications by creating and launching a compute instance by first selecting an OpenStack flavor subject to memory and storage quota requirements. Because MOC users have root access, they can install development tools themselves. However, we do not encourage this approach because downloading and installing Vitis is tedious and can take a few hours to complete.

Instead, the pre-installed Vitis image that we provide can be used to create an instance and start developing applications in a short amount of time. As an example, we have used the publicly available FINN compiler to implement a quantized neural network inference accelerator for a single U280 target. Under the FINN workflow, the user needs to select a pre-trained neural network model, quantize it using Brevitas, and perform various transformations until it is ready for bitstream generation. Following this, Vitis should be used to generate a platform specific binary file.

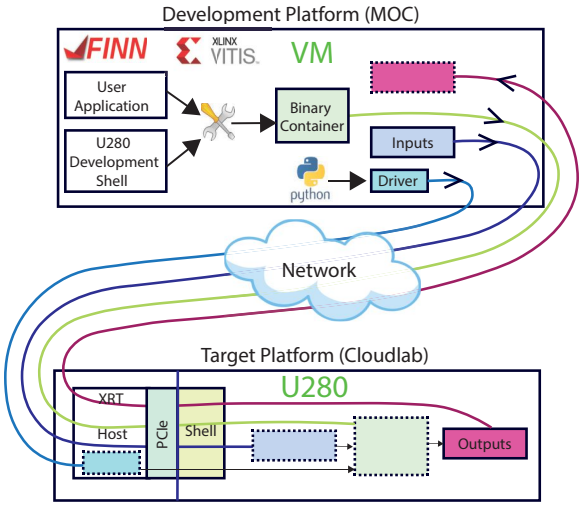

Figure 2. Tool Flow for the OCT

Cloudlab gives researchers a bare metal machine to do cloud research on and also to ensure security and privacy between subsequent users. Similar to MOC, Cloudlab users will also need to request an account and get approval to use the platform. The FPGAs are not bare metal, but have the Xilinx U280 XDMA shell installed. The shell is considered a trusted partition which provides platform security as well as communication infrastructure such as PCIe between the FPGA and the host. The FPGA binary that will reside in the user partition is programmed via the shell using dynamic function exchange (DFX). The user partition is considered untrusted, and all the transactions to and from this partition are monitored using built-in firewalls to ensure the security of the shell.

Once the application is developed and bitstream generation is completed at the MOC end, the bitstream, input stimulus, and a PYNQ driver are transferred over the network to the Cloudlab server. The driver is used to set various attributes of the accelerator kernel such as batch size, execution mode, etc. Xilinx Run Time (XRT) is installed on the Cloudlab host which provides a low level communication layer between the host and the FPGA. After remotely executing the accelerator kernel, users can examine the results using the same MOC server.

3. Next Steps

One of the goals of the next phase is to deploy Intel FPGAs with a basic tool chain, analogous to the one currently supported for the Xilinx FPGAs, and also running COPA (COnfigurable network Protocol Accelerator). COPA provides a customizable framework that integrates communication and computation on an FPGA and incorporates SmartNIC capabilities [5]. The hardware environment provides networking and accelerator infrastructure while the software abstracts the FPGA from the underlying application or middleware. The API is based on the OpenFabric Interface (OFI) with extensions. COPA configured FPGAs can be used as either SmartNICs or standalone nodes. COPA supports two accelerator models, *in-line* for transforming streaming data and *look-aside* which is a traditional SmartNIC or accelerator model. In recent work, OpenSHMEM has been integrated into COPA [6]; this will provide direct support for widely used HPC Middleware.

We envisage using the Intel/COPA FPGAs to support a number of research project types. One is simply as a robust network interface. A second is as a shell to develop SmartNIC applications; e.g. we have implemented the lossy compression framework SZ as a look-aside application (GhostSZ). A third is to develop additional middleware support, e.g., for MPI collectives. A fourth is to integrate one or more of these into an HPC codesign flow.

4. Conclusion

The Open Cloud Testbed is a resource available now for Us researchers looking for FPGAs to support cloud research. The testbed and software stacks involve researchers from both the cloud and FPGA community and provide flexibility in the form of bare metal nodes for cloud research experiments, FPGAs programmable by users, and FPGAs connected directly to the network and to one another.

Acknowledgments

This project is supported in part by the National Science Foundation under Grant No. CNS 1925658.

References

[1] "Open Cloud Testbed(OCT) – Mass Open Cloud." [Online]. Available: https://massopen.cloud/connected-initiatives/open-cloud-testbed/

[2] Xilinx, "XUP Vitis Network Example (VNx)," 2020. [Online]. Available: https://github.com/Xilinx/xup_vitis_network_example

[3] ETH Zurich, "Vitis with 100 Gbps TCP/IP Network Stack," 2020. [Online]. Available: https://github.com/fpgasystems/Vitis_with_100Gbps_TCP-IP

[4] "Open Source Cloud Computing Infrastructure - OpenStack." [Online]. Available: https://www.openstack.org/

[5] V. Krishnan, O. Serres, and M. Blocksome, "COnfigurable Network Protocol Accelerator (COPA)," *IEEE Micro*, vol. 41, no. 1, p. 8–14, Jan. 2021.

[6] D. Ozog, A. Kot, and V. Krishnan, "Accelerating HPC Runtimes such as OpenSHMEM with COPA," in *OFA Virtual Workshop*, 2021.