High-Performance Custom Computing with FPGA Cluster as an Off-loading Engine

Takaaki Miyajima, Tomohiro Ueno, Atsushi Koshiba, Jens Huthmann, Kentaro Sano, and Mitsuhisa Sato
Center for Computational Science, RIKEN
Kobe, Hyogo, Japan
takaaki.miyajima@riken.jp

ABSTRACT

A heterogeneous system with Field Programmable Gate Arrays (FPGAs) is gathering attention in High-Performance Computing (HPC) area. We have been researching and developing a high-performance custom computing with FPGA cluster as an off-loading engine for an existing HPC system. In this research poster, we describe the motivation of our research and present research topics on a software bridge between the FPGA cluster and existing HPC servers, and dedicated inter-FPGA networks.

CCS CONCEPTS

• **Computer systems organization** → **Embedded systems**; *Redundancy*; Robotics; • **Networks** → Network reliability.

KEYWORDS

FPGA cluster, computer systems organization, software bridge

ACM Reference Format:
Takaaki Miyajima, Tomohiro Ueno, Atsushi Koshiba, Jens Huthmann, Kentaro Sano, and Mitsuhisa Sato. 2018. High-Performance Custom Computing with FPGA Cluster as an Off-loading Engine. In *Woodstock '18: ACM Symposium on Neural Gaze Detection, June 03–05, 2018, Woodstock, NY*. ACM, New York, NY, USA, 2 pages. https://doi.org/10.1145/1122445.1122456

1 INTRODUCTION

As the CMOS technology scaling is slowing down, hardware customization for target applications is gathering attention in HPC area. Recent large scale FPGAs can be applied to such purpose. An FPGA is a reconfigurable device which allows us to perform non-Von Neumann computing with a specialized architecture. It can provide a combination of high computation throughput, low latency, and good power efficiency by customizing data path, cache, and memory subsystem. An earlier study reported that many applications such as computational fluid dynamics [9] or data-base[3] are accelerated by using single FPGA. Furthermore, some kind of applications have very low efficiency in terms of performance and scalability for already existing HPC systems without FPGAs. Consequently, we believe that one of the promising approaches is a combination of an existing HPC system and multiple FPGAs to cover wider applications by off-loading some of them. Although a few examples of large scale heterogeneous systems with FPGA are emerging [2][6], it is still challenging to efficiently utilize FPGAs combined with an existing HPC system. We have the following two main problems to achieve this goal.

(1) How should we combine FPGA cluster with an existing HPC system?
(2) How should we achieve high performance custom computing by off-loading tasks to FPGA clusters?

We describe a fundamental ideas of our system and present current status in the following sections.

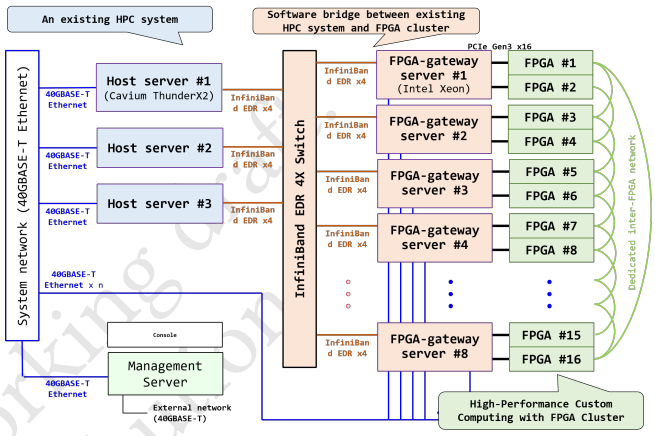

Figure 1: Our system consists of three main part; an existing HPC system, software bridge for FPGA cluster, and high-performance custom computing with FPGAs.

2 COMBINE FPGA CLUSTER WITH AN EXISTING HPC SYSTEM

For a feasibility study, we choose the most practical way to combine FPGA cluster with an existing HPC system such as Fugaku, the Japanese next-generation HPC system. Inside of the FPGA cluster, FPGAs and host CPUs are tightly coupled. FPGA cluster is loosely connected to an existing HPC system since most chassis of current HPC system doesn't have extra room to install FPGA card. In this situation, the FPGA cluster requires some features. First, feature to enable host servers to access remote FPGAs to use remote FPGAs transparently and efficiently. Second, feature to make

Woodstock '18, June 03–05, 2018, Woodstock, NY

ACM ISBN 978-1-4503-9999-9/18/06…$15.00
https://doi.org/10.1145/1122445.1122456

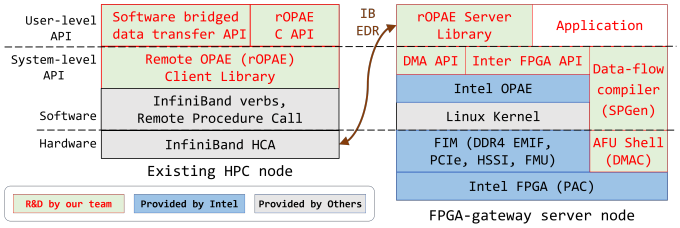

Figure 2: System stack for our system.

existing HPC node can handle arbitrary FPGAs. Third, feature to decouple a 1:1 ratio of host CPUs and FPGAs.

Our FPGA cluster is a high-performance custom computing part and used as an off-load engine from the viewpoint of an existing HPC system. Figure 1 illustrates the architectural overview of our system. We assume that some kernels run on FPGAs but the others run on the host servers. An existing HPC system controls manages and communicates with the FPGA cluster.

So as to combine FPGA cluster with an existing HPC system while satisfying above described requirements, we have developed a bridging function. Figure 2 illustrates a system stack of the bridging function. Intel OPAE [1] is an open source program to control, manage and communicate with FPGA on a single node. Remote OPAE (rOPAE) is an extension of OPAE which enables the same functions from a remote node. rOPAE server library on FPGA-gateway server node works as a bridge. It handles messages or data from rOPAE client library on the host node and control FPGAs. It is flexible but causes overhead since the bridge is similar to a software switch. Software bridged data transfer (SBDT) function is developed to transfer data between two parts[7]. Preliminary evaluation on an older version FPGA shows that SBDT achieved 81.0% of theoretical bandwidth of PCIe Gen3 x8.

3 HIGH PERFORMANCE CUSTOM COMPUTING WITH FPGA CLUSTER

It is also a challenging problem how should we achieve high-performance custom computing with FPGA cluster. There are many problems to be solved for this goal. We have been researching and developing research topics such as a dedicated inter-FPGA network [5], programming model [4], and compiler for FPGAs[8]. Regarding the inter-FPGA network, we evaluate the is three kinds of networks shown in Figure 3; (a) 1D ring network without a router, (b) 1D or 2D torus network with a router, and (c) Ethernet-based network with switch. Research of programming model to transparently use FPGAs from the viewpoint of the programmer is in progress. We have been developing a communication system on the FPGA node for (b) and (c) to measure the sustained latency and bandwidth. It is designed for low-latency and high-throughput. The system consists of a router, a flow controller, a serial transceiver, and a remote DMA controller.

4 CONCLUSION

It is still a challenging how do we efficiently utilize multiple FPGAs and combine them with an existing HPC system. We are researching and developing multiple research topics to achieve this

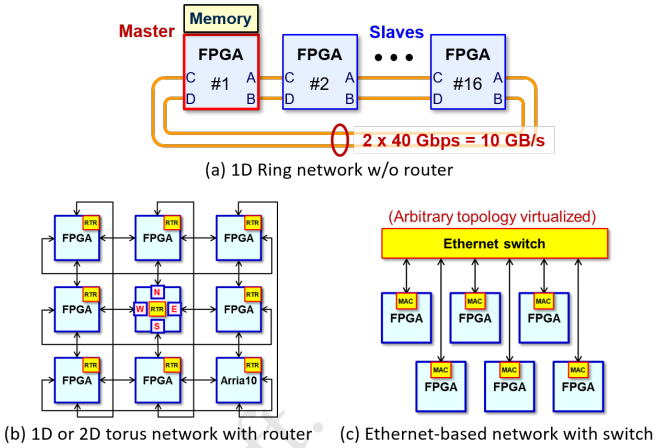

Figure 3: Three kinds of inter-FPGA networks for tightly-coupled FPGA cluster

goal. Current status of two research topics; software bridge for two parts and high-performance custom computing with FPGA cluster are presented.

ACKNOWLEDGMENT

This work was partially supported by JSPS KAKENHI Grant Numbers 17H01706, and 19K20282.

REFERENCES

[1] [n.d.]. Open Programmable Acceleration Engine, Intel - GitHub. https://github.com/OPAE.

[2] A. M. Caulfield, E. S. Chung, A. Putnam, H. Angepat, J. Fowers, M. Haselman, S. Heil, M. Humphrey, P. Kaur, J. Kim, D. Lo, T. Massengill, K. Ovtcharov, M. Papamichael, L. Woods, S. Lanka, D. Chiou, and D. Burger. 2016. A cloud-scale acceleration architecture. In *2016 49th Annual IEEE/ACM International Symposium on Microarchitecture (MICRO)*. 1–13. https://doi.org/10.1109/MICRO.2016.7783710

[3] E. S. Fukuda, H. Inoue, T. Takenaka, Dahoo Kim, T. Sadahisa, T. Asai, and M. Motomura. 2014. Caching memcached at reconfigurable network interface. In *2014 24th International Conference on Field Programmable Logic and Applications (FPL)*. 1–6. https://doi.org/10.1109/FPL.2014.6927487

[4] Jinpil Lee, Tomohiro Ueno, Mitsuhisa Sato, and Kentaro Sano. 2018. High-productivity Programming and Optimization Framework for Stream Processing on FPGA. In *Proceedings of the 9th International Symposium on Highly-Efficient Accelerators and Reconfigurable Technologies, HEART 2018, Toronto, ON, Canada, June 20-22, 2018*. 5:1–5:6. https://doi.org/10.1145/3241793.3241798

[5] Antoniette Mondigo, Kentaro Sano, and Hiroyuki Takizawa. 2018. Performance Estimation of Deeply Pipelined Fluid Simulation on Multiple FPGAs with High-speed Communication Subsystem. In *29th IEEE International Conference on Application-specific Systems, Architectures and Processors, ASAP 2018, Milano, Italy, July 10-12, 2018*. 1–4. https://doi.org/10.1109/ASAP.2018.8445100

[6] Fujita Norihisa, Kobayashi Ryohei, Yamaguchi Yoshiki, and Boku Taisuke. 2019. Parallel Processing on FPGA Combining Computation and Communication in OpenCL Programming. In *2018 IEEE International Parallel and Distributed Processing Symposium Workshops (IPDPSW)*.

[7] Miyajima Takaaki, Hirao Tomoya, Miyamoto Naoya, Son Jeongdo, and Sano Kentaro. 2019. A software bridged data transfer on a FPGA cluster by using pipelining and InfiniBand verbs. In *Proceedings of the 9th International Symposium on Highly-Efficient Accelerators and Reconfigurable Technologies (HEART 2019)*.

[8] Luzhou Wang, Kentaro Sano, and Satoru Yamamoto. 2012. Domain-Specific Language and Compiler for Stencil Computation on FPGA-Based Systolic Computational-Memory Array. In *Reconfigurable Computing: Architectures, Tools and Applications - 8th International Symposium, ARC 2012, Hong Kong, China, March 19-23, 2012. Proceedings*. 26–39.

[9] H. R. Zohouri, N. Maruyama, A. Smith, M. Matsuda, and S. Matsuoka. 2016. Evaluating and Optimizing OpenCL Kernels for High Performance Computing with FPGAs. In *SC '16: Proceedings of the International Conference for High Performance Computing, Networking, Storage and Analysis*. 409–420. https://doi.org/10.1109/SC.2016.34