FPGA Memory System for HPC Applications using Addressable Cache

Norihisa Fujita
Center for Computational Sciences,
University of Tsukuba
Tsukuba, Ibaraki, Japan
fujita@ccs.tsukuba.ac.jp

Ryohei Kobayashi
Center for Computational Sciences,
University of Tsukuba
Tsukuba, Ibaraki, Japan

Yoshiki Yamaguchi
Faculty of Engineering, Information and Systems,
University of Tsukuba
Tsukuba, Ibaraki, Japan

Taisuke Boku
Center for Computational Sciences,
University of Tsukuba
Tsukuba, Ibaraki, Japan

1 ABSTRACT

In our previous work [1], we implemented an astrophysics application for the early universe for a Field Programmable Gate Array (FPGA) cluster. It is written in OpenCL High-Level Synthesis (HLS) and uses FPGA's high-speed and low-latency direct optical links for inter-FPGA communication on different nodes. We conducted that FPGAs can run parallel applications efficiently thanks to their high-performance direct inter-FPGA communication. However, the memory bandwidth of an FPGA is the bottleneck to implementing an HPC application on it. The FPGA board used in [1] has only 4 channels of DDR4 memory (76.8GB/s), whereas other accelerators used in the HPC area have more than 1TB/s of memory bandwidth.

High Bandwidth Memory (HBM) 2 is a high bandwidth memory and is used in several accelerators such as NVIDIA A100 GPU. FPGA is also one of these accelerators. Intel Stratix 10 MX FPGA has two HBM2 stacks providing up to 512GB/s of memory bandwidth and 16GB of memory capacity. The architecture of HBM2 memory is different from the conventional DDR4 memory. HBM2 aggregates many slow memory channels to achieve high performance. Each memory channel in an FPGA has only 16GB/s of bandwidth, and an FPGA has 16 pseudo-channels per HBM2 stack and 32 pseudo channels in total. Although an FPGA does not have a sophisticated memory network, it must handle all memory channels simultaneously to obtain maximum performance from HBM2 memory. This is a big challenge for FPGAs equipping HBM2 memory.

Because of HBM2 aggregated architecture, we have to introduce a new memory architecture on an FPGA to utilize HBM2 memory. We propose a new memory system for FPGA and HPC applications using addressable caches. We believe that automatic cache system like CPUs is not suitable for HPC FPGAs. Resources in an FPGA are shared by the memory system and the application kernel. If we use an automatic cache system that consumes a lot of resources, resources for the computation are reduced. To solve this problem, we introduce addressable caches in our system. They have data copy controllers to transfer data between caches and memories. We describe how to copy data explicitly because these caches are not automatic. In addition to resource consumption, we can estimate the performance of the system easier than that of an automatic system. We believe this characteristic improves the memory performance for HPC applications. Our system also has crossbars to maximize the performance and flexibility of data transfer from and to HBM2 memory. In this poster, we show the work-in-progress implementation and the performance evaluation of the proposed memory system.

2 RELATED WORK

The use of HBM2 on an FPGA is widely studied. [4] implements an HPC Challenge benchmark suite on an FPGA using OpenCL and applies HBM2. [5] and [2] report how to implement neural networks on an FPGA with HBM2. In the abovementioned research, applications are tightly connected to the specific HBM2 memory channel, and there is no flexibility between an application and the HBM2 channels.

In [3], they implement bucket sort and merge sort on a Xilinx Alveo U280 FPGA board. They introduced their original memory network called "HBM Connect" in the paper. HBM Connect is an all-to-all memory network between an application and memory. It has buffers for burst memory access but does not have any cache. The originality of our research is that we introduce crossbars and addressable caches between an application and the HBM2. Crossbars improve the flexibility of memory access in an application. Caches improve memory efficiency and reduce the memory access complexity in an application.

REFERENCES

[1] Norihisa Fujita, Ryohei Kobayashi, Yoshiki Yamaguchi, Taisuke Boku, Kohji Yoshikawa, Makito Abe, and Masayuki Umemura. 2020. OpenCL-enabled Parallel Raytracing for Astrophysical Application on Multiple FPGAs with Optical Links. In *2020 IEEE/ACM International Workshop on Heterogeneous High-performance Reconfigurable Computing (H2RC)*. 48–55. https://doi.org/10.1109/H2RC51942.2020.00011

[2] R. Kuramochi and H. Nakahara. 2020. An FPGA-Based Low-Latency Accelerator for Randomly Wired Neural Networks. In *2020 30th International Conference on Field-Programmable Logic and Applications (FPL)*. 298–303. https://doi.org/10.1109/FPL50879.2020.00056

[3] Young kyu Choi, Yuze Chi, Weikang Qiao, Nikola Samardzic, and Jason Cong. 2021. HBM Connect: High-Performance HLS Interconnect for FPGA HBM. In *FPGA '21*.

[4] M. Meyer, T. Kenter, and C. Plessl. 2020. Evaluating FPGA Accelerator Performance with a Parameterized OpenCL Adaptation of Selected Benchmarks of the HPCChallenge Benchmark Suite. In *2020 IEEE/ACM International Workshop on Heterogeneous High-performance Reconfigurable Computing (H2RC)*. 10–18. https://doi.org/10.1109/H2RC51942.2020.00007

[5] S. K. Venkataramanaiah, H. S. Suh, S. Yin, E. Nurvitadhi, A. Dasu, Y. Cao, and J. S. Seo. 2020. FPGA-based Low-Batch Training Accelerator for Modern CNNs Featuring High Bandwidth Memory. In *2020 IEEE/ACM International Conference On Computer Aided Design (ICCAD)*. 1–8.