Project Office for Exascale Computing System Research and Development

Head: BOKU Taisuke, Professor, Ph.D., Director of CCS

Developing advanced fundamental technologies for ExaFLOPS and beyond

A parallel supercomputer system’ s peak computing performance is expressed as follows: node (processor) performance × number of nodes. Thus far, performance improvement has been pursued by increasing the number of nodes, but owing to problems such as power consumption and a high failure rate, simply increasing the number of nodes as a means of performance improvement is approaching its limit. The world’ s highest performing supercomputers are reaching the exascale, but to reach this level and beyond, it is necessary to establish fault-tolerant technology for hundreds of thousands to millions of nodes and to enhance the computing performance of individual nodes to the 100TFLOPS level. To achieve the latter, a computation acceleration mechanism is promising, which would enable strong scaling of the simulation time per step.

In the Next Generation Computing Systems Laboratory, we are conducting research on fundamental technologies for the next generation of HPC systems based on the concept of multi-hybrid accelerated supercomputing (MHAS). FPGAs play a central role in this research and are applied to the integration of computation and communication against the backdrop of dramatic improvements in the computation and communication performance in these devices. Since 2017, we have been developing and expanding PPX (Pre-PACS-X), a mini-cluster for demonstration experiments based on the idea of using FPGAs as complementary tools to achieve the ideal acceleration of computation for applications where the conventional combination of CPU and GPU is insufficient. Cygnus—the 10th generation supercomputer of the PACS series—was developed as a practical application system and began operating in the 2019 academic year. The following is an introduction to the main technologies being developed here.

Developing applications that jointly use FPGAs and the GPU

In collaboration with the Division of High-Performance Computing Systems and Division of Astrophysics, we developed the ARGOT code, which includes the radiation transport problem in the simulation of early-universe object formation, so that it would run fast on a tightly coupled GPU and FGPA platform. The entire ARGOT code can now be run on a 1-node GPU+FPGA co-computation, which is up to 17 times faster than a GPU-only computation. Additionally, we developed a DMA engine that enables high-speed communication between the GPU and FPGAs without using the CPU. Furthermore, through joint research with the Division of Global Environmental Science and the Division of Quantum Condensed Matter Physics, we are working on GPU/FPGA acceleration of application codes being independently developed at JCAHPC.

OpenCL interface for FPGA-to-FPGA high-speed optical networks

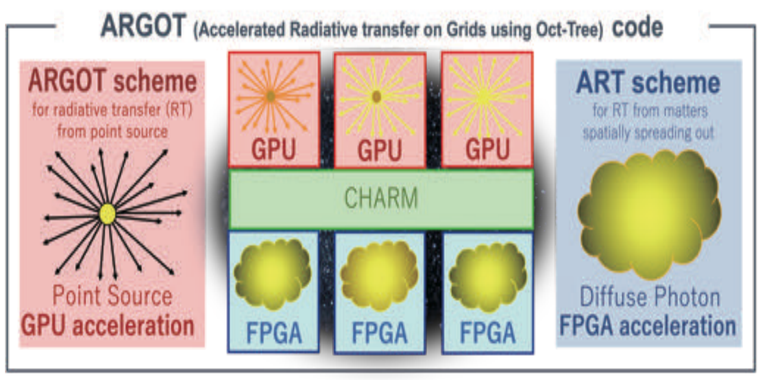
We developed CIRCUS (Communication Integrated Reconfigurable CompUting System) as a communication framework that facilitates the use of FPGA-to-FPGA networks with physical performance levels of 100 Gbps. It is easy to use at the user level and achieved high-performance communication (90 Gbps—90% of the theoretical peak performance of 100 Gbps) on the Cygnus supercomputer. Additionally, we applied this framework to the FPGA offload portion of the ARGOT code and confirmed that parallel FPGA processing can be performed smoothly.

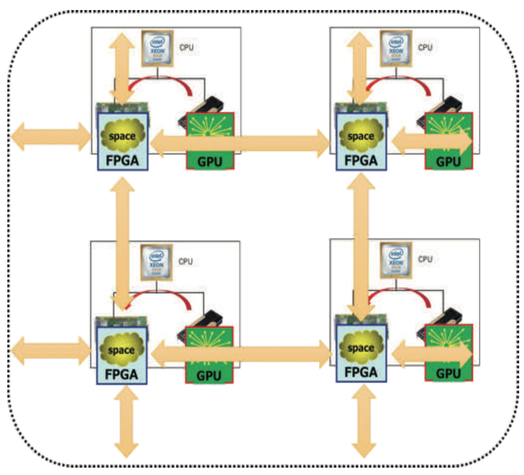
Unified GPU–FPGA programming environment

In collaboration with the Division of High-Performance Computing Systems and Division of Astrophysics, we developed the ARGOT code, which includes the radiation transport problem in the simulation of early-universe object formation, so that it would run fast on a tightly coupled GPU and FGPA platform. The entire ARGOT code can now be run on a 1-node GPU+FPGA co-computation, which is up to 17 times faster than a GPU-only computation. Additionally, we developed a DMA engine that enables high-speed communication between the GPU and FPGAs

Parallelized ARGOT code on GPUs & FPGAs

The figure below shows an overview of the multi-node parallelization of the GPU-FPGA ARGOT code. While each node of Cygnus is equipped with multiple GPUs and FPGAs, current version of code applies single GPU and FPGA for each MPI process to be assigned to a node. The ART method runs on the FPGA while the rest of the code including ARGOT method runs on the GPU. Two communication channels among multiple nodes are used where GPUs and CPUs communicate via MPI and inter-FPGA communication is taken by CIRCUS, to create a large computation and communication pipeline over multiple FPGAs. The original ARGOT code for GPU is written by CUDA + MPI, and we enhanced it with OpenCL programing for FPGA with CIRCUS feature.





Left: GPU+FPGA co-computation performance in early-universe object-formation simulation
Right: Overview parallelized ARGOT code on GPUs and FPGAs cooperability