Direct data communication in heterogeneous systems

1 Introduction

Heterogeneous system is defined as a unified platform in which different kinds of processors are integrated to leverage their unique capabilities. As accelerators or co-processors for CPUs, FPGAs and GPUs are commonly used in heterogeneous architectures due to their outstanding performance in high-speed computing and task parallelism. Typical applications, such as image processing, climatic modeling, computer vision and so on, often require rapid and real-time data communication among CPUs, GPUs and FPGAs. For these applications, high bandwidth and low latency in data transfer are vital for their performance.

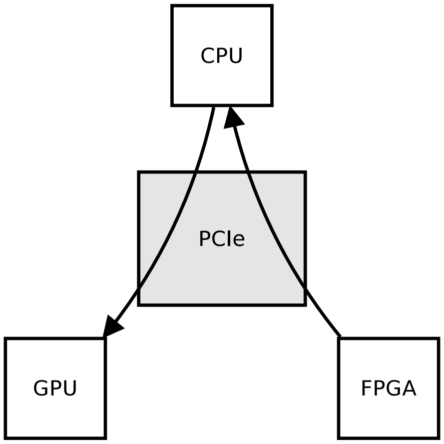

Figure 1: Indirect transfer

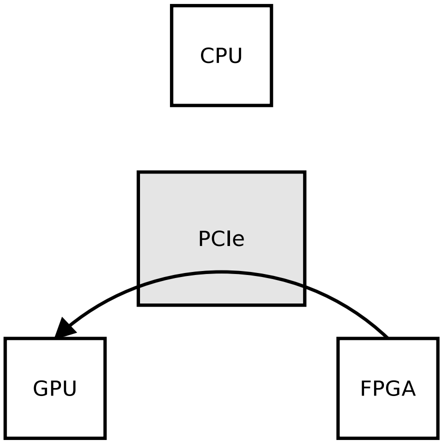

Figure 2: Direct transfer

Current methods for data communication between CPU and the accelerator devices are provided by corresponding vendors. To our best knowledge, at present no standard or vendor-provided methods exist for direct data communication between two accelerator devices. Communication between accelerators from different vendors, therefore, has to take slow and cumbersome steps including a round trip via CPU (shown in Figure 1) which obviously brings much higher latency and unnecessary communication overhead. As the consequence, direct data transfer between FPGA and GPU (shown in Figure 2), is required to avoid the bottleneck in the overall system.

2 Motivation and Goals

We want to implement direct data communication between different accelerators using the popular Open Computing Language (OpenCL) programming framework. OpenCL specifies a high-level abstraction for low-level hardware instructions, and thus it enables to scale computations among different bands of CPUs, GPUs and FPGAs without changing the source code.

Previous work has been done to enable the data transfer using a Nvidia GPU and an Altera FPGA, with one direction passed. Based on this framework, we want to drive data communication using GPUs from other bands, like AMD. Moreover, with OpenCL as a general programming framework, this direct data communication can be further used in heterogeneous systems to speedup more common applications.

3 Your tasks

- Enable the current data transfer method with the other direction passed.
- Extend the methodology using different types of accelerators, like GPUs (Nvidia and AMD) and FPGAs.

4 Requires

- Good knowledge on C++ programming skills
- Basic knowledge about OpenCL

5 Contact

Xiebing Wang
Institut für Informatik VI, TUM
Room: MI 03.07.059
E-mail: wangxie@in.tum.de
Phone: +49.89.289.18128