View metadata, citation and similar papers at core.ac.uk

brought to you by CORE

provided by Ghent University Academic Bibliography

ParaFPGA 2013: Harnessing Programs, Power and Performance in Parallel FPGA applications

Erik H. D'Hollander[*], Dirk Stroobandt[*] and Abdellah Touhafi[†]
[*]*ELIS Department, Ghent University, Belgium*
{erik.dhollander,dirk.stroobandt}@elis.ugent.be
[†]*ETRO Department, Vrije Universiteit Brussel, Belgium*
abdellah.touhafi@etro.vub.ac.be

Abstract—Future computing systems will require dedicated accelerators to achieve high-performance. The mini-symposium ParaFPGA explores parallel computing with FPGAs as an interesting avenue to reduce the gap between the architecture and the application. Topics discussed are the power of functional and dataflow languages, the performance of high-level synthesis tools, the automatic creation of hardware multi-cores using C-slow retiming, dynamic power management to control the energy consumption, real-time reconfiguration of streaming image processing filters and memory optimized event image segmentation.

Keywords: high-level synthesis, C-slow retiming, dataflow languages, routing, image processing, roofline model, polyhedral computations

Introduction

The mini-symposium ParaFPGA started in 2007 and is held every two years in conjunction with the parallel computing conference. The main topic of the symposium is the use of parallel techniques in FPGAs for high performance computing.

Nowadays the topic of Exascale computing is discussed from many different perspectives. Exascale computers will require low power, highly parallel computing devices and these requirements make FPGAs very suitable candidates. However, this wisdom is not very well known. A search in Google scholar at the date of the symposium delivers only 502 publications on the intersection of the keywords Exascale and FPGA.

The different research groups of the FPGA community have made already a lot of progress towards high performance computing, each in their own field of research. However there are many FPGA related research fields such as high-level synthesis, partial reconfiguration, routing, image processing applications, neural networks and so on. Therefore, an orchestrated composition of the advances in each of the FPGA related research fields is needed. This is the main purpose of ParaFPGA as it will help to reflect the potential of FPGAs in high-performance computing.

The contributions of this mini-symposium are presented in the following section.

1. Harnessing programs, power and performance

The rush to more computational power for solving grand problems hasn't stopped with the clock frequency ceiling and power walls in modern multicore processors. However, the general consensus is that a new and multidisciplinary approach will be needed to maintain a computational growth in-line with what we experienced in the last 20 years. In an effort to narrow the gap between the architecture and the application, specialized processors such as GPUs have shown to be very effective. Field programmable gate arrays have the additional advantage to embed an algorithm in hardware on the fly, tailored to a specific application. FPGAs have been very successful in specific applications and the speed and the development tools are constantly improving. In ParaFPGA 2013 a number of key issues related to high-performance computing with FPGAs have been highlighted.

The exploitation of locality in an algorithm is of crucial importance for achieving performance and energy efficiency [1]. Whereas the instructions are embedded in the FPGA fabric, routing the computational elements is much improved by detailed scheduling and placement information. This technique is used in the PARO high-level synthesis framework. In [2], Frank Hannig presents a domain specific language based on the polyhedral model to generate automatically massively parallel

FPGA accelerators. The functional language creates very regular computational structures and the data locality is improved using advanced loop transformations.

Space and time locality information is also available in the data dependency algebra (DDA). In [3] Eva Burrows specifies the supply and receive ports of each computational element in the algorithmic description of an FPGA design. The duality of requests and supplies is presented in a space-time diagram specifying the parallel operations as well as their interconnections. This information is used to improve the routing and placement of the computational units and to reduce the floor plan of the FPGA.

Optimizing the utilization and finding the right balance between resource consumption and performance can be done at different levels of the synthesis hierarchy. Tobias Strauch uses C-Slow retiming at the RTL level to automatically pipeline a core into multiple independent cores with a modest area increase [4]. As an example, a 4 core RISC processor is implemented on an area of less than 2 cores.

High-level synthesis (HLS) tools enable complex pipelining and loop transformations to maximize the performance of a single core at the expense of a large resource footprint. When multiple identical cores can operate in parallel, it may be beneficial to replicate a suboptimal but less resource-hungry design a large number of times to obtain a greater global performance. This approach is taken by Bruno da Silva et al. in [5]. The performance of the cooperating IP-cores is estimated by extending the roofline model [6] to take into account the combined effect of HLS optimizations, scalability, I/O and parallel computations.

At the other end of high-level synthesis, the flexibility of FPGAs gives great power to the developer of low-level hardware. In [7], Kobiri et al. have developed a real-time image segmentation design which entails irregular, non cached memory accesses. Using deep pipelining, performances of 20-30 frames per second are obtained, which is in excess of GPUs and involves a lower power penalty. Along the same vein, fast reconfigurability is put at work in an image filter bank developed by Kurita et al. [8]. The system allows to put filters in a pipeline at run time without disturbing the video stream.

Future high-performance systems will require an exorbitant energy budget. The number of energy and power related HPC research papers has risen sharply in the last five years [9]. Accelerators such as FPGAs have a duty cycle which is only a fraction of the overall computational time. In [10], Khurram Shahzad et al. addresses voltage control to minimize the power consumption when the FPGA is inactive.

2. Acknowledgement

The organizers of the mini-symposium are indebted towards the ParaFPGA program committee for reviewing the contributions and providing useful suggestions for the authors in preparation of their final papers.

References

[1] André M. DeHon, “Location, location, location: the role of spatial locality in asymptotic energy minimization,” in *Proceedings of the ACM/SIGDA international symposium on Field programmable gate arrays*, 2013, pp. 137–146.

[2] Frank Hannig, “High Level Synthesis Revised: Generation of FPGA Accelerators from a Domain-Specific Language using the Polyhedron Model,” in *ParaFPGA2013 mini-symposium, proceedings of the ParCo2013 conference, Advances in Parallel Computing*, vol. 25, Munich: IOS Press, 2013.

[3] Eva Burrows, “Compiling a Dataflow-based Language Abstraction onto an FPGA,” in *ParaFPGA2013 mini-symposium, proceedings of the ParCo2013 conference, Advances in Parallel Computing*, vol. 25, Munich: IOS Press, 2013.

[4] Tobias Strauch, “Timing Driven C-Slow Retiming on RTL for MultiCores on FPGAs,” in *ParaFPGA2013 mini-symposium, proceedings of the ParCo2013 conference, Advances in Parallel Computing*, vol. 25, Munich: IOS Press, 2013.

[5] Bruno da Silva, An Braeken, Erik H. D’Hollander, and Abdellah Touhafi, “Performance and Resource Modeling for FPGAs using High-Level Synthesis tools,” in *ParaFPGA2013 mini-symposium, proceedings of the ParCo2013 conference, Advances in Parallel Computing*, vol. 25, Munich: IOS Press, 2013.

[6] Samuel Williams, Andrew Waterman, and David Patterson, “Roofline: an insightful visual performance model for multicore architectures,” *Communications of the ACM*, vol. 52, no. 4, pp. 65–76, 2009.

[7] Daichi Kobori and Tsutomu Maruyama, “Interactive Graph Cuts using FPGA,” in *ParaFPGA2013 mini-symposium, proceedings of the ParCo2013 conference, Advances in Parallel Computing*, vol. 25, Munich: IOS Press, 2013.

[8] Hisaaki Kurita and Tsutomu Maruyama, “An Image Filter System based on dynamic partial reconfiguration on FPGA,” in *ParaFPGA2013 mini-symposium, proceedings of the ParCo2013 conference, Advances in Parallel Computing*, vol. 25, Munich: IOS Press, 2013.

[9] Dimitrios S. Nikolopoulos, “Programming the Energy--Efficiency of High-Performance Computing Systems,” Keynote talk at the International Conference on Energy-Aware High Performance Computing, Dresden, 02-Sep-2013.

[10] Khurram Shahzad and Bengt Oelmann, “Investigation of Energy Consumption of an SRAMbased FPGA for Duty-Cycle Applications,” in *ParaFPGA2013 mini-symposium, proceedings of the ParCo2013 conference, Advances in Parallel Computing*, vol. 25, Munich: IOS Press, 2013.