Comparing CPU and FPGA Application Performance

Volodymyr Kindratenko, David Pointer[‡], David Raila, and Craig Steffen
National Center for Supercomputing Applications (NCSA)
University of Illinois at Urbana-Champaign (UIUC)
1205 W. Clark St. Room 1008
Urbana, IL 61801
{kindr, pointer, raila, csteffen}@ncsa.uiuc.edu

Abstract: When comparing CPU based application performance to FPGA based application performance, we suggest that system dependant FPGA overhead time must be included in the comparison.

1 Introduction

In the Innovative Systems Lab (ISL) at NCSA, emerging technology researchers and domain scientists team up to evaluate new architectures with high value scientific applications and real world data sets. One emerging system architecture undergoing close scrutiny in ISL is high performance reconfigurable computing (HPRC) as embodied by Cray, Nallatech, SGI, and SRC. HPRC technology is based on the combination of conventional CPUs and Field Programmable Gate Array (FPGA) devices. This technology merger enables software developers to exploit coarse-grain functional parallelism through conventional parallel processing as well as fine-grain parallelism through direct hardware execution on FPGAs.

When we started evaluating HPRC systems, we struggled to define exactly how to quantify an application's performance on a traditional CPU system to the application's performance on an HPRC system. The guiding principle that led us to a solution was simply: which performance comparison method had the highest value to the applications science? If we say that an application on an FPGA has a 3x speedup relative to a CPU, we wanted the application scientist to know, for instance, that their 3 hour CPU based application would run in 1 hour on an FPGA. This white paper describes our approach to the problem of comparing an application's performance on these two architectures.

2 Application Performance Measurement

In this section, we discuss the specific time components of CPU based applications and FPGA based applications relative to each other and our method of comparing application performance.

2.1 CPU and FPGA Application Time

In any application, there is some time spent to organize the data before performing computations on the data. Similarly, there is some time spent organizing or re-arranging data after the CPU has finished computation. The sum of these three times we refer to as "overall application time" (Figure 1).

[‡] Corresponding author

The time spent by the CPU actually performing calculations on the data we refer to as "computation time". We assume that it is the application computations which will be ported to an FPGA.

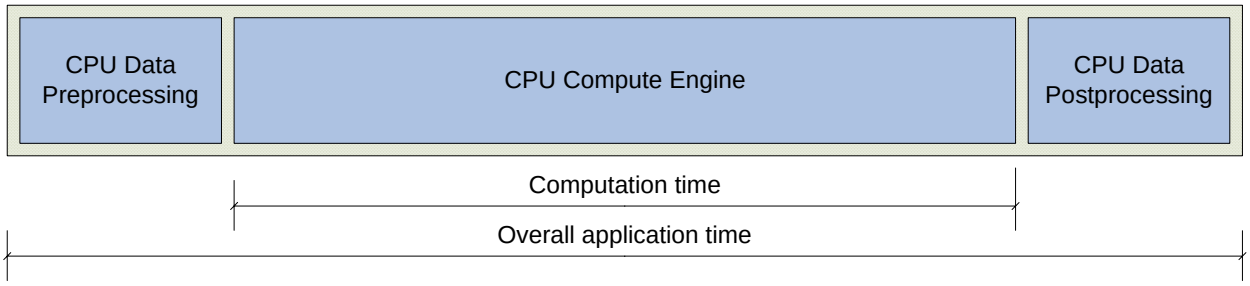

Figure 1 - CPU Based Application Time

In an FPGA, the computation time is only a part of the time required to perform useful work on an FPGA. There is also time required to initiate communication with the FPGA and load the FPGA bit stream ("load FPGA design" in Figure 2), the time to move the application data set from the CPU memory to the local FPGA memory ("transfer data to FPGA memory"), and the time to move the result of the FPGA based computations from local FPGA memory to CPU memory ("transfer data from FPGA memory"). These three items we lump together into the term "FPGA overhead".

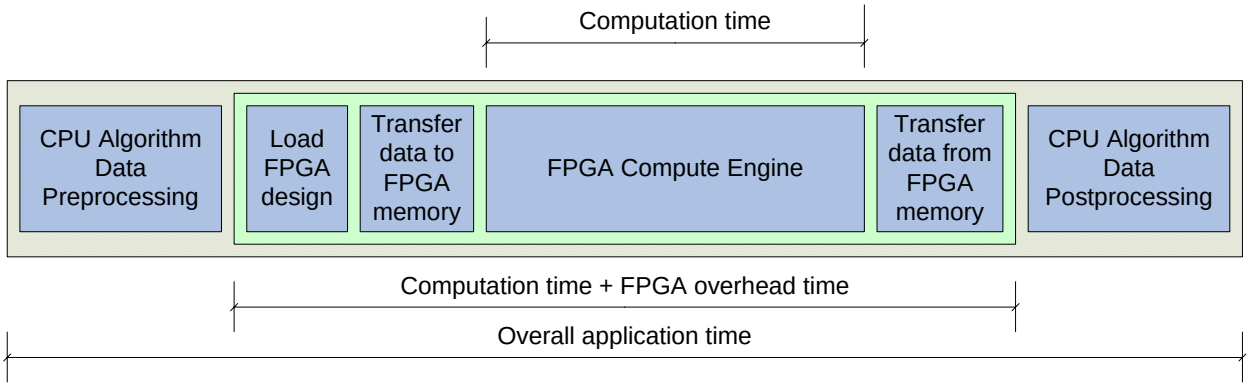

Figure 2 - FPGA Based Application Time

We assume that any CPU based data pre- and post-processing time is constant between the two views of application time.

2.2 Relative Application Performance

Simply comparing "computation time" between an application on a CPU system and the same application on an FPGA based system does not meet our guiding principle of providing a meaningful performance comparison number to the application scientists. The FPGA overhead

time must be taken into account. For instance, during our work on the NAMD[1] application, we found that the FPGA computation time was 4.5x over the CPU computation time[2]. We also found that the FPGA computation time *plus* the FPGA overhead time was 3x over the CPU computation time. The 4.5x figure is meaningless to the application scientist as they will not see an actual 4.5x performance improvement on their application running on an FPGA system. The application scientist can expect an actual 3x performance improvement.

3 Summary

When we state that a given application runs N times faster (or slower) on an FPGA system, we are stating that the combination of the FPGA overhead *and* the FPGA computation time is N time faster (or slower) than the CPU computation time.

4 Acknowledgement

This material is based upon work supported by the U.S. National Science Foundation under Award No. SCI 05-25308.

Any opinions, findings, and conclusions or recommendations expressed in this publication are those of the author and do not necessarily reflect the views of the U.S. National Science Foundation.

[1] J. Phillips, R. Braun, W. Wang, J. Gumbart, E. Tajkhorshid, E. Villa, C. Chipot, R. Skeel, L. Kale, and K. Schulten, *Scalable Molecular Dynamics with NAMD*, Wiley Interscience (www.interscience.wiley.com), 26 May 2005.

[2] V. Kindratenko, and D. Pointer, A case study in porting a production scientific supercomputing application to a reconfigurable computer, submitted to the IEEE Symposium on Field-Programmable Custom Computing Machines (FCCM'06)