Harmonica: An FPGA-based Data Parallel Soft Core

Chad Kersey, Sudhakar Yalamanchili
School of Electrical and Computer Engineering
Georgia Institute of Technology
Atlanta, Georgia
{cdkersey, nimitnigania, sudha}@gatech.edu

Hyojong Kim, Nimit Nigania, Hyesoon Kim
College of Computing
Georgia Institute of Technology
Atlanta, Georgia
{hyojong, nimitnigania}@gatech.edu, hyesoon@cc.gatech.edu

***Index Terms*—FPGA; GPGPU; SIMT; data parallel; processor near memory; C++**

I. Introduction

General-purpose GPUs or *GPGPUs* have taken their place in the market, being present in 38 of the Top 500 supercomputers [5]. In the same way that the emergence of FPGAs in the 1980s led to a demand for soft cores with instruction sets similar to the CPUs of the day, we anticipate a similar demand in the 2010s for soft cores with GPGPU instruction sets. These architectures are distinguished by their *SIMT*, single-instruction-multiple-thread, execution model, acheiving throughput by running multiple threads of execution simultaneously across multiple functional units, keeping separate register values for each *lane* of execution.

II. Contributions

A. Harmonica

Current soft cores come in many sizes, from microcontrollers to superscalar, often providing some level of configurability. It is likely that the range of sizes demanded from current soft cores will also be seen in the demand for SIMT soft cores. To answer the need in the research community for soft GPGPU cores of many sizes and feature sets, we have developed Harmonica, a configurable SIMT soft-core with a parameterized instruction set architecture, to run GPGPU applications. By developing a configurable SIMT soft core, we hope to extend the scope of rapid prototyping, architecture exploration, and the development of custom accelerators, especially in design spaces incorporating novel memory system technologies like Micron's Hybrid Memory Cube [3].

B. HARP

Along with Harmonica we introduce the HARP (Heterogeneous Architecture Research Prototype) family of instruction set architectures. This is a parameterized family of architectures enabling unprecedented flexibility in the implementation of SIMT cores. The HARP ISAs allow the use of a short string to describe the SIMD width, number of general-purpose and predicate registers, and instruction encoding used. Using this string, the toolchain targets the assembly and linking of input programs to the appropriate core type.

This core is configurable in the sense that dimensions such as SIMD width and word size can be configured as well as the set of and types of functional units present. This not only saves space by allowing the removal of unnecessary functional units, but also allows the development of functional units using vendor-provided IP for performance-critical operations like fixed or floating point multiply.

C. CHDL

Harmonica is implemented using a C++ hardware design library partially influenced by JHDL [2] and Chisel [1] which we call CHDL. This effort is unrelated to the coincidentally homonymous earlier work [4], an FPGA development platform also built in C++. CHDL provides a way to describe the structure of hardware using a common programming language. This structure can then be used to produce netlists in several formats, including Verilog, which can then be parsed by traditional FPGA tool flows. The use of CHDL enables configurability at compile time with C++ template metaprogramming, ultimately improving performance during elaboration and simulation.

D. FPGA Prototype

To validate and verify Harmonica, an FPGA prototype integrating Harmonica cores, a cache module written in Verilog, and a DDR2 controller has been constructed. This early prototype is intended to be one of the first steps in the Cymric microarchitecture project, which has the mid-term goal of running the same configuration of Harmonica cores with stacked DRAM. This prototype has been demonstrated running at a 62.5MHz core frequency on an Altera Stratix III.

References

[1] J. Bachrach *et al.*, "Chisel: constructing hardware in a scala embedded language," in *Proceedings of the 49th Annual Design Automation Conference*. ACM, 2012, pp. 1216–1225.

[2] P. Bellows and B. Hutchings, "Jhdl-an hdl for reconfigurable systems," in *FPGAs for Custom Computing Machines, 1998. Proceedings. IEEE Symposium on*. IEEE, 1998, pp. 175–184.

[3] J. Jeddeloh and B. Keeth, "Hybrid memory cube new dram architecture increases density and performance," in *VLSI Technology (VLSIT), 2012 Symposium on*. IEEE, 2012, pp. 87–88.

[4] K. Kornmesser, A. Kugel, and R. Manner, "The fpga development system chdl," in *Field-Programmable Custom Computing Machines, 2001. FCCM'01. The 9th Annual IEEE Symposium on*. IEEE, 2001, pp. 271–272.

[5] E. Strohmaier, "Top500 supercomputer," in *Proceedings of the 2006 ACM/IEEE Conference on Supercomputing*, ser. SC '06. New York, NY, USA: ACM, 2006. Available: http://doi.acm.org/10.1145/1188455.1188474