

Reconfigurable Computing and FCCM: What have we done in 20 years, and what will Reconfigurable Computing mean in 2032?

FCCM 2012 Sunday Workshop

Co-Chairs: David Andrews, University of Arkansas
Paul Chow, University of Toronto

This workshop is a call to arms to the FCCM community to help establish a new research agenda for the next two decades. The workshop will start with two talks. The first will provide a post mortem over how the discipline has evolved, and consider the appropriateness and effectiveness of the advancements made in the community over the last twenty years. The second talk proposes a forward looking vision with the goal of stimulating discussion to uncover the research challenges that should be addressed if the discipline is to move forward over the next twenty years. After the talks, attendees will break up into working groups where they will help to develop a set of community-based research challenges for the future. The outcomes of the working groups will then be shared and discussed in the final session of the workshop.

A Vision for the Next Twenty Years

Platform FPGA densities now provide over 1 Million LUTs, a sufficient density to turn a single-chip FPGA into a complete multiprocessor system on chip (MPSoC). As FPGAs continue to follow Moore's Law, density levels will allow hundreds to thousands of heterogeneous programmable processors as well as custom accelerators to be configured within a single chip.

While the performance potential of these next-generation chips will be significant, will this level of gate density also be bringing with it a paradigm shift in required system designer skills? The complexity levels associated with earlier generations of FPGAs were small enough to be handled by designers with knowledge of hardware description languages and low-level digital design skills. However, the density levels of next-generation FPGAs will be simply too great to be handled by low level-design skills and even C-to-gates languages. Will the next generation designers be required to possess design skills more aligned with parallel computer architectures instead of lower-level digital design? If so, then will designers be required to be knowledgeable in the design of complex multi-tiered memory hierarchies composed of global, shared, and private memories, as well as cache organizations, hierarchies and protocols instead of circuit delays and fan outs?

The near term ability to integrate hundreds to thousands of processors is exciting from a performance perspective. Current manual assembly approaches within vendor specific CAD tools can be used to design systems with tens of processors. However they certainly will not be appropriate for dealing with the complexities of designing and integrating parallel processing architectures with hundreds to thousands of processors, complex interconnect networks and multi-tiered partitioned memory. Will new capabilities in vendor-neutral architecture automation evolve? Can these new methods result in portable architectures that finally allow fair comparisons between vendor specific components?

Assembling soft IP processors, accelerators, buses, memories, and support components is a time consuming process, but in reality only represents a small percentage of any overall effort to create a usable MPSoPC system. High-level parallel programming models and software protocol stacks are important infrastructure that brings performance, portability, and productivity. Can the FPGA community adopt more standard high-level parallel programming models without sacrificing significant performance? Adopting standard programming models and protocol stacks will bring new requirements for middleware and run-time systems support. The shift from scalar to parallel processors within our modern many-core era is already bringing new challenges associated with scalability and processor heterogeneity. In addition to scalability, heterogeneity brings new challenges for compilation and run-time systems to resolve differences in processor ISAs, synchronization primitives, Application Binary Interfaces (ABIs), and cache coherency protocols. Will standardization of run-time systems and new compilation techniques become a reality, freeing design teams from each having to “roll their own”? Will standard software-centric debug capabilities evolve to support each MPSoPC platform?

The purpose of this workshop is to engage the community to explore this vision and how it might affect research directions and change the use cases of next-generation FPGAs.

Will the fact that you are using an FPGA ultimately dissolve from your consciousness? Will this use case kill or stimulate more interest in classic accelerator and co-processor based reconfigurable computing?

Format

1. Introduction and Welcome talk (10 minutes), Paul Chow, University of Toronto
2. History and the Past 20 years (30 minute talk/discussion), Mike Butts, Compute Forest, USA
3. Controversial vision and research challenge for the next twenty years (30 minute talk/discussion), David Andrews, University of Arkansas
4. Breakout sessions with suggested questions/topics to discuss (45 minutes)
5. 15 minute break
6. Presentation of findings from breakout session. (60 minute discussion)