Thrifty: An Exascale Architecture for Energy-Proportional Computing

Josep Torrellas (University of Illinois), Daniel Quinlan (Lawrence Livermore National Laboratory),
Laura Carrington (University of California, San Diego), Wilfred Pinfold (Intel)
http://extremescale.cs.illinois.edu/thrifty
March 2013

This work proposes a novel exascale architecture called *Thrifty*. Thrifty aims to provide a platform for highly-efficient, energy-proportional computing — one where energy is spent proportionally to the work done, and in the resources that are the bottleneck. Our objectives are:

Power/Energy Efficiency: Attain a major efficiency gain over current systems by developing: (1) Novel circuits and architecture technologies for low supply voltage, and fine-grain management of static and dynamic power; (2) A power/energy-aware source-to-source compiler that uses static and dynamic code analysis to drive the power-management hardware and auto-tunes code for power/energy efficiency; (3) Application models for efficient energy-proportional computing and a means to insert power-management pragmas in the application.

Resiliency: Reduce the waste due to faults and recovery by developing: (1) New circuits and architecture technologies for high resiliency at low supply voltage and efficient error detection and tolerance; (2) A novel architectural scheme for energy-efficient, incremental, in-memory, scalable checkpointing; (3) Compilation and applications support to drive the checkpointing scheme.

Performance: Attain a major performance increase over current machines by: (1) Architecting many performance-enhancing features in Thrifty, including primitives for fine-grain synchronization and communication; (2) Developing a compiler that efficiently drives the performance features of Thrifty and auto-tunes programs for performance; (3) Identifying application *Idioms* and mapping them efficiently on Thrifty.

This handout describes the progress made since August 2012.

Recent Results on Power/Energy Efficiency

• **Energy Management Framework that Combines Architecture, Compilation and User Annotations**.

We have developed an Energy Management Framework composed of an API that enables the compiler or the programmer to change hardware configuration parameters in the chip. With the API, we can: (1) Select the Voltage and Frequency Bin for each of the processors and for each of the L2 caches in the chip, and (2) Turn-off (power-gate) and turn-on various resources in the chip.

• **Energy-Efficient Thrifty Chip Organization for Near-Threshold Computing (NTC):** "EnergySmart: Toward Energy-Efficient Manycores for Near-Threshold Computing", by Ulya R. Karpuzcu, Abhishek Sinkar, Nam Sung Kim, and Josep Torrellas. In International Symposium on High Performance Computer Architecture (HPCA), February 2013.

We examine the types of power and energy efficiency issues that appear when we consider multiple voltage domains in such a chip. We recommend a simple chip design.

• **Intelligent Refresh of On-Chip Memory Hierarchies:** "Refrint: Intelligent Refresh to Minimize Power in On-Chip Multiprocessor Cache Hierarchies", by Aditya Agrawal, Prabhat Jain, Amin Ansari, and Josep Torrellas. In International Symposium on High Performance Computer Architecture (HPCA), February 2013.

To minimize power in the on-chip cache hierarchy, we use embedded DRAM, and describe an intelligent, energy-efficient refresh algorithm.

• **Chip Architecture Designed from the Ground Up for Energy Efficiency:** "Runnemede: An Architecture for Ubiquitous High-Performance Computing", by Nicholas P. Carter, Aditya Agrawal, Shekhar Borkar, Josep Torrellas, and others. In International Symposium on High Performance Computer Architecture (HPCA), February 2013.

We designed an architecture that is built from the ground up for energy efficiency. All of the layers of the

computing stack are co-designed to consume the minimum possible energy.

Recent Results on Resiliency

• **Reliable Energy-Efficient On-Chip Networks:** "Designing Reliable, Ultra-Energy-Efficient On-Chip Networks Under Process Variation at Near-Threshold Voltage", by Amin Ansari, Asit Mishra, Jianping Xu, and Josep Torrellas. In University of Illinois Technical Report, 2012.

We dynamically change the voltage of each domain of the chip based on the fault rate that we observe in the network: we increase the voltage if the error rate is too high, or reduce it otherwise.

• **Compiler-Driven Code Transformations for Power Reduction:** "ROSE::FTTransform – A Source-to-Source Translation Framework for Exascale Fault-Tolerance Research", by J. Lidman, D. J. Quinlan, C. Liao, and S. A. McKee, Workshop on Fault-Tolerance for HPC at Extreme Scale (FTXS 2012), Boston, 2012.

ROSE::FTTransform is a source-to-source translator based on ROSE that adds to the source code both fault detecton and fault handling mechanisms. This is accomplished by performing redundant computations.

Recent Results on Performance

• **Software-Managed Cache (SMC) Framework that Combines Architecture, Compilation and User Annotations:** "Automatic Generation of Coherence Instructions for Software-Managed Multiprocessor Caches", by Sanket Tavarageri, Wooil Kim, Josep Torrellas, and P Sadayappan. Submitted for publication.

We have developed a Software-Managed Cache (SMC) Framework composed of an API that enables the compiler or the programmer to manage cache coherence through *write back* and *invalidate* instructions.

• **Potential Performance and Energy Benefits of Scratchpad Memories (SPM):** "Efficient HPC Data Motion via Scratchpad Memory", by K. Seager, A. Tiwari, M. Laurenzano, J. Peraza, P. Cicotti, and L. Carrington. International Workshop on Data-Intensive Scalable Computing Systems (DISCS-12), 2012.

Using an approximate model for the behavior of SPMs, we show that applications can benefit from hardware containing both scratchpad and traditional caches.

• **Fine Grain Synchronization with Full/Empty Bits:** "Exploring Alternatives to Hardware Support for Fine-Grain Synchronization", by Benjamin Ahrens, Roger Golliver, and Josep Torrellas. Submitted for publication, 2013.

Rather than supporting hardware F/E bits, we propose to emulated them with the well-known Compare&Swap instruction and special software algorithms.

• **Asymmetric Multicores:** "Illusionist: Transforming Lightweight Cores into Aggressive Cores on Demand", by Amin Ansari, Shuguang Feng, Shantanu Gupta, Josep Torrellas, and Scott Mahlke. International Symposium on High Performance Computer Architecture (HPCA), February 2013.

We propose an adaptive asymmetric multicore where, to accelerate the performance of many lightweight cores, a few aggressive cores run a distilled version of the threads and generate hints for the lightweight cores.

• **Core Assignment:** "The Case for Colocation of HPC Workloads", by A. Breslow, L. Porter, A. Tiwari, M. Laurenzano, L. Carrington, D. Tullsen, and A. Snavely. Concurrency and Computation: Practice and Experience, February 2013.

We proposed Job Striping, a technique that significantly increases performance over the current allocation mechanism by co-locating pairs of jobs from different users on a shared set of nodes.