Hadi Esmaeilzadeh | Research Statement

My experience as a graduate student spans several areas in computing including transistor-level circuit design, logic and VLSI system design, chip manufacturing and online test, processor architecture design, programming languages, neural networks, and application and workload studies. My primary area of interest is computer architecture. My other strong interests include programming languages, machine learning, and mixed-signal VLSI design. I am looking forward to collaborating in all of these areas. More specifically, I am interested in cross-stack and technology-driven innovations that improve the performance and energy efficiency of computer systems for emerging applications. These innovations include solutions that span the entire computing stack form underlying elements of computing (e.g., transistors) to applications.

My PhD research has focused on (1) identifying the impact of technology and application trends on computer architecture design and (2) proposing new possible directions to tackle the fundamental challenges emerging from these trends. I first studied the interplay between two revolutions in the past decade, the rise of multiprocessors and the rise of managed languages such as Java. Then, in a study of the dark silicon phenomenon, I investigated whether increasing the number of cores every technology generation will sustain historical performance improvements. This study highlights that radical departures from conventional approaches are necessary to sustain traditional speedup trends due to power constraints. I focused on solutions that leverage approximation in general-purpose computing and trade accuracy or quality of results for better performance and energy efficiency. I designed an architectural framework, from the ISA (Instruction Set Architecture) to the microarchitecture, for conventional processors to trade accuracy for efficiency. I then introduced a new class of accelerators that map a hot code region from a von Neumann model to a neural model and provide significant performance and efficiency gains. My interests for future research include developing analog and alternative technologies for general-purpose computing, disciplined approximate circuit design, energy-budgeted embedded computing, and architecture support for big data and machine-learning applications.

Dissertation Research

Measured power and modern workloads. Microprocessor design has been transformed over the past decade by the limits of chip power, leading to multicore processors. At the same time, managed languages (e.g. Java) and an entirely new software landscape emerged, dramatically changing how software is developed. Researchers most often examine these changes in isolation. I studied the interplay between these two revolutions by measuring power, performance, energy, and effects of technology scaling on real hardware [1]. These measurements and analyses quantify many hardware and software trends, pointing to new research directions. (1) Performance, power, or energy of the native benchmarks (mostly implemented in C/C++) does not predict the results of modern applications (mostly implemented in managed languages such as Java). Thus, the architects must extend their conventional evaluation methods to include workloads developed in managed languages. (2) Traditionally, power and energy have been the domain of architects. However, the diverse application power profiles suggest that there is opportunity for software developers to participate in power optimization. (3) To facilitate the involvement, both industry and the research community should systematically integrate power measurement in the software development stack. John Hennessey and David Patterson used our data in the most recent edition of their canonical computer architecture textbook [2]. Others in the research community have used our methodology to design and study software systems. Our paper was selected as an IEEE Micro Top Pick from the 2011 computer architecture conferences [3] and also for a Research Highlight in the Communications of ACM journal [4].

Dark silicon and multicore scaling. Starting in 2004, the microprocessor industry shifted to multicore scaling—increasing the number of cores per die each generation—as its principal strategy for continuing performance growth. However, while transistor count increases continue at traditional Moore's Law rates, the per-transistor speed and energy efficiency improvements have slowed dramatically. My study brought together transistor technology, processor core, and application models to understand whether multicore scaling can sustain the historical exponential performance growth in this energy-limited era [5]. Our study showed that as the number of cores increases, power constraints will prevent running all of the cores on a die unless they

are significantly throttled. Power density trends will require an increasing fraction of the chip to be powered off with each new technology generation, a phenomenon now widely called "dark silicon". The low utility of this dark silicon may prevent both scaling to higher core counts and ultimately the economic viability of continued silicon scaling. The results show that multicore scaling provides much less performance gain than conventional wisdom had suggested before this paper. This paper was selected as an IEEE Micro Top Pick from the 2011 computer architecture conferences [6], for a Research Highlight in the Communications of ACM journal [7], and was invited for publication in ACM Transaction on Computer Systems (TOCS) [8]. The New York Times also profiled our paper on the front page (Aug. 1, 2011) [9]. To surpass the dark silicon performance barrier highlighted by our work, significant departures from conventional techniques are needed that provide considerable energy efficiency in general-purpose computing. One such approach is abstractions and architectures that allow probabilistic and approximate computing, which has been the focus of my dissertation.

Variable-precision architectures. There is an emerging opportunity due to the synergy between applications that can tolerate approximation and the unreliability in the computation fabric as technology scales down. Relaxing the high tax of providing perfect accuracy at the device, circuit, and architecture level can provide a opportunity to improve performance and energy efficiency for the domains in which applications can tolerate approximation. This work provides an architectural framework for conventional general-purpose processors to trade accuracy for energy [10]. I introduced a variable-precision Instruction Set Architecture (ISA) that allows conventional von Neumann processors to interleave approximate and precise instructions. This ISA allows the compiler to convey what can be approximated without specifying how. This abstraction allows the microarchitecture to freely choose from a range of approximation techniques without exposing them to the software. I also proposed the dual-voltage Truffle microarchitecture that implements this variable-precision ISA. I also designed Truffle's novel dual-voltage on-chip memory structures as well as its microarchitectural extensions down to the transistor level.

Neural Processing Units. I designed a new acceleration technique that leverages a simple programmer annotation ("approximable") to transform a hot code region from a von Neumann model to a neural model. I proposed an algorithmic transformation that automatically selects and trains a neural network to mimic a region of imperative code [11]. One of the most important findings of our work is that neural networks can learn the behavior of regions of imperative code. After the learning phase, the compiler transparently replaces the original code with an invocation of a low-power accelerator. The transformation is beneficial because neural networks are parallel structures with efficient hardware implementations. Leveraging this transformation, I introduced a new class of approximate accelerators, called Neural Processing Units (NPUs), with implementation potential in both the digital and the analog domain. Even though hardware neural networks are far from being a new idea, prior research has not considered tightly integrating neural hardware with the processor due to the lack of abstractions allowing applications to benefit from the integration. My prior experience in design and utilization of hardware neural networks [12] helped me to identify this missing link and propose the algorithmic transformation that enables neural hardware to operate beyond their conventional use case and accelerate imperative code. While traditional techniques, such as dynamic voltage and frequency scaling, trade performance for efficiency, my work explored the approximation axis in the general-purpose computing design space and showed significant gains both in performance and energy when the abstraction of full precision is relaxed. My work formed the core of the NPU project, which is an important part of Professor Ceze's broader approximation project.

VLSI design and test. During my master studies, my primary research area was VLSI design and test. My contribution in this area includes online processor self-testing [13], circuit design for low-power testing [14], test-aware high-level synthesis [15], design and integration of hardware neural networks in system-on-a-chip for embedded systems [12], and language extensions for object-oriented hardware-software co-design [16]. My expertise in different aspects of VLSI circuits and systems design helped me to have a realistic view in my research in computer architecture and always consider the silicon implementation of my innovations.

Future Research Directions

CMOS technology scaling trends and the resulting dark silicon phenomenon pose a great challenge to the computing community. However, there is a silver lining for architects to innovate in general-purpose computing and deliver the performance and efficiency gains that can work across a wide range of problems.

Complementary to dark silicon and the looming end of Moore's law is the changing landscape of computing to a cloud-mobile model. In this landscape, both mobile and cloud services are aiming to provide a more targeted and personalized experience for users. I believe that, in this new landscape, computer architecture design must be driven by applications while considering technology trends. To this end, I will work on architecture support for integrating machine learning into everyday computing and for services that rely on big data. On the other hand, I will also develop alternative technologies for general-purpose computing through machine learning. Energy efficiency is and will be the primary concern in system design. I will work on circuit design techniques that systematically trade quality of results (accuracy) for energy and performance. I will also work on comprehensive solutions to enable tiny embedded devices to perform general-purpose processing and become a cohesive part of the new landscape of computing. My plan is to bring together innovations across the stack of computing from physical elements to applications.

Architecture support for big data and machine learning. I will focus on the design of server systems that energy-efficiently accelerate algorithms (e.g., machine learning algorithms) that process big data and enable new possibilities in many domains such as medicine. Since benchmarks shape the foundation of a research effort, the first step in this direction is identifying and providing benchmarks that represent these algorithms. With my experience in workload analysis, I will characterize these benchmarks to identify their constraints and requirements. After determining the bottlenecks and opportunities, I will focus on designing computing systems that provide the required capabilities. For example, many of the machine-learning algorithms that target big data applications rely on statistical optimization. Providing specialization for a subset of these algorithms is a promising starting point. Another direction that I plan to pursue is scaling out the machine-learning kernels to large and warehouse-scale computing fabrics. Providing a large amount of compute power for these kernels can enable applications that are not possible with limited compute power. The end of Moore's law will likely dictate specialization in the computing fabric. I also plan to focus on hardware acceleration and specialization for machine learning in mobile devices. As many system-on-a-chip designs and embedded systems include specialized hardware for signal processing and graphics, the time has come to include specialized units for machine learning. Specialization for machine learning can enable mobile and embedded systems to provide a more personalized human–machine interaction.

General-purpose computing with alternative technologies. There is a potential new space of algorithmic transformations that leverage learning to mimic regions of code. This space of learning-based transformation can enable the use of alternative technologies for general-purpose computing. I will work on learning techniques (e.g., support vector machines or deep belief networks) that can replace regions of code. I will also develop hardware accelerators for these machine-learning models. A promising starting point is developing an analog neural substrate to learn and replace regions of imperative code. The use of analog and alternative storage technologies such as memristors in this framework is another direction that I will pursue.

Disciplined approximate hardware design. Systematically relaxing the abstraction of accuracy at the circuit level can unleash more efficiency gains than current approximate computing techniques leverage. I plan to work on extensions to hardware description languages (HDLs) that allow the designer to relax accuracy requirements on non-critical parts of the design. The synthesis tools then can leverage unsafe optimizations on these parts and trade accuracy for efficiency and speed in a disciplined manner.

Energy-budgeted embedded computing. Enabling tiny embedded devices, with limited or no battery, to perform general-purpose computation on harvested energy is another promising research direction. I plan to work on domain-specific languages that allow explicit energy budgeting. I will focus on programing models that expose energy consumption, allowing developers to explore and choose alternative algorithms and implementations and to trade performance or accuracy (quality of results) for energy. I will work on designing domain-specific processors and microarchitectural mechanisms that adapt program execution when energy becomes scarce. Furthermore, I plan to research reconfigurable accelerators for these particularly energy-limited systems and to develop programming abstractions for the accelerators. I will also work on utilizing near-threshold voltage technology for this specific domain of embedded computers where performance is not the primary concern.

Bibliography

[1] H. Esmaeilzadeh, T. Cao, X. Yang, S. Blackburn, and K. McKinley, "Looking Back on the Language and Hardware Revolution: Measured Power, Performance, and Scaling", in *International Conference on Architectural Support for Programming Languages and Operating Systems (ASPLOS)*, pp. 319—332, March 2011.

[2] J.L. Hennessy and D.A. Patterson, *Computer Architecture: A Quantitative Approach,* Fifth Edition. The Morgan Kaufmann Series in Computer Architecture and Design, September 2011.

[3] H. Esmaeilzadeh, T. Cao, X. Yang, S. Blackburn, and K. McKinley, "What Is Happening to Power, Performance, and Software?," in *IEEE Micro Top Picks from 2011 Computer Architecture Conferences*, vol. 32, no. 3, pp. 110—121, May/June 2012.

[4] H. Esmaeilzadeh, T. Cao, X. Yang, S. Blackburn, and K. McKinley, "Looking Back and Looking Forward: Power, Performance, and Upheaval," in *Communications of ACM Research Highlights*, vol. 55, no. 7, pp. 105—114, May/June 2012.

[5] H. Esmaeilzadeh, E. Blem, R. St. Amant, K. Sankaralingam, and D. Burger, "Dark Silicon and the End of Multicore Scaling," in *International Symposium on Computer Architecture (ISCA),* pp. 365–376, June 2011.

[6] H. Esmaeilzadeh, E. Blem, R. St. Amant, K. Sankaralingam, and D. Burger, "Dark Silicon and the End of Multicore Scaling," in *IEEE Micro Top Picks from 2011 Computer Architecture Conferences*, vol. 32, no. 3, pp. 122–134, May/June 2012.

[7] H. Esmaeilzadeh, E. Blem, R. St. Amant, K. Sankaralingam, and D. Burger, "Power Challenges May End the Multicore Era," to appear in *Communications of ACM Research Highlights*, vol. 56, no. 1, January 2013.

[8] H. Esmaeilzadeh, E. Blem, R. St. Amant, K. Sankaralingam, and D. Burger, "Power Limitations and Dark Silicon Challenge the Future of Multicore," in *ACM Transactions on Computer Systems (TOCS)* vol. 30, no. 3, pp. 11:1–11:27, August 2012.

[9] John Markoff, "Progress Hits Snag: Tiny Chips Use Outsize Power," The New York Time, August 1, 2011. URL: http://www.nytimes.com/2011/08/01/science/01chips.html

[10] H. Esmaeilzadeh, A. Sampson, L. Ceze, and D. Burger, "Architecture Support for Disciplined Approximate Programming," in *Architectural Support for Programming Languages and Operating Systems (ASPLOS)*, pp. 301–312, March 2012.

[11] H. Esmaeilzadeh, A. Sampson, L. Ceze, and D. Burger, "Neural Acceleration for General-Purpose Approximate Programs," in *International Symposium on Microarchitecture (MICRO),* pp. 449–460, December 2012.

[12] H. Esmaeilzadeh, P. Saeedi, B.N. Araabi, C. Lucas, S.M. Fakhraie, "Neural Network Stream Processing Core (NnSP) for Embedded Systems," in International Symposium on Circuits and Systems (ISCAS), pp. 2773-2776, May 2006.

[13] S. Shamshiri, H. Esmaeilzadeh, Z. Navabi, "Instruction-Level Test Methodology for CPU Core Self-Testing," in *special issue of ACM Transactions on Design Automation of Electronic Systems (TODAES) on "Design Validation of Large Systems,"* vol. 10, no. 4, pp. 673–689, October 2005.

[14] H. Esmaeilzadeh, S. Shamshiri, P. Saeedi, Z. Navabi, "ISC: Reconfigurable Scan-Cell Architecture for Low Power Testing," in *Asian Test Symposium (ATS),* pp. 236-241, December 2005.

[15] S. Safari, A. H. Jahangir, H. Esmaeilzadeh, "A Parameterized Graph-Based Framework for High-level Test Synthesis," *Integration, the VLSI Journal,* vol. 39, no. 4, pp. 363–381, July 2006.

[16] H. Esmaeilzadeh, N. Shahidi, E. Ebrahimi, A. Moghimi, C. Lucas, Z. Navabi, "Cim++: A C++Library for Object Oriented Hardware Design," in International Journal of Science and Information Technology (IJSIT), Lecture Notes of 1st International Conference on Informatics, vol. 1, no. 2, pp. 35–41, September 2004.