Prof. Mark D. Hill

Computer Sciences Department University of Wisconsin-Madison 1210 W. Dayton Street Madison, Wisconsin 53706

Phone: 608-262-2196 Fax: 608-262-9777 E-mail: markhill@cs.wisc.edu Web: http://www.cs.wisc.edu/~markhill

May 7, 2014

Dear Cisco Fellowship Selection Committee:

I strongly recommend **MR. JASON POWER** for a Cisco Fellowship*.* He has demonstrated both accomplishments and potential for making emerging heterogeneous CPU-GPU systems more widely effective. He is my best student in several years and a U.S. citizen, which I believe is preferred for this fellowship.

**Background and Teaching.** I am Ph.D. co-advisor of Jason Power with Prof. David Wood. In addition to co-supervising his research, I know Jason well from the work he has done assisting my teaching of programming multicore processors (CS 758: <http://www.cs.wisc.edu/~markhill/cs758/Fall2012/wiki/>) and parallel computer architecture (CS/ECE 757: <http://www.cs.wisc.edu/~markhill/cs757/Spring2014/wiki/>). For CS 758, Jason largely handled parallel programming assignments—revising assignments, answering questions and grading them—and was a consultant on course research projects. He did a first-rate work.

**Computer Architecture.** Jason and I share the field of computer architecture whose importance is arguably growing. During the late 20th century, computer architects harvested the bounty of transistors provided by Moore’s Law to make computers ever faster and cheaper in a manner largely separable from much of the rest of computer science and electrical engineering. In this century, computer advances will require more synergy among experts in many sub-fields, as transistor scaling has issues, energy is a first-order design consideration, and application trends diverge from the handheld to the cloud. See *21st Century Computer Architecture* (<http://cra.org/ccc/docs/init/21stcenturyarchitecturewhitepaper.pdf>).

**Research Big Picture.** Jason’s work focuses on *heterogeneous systems* that combine latency-optimized conventional CPU cores with numerous throughput-optimized *graphics processing unit (GPU)* cores. Heterogeneous systems promised higher performance at lower energy than CPU-only systems, but can be hard to program, which currently squanders part of their potential.

**CPU-GPU Coherence.** Jason’s work seeks to make heterogeneous systems better in several ways. First, following from his AMD internship, Jason developed a coherence protocol to facilitate sharing among CPU and GPU cores. Just using a conventional CPU coherence protocol would not work well, because the GPU cores’s high request bandwidth would have overwhelmed CPU caches, even if the data are rarely present. For this reason, Jason adapted region coherence—originally used by CPUs for different reasons—to providing CPU-GPU coherence with an order of magnitude fewer coherence requests. This worked because region coherence groups cache blocks into regions (e.g., 16 blocks per region) and often gets coherence permissions once per region instead of once per block. This work was published in the premier computer architecture venue of MICRO 2013.

**GPU Address Translation.** Second, Jason analyzed virtual-to-physical address translation (memory management units or MMUs) for GPUs. Having GPUs logically translate addresses like CPUs enables CPU-GPU programs to share data in rich pointer-based data structures without recalculating stored pointers between CPU and GPU use. Actually performing GPU address translation with the same micro-architecture mechanisms at CPUs, however, does not work well, because GPU cores can generate hundreds of memory references per cycle (versus 1-2 per CPU) and longer-latency MMU misses tend to come in bursts of many tens (versus rare CPU bursts). Through analysis, Jason developed an effective GPU MMU using a synergy of good engineering decisions, e.g., level-one TLB after the memory coalescer, shared highly-threaded page-table walker, and optional page walk cache. This work appeared in the first-rate computer architecture venue of HPCA 2014.

**Gem5-gpu Simulator.** Third, Jason co-led the release of the open-source *gem5-gpu* simulator. This simulator combines the full-system software, CPU modeling, and memory hierarchical tools of our gem5 simulator and the GPU modeling of the University of British Columbia’s GPGPU-Sim simulator. This fusion enables the analysis of heterogeneous systems running operating systems, as was done for the GPU address translation study just discussed. While gem5-gpu resulted in an IEEE Computer Architecture Letters [CAL 2014] paper, its real value is facilitating other researchers. Newton said, “If I have seen further it is by standing on the shoulders of giants.” In computer architecture ideas work similarly, while tools, such as gem5-gpu, accelerate progress by reducing redundant work building research infrastructure.

**Ongoing Work.** Fourth, Jason’s Ph.D. is under development. It will include some of the above work, especially on GPU address translation, as well an exciting new work on future GPUs implemented with multiple chips stacked on a Silicon interposer. Interesting design spaces include non-client systems, such as servers and even Internet routers. To this end, Jason has been working with database professor Jignish Patel to see how Patel’s dictionary-encoded decision support database systems might run on GPUs. Initially, the answer was, “Not well.” However, Jason vastly improved the performance. Importantly, this work has uncovered potential opportunities to improve heterogeneous system hardware and perhaps database software. I look forward to supervising Jason as he develops this potential.

Due to his accomplishments and potential, I strongly recommend **MR. JASON POWER** for a Cisco Fellowship*.*

Sincerely,



Mark D. Hill

Gene M. Amdahl Professor of Computer Sciences

Professor of Electrical and Computer Engineering

ACM Fellow

Fellow of the IEEE

Biography at <http://www.cs.wisc.edu/~markhill>