

John W. Haskins, Jr., Ph.D.

john.haskins.jr@gmail.com

Objective

To secure a challenging position researching and developing high-performance computing systems and software.

Skills

Expertise in:

- C and assembly language (x86, ARM)
- application performance tuning, optimization
- multithreaded programming
- Linux/UNIX use and administration
- microarchitecture simulation

Direct experience with:

- MIPS, 680x0, and SPARC assembly language
- C++, Python, Perl, Lisp, Scheme
- CUDA, Verilog, HTML
- Synplicity, ModelSim, Xilinx ISE Foundation, Git
- Android, ChromeOS development

Work Experience

Qualcomm; Raleigh, NC

2009 – present

Staff Engineer

- led technical development of statistical methods for determining (dis)similarities between benchmark applications
- worked as part of a team to port Flash Lite to Android for HP AirLife netbook
- developed high-performance copy routines for graphics processing on Scorpion processors
- developed optimizations for V8, the Android JavaScript engine, on Krait processors

NVIDIA Corporation; Durham, NC

2007 – 2008

Senior Systems Software Engineer

- development, debugging, and maintenance of user-level driver for the Compute Unified Device Architecture (CUDA) general-purpose computing on graphics processing unit (GPGPU) tools
- integrated CUDA into the mainline NVIDIA graphics driver
- led upgrade of CUDA tracing infrastructure
- performance analysis of CUDA-capable hardware
- collaborated in CUDA-based OpenCL library development

Institute for Defense Analyses Center for Computing Sciences; Bowie, MD

2002 – 2007

Research Staff Member

- led Xen virtualization research effort to analyze performance of virtual kernels, adding and extending low-level benchmarking functionality
- led asynchronous ASIC design and implementation on Xilinx FPGA
- researched probabilistic methods for solving systems of equations
- analyzed cache transaction timings in 64-bit microprocessors

University of Virginia Department of Computer Science; Charlottesville, VA

1998 – 2002

Graduate Research/Teaching Assistant

- managed small team of teaching-, and lab assistants
- taught undergraduate lab sections and proctored lab assignments, held office hours, graded assignments
- occasionally substitute taught graduate-level computer architecture class
- created, and led research efforts: Differential Multithreading, Minimal Subset Evaluation, Memory Reference Reuse Latency, and μ OS embedded kernel

Institute for Defense Analyses, Center for Computing Sciences; Bowie, MD

1997 – 1998

Research Intern

- researched Linux file system
- developed low-level Ext2 metadata manipulation primitives

Continued...

John W. Haskins, Jr., Ph.D.

Education

Doctor of Philosophy in Computer Science University of Virginia, May 2003
Dissertation: *Accelerating Sampled Microarchitecture Simulation: Rapid Warm Up for Simulated Hardware State*
Advisor: Kevin Skadron

Master of Computer Science University of Virginia, May 2000
Thesis: *μOS: A Multithreaded, Dynamically Reconfigurable Embedded Operating System Kernel*
Advisor: John A. Stankovic

Bachelor of Science in Computer Science Georgia Institute of Technology, December 1997

Open Source Software Projects and Contributions

3SAT

www.cs.virginia.edu/~jwh6q/3sat-web/

3-variable conjunctive normal form satisfiability solver, and simple arbitrary-precision integer library; package contains

- source code for 3SAT solver
- source for integer library used by the solver
- sample CNF circuits

STAX

www.cs.virginia.edu/~jwh6q/stax-web/

Programming language for manipulating input strings in a multiple-stack finite state machine; package contains

- STAX source code: lexer, parser, bytecode interpreter
- STAX-to-bytecode compiler
- bytecode virtual machine for executing STAX programs
- sample STAX programs

LWT

www.cs.virginia.edu/~jwh6q/lwt-web/

Light-weight user-level thread library for Linux/x86 with C interface; package contains

- LWT source code (C and x86 assembly language)
- sample programs that use LWT

SS3B_MRRL

www.cs.virginia.edu/~jwh6q/mrrl-web/

Extension of the SimpleScalar simulation framework for performing Memory Reference Reuse Latency (accurate, accelerated cache/branch predictor initialization for execution-driven simulation) warmup; package contains

- source code for MRRL-enabled SimpleScalar out-of-order microarchitecture simulator
- MRRL benchmarking tools

Conference and Journal Publications

J. W. Haskins, Jr., and K. Skadron. "Accelerated Warmup for Sampled Microarchitecture Simulation." In *ACM Transactions on Architecture and Code Optimization*, Mar. 2005.

K. R. Hirst, J. W. Haskins, Jr., and K. Skadron. "dMT: Inexpensive Throughput Enhancement in Small-Scale Embedded Microprocessors with Differential Multithreading." In *IEE Proceedings on Computers and Digital Techniques*, Jan. 2004.

J. W. Haskins, Jr., and K. Skadron. "Memory Reference Reuse Latency: Accelerated Warmup for Sampled Microarchitecture Simulation." In *Proceedings of the International Symposium on Performance Analysis of Systems and Software*, Mar. 2003.

J. W. Haskins, Jr., and K. Skadron. "Minimal Subset Evaluation: Rapid Warm-up for Simulated Hardware State." In *Proceedings of the 2001 International Conference on Computer Design*, Sept. 2001.

L. F. Friedrich, J. Stankovic, M. Humphrey, M. Marley, and J. W. Haskins, Jr. "A Survey of Configurable Component-based Operating Systems for Embedded Applications." *IEEE Micro*, May 2001.

J. W. Haskins, Jr., K. R. Hirst, and K. Skadron. "Inexpensive Throughput Enhancement in Small-Scale Embedded Microprocessors with Block Multithreading: Extensions, Characterization, and Tradeoffs." In *Proceedings of the 20th IEEE International Performance, Computing, and Communications Conference*, Apr. 2001.

J. W. Haskins, Jr., and K. Skadron. "Differential Multithreading: Recapturing Pipeline Stall Cycles and Enhancing Throughput in Small-scale Embedded Microprocessors," *Proceedings of the Workshop on Complexity-effective Design* held in conjunction with the International Symposium on Computer Architecture, June, 2000.