A Tutorial on MARSS: A Cycle-Accurate Full System Simulator for Multicore Architectures

Introduction:
The microarchitecture and architecture research community has relied on cycle accurate simulators for validating new techniques that are being actively investigated and developed in this community. There is also an emerging need to go beyond the memory and processor components in modeling and simulating a computing platform at full system level, encompassing system level hardware components as well as critical software components such as the OS and libraries. It is thus important to have a full system simulation tool that incorporates realistic simulation models for other systems level components such as the chipset, DRAM, network interface cards and peripheral devices in addition to accurate simulation models for single and multicore processors implementing the x86 ISA. We have developed an open source full system simulation tool, called MARSS – Micro Architectural and System Simulator, to meet this critical need. Current public release of MARSS supports single core and multi-core X86 platforms, running Linux OS. Unmodified X86 binaries can be simulated or emulated in MARSS. MARSS is currently being extended to support the ARM ISA in an Android environment.

MARSS extends the widely used QEMU framework – a full system emulator that supports multiple ISA, with cycle accurate simulation framework based on PTLsim. MARSS provides a modular framework to implement Core and Cache models like Out-Of-Order core, Atom like In-Order core, Write-back/Write-through caches, MESI and MOESI coherent Caches, Bus and Switch interconnects. This modular framework also allows users to easily integrate other open-source or proprietary tools, for example DRAMSim2 (detailed DRAM simulator) uses MARSS as front-end to generate traffic to DRAM model.

MARSS has attracted not only the academic researchers but also industrial researchers from Intel, Rambus, Cray Systems and others as viable open-source alternative to many existing proprietary and closed source tools. Researchers from Rambus have evaluated MARSS's performance with real hardware and are working towards improving its co-relation with hardware. Intel research labs have utilized MARSS framework to develop multicore memory tracing tool and they are working to integrate it with x86 Android Emulator.

Full System simulation framework of MARSS with support for unmodified OS and libraries enables researchers to evaluate new programming languages focused on parallel and heterogeneous architectures and power consumption of the system. MARSS also provides attractive platform to evaluate hardware-software co-design. Because of such features MARSS
provides a common platform for researchers in field of architecture, operating system, compilers, programming languages, etc. and PACT attracts researchers from all these fields. Hosting MARSS tutorial at PACT will provide great opportunity for attendees to have hands-on experience with MARSS.

A full day tutorial on MARSS is also planned for International Symposium on Computer Architecture (ISCA) 2012 in Portland, OR.

Website: http://marss86.org

Contact Info: ghose@cs.binghamton.edu  apatel@cs.binghamton.edu

This full-day tutorial focuses on the MARSS simulation environment and is organized into the following sections:

**PART 1**: Introduction to MARSS and getting started on MARSS, demos (Ghose, Patel - SUNY-Binghamton)

**PART 2**: Internal components of MARSS:

- Full system framework and Processor Core models (Patel - SUNY- Binghamton)
- Coherent Caches, Interconnections and Simple RAM models (Afram - SUNY- Binghamton)

**PART 3**: Modifying MARSS for specific projects and examples with:

- Modifying the processing core (Afram, Patel, SUNY-Binghamton)
- Coherent caches (Patel, Afram, SUNY-Binghamton)
- OS/System level modifications (Afram, Patel, SUNY-Binghamton)

**PART 4**: Industrial experience with MARSS: Rambus and Intel

- Rambus: Validating MARSS performance with real platforms
- Intel: Developing Multicore Memory Tracing capabilities in MARSS and integrating MARSS to x86 based Android Emulator

**PART 5**: Ongoing development activities and closing remarks (Patel, Afram, SUNY- Binghamton)
Who Should Attend:

Academic and industry researchers who have a need to use and modify a full system, cycle-accurate simulator that implements pervasive ISAs (x86 and ARM), and whose simulation results can be validated in many situations on widely available platforms/OSs. MARSS is of significant value in designing, developing and evaluating systems that range from server platforms to portable devices.

Organizers and Presenters:

**Kanad Ghose, SUNY Binghamton** : Kanad is a Professor and Chair in the Computer Science Department at SUNY Binghamton and holds a PhD in Computer Science from Iowa State University. His research interests are in power-aware microarchitectures and systems. His research group developed and released several microarchitectural simulators for the research community (msim, PTLsim and MARSS). MARSS is the first-ever public domain full system microarchitectural simulator for the X86 ISA, providing cycle-accurate full system simulation facilities.

**Avadh Patel, SUNY Binghamton** : Avadh is currently pursuing his PhD degree at SUNY Binghamton in the area of multicore server architectures. Avadh is the principal architect of MARSS. Avadh started working on MARSS for his PhD thesis in 2009 and he is maintaining MARSS as an open-source project since.

**Furat Afram, SUNY Binghamton** : Furat is currently pursuing his PhD degree at SUNY in the area of Computer Architecture and energy-aware data center design. Furat is the co-designer of MARSS.

**Hongzhong Zheng, RAMBus** : Hongzhong received the BS and MS degrees in electrical engineering and computer science from Huazhong University of Science and Technology, China, and the PhD degree in electrical and computer engineering from the University of Illinois at Chicago in 2009. He is currently a senior member of technical staff at Rambus, Inc. His research interests include computer architecture, energy-efficient computing system designs, novel memory architecture and performance modeling. He is a member of the ACM and the IEEE.

**James Tringali, RAMBus** : James Tringali is currently a Senior Manager of Research at Rambus and a charter member Rambus Labs. He has been designing and programming computer systems for over 30 years. Over the course of his career his design work has spanned from scientific, visual and personal computers to embedded systems for high volume consumer applications. James graduated from University of California, Irvine with a B.S. Electrical Engineering and has since spent time honing his craft at Rockwell, FileNet, MIPS, Apple, Silicon Graphics, Matrix Semiconductor and SanDisk. Three of these companies were startups when James joined, and he was a founding member at two of them, FileNet and Matrix Semiconductor.
James Greensky, Intel: James Greensky received his BS and MS in computer science from the University of Minnesota. Currently, he is a researcher in the UHPC Microarchitecture Lab within Intel Labs.