Mahita Nagabhiru

https://mnagabh.wordpress.ncsu.edu | 2717 Brigadoon Dr, Apt 22, Raleigh-NC 27606 | +1 2035508017 | mnagabh@ncsu.edu

EDUCATION

PhD in Computer Engineering — Aug-2023 (expected)
North Carolina State University, USA — GPA: 3.67/4.0
B.E (Hons) Electronics and Instrumentation — May-2011
Birla Institute of Technology and Science, Pilani, India — GPA: 8.23/10.0

RESEARCH WORK

My research work aims at inspecting the H/W support for lock-free programming specifically looking into newer atomics like MCAS (multi-word compare-and-swap) and their implementation in H/W entirely.

Lock-free programs are a class of **"non-blocking" multithreaded** algorithms that are used in critical pieces of software like OS kernel and databases; unlike in lock-based programs, multiple threads in lock-free programs can enter **critical sections** concurrently; on encountering a conflict, only one thread commits while others fail and retry. Ensuring lock-freedom is notoriously difficult even in simple structures like **linked-lists** today as H/W supports only single-word atomics like CAS (compare-and-swap) and hence reasoning about **linearizability** and proving it based on simulations alone is not exhaustive. Additionally, it suffers from non-trivial memory reclamation problems and ABA problem. This process becomes much easier if the hardware provides efficient MCAS/DCAS/CASN features.

I developed a lock-free microbenchmark for this research by bringing together existing complex protocols into a similar format. Additionally, I extended ARM-TME (transactional memory extension) model on gem5 simulator to model an MCMS (multi-compare multi-swap) instruction as one possible solution for guaranteeing lock-freedom entirely in H/W.

RELEVANT COURSES

Computer micro-architecture, Architecture of Parallel Computers (cache coherence and memory consistency), Compilers, Digital ASIC Design, Operating Systems Principles, GPU Architecture (GPGPUs), Quantum Computing

COURSE PROJECTS

Computer Architecture:

1. Designed a cycle accurate simulator for **Dynamic Instruction Scheduling in out-of-order Superscalar Processor** using C++; features included multi-functional execution units, configurable scheduler and superscalar widths.
2. Designed a Simulator for a **Multilevel Cache** in C++ with **WBWA, WTNA write policies** and **LRU replacement policy**. Used the generic Cache module to have a Victim cache for L1 Cache. Further enhanced this Cache module to implement **bus based Cache Coherence Protocols (MSI, MESI, MOESI, Dragon)** in a Symmetric Multiprocessor (SMP)-like environment with **L2** as **shared cache**.
3. Simulated a configurable **branch predictor** with BTB for **Bimodal, Gshare and Hybrid** predictors and analyzed for various configurations.

PROFESSIONAL EXPERIENCE

Arm Holdings, Cambridge, UK — June-2022 to Aug-2022, 3mons
Research Intern

Areas of work: Developed DCAS lock-free microbenchmarks setting the foundation for evaluation of traditional lock-free programs with single atomics like single-CAS vs multi-word atomicity semantics. Preliminary inspection of H/W support for multi-word atomicity semantics modeled on gem5 simulator.

Arm Holdings, Cambridge, UK — June-2019 to Aug-2019, 3mons
Intern

Areas of work: Created a framework for workload characterization for the SVOS team. Started a setup to analyze the payloads used within the team to assess if they provided the right stresses for various validation purposes and proposed appropriate extensions to this setup.

Graphene Semiconductor Services Pvt Ltd, Bangalore, India — Apr-2014 to May-2015, 1yr 2mons
Member Technical Staff

Areas of work: On-site customer support to crucial customers like Qualcomm, Sandisk and Lantiq Communications (now Intel) as a Static Timing Analysis expert. Activities involved driving timing closure of the chips, complete constraints development and verifications and provide CTS strategies for the PnR team.

Texas Instruments, Bangalore, India — June-2011 to Apr-2014, 2yrs 11mons
Design Engineer

Areas of work: Worked on 28nm and 40nm technology nodes in various Physical Design and Static Timing Analysis roles. Ownership of the design library for the entire SoC team, PnR experiments for various Sub-blocks, Netlist to GDSII flow flush of a crucial test-chip and development of timing constraints for various block/top level of the SoCs.

US PATENT: Placement aware clock gate cloning and fan-out optimization- number: **8,661,374**

Technical Skills

Programming: C/C++ *OLDER:* Tcl, Perl, Shell (bash/tcsh), Verilog, openMP, CUDA. **Tools:** GDB, gem5 simulator, *OLDER:* Synopsys PrimeTime, Synopsys Design Compiler, Synopsys Talus and IC compiler, Mentor Graphics ModelSim, Apache RedHawk.