# Saurabh Labde

Digital Design · Verification · Computer Architecture
2510 Avent Ferry Road, Apt# 205 , Raleigh , NC - 27606
+1 (984)-218-6364 | sslabde@ncsu.edu | saurabhlabde29 | saurabhlabde

## Summary

Graduate Student at NC State University pursuing MS in Computer Engineering. Proficient in Verilog, System Verilog, C, C++ and python scripting. Equipped with strong domain knowledge and experience in **complex logic design, RTL coding, low power design, static timing analysis, DFT, CPU/ASIC/SOC verification infrastructure using UVM methodology with SV Assertions and high-performance CPU/GPU micro-architecture** demonstrated through several academic and personal projects, relevant coursework and research volunteer work.

## Education

**North Carolina State University** — *Raleigh, NC*
M.S. in Computer Engineering — *Aug. 2017 - Expected May. 2019*

- **Relevant Courses:** ASIC and FPGA Design(Verilog), ASIC Verification(System Verilog), Advanced Verification with UVM, Advanced Micro-architecture , Microprocessor Architecture , Architecture of Parallel Computers(Multi-core & Many-core systems), Advanced Computer Architecture(GPU & Vector Processors), Operating Systems, SOC Verification Using System Verilog (Online)

**University Of Mumbai** — *Mumbai, India*
B.E. in Electronics and Communications Engineering — *Aug. 2013 - May.2017*

## Skills

**Languages** C, C++, Verilog, System Verilog, Python, Ruby, Perl, Assembly (x86, MIPS, ARM), Shell , CUDA, Java
**Tools** ModelSim, QuestaSim, Veloce Emulator, Synopsys Design Vision, MATLAB, Xilinx ISE, Git, Make Utility, GDB, Valgrind, LLDB
**Platforms** Linux, Unix(MAC OS), Windows, XINU-OS, GPGPU-sim, SNIPER-sim, GEM5-sim

## Projects

### Verification *(EDA Tools: ModelSim, QuestaSim, Veloce Emulator. Languages: System Verilog , Python Scripting)*

**Emulation Compatible Chip Level Verification Environment for LC3 Microcontroller using UVM Framework** — *Dec. 2018*

- Constructed a hierarchical, re-usable and emulation compatible chip-level UVM based test bench for the verification of LC3 Micro-controller which includes environments, agents, BFM's, predictors, coverage collectors , stimulus generators and scoreboards.
- Designed a detailed test plan and achieved 100% coverage through constrained random testing, assertions and directed test cases.

**UVM based Block Level Verification IP for AMBA-APB bus Protocol** — *Aug. 2018*

- Designed and Implemented a hierarchical and re-usable verification environment for APB protocol using UVM class libraries
- The package includes APB bridge master and APB slave agents, drivers , monitors, environment, sequences and sequencers.

**Functional Verification of 5-stage Pipelined LC3 Micro-controller** — *April. 2018*

- Designed a layered test bench and a cycle accurate reference model to verify the data and control path of LC3 micro-controller with a comprehensive instruction set. Full functional coverage was achieved using Constrained random testing and assertions.

### RTL Design *(EDA Tools: ModelSim, Synopsys Design Vision. Languages: Verilog )*

**Hardware accelerator for simplified convolutional neural network** — *Nov. 2017*

- Designed a synthesizable ASIC implementing the two staged convolutional neural network arithmetic on inputs read from SRAM .
- The design was verified for functional correctness and synthesized to achieve minimum area and clock period.

**Synthesizable ASIC for SHA-256 Cryptographic Hash Algorithm** — *Jan. 2019*

- Implemented the SHA-256 hash algorithm using Verilog HDL . Synthesized the RTL to gate level using Synopsys design vision and ensured the setup and hold time constraints are met. Leveraged a pipelined architecture to optimize the design for performance.

### CPU / GPU Architectural Simulators and Operating Systems *(Languages: C , C++. Platforms: GPGPU-Sim, XINU OS )*

**Dynamic Instruction Schedulling in an OOO Superscalar Processor** — *Nov. 2017*

- Implemented a cycle accurate simulator for an out-of-order superscalar processor based on Tomasulo's algorithm that fetches, dispatches, issues and executes N instructions per cycle.The simulator successfully handled RAW, WAW and WAR hazards .

**Multi-level Cache Hierarchy Simulator** — *Sept. 2017*

- Modelled a trace driven, highly parameterized two-level cache hierarchy simulator with a support for L1 victim cache.
- Implemented LRU ,LFU cache replacement policies along with write through - not allocate and write back - allocate write policies.

**Phase Aware Warp Schedulling (PAWS) in GPGPU** — *May. 2018*

- Extended GPGPU-sim to include Phase Aware Warp schedulling policy which performs closer to the best of GTO and RR policies. Benchmarked with CUDA SDK, Rodnia and Parboil kernels and achieved an 8% improvement in the IPC.

**Dynamic Branch Predictor Simulator** — *Oct. 2017*

- Developed a simulator to model bimodal, gshare and hybrid branch predictors with Branch Target Buffer(BTB).

**Cache Coherence Protocol Simulator** — *Oct. 2017*

- Developed a trace driven multiprocessor simulator supporting MSI, MOSI, MOESI bus based cache coherence protocols.

**Virtual Memory Management System in Xinu Operating System** — *Dec. 2018*

- Extended the x86 based Xinu OS to support a virtual memory system with demand paging and lazy allocation optimizations.

## Work Experience

**ARPERS Research Group - NC State University** — *Raleigh, NC*
CPU Architecture research volunteer — *Jun. 2017 - Aug. 2018*

- Ported a set of persistent memory microbenchmarks to work with SNIPER multicore simulator using simulator hooks.
- Created Python scripts to automate the compilation and generation of CPI and power visualizations.