

Barry (Muqi) Xu

+1 (217) 637-3442 | muqixu@outlook.com | [linkedin.com/in/muqi-xu](https://www.linkedin.com/in/muqi-xu)

Education

University of Illinois Urbana-Champaign

Expected Dec. 2026

Bachelor of Science in Computer Engineering

GPA: 3.73/4.0

- **Relevant Coursework:** Computer Organization and Design, Applied Parallel Programming, Computer Systems Engineering, Digital Systems Laboratory, Intro to VLSI System Design, IoT and Cognitive Computing.

Carnegie Mellon University

Expected May 2028

Master of Science in Electrical and Computer Engineering

Starting Jan. 2027

Technical Skills

Programming: SystemVerilog, C/C++, Python, Shell, Makefile

Tools: Synopsys VCS, Verdi, Xilinx Vivado/Vitis, GDB, Cadence Virtuoso, Git

Protocols: AHB, MESI, I2C, UART

Verification & Methodology: UVM, SVA, Functional Coverage, Constrained-Random, Computer Architecture, OOP, Waveform debug, VLSI, LLM-driven Verification Assistance

Experience

Design Verification Intern

Jan. 2026 – Present

VortexNet LLC

- Designed and verified a UVM-based environment for a 4×4 INT8 matrix multiplication tile with programmable weights and scaling, ReLU activation, saturation, and valid/ready streaming interfaces.
- Built reusable **agents, drivers, monitors, sequencers**, a bit-accurate **reference model**, and a self-checking **scoreboard** using UVM analysis ports to validate datapath and interface behavior.
- Wrote **25+ SystemVerilog Assertions** for valid/data stability under backpressure, configuration-handshake sequencing, and busy/config mutual exclusion using concurrent properties, temporal implication, cycle delays, and `$stable`.
- Built **Claude Code Skills for SVA generation**, guiding LLMs to translate design requirements into validated assertions with reset, coverage, and vacuity checks.
- Created **20 directed, constrained-random, and saturation-stress sequences** with randomized output backpressure; collected functional coverage across weight, scale, and ReLU scenarios and used Verdi waveform debug to isolate corner-case failures.

Selected Projects

Two-Way Superscalar Out-of-Order RISC-V CPU Design and Verification

Jan 2026 - August 2026

- Designed and verified a two-way superscalar **out-of-order RV32IM processor** with register renaming, ROB, RAT/RRAT, free list, physical register file, reservation stations, and split load/store queues.
- Built **10 directed assembly tests** and a **random instruction generator** to validate instruction execution, data dependencies, simultaneous issue, branch recovery, pipeline flushes, load-store ordering, and execution-unit contention.
- Generated **instruction/commit logs** to trace architectural state and localize failing instructions, then used **Verdi waveforms** to root-cause ROB, rename, and memory-system bugs.
- Wrote **SystemVerilog Assertions** for in-order retirement, pipeline-flush correctness, legal ROB behavior, interface stability, and architectural register invariants; used assertion failures to identify control and timing bugs.
- Achieved **100% ISA functional coverage** and used coverage results, performance counters, and Verdi waveform debug to isolate architectural and RTL issues; optimized the implementation to reach **575 MHz**.

Four-Way Set-Associative Cache Design and UVM Verification

Jan 2025 - May 2025

- Designed and verified a 4-way, 16-set cache with 32-byte cache lines, **write-back** and write-allocate policies, byte-masked CPU accesses, a **Tree-PLRU** replacement policy, and a 256-bit memory interface.
- Developed a reusable **UVM environment** with an active CPU-side agent, a reactive memory-side agent, sequencers, drivers, monitors, and TLM analysis connections to verify cache requests, responses, allocations, and writebacks.
- Refactored the behavioral cache model into a self-checking **reference model and scoreboard** that tracked data, tags, valid/dirty state, and Tree-PLRU history.
- Created **20 directed and constrained-random sequences** targeting read/write hits, clean and dirty misses, same-set conflicts, fill-and-evict behavior, Tree-PLRU replacement. Achieving **100% functional coverage**.

UVM-Based AMBA AHB Verification IP

August 2026 - present

- Developed and extended a reusable **UVM verification IP** with configurable **master and slave agents**, sequencers, drivers, monitors, virtual sequences, register adapters, and active/passive operating modes.
- Supported **SINGLE, incremental, and wrapping bursts**; variable transfer sizes; aligned addresses; **1-KB boundary constraints**; BUSY/IDLE insertion; and HREADY-controlled wait states through constrained transaction generation.
- Implemented **pipelined address and data-phase handling** with read/write response collection, configurable abort-after-error behavior, **on-the-fly reset recovery**, callbacks, and sequencer response support.
- Integrated **functional coverage** for direction, size, burst, HTRANS, and HRESP behavior, along with **throughput analysis, UVM RAL integration**, and direct, back-to-back, and register-model test sequences.