Qifa(Richard) Wang

qifaw2000@gmail.com | 617-816-5834 | qifa-wang-28a180170 | rthelionheart24

Objective: Motivated engineering graduate passionate about computer architecture, SW/HW co-design, accelerated computing, HPC, quantum computing, and software engineering. Quick to adapt to new technologies and concepts.

Education

Computer Science & Engineering — ***Master of Science in Engineering*** Aug 2023 - May 2025
Rackham Graduate School, University of Michigan GPA: 4.00
- Graduate Student Instructor for Quantum Computing and Parallel Computing

Computer Science, minors in Math and Physics — ***Bachelor of Science in Engineering*** Aug 2020 - May 2023
College of Engineering, University of Michigan *Summa Cum Laude, Dean's List, University Honors*, GPA: 3.77

Skills

- **Coursework:** Computer Architecture and Microarchitecture, Parallel Computing and Architecture, GPU Programming, Machine Learning, Quantum Computing and Architecture, Compiler Design, Data Structure and Algorithms, Operating System, Web Systems
- **Languages:** C/C++, Python, Go, Rust, Java, CUDA, OpenMP, MPI, Verilog/SystemVerilog, Chisel, Tcl, Javascript, TypeScript
- **Frameworks and tools:** PyTorch, CUDA-Q, Qiskit, AWS, Docker, Chipyard, Synopsys (DC, VCS, Verdi), GNU, LLVM, Valgrind
- **Multilingual:** English, Mandarin, Cantonese, French

Work Experience

Apple Inc. — ***Hardware Technology Intern*** May 2024 - Aug 2024
- Implemented PPROC method to evaluate hardware coverage on pixel processing module down to bit-field-level. Output indicated 74% coverage and identifies 22% critical under-covered areas; arch test suites optimized upon communication with architecture team and full coverage achieved. Output exposed critical bugs in architectural C-model that was later resolved by maintainers.
- Constructed heuristic-based algorithm using PPROC metadata to optimize arch test coverage using dynamic parameter adjustment. Integrate hardware coverage PPROC method into **C/C++** bare-metal testing infrastructure and architectural C-model to run and provide feedback in real time.

Werfen — ***Software & Algorithm Development Intern*** May 2021 - Aug 2021
- Improved Mercury, the point-of-care diagnostic tool for blood coagulation. Engineered core embedded modules to process customized medical algorithm and monitor blood data; engineered GUI module for real-time feedback and data visualization using **C/C++ and Qt framework**. Built from scratch a **real-time data processing pipeline** to handle, process data from medical devices and display results on GUI; prototype served as starting point for the next-gen product.

Technical Projects

Architecture — ***R10K Out-of-Order Processor based on RISC-V ISA*** Jan 2024 - May 2024
- Designed and implemented out-of-order superscalar processor with N-way execution at **RTL level** using **SystemVerilog**, featuring **Tomasulo's algorithm**, instruction and data caches (prefetching, associative, and non-blocking with victim cache), G-share best branch predictor, return address stack, and reservation stations.
- Optimized performance with early tag broadcast and robust memory hierarchy including load-store queue with data forwarding. Achieved 30% improvement in CPI and clock period of 15.5ns. Verified and tested the design using **Synopsys DC** and exhaustive benchmarks including alexnet, dft, matrix multiplication, and saxpy.

GPU — ***Batched Quantum Circuit Simulation on CUDA-ready GPU*** Sep 2023 - May 2024
- Developed a **CUDA-based** quantum simulation framework for performing batched quantum experiments on GPUs, enabling parallel execution of quantum circuits with varying gate counts and types. Implemented **synchronization strategy** to handle non-deterministic quantum operators and efficient **shot-branching**.
- Achieved **superlinear speedup** in runtime performance and optimized memory usage through task batching and state management.

Compiler — ***Compiler Optimization for CUDA Memory Coalescing (COALDA)*** Sep 2023 - Dec 2023
- Developed a IR-level CUDA compiler optimization tool using **AST and canonical forms** to transform uncoalesced memory accesses into coalesced patterns. Implemented NVCC-LLVM pipeline that allows for seamless integration with existing CUDA toolchains and optimziation to LLVM IR.
- Achieved 9x L2 cache writeback reduction and 6.4x read bandwidth improvement, validated through **NVIDIA Nsight Compute**.

Operating System — ***Linux-Based Operating System*** Jan 2023 - May 2023
- Developed a **custom thread library** to support multi-cpu, multi-threaded execution using **C/C++**.
- Implemented a **kernel pager** for efficient management of applications' virtual memory using both Swap and physical memory.
- Engineered a **multi-threaded network file server** for reliable data exchange. Designed a hierarchical file system with secure file ownership and permissions and fine-grained locking mechanisms.