Yufei Shi

412–251–8844 yfshi@cmu.edu yufei-shi yshi02

Education

Carnegie Mellon University **Pittsburgh, PA**
Bachelor of Science in Electrical and Computer Engineering, GPA: 3.67 Expected May 2024

- **Teaching assistant**: Intro to Computer Systems (4 semesters); Computer Systems and HW-SW Interface (1 semester)
- **Courses**: Intro to Computer Architecture, Parallel Computer Architecture & Programming, OS Design & Implementation

Experience

Research Assistant to Prof. Brandon Lucia **Pittsburgh, PA**
Carnegie Mellon University, ABSTRACT Research Group January 2023 — Present

- Evaluated and improved Coarse-Grained Reconfigurable Architecture (CGRA), which is a hardware dataflow architecture that allows reconfiguration of processing elements for different tasks and applications.
- Analyzed CGRA's memory consistency model by developing testing programs with diverse memory access patterns, tracing their execution on a simulator and identifying violations of sequential consistency in the trace.
- Added support for lightweight threads in the dataflow execution simulator by extending the ISA with a new processing element, implementing thread dispatch synchronization, and resolving legacy bugs within the simulator.
- Developed a testing framework, validated a set of workloads including dense and sparse matrix multiplication kernels, and benchmarked the execution efficiency of a CGRA with lightweight threading across these workloads.
- Contributed to the design of a programming language for expressing parallelism in CGRA applications by writing programs using current parallel programming methods and comparing their efficacy to the language design.

Projects

Parallel Mesh Collision Detector | C++, CUDA, GDB, OpenMP, Open3D **Apr-May 2023**

- Developed a parallel algorithm to accurately determine the minimum distances between convex meshes, which can be used to detect potential collisions between objects in real-time for tasks such as robotic arm motion planning.
- Implemented Gilbert-Johnson-Keerthi algorithm and optimized it by parallelizing its support function in CUDA.
- Developed a dynamic visualization framework that simulates object motion and shows distances between objects.
- Achieved a 20x speedup over the baseline by combining CUDA and OpenMP with additional optimizations.

In-Order 2-Way Superscalar RISC-V Processor | C++, SystemVerilog, Synopsys VCS & DC **Jan-Apr 2023**

- Designed and implemented an RV32I processor featuring a 2-way superscalar in-order 5-stage pipeline.
- Implemented branch prediction and data forwarding capabilities to alleviate data hazards in the 5-stage pipeline, and ensured their seamless integration with the final superscalar pipeline to maximize IPC of the processor.
- Conducted rigorous timing and power optimizations over design iterations by iteratively analyzing synthesis report and making adjustments to the design; achieved a 15% IPS increase while reducing power consumption by 80%.

Memory Hierarchy Simulator | C++, GDB, Pin Tool, Python **Sep-Oct 2022**

- Developed a memory hierarchy simulator supporting two levels of cache to analyze memory hierarchy design.
- Implemented flexible configuration for each cache level, including size, sets, ways, block size, and eviction policies.
- Conducted design space exploration using the simulator, performed trade-off analyses for AMAT, power, and area, and identified the Pareto optimal cache configuration for applications in the SPEC2017 intspeed benchmark suite.

Skills

Programming Languages: C, C++, Python, Rust, Shell, x86 Assembly
Hardware Design Tools: SystemVerilog, Synopsys VCS, Synopsys Design Compiler, Intel Quartus
Developer Tools: GDB, Git, Vim, VS Code, Valgrind, Make, Regex, Anaconda, Various Linux Distros, Pin Tool
Technologies: MATLAB, SOLIDWORKS, NumPy, Matplotlib, OpenMP, MPI, CUDA, OpenGL, HTML, LaTeX