Chenyang Zhou

cz2791@columbia.edu

chenyang-zhou chz05 My Website

Research Interest

I combine expertise in computer architecture, compilers, and system design to pursue two parallel research interests: developing high-level hardware description languages and compiler-driven frameworks that make hardware modeling, verification, and simulation more intuitive and productive, and advancing accelerator design and optimization to improve the performance, efficiency, and programmability of emerging AI and data-centric workloads.

Education

Columbia University *Aug 2023 – Dec 2024*
MS in Computer Science

- GPA: 3.4/4.0
- **Relevant Coursework:** Computer Architecture Research Project(A), Embedded Systems(B+), Computer Networks(A-).

University of California San Diego *Sept 2021 – June 2023*
BS in Computer Science

- GPA: 3.6/4.0
- **Relevant Coursework:** Computer Architecture(A), Processor Design Project(A-), Algorithm(A), Operating System(A), Digital System(A).

Experiences

Research Intern, KAUST - Simulator *Remote*
Mentor: Prof. Jian Weng *May 2025 – present*

- Contributed to **Assassyn** (accepted at ISCA 2025), an event-driven hardware design framework that simplifies RTL development by abstracting low-level timing and parallelism.
- Developed the framework's **DRAM subsystem** by integrating **Ramulator 2.0**, implementing a custom C++ wrapper for libramulator, and extending the Python code-generation layer to automatically emit correct Rust and Verilog DRAM modules with only **5** lines of user code.
- Authored **5,000+** lines of production-level code and validated the new memory model against Ramulator's test suite, strengthening cross-layer skills in system design and memory modeling.

Research Intern, UCSD PICASSO LAB - System for LLM *La Jolla, CA*
Mentor: Prof. Yufei Ding *March 2025 – present*

- Profiled and analyzed data movement in large-scale Mixture of Experts (MoE) LLMs (200B–671B) using 150GB+ of trace data from 24,000+ diverse inference requests. (Submitted to ASPLOS 2026).
- Derived six key system-level insights from temporal and spatial profiling to guide next-generation serving system designs.
- Implemented and validated architectural optimizations on wafer-scale GPUs, achieving up to 6.3× performance gains on DeepSeek V3 and 4.0× on Qwen3, while building an open-source profiling and simulation framework.

Research Assistant, Columbia University - Accelerator Simulator *New York, NY*
Mentor: Prof. Tanvir Ahmed Khan *Feb 2024 – Dec 2024*

- Developed a compiler-based simulation infrastructure using MLIR, achieving rapid simulation and detailed architectural insights for multi-accelerator systems under the mentorship of Prof. Tanvir Ahmed Khan.
- Designed scalable modeling techniques to overcome speed and fidelity limitations of existing data center accelerator simulators.

Teaching Assistant, Columbia University *New York, NY Sept 2023 – Dec 2024*

- Served as a Teaching Assistant for Computer Architecture, supporting the course through grading, exam preparation, and weekly office hours to guide students through complex architectural concepts.
- Identified and resolved students' technical issues by providing hands-on debugging, targeted guidance, and clear, concept-driven explanations to help them quickly get unstuck and deepen their understanding.

Publications

Orders in Chaos: Enhancing Large-Scale MoE LLM Serving with Data Movement Forecasting Oct 2025

Zhongkai Yu, Yue Guan, Zihao Yu, ***Chenyang Zhou***, Shuyi Pei, Yangwook Kang, Yufei Ding, Po-An Tsai

arXiv preprint arXiv:2510.05497, 2025

Projects

Operating System Application (Java) *nachos*

- Designed and implemented logics for Nachos operating system to support multiprogramming, enabling the execution of multiple threads concurrently.
- Implemented file system I/O interfaces in a multi-threaded scenario following documented semantics and specifications, to achieve processes' asynchronous file system access capabilities.
- Implemented UNIX-like exec, join, and exit commands for sub-process creation and orchestration.

DEET Debugger (Rust) *DEET*

- Implemented a fully functioning Rust debugger using ptrace, supporting breakpoints, register inspection, single-stepping, and ELF symbol parsing.
- Designed and built robust Rust abstractions for process control, error handling, and resource management using ownership, borrowing, lifetimes, and trait-based interfaces.
- Implemented and optimized custom systems-level data structures in Rust, demonstrating strong command of ownership, borrowing, and performance-oriented design.

Screaming bird (SystemVerilog/C) *screaming bird*

- Built a voice-controlled Flappy Bird system on the DE1-SoC, integrating microphone input, real-time signal processing, VGA graphics, and game logic into a full hardware/software design.
- Implemented a VGA graphics pipeline with timing generation, BRAM-backed sprite buffers, and a sprite-rendering FSM for smooth 640×480 display.
- Developed the C-based game engine, handling physics, collision detection, pipe generation, and hardware interaction via the Avalon memory-mapped interface.

Advanced Processor Architecture Design (SystemVerilog/C) *MIPS*

- Implemented and designed Victim Cache, Cache Set Dueling, Tournament Predictor and Out-of-order Execution in MIPS processor.

Awards and Honors

Provost Honors: 5 times in University of California San Diego.

Dean's List: 5 times in De Anza College (Cupertino, CA).

Technologies

Languages: C++, C, Java, Python, SQL, Rust, JavaScript, Verilog, SystemVerilog

Technologies: MLIR, LLVM, Linux, Git, Docker