Wonjong Peter Lee

+82)10-3021-0751 | wonjongbot@gmail.com | linkedin.com/in/wonjongbot | github.com/wonjongbot

Education

University of Illinois Urbana-Champaign — Champaign, IL
Bachelor of Science in Computer Engineering — *Aug. 2021 – May 2027*

- GPA: 3.92/4.0
- Dean's List: Fall 2021, Spring 2022, Spring 2023
- Relevant Coursework: Computer Architecture, Computer Systems Engineering, Digital Systems Laboratory, Analog Signal Processing, Data Structures, Linear Algebra, Discrete Mathematics
- Currently on a gap year to fulfill mandatory Korean military service.

Experience

Sergeant — Sep. 2023 – Mar. 2025
The Republic of Korea Army — *Paju, Korea*

Design Verification Intern — Mar. 2023 – Aug. 2023
Rivian Automotive, Inc. — *Champaign, IL*

- Designed a high-performance Python infrastructure to monitor DUT simulation and provide live performance evaluation; integrated a Grafana dashboard frontend with MongoDB.
- Overhauled an existing UVM testbench to simulate more robust and realistic workloads, improving test coverage.
- Enabled the tool to process over 1 million transactions/DUT port per simulation run, effectively identifying critical design bottlenecks early on.
- Collaborated with the Verification Lead to investigate abnormalities detected by the infrastructure, documenting issues via Jira for the design team.

Undergraduate Course Assistant — Jan 2023 – Aug. 2023
University of Illinois Urbana-Champaign — *Champaign, IL*

- Held weekly office hours to mentor and support roughly 150 students in ECE385: Digital Systems Laboratory.
- Guided students through lab assignments, including the design and implementation of a multicycle RISC processor datapath, a VGA text mode controller, and an SoC platform using the NIOS II soft CPU core.

Projects

tageBuilder - TAGE predictor simulation and evaluation tool | *Python* — Nov 2024 – Present

- Developed a Python-based simulator and performance evaluation tool for TAGE-like branch predictors to aid high performance CPU design decisions.
- Implemented a parameterized behavioral predictor model, enabling fast prototyping via user-defined YAML specs.
- Leveraged Numba JIT, Numpy and Pandas to process ∼700k branch/sec per process – achieving performance within 5x of Championship Branch Prediction 2016's C++ simulation suite (CBP-5) – all with the enhanced evaluation features.
- Enabled batch prediction and streamlined data processing with compatibility for 200+ BT9-formatted branch trace files, generating comprehensive metrics and visualizations(MPKI, accuracy, memory area, misprediction statistics, branch class analysis) with plots and CSV export.

Unix like Operating System | *C, QEMU* — April 2023 – May 2023

- Collaborated within a team of four to design and implement a UNIX-like operating system for an x86 uniprocessor system. Implemented as part of ECE 391: Computer Systems Programming.
- Developed key OS features including a read-only filesystem, multithreading, virtual memory management, interrupt handling, device drivers, and support for system calls (e.g., execute, read, write).
- Achieved 4th place overall in the course's OS design competition.

SoC Arcade Game Core on FPGA | *Systemverilog, C, Quartus* — Nov. 2023 – Dec. 2023

- Designed and implemented a NIOS II-based SoC on FPGA, optimized to run a 2D top-down multiplayer shooter game that integrates custom hardware with embedded game logic.
- Developed custom ROM modules for sprite storage, implemented palette-based graphics techniques to conserve FPGA area, and cached sin/cos values to accelerate sprite rotation calculations.
- Achieved efficient real-time rendering by "racing the beam," enabling full scene rendering with colorful sprites and animations without a dedicated framebuffer.

Technical Skills

Languages: Python, C/C++, Systemverilog (UVM), x86 ASM
Technologies:FPGA, Quartus, Modelsim
Libraries: pandas, NumPy, Matplotlib