Po-Hsun Lai

Taiwan cseslowpoke@gmail.com 0909 955 202 Po-Hsun Lai cseslowpoke

Education

National Yang Ming Chiao Tung University *Hsinchu, ROC*
M.S. in Computer Science, Advisor: Yi-Ping You. *GPA: 4.23/4.3* *Sept 2024 – Present*

- Focus on compiler optimization

National Taiwan Ocean University *Keelung, ROC*
B.S. in Computer Science *Sept 2020 – June 2024*

Awards

Silver Medal, ICPC Asia Taoyuan Regional Programming Contest *2022*
Achieved the Silver Medal, ranking 28th out of 100 teams, demonstrating strong problem-solving and algorithmic skills.
Bronze Medal, ICPC Asia Taipei Regional Programming Contest *2021*
3rd, TSMC IT CareerHack Digital Excellence X Generative AI *2024*
Developed AI-driven GCP log monitoring for automated scaling and real-time anomaly detection, enhancing rapid development, MVP design, and teamwork.

Projects

LLVM Project: Contributor *Aug 2024 - Present*

- Implemented NaN detection in NSAN within **compiler-rt**, enhancing runtime floating-point error detection for invalid computations.

Compiler Design: RISC-V Code Generation *2024*
Developed a RISC-V compiler from scratch, implementing the full compilation pipeline:

- Built a lexical analyzer using Flex and a syntax parser using Bison.
- Generated an Abstract Syntax Tree (AST) with semantic analysis and type checking.
- Compiled code into RISC-V assembly, supporting arrays and floating-point operations

CS 6120: Advanced Compiler *2024 - Present*
Implemented several compiler middle-end optimizations, including dead code elimination, local value numbering, and data-flow analysis...
Superscalar RISC-V Processor Design *2024*
Designed and implemented a two-wide superscalar in-order RISC-V processor, enhancing a baseline single-issue design to achieve dual-issue execution per cycle.

- Implemented control logic, instruction steering, and a scoreboard to manage data hazards and allocate functional units efficiently.
- Integrated instruction and data caches, including a 2KB direct-mapped design and an alternative 4KB 2-way set-associative cache.
- Employed write-allocate and LRU policies for cache management and implemented spill-before-fill for dirty block replacements.

Experience

NYCU CSIC30098 Compiler Design TA *Feb – Jun 2025*
NYCU CSCS10012 Introduction to Compiler Design TA *Sep 2024 – Jan 2025*

Technologies

Languages: C/C++, Python, Verilog
Software & Tools: Linux, Git, Make, CMake, GDB, Docker, Neovim, Clang/LLVM