Gaohong Liu

206.954.6826 | gaohol@uw.edu | github.com/cheimu

SUMMARY OF QUALIFICATION

- Proficient in Computer Architecture, Operating System, Networking System, Embedded Computing and VLSI System design: CPU and GPU architecture, Computation/Algorithm Accelerator Design, Basic Implementation of Unix-like Operating system, TCP/IP, Embedded System Software design, development, and test, Digital IC design.
- Programming Language: System Verilog, C/C++, Java, Python, Chisel, Scala, ML.
- Strong design and testing skill with demonstrated experience and background using Software such Quartus, Code Compose Studio, Cadence Virtuoso, Synopsis Compiler and Library, Spice, MATLAB, Mathematica and Hardware such as Xilinx and Intel FPGA, Arduino Development Board and Raspberry Pi Development Platform.

EDUCATION

University of Washington, Seattle, WA Cumulative GPA: 3.70 Expected Graduation Date: June 2019

Bachelor of Science in Electrical Engineering, VLSI & Embedded Systems Track (EE)

Bachelor of Science in Physics, Comprehensive Track (PHYS)

Emphasis: Computer Architecture, Digital VLSI Design, Embedded Computing, Operating System, Networking System.

RELEVANT EXPERIENCE

Research Assistant, Bespoke Silicon Group, University of Washington, Seattle WA March 2018 – August 2018

- Beginning with RISC-V ISA, fully investigated Architecture and Micro-architecture of Ariane version, Rocket Core, SiFive E54 and U51 of RISC-V and ran several bench marks such as Dhrystone and Rsort, etc. Successfully figured out the reasons which could influence the performance. (Chisel and System Verilog)
- Implemented a 4-Issue 7-Stage Out-Of-Order Superscalar RISC-V Microprocessor using customized Tomasulo Algorithm which is both performance and energy proper for Internet of Things. Currently working on testing to support Linux System.

Member, Computation Accelerator ASIC Design Team, University of Washington, Seattle WA January 2018 – June 2018

- Implemented Streaming Convolutional Neural Networking Accelerator on ASIC with completed I/O pads which can be used for image recognition in lower cost and smaller core area using SAED90nm Technology. (System Verilog)
- Used Bespoke Silicon Group open sources Hardware IP Cores and passed completely formal verification and source code can be loaded on Intel Altera FPGA to recognize streaming image edge at 60FPS.

Firmware Engineer, Sensor Systems Lab & WiBotic, Inc, University of Washington, Seattle WA June 2018 – August 2018

- Maintained firmware of Battery-Free Phone including updating firmware code to adapt new IDE and compiler and making Battery-Free Phone accept signal from USRP. (C)
- Modified and improved communication model and microwave transmission architecture which used on USRP to send signal such as message and audio file to Battery-Free Phone. (Python)

Member, CPU Design Team, University of Washington, Seattle, WA September 2017 – December 2017

- Implemented 64-bit 5-stage ARM pipeline CPU using System Verilog and ran on FPGA. (System Verilog)
- Designed 16-bit SAPR Register File and Brent-Kung Tree ALU layout manually using TSMC65nm technology so that critical path, efficiency, space utility and power consumption get optimized. (Verilog designed and Manually drew the layout)

Individual Project, System Programming Course, University of Washington, Seattle WA June 2018 – August 2018

- Successfully implemented a file search engine and web search engine using self-implemented, optimized doubly linked list, Hash Map, hash table, AVL Tree, and Binary Heap and Dijkstra Algorithm. (C/C++, JAVA)
- To speed up the file search engine, wrote code that takes an in-memory inverted index produced by file search engine and wrote it out to disk in an architecture-neutral format. (C/C++)
- Implemented a multithreaded Web Server which allows user following TCP/IP to use this search engine and fix security issue such as Cross-Site Scripting flaw and Directory Traversal Attack. (C++ and HTTP)

Member, Embedded System Development, University of Washington, Seattle WA March 2018 – June 2018

- Successfully developed a remote Human Body Physiological Indexes Measurement Equipment using Arduino ATMega2560 and Uno as developing platform. Wrote a ROTS Kernel on Arduino ATMega2560 to schedule and issue commands, manage memory, and handle communication. Used Arduino Uno as Peripheral System to measure temperature, diastolic pressure, systolic pressure, pulse rate, respiration rate and EKG. (C++)
- Implemented high fidelity communication between ATMega2560 and Uno allowing remote interaction using PUTTY.