ISHAN JOSHI

Computer Engineering Student
4th year - University of British Columbia

ishanjoshi23.github.io | joshi.ishan23@gmail.com
+1 (236)-889-2160 | github.com/ishanjoshi23
Vancouver, Canada | /in/-ishanjoshi

SKILLS

Languages: C, C++, Python, Java, SQL, asm (x86, ARM), HTML, CSS, Javascript
Hardware: Verilog/SystemVerilog, CMOS, Altera FPGA, STM32 MCUs, Adruino, Electronic Lab Equipment
Tools: Git, GDB, Unix, Modelsim & Quartus Waveform Simulation/Synthesis, Cadence (NC-Sim, RC Compiler, Virtuoso, Innovus), Docker, FreeRTOS, TCP/IP Networking

EDUCATION

9/2020 - 4/2025 **Computer Engineering - Bachelor of Applied Science** **University of British Columbia**
Relevant Coursework: Digital Design (100%), Microcomputers (95%), Operating Systems (95%), Algorithms (94%), AI (93%), Computer Architecture (93 %), VLSI Design (91%), Computer Communications (91%), Microcomputer Systems Design (current), Cybersecurity (current), Machine Learning (current)

EXPERIENCE

5/2022 – 12/2022 **Firmware Test Engineer Intern** **NETINT Technologies Inc, Burnaby, BC**
- Worked closely with 40+ architects, ASIC design, and firmware engineers to ensure the successful implementation and verification of firmware for several products. Used Jira for agile development.
- Developed and executed 1000+ Python/Shell test scripts and automation tests for design verification through Jenkins, encompassing unit, regression, compliance and HIL testing of the company's several video transcoder and SSD products.
- Gained understanding of 5+ video codecs and FFmpeg and Gstreamer video command-line tools.

9/2022 – 4/2023 **Teaching Assistant: CPEN 212 & CPEN 211** **Department of Electrical and Computer Engineering, UBC**
- CPEN 212: Facilitated project evaluations and conducted office hours for 160 students, providing guidance on computer architecture, hardware/software interfaces, and memory management.
- CPEN 211: Partnered with colleagues to offer structured office hours, addressing inquiries related to digital circuit and microprocessor hardware design, SystemVerilog, and ARM assembly coding for 330 students.

ENGINEERING STUDENT TEAMS

9/2021 – Present **Embedded Firmware Team Lead** **UBC Solar**
- Leading team of 8 for the development and validation of C firmware for STM32 microcontrollers in key vehicle systems, leveraging ADC, CAN bus, Interrupts, and HAL for our 3rd generation solar car.
- Enhanced system performance and multitasking efficiency by applying FreeRTOS architecture to suitable components including the Main Control Board and Telemetry Board.
- Developed a linux-based telemetry system *Sunlink* in Python to post data to InfluxDB using encrypted HTTP messages and stream live data to Grafana dashboards.
- Firmware: github.com/UBC-Solar/firmware_v3 Sunlink: github.com/UBC-Solar/sunlink

PROJECTS

SystemVerilog, DE1-SoC FPGA **Pipelined RISC-V Processor**
- Designed and implemented a pipelined datapath and finite state machine for a RISC-V processor.
- Enhanced functionality by supporting memory I/O operations with FPGA to fetch instructions and data.
- Achieved Turing completeness by implementing support for conditional execution.

SystemVerilog, DE1-SoC FPGA **Simple iPod on FPGA**
- Implemented Finite State Machines to read flash memory and PS2 keyboard input to play sound samples output from a given digital-to-analog converter and control song playback on an FPGA.

SystemVerilog, Cadence Innovus **Home Security System: Custom FSM + Place and Route**
- Designed and implemented a custom Finite State Machine modelling a home security system, executing the standard-cell layout using place and route techniques in Cadence Virtuoso and Innovus.

C, assembly **OS161 Operating System**
- Implemented parts of the OS161 operating system, including synchronization primitives, virtual memory management, and over 10 system calls such as sbrk, fork, execv, waitpid, read, write, dup2, and more.
- Achieved excellent understanding of the workings of a UNIX-like operating system and kernel processes.

C++, ChampSim Simulator **Cache Replacement Policies, Branch Prediction & Pipeline Friendly Code**
- Executed multiple cache replacement strategies (LRU, LIP, DIP, BIP, PLRU) to optimize hit rate.
- Implemented gselect and correlated branch predictors to explore branch prediction methodologies.
- Optimized matrix-multiplication code for improved CPI and cache hit rate in the ChampSim simulator

AWARDS & SCHOLARSHIPS

- UBC Dean's Honour List, 2021 & 2022
- MacKenzie Swan Memorial Scholarship, 2022
- Jim and Helen Hill Memorial Service Award, 2022
- Graduating Class of Electrical Engineering 1971 Service Award, 2022