

**Summary** — I combine research, industry, teaching experience, and hands-on projects to address complex problems in hardware architectural design, verification, CAD optimization, and FPGA applications for scalable digital systems.

## Education

**University of Toronto (UofT)**, Toronto, Canada

Masters in Applied Science (MAsc) in ECE [GPA: 3.8/4]

**Advisor:** Prof. Jason Anderson

September 2023 - Present

→ **Relevant Coursework:** ECE552 (Computer Arch.) || ECE1718 (Advanced Computer Arch.) || ECE1756 (Reconfigurable Computing & FPGA Arch.) || ECE1755 (Parallel Computer Arch. & Prog.) || ECE1387 (CAD for Digital Circuits Synthesis)

→ **Teaching Experience:** ECE1387: CAD for Digital Circuits [Fall'24] || ECE 532: Digital Systems Design [Winter'24] [Summer'24] [W'25] || CSC258: Computer Organization [F'24] || ECE 241: Digital Systems [F'23] [F'24]

**Bachelor's in Engineering (EE)**, VJTI, Mumbai, India [GPA: 9.13/10]

July 2019 - May 2022

## Publications

[1] H. Wei, O. Bhilare, H. Waqar, J.H. Anderson, "CAD Techniques for NoC-Connected Multi-CGRA Systems," *HEART '24* [link]

[2] O. Bhilare, et al., "DEEPFAKE CLI: Accelerated Deepfake Detection Using FPGAs," *PDCAT 2022* [link]

## Projects

**Determining and bridging Area gap between ASIC and CGRA** (Master's Thesis)

Aug 2024 - Present

→ Analyzing programmability costs in modern CGRA architectures by comparing Power, Performance, and Area results from diverse benchmark suites run on both ASIC and CGRA platforms

→ Conducting architectural exploration to identify trade-offs between programmability and performance in CGRA designs

→ Bridging the power gap in CGRA architectures by leveraging fracturable processing elements and analyzing their impact on Performance and Area (Work-in-progress)

**NoC-Connected Multi-CGRA Systems**

Jan 2024 - May 2024

→ Developed a partitioning-based CAD flow for NoC-connected large CGRAs, reducing run time by 33× compared to traditional CAD methods

→ Integrated and customized an open-source min-cut partitioner, TritonPart, to fit specific CGRA CAD requirements

→ Enhanced CAD flexibility with a multi-stage framework, enabling the use of advanced mapping algorithms alongside the partitioning stage, simplifying CAD exploration for large CGRAs

## Research Experience

**Processor Architecture Laboratory (LAP), EPFL, Switzerland**

Prof. Paolo Ienne, Mr. Andrea Guerrieri

Summer@EPFL

June 2022 - August 2022

→ Developed a framework to enable **Dynamatic** (a dynamic HLS compiler) to access a multi-cycle memory system via a custom AXI interconnect, integrated with Dynamatic's out-of-order load-store queues (LSQs)

→ Added burst support within Dynamatic's elastic environment using a custom Burst & Outstanding Manager (BOM) module

→ Enhanced memory accessing capabilities of Dynamatic with only a 2% increase in LUT and a 4% in FF usage

**Shakti Lab, RISE Group, IIT Madras, India**

Prof. V. Kamakoti, Mrs. Lavanya J

Research Verification Intern [REPORT]

March 2021 - July 2021

→ Designed and developed a framework for verifying SHAKTI, India's first indigenous RISC-V processor family, on FPGAs

→ To verify the processor, the framework produces random assembly tests, executes them on a softcore processor on an FPGA, and compares the execution signatures from FPGA with the golden signature from Spike, an ISA emulator

→ The framework enhanced verification speed, executing 500k RISC-V instructions in just 64 seconds on the FPGA softcore— a process that would need hours in conventional techniques— while preserving visibility and control within the flow

## Work Experience

**AMD**

**Bangalore, India**

Co-Op + Silicon Design Engineer I

Dec 2021 - July 2023

→ Contributed to SoC-level verification of Design for Debug (DFD) IPs in 3 AMD processors, aiding in successful tapeouts

→ Verified debug infrastructure in IPs, including USB, memory, and CPU cores, etc., using constrained-random test cases, ensuring functional correctness

→ Developed and ported a verification testsuite for DFD IPs across AMD processors, ensuring maximum test coverage

## Skills

**Language:** Verilog, VHDL, C, C++, Python

**Systems:** Git, Linux

**EDA Tools:** Quartus Prime, Xilinx Vivado, Microchip Libero, Yosys, Icarus Verilog, Synopsys Design Compiler, Cadence Innovus