Sheik Dhawood Ashfaq Asick Ali

Los Angeles, CA | +1 (213) 675-3715 | linkedin.com/sda00 | asickali@usc.edu

Education

University of Southern California, *Los Angeles,CA* May 2025
Masters of Science in Electrical and Computer Engineering, ***MS Honors Program*** **GPA 3.95**
Courses: *Digital System Design, Computer Systems Architecture, Systems for ML, MOS VLSI Circuit Design*

Amrita Vishwa Vidyapeetham, Coimbatore, *India* May 2021
Bachelor of Technology in Electrical and Electronics Engineering **GPA 8.74/10**
Courses: *Data Structures & Algorithms, Embedded Systems, DSP, PIC Microcontroller.*

Technical Skills

Computer Architecture: RTL design, GPU, ASIC, VLIW, SoC, Chip Multiprocessor, FPGA, DDR SDRAM, DFT, Virtual Machine, Object Oriented Programming, CDC, Cache Coherence, Intel SGX, UART
Programming: SystemVerilog, Verilog, VHDL, C++, Python, CUDA, C, Assembly Language
Tools: Questasim, Xilinx Vivado, Iverilog, VCS, Genus, Innovus, Quantus, Joules, Conformal, Gem5, Intel's PIN API, Cadence Virtuoso, HSPICE, MATLAB, PROTEUS, GitHub, Arduino, Unix Shell, Xcode, VS Code, Bazel
Libraries: TensorFlow, PyTorch, Pthreads, NumPy, Matplotlib

Academic Projects

Smith-Waterman Accelerator for DNA Sequencing | *SystemVerilog, SVA, Tape-out project* Dec 2024
- Designed and implemented a modular 10x10 PE array, leveraging a **pipelined design** reducing computational complexity from O(MxN) to O(M+N-1), resulting in a **500x speedup** for 1,000-length sequences.
- **Led a team** of 5 as the RTL Lead, directing RTL design and overseeing verification using SystemVerilog methodologies, synthesis, placement, and routing for Intel's 16nm tape-out.
- Demonstrated exponential scalability for industry-standard sequence lengths.

PCIe Physical Layer Implementation | *PCIe protocol, Verilog, RTL design, Chipscope* Jul 2024
- **Coded** PCIe physical layer receiver module for both upstream and downstream in Verilog, including 8b/10b encoder/decoder, elastic buffer, and deskew FIFO.
- Utilized ILA chipscope for comprehensive signal analysis, including skew measurement and elastic buffer behavior.
- Validated receiver functionality through simulation, ensuring accurate data recovery.

Tomasulo 3-based Out-of-Order MIPS CPU | *OoO execution, Tomasulo algorithm, VHDL* Jun 2024
- **Engineered** high-performance out-of-order CPU leveraging Tomasulo algorithm for dynamic instruction scheduling. Supported core set of MIPS instructions.
- Created critical pipeline components - **Branch prediction buffer,** Store address buffer, **Copy-free checkpoint,** Register Order Buffer, Dispatch Unit, and Issue Logic. Ensured efficient instruction flow and handling dependency.
- Validated CPU's functionality using multiple instruction streams. Verified correctness of OoO execution.

AXI4 Multi-Initiator/Responder Bus Interface | *AXI protocol, Bus interface development, Verilog* Jun 2024
- **Developed** AXI4-compliant mesh network interconnect for multi-processor SoC in Verilog.
- Orchestrated OoO packet delivery for improved performance while ensuring in-order completion of requests.
- Executed expertise in AXI4 protocol, mesh network design, and RTL design for SoC communication.

Optimizing RISC-V CPU Performance: A Gem5 Simulation Study | *Gem5 tool, C++* Feb 2024
- **Spearheaded** a project using the Gem5 simulation tool to enhance a RISC-V CPU, achieving average speedup across four benchmarks of **2.13x**, all within **50%** of given area constraint.

Memory-Efficient Fine-tuning of LLaMA 2 on Single-Core A100 GPU | *VS Code, GitHub* Feb 2024
- Focused on instruction tuning to adapt pre-trained LLaMA2 7B model. Exploring efficient fine-tuning techniques such as gradient accumulation, checkpointing, mixed precision training, and low-rank adaptation to refine LLMs in resource-constrained environments, particularly addressing memory usage concerns.
- Resulted in a **2x speedup** in training time and consuming less than **41% memory**, enabling effective adaptation of model in resource-constrained environments.

Performance Analysis of Branch Predictors: A Study Using Intel's PIN API | *PIN tool, C++* Jan 2024
- Conducted a comprehensive performance analysis of four different branch predictor algorithms with Intel's PIN API and assembly programming, providing valuable insights into predictor's efficiency.
- Coded in C++ to simulate behavior of each branch predictor, is then fed as input to PIN tool, convincingly exhibiting accuracy of each predictor.

Experience

Tata Consultancy Services | *System Engineer, India* June 2021 – June 2023
- Exercised strong problem-solving and analytical skills to manage and resolve complex technical issues of end users across CITI's production applications, **reducing average time for resolution by 25%.**
- Ensured smooth operation of production applications by deploying new software, upgrades, and application code to remediate known defects, with a **0% downtime record.**
- **Boosted customer satisfaction by 10%** by collaborating with stakeholders to provide enhancements and weekly patches across all environments, cutting down performance issues related customer support tickets by **20%.**