Harshith Kantamneni

kantamneniharshith@gmail.com | +1 (414) 916-5799 | linkedin.com/in/hk4231 | Madison, WI 53726

M.S. ECE student at UW–Madison specializing in Computer Architecture, RTL Design, and High-Performance SoC Systems

Skills

- **Languages:** Verilog, SystemVerilog, C, C++, Python
- **Tools:** ModelSim, Quartus, gem5, McPAT, MATLAB, Simulink, Git
- **Domains:** RTL Design, Microarchitecture, SoC Performance Modeling, FPGA Prototyping, Fault-Tolerant Computing
- **Protocols:** I2C, SPI, UART, CAN, WiFi, Bluetooth

Education

University of Wisconsin-Madison — *Madison, USA*
M.S. in Electrical and Computer Engineering — Sep 2024 – Dec 2025

- Relevant Coursework: Computer Architecture, Advanced Computer Architecture II, High-Performance Computing, Digital System Design, Fault-Tolerant Computing, Machine Learning

Vellore Institute of Technology — *Amaravati, India*
B.Tech in Electronics and Communication Engineering — Nov 2020 – May 2024

Certifications

- **NVIDIA Deep Learning Institute (DLI) – Getting Started with Accelerated Computing using CUDA C++**, 2025

Hardware Design Projects

Architectural Performance Modeling using gem5 — Sep 2025 – Present
gem5, Python, McPAT

- Profiled and compared **TimingSimple**, **Minor**, and **O3 CPU** architectures in gem5 to evaluate performance (CPI), latency, and energy trends.
- Executed and analyzed vector arithmetic workloads including **IAXPY, SAXPY, and DAXPY** to study instruction mix, cache/memory behavior, and sensitivity to hierarchy changes.
- Automated simulation, metric extraction, and power estimation with Python + McPAT to enable scalable SoC-level exploration.
- Mapped architectural observations to RTL-level trade-offs to inform pipeline and control-path design decisions.

5-Stage Pipelined RISC CPU (WISC-F24) — Aug 2024 – Dec 2024
Verilog, ModelSim

- Implemented a pipelined CPU with hazard detection, forwarding, and branch handling logic.
- Verified functionality with a cycle-accurate testbench achieving **100% instruction coverage**; waveform-driven debug for stall/flush corner cases.
- Applied modular RTL methodologies, parameterized components, and synthesis-friendly coding practices.

FSM-Based Knight's Tour Solver (FPGA Accelerator) — Aug 2024 – Dec 2024
SystemVerilog, Quartus, ModelSim

- Designed a pipelined FSM accelerator achieving **333 MHz timing closure** on FPGA.
- Integrated UART/SPI interfaces for host control and real-time command handling.
- Performed **post-synthesis gate-level verification** to ensure timing and functional correctness.