Deekshith Anantha

Raleigh, NC | +1(984) 382-1068 | ananthadeekshith@gmail.com | linkedin.com/in/deekshith-anantha | Portfolio

EDUCATION

North Carolina State University, Raleigh — **Aug 2025 - May 2027**
Master of Science (MS), Computer Engineering — *4.00/4.00*
- **Coursework:** Microprocessor Architecture, Architecture of Parallel Computers, Advanced Microarchitecture, Embedded Systems Architecture, Object-Oriented Design and Development, Neural Networks, Operating System

RV College of Engineering, Banglore India — **Aug 2019 - Jun 2023**
Bachelor of Engineering, Electronics and Telecommunication — *7.9/10*

TECHNICAL SKILLS

- **Programming Languages**: C, C++, Ruby, Python, Robot Framework, Shell Scripting, Perl, Linux/Unix
- **Tools/Frameworks**: Lauterbach Trace32, GDB, Astree/polyspace, Keil, Linux Kernel, MATLAB, Simulink, Git, Jira, Jenkins
- **Methodologies**: Agile Methodology, Component based development, Requirement Engineering, Requirement Based Testing
- **Embedded & Real-Time Systems**: Bare-Metal Programming, RTOS, Microcontrollers, Firmware Architecture, Low-Level Drivers, Memory Management, Communication buses
- **Concepts**: Computer Architecture (GPU/CPU), Operating Systems, RISC-V, Object Oriented Programming, Cache Memories, Virtual memory, Cache Coherency, GPU programming, WCET, Signal processing, Data Pipelining, DSP, Data Structures and Algorithms

WORK EXPERIENCE

AIRBUS

Embedded Software Engineer — *Banglore India* · **Jul 2023 - Jul 2025**
- Developed and optimized firmware for the **Flight Warning Computer** (A320 aircraft) using **C/C++** and RTOS, demonstrating deep understanding of **hardware-software integration** and embedded system performance.
- Conducted **requirement-based testing** and **real-time hardware debugging** using Lauterbach **Trace32** and **GDB**, mirroring **post-silicon validation** and hardware bring-up processes.
- Collaborated with cross-functional hardware and verification teams to ensure **real-time system reliability and SoC-level data communication consistency.**
- Engineered and maintained **CI/CD pipelines** in Jenkins, improving build and deployment efficiency by **50%** through automated testing and system-level integration.
- Reinforced experience in **embedded systems**, **testing automation**, and **system-level debugging**.

Engineering Intern — *Banglore India* · **Mar 2023 - Jul 2023**
- Designed automation scripts using **Python** to streamline **firmware configuration and verification** under **DO-178C** guidelines.
- Developed tools to improve Change and Configuration Management processes, aligning with **system-level validation** workflows.
- Automated migration of **34** software components to the latest toolchain using **Python** and Shell scripting, reducing migration time by **80%**.
- Enhanced cockpit simulation by optimizing over **15,000** signals using Python profiling, leading to a **15%** boost in computational performance and reduced system latency.

PROJECTS

Cache and Memory Hierarchy Simulator with Stream Buffer Prefetching (C++)
- Developed a configurable simulator for multi-level cache hierarchies (L1, L2) supporting adjustable size, associativity, and block parameters.
- Implemented a **stream buffer prefetching unit** to model speculative data access and analyzed performance metrics such as hit/miss rates and average access time. Evaluated system-level trade-offs in **latency, bandwidth, and energy** using CACTI models, reinforcing concepts in **performance modeling and firmware-level optimization**.

Superscalar Pipeline Simulator (C++)
- Developed a cycle-accurate simulator for an **out-of-order** (OOO) superscalar processor to model **dynamic instruction scheduling.**
- **Implemented core microarchitectural components essential for OOO execution**, including a Reorder Buffer (ROB), a unified Issue Queue (IQ), and a Rename Map Table (RMT). **Modeled the complete multi-stage pipeline** (Fetch, Decode, Rename, Register Read, Dispatch, Issue, Execute, Writeback, and Retire) to resolve true data dependencies and structural hazards.
- **Analyzed processor performance by processing instruction trace files** to generate detailed cycle-by-cycle timing reports and calculate final metrics like Instructions Per Cycle (IPC).

Branch Predictor Simulator (C++)
- Designed and implemented a **RISC-V branch prediction simulator** in C++ supporting **bimodal**, **gshare**, and **hybrid predictor** architectures.
- Modeled dynamic two-bit counter tables and chooser logic to measure **misprediction rates**, accuracy, and pipeline efficiency across configurations.- Optimized simulator runtime and validated against benchmark traces to study trade-offs in **accuracy, hardware complexity, and pipeline performance**.

Adaptive Modulation and Detection using Machine Learning
- **Designed a machine learning model** to perform automatic modulation classification (AMC) on signals using adaptive modulation techniques.
- **Developed and trained a deep neural network (DNN)** to predict the modulation scheme (e.g., QPSK, 16-QAM, 64-QAM) of a received signal, overcoming the errors and quality loss of traditional demodulation.
- **Utilized Python, MATLAB, and Jupyter Notebook** to pre-process signal datasets, build the ML architecture, and train the model to minimize loss.
- **Evaluated model performance** by comparing its classification accuracy against other established architectures, including CNN, LSTM, and PCA.