Kalyan Sriram

Final year undergraduate student in computer engineering and computer science with a passion for computer architecture. Deeply interested in processor microarchitecture, digital logic, compilers and programming language theory, and distributed systems.

kgsriram@wisc.edu
+1 (925) 523-8956
linkedin.com/in/kalyangsriram
github.com/coderkalyan

EDUCATION & NOTABLE COURSEWORK

University of Wisconsin–Madison (2022–2026)
BS, Computer Engineering and BS, Computer Science with Honors (GPA: 3.87)

- CS 759: (Graduate, Spring) HPC & GPU Programming; CS 577: Algorithms; CS 537: Operating Systems; CS 536: Compilers
- ECE 757: (Graduate) Computer Architecture; ECE 551: Digital Design & Verilog; ECE 340: Electronic Circuits
- CS 532: Matrix Methods for AI/ML; Math 340: Linear Algebra & Differential Equations; CS 564: Databases
- **Summer Coursework (UC Berkeley)**: CS 161: Computer Security; CS 61C: Great Ideas in Computer Architecture

RESEARCH & TEACHING EXPERIENCE

Summer@EPFL Research Fellow, École Polytechnique Fédérale de Lausanne (Summer 2025): Lausanne, Switzerland

- Joined the Processor Architecture Lab (Dr. Paolo Ienne) through the selective (1.3%) Summer@EPFL research fellowship.
- Studied novel reconfigurable fabrics (CGRAs) on circuits generated by high level synthesis compared to traditional FPGAs.
- Built software to verify functional correctness and analyze performance of custom reconfigurable fabrics (FPGAs, CGRAs).

Teaching Assistant, University of Wisconsin–Madison (Spring 2025 - Current): Madison, WI

- Teaching the upper level CS 552: Computer Architecture for past two semesters to a group of 100 undergraduate students.
- Modernizing the syllabus around RISC-V and developing new course material (lecture material, exercises, projects).
- Revamping the course project with automatic grading, better documentation and testing, and FPGAs/synthesis.

Senior Research Thesis, University of Wisconsin–Madison (Fall 2025 - Current): Madison, WI

- Research involves characterization and mitigation of timing side channels in Networks on Chip (NoCs).
- Exploring microarchitectural techniques (speculation, path diversity) to improve performance/security tradeoff in NoCs.

Research Assistant, Embedded Systems Lab, University of Wisconsin–Madison (Fall 2023): Madison, WI

- Developed/optimized a wireless circuit board for monitoring heat stress in dairy cattle using LoRa, RFID, and STM32.

Presenter, PX4 Developer Summit (2023): New Orleans, LA
UAV Communication: Exploring Protocols, Innovations, & Best Practices

- Presented on the use of CAN, Ethernet, and UART links between avionics with DroneCAN, Cyphal, MAVLink, and ROS DDS.

WORK EXPERIENCE

Software Engineering Intern, Vehicle Avionics Network, SpaceX Starlink (Fall 2025 - Current): Redmond, WA

- Work spans across telemetry infrastructure, embedded systems software, networking, and in-space fault detection/recovery.
- Focusing on improvements to satellite telemetry with the goal of reducing telemetry bandwidth through compression.
- Gained experience in code review by reviewing changes to C++ flight software, catching errors in correctness and performance.

Software Engineering Intern, Satellite Software, SpaceX Starlink (Summer 2024): Redmond, WA

- Profiled (perf), analyzed (assembly), and implemented targeted improvements to satellite flight software performance.
- Designed a system for granular software updates to improve satellite update time and reduce network traffic to space (>50%).
- Developed and optimized code in C++ for Linux, Xilinx Zynq (ARM64 + FPGA) processors, and STM32 microcontrollers.

INDEPENDENT HARDWARE & SOFTWARE PROJECTS

- Warp: RISC-V (RV64GCB) in-order, dual-issue core in Verilog with full pipelining and branch prediction. Working on pipeline optimizations for throughput improvement, coherent cache, and vector unit.
- Boa: High performance interpreter/JIT targeting a small subset of Python 3. Significantly outperforms CPython 3.12 (5x) on arithmetic and control-flow heavy benchmarks. Working on integrating and testing a tiered JIT and expanding syntax support.
- Femto: A compiler for a systems programming language, written in Zig. Includes a hand-written lexical analyzer and syntax parser, intermediate representation for semantic and type analysis, and a backend using LLVM.
- PicoFusion: Compact AHRS module combining a microcontroller running PX4 Autopilot with a 9DOF inertial measurement unit on an embeddable PCB, enabling high precision orientation estimates, built-in calibration, and Kalman Filter estimation.

SKILLS & TOOLS

- **Systems programming**: C/C++, Rust, Java, Python, x86/ARM/RISC-V assembly, Zig, Go
- **VLSI**: Verilog/SystemVerilog, Chisel, formal verification, VPR place and route, Yosys synthesis suite, high level synthesis
- **Circuit design**: PCB design and fabrication (KiCAD), THT & SMT soldering (hand & reflow)
- **Embedded**: Linux, IP networking, STM32, Zephyr & NuttX RTOS, Proxmox/KVM/QEMU hypervisor, Docker