DIR-V Symposium Hackathon

Track: Systems and Software

Objective:

To empower participants to explore RISC-V software and system-level solutions. The hackathon aims to foster problem-solving in performance optimization and systems software development.

How: Hands-on Online

Participants will

- Gain hands-on experience with RISC-V tools (Spike, Verilator, RV32/64 gcc).
- Understand performance and code-size trade-offs in embedded systems.
- Learn advanced compiler optimizations, pragma usage, and extended assembly integration.

Who can Participate:

Students, researchers, developers, and open-source hardware enthusiasts passionate about RISC-V technology.

Problem Statement:

Participants will optimize and benchmark the CoreMark (bare-metal) source code using RV32/64 gcc build tools, Shakti Verilator, and Spike simulator.

- Simulate with Spike to obtain golden outputs and measure base cycles on Verilator for a specific iteration count (e.g., 100).
- Develop two optimized binaries:
  - **Performance**: Maximize CoreMark/MHz (CM/MHz).
  - **Code Size**: Minimize static code size
- The participant with the highest performance (CM/MHz) and least code size will be adjudged the winner
- Participants can use any compiler flags, any pragma & attributes in source code. They can also do source code modifications in C and use extended assembly (in the coremark source code).
- They should run the optimized binaries on spike and prove that they are functionally equivalent to original code.
- For performance (CM/MHz) they need to run on Verilator to measure cycles and compare to the base numbers.
- They have to submit a detailed report on the specific optimizations applied; and the percentage cycle or code size improvements achieved at each step