

CF-RV-25-13

PROBLEM STATEMENT:

C-Class core/debug RTL implementation and simulation: Implement atleast one instruction trigger (hardware breakpoint **mcontrol6**) in C-Class. Evaluate with manual 3-window simulation with simulator, openocd and gdb. Also, test trigger programs from risc-v debug tests (setup provided by Shakti team). Extra credit: Test on FPGA.

GENERAL FLOW:

- 1) Getting familiarized with concepts(instruction triggers, working of 3 window simulation setup,C-class core).
- 2) Studying RISC V debug module version 1.0 to understand the mcontrol6 trigger config.
- 3) Understanding the existing debug logic.
- 4) Implementation of mcontrol6 by making changes with the respective modules from the C-class core. (before proceeding, we will be confirming about these modification points through the discord group)
- 5) After implementation, the 3 window simulation will be set up to verify the working.
 - i) Simulator - Verilator - to simulate the modified core
 - ii) Openocd - bridges verilator and GDB
 - iii) GDB - it controls the mcontrol6 trigger config.

As of now, we have planned to modify the **stage1.bsv** file in the C-class core repo. We intend to modify the file by adding the **trigger check logic** inside the check_trigger() function using trigger configuration signals (tdata1, tdata2) that represent mcontrol6 format. Following that, we will add a logic to decode the tdata1(mcontrol6 format) into its fields (such as select, match..) and send it to the check_trigger() logic to raise a trap when pc or instruction matches tdata2 based on select, match values. (we have to figure out where to apply this decoding logic yet).

3-window simulation: The trigger configuration data will be written via GDB using OpenOCD then we simulate the modified RTL using Verilator. We run the program, configure the trigger with GDB, and verify if the core halts at the correct breakpoint address.