Feb 16, 2023 | RISC-V Control Transfer History TG Meeting

Attendees: tech.meetings@riscv.org Beeman Strong Bruce Ableidinger

Notes

- **Attendees**: Beeman, JohnS, StasN, Snehasish, BruceA, DavidW, AtishP
- **Slides/video** here
- Considering tradeoffs between CSR and MMIO interfaces
- Latency
  - CSRs can be pretty fast, but indirect access will limit ability to take advantage of speculative/OOO execution
  - MMIO accesses typically order of magnitude slower, but CTR-specific optimizations exist that could make it competitive
  - MMIO could also be read by DMA, but may have conflicts
  - Ved: CTR typically implemented deep in the CPU core, would be hard to implement a wider ifc
    - But could build CTR within Trace Encoder, outside the core
    - Trace is usually not context switched, only the buffer/pointer. But CTR is switched, so benefit to being in the CPU.
- External Access
  - Not sure how important debugger access to CTR without halting is
    - Robert: reading PC out of CPU without halting was used heavily, can read it quickly. Up to 10K samples per sec for a few regs. Gets more complicated with >1 CPUs.
  - Implementation could opt to make CSR-based CTR state accessible through JTAG
  - OOB profiling is an interesting usage, but needs lots more infrastructure. Probably warrants a TG, many unknowns, hard to target
    - Ved: OOB profiling agent would need satp and hgatp for context
      - Agreed, need more state to be valuable
- Trace encoder integration
  - TE has all info needed for required CTR fields
  - TE doesn't have mispredict indication today, but easy to add
- Supporting >1 transfers per cycle, can be challenging to support, for both trace and CTR
- See any difference between call-stack mode and default mode for this discussion?
  - Not really, save for context switch latency implications
- DavidW: CTR info could be a source for side-channel if leaked. MMIO seems potentially more vulnerable.
- Robert: implementation cost is trivial if building on top of trace. Don't think impact to Linux perf of allowing either option is so bad.
- Ved: if a CPU has built the ingress port, could have CPU logic near the port that could capture CTR. So could be built on either side of that ingress.
- Stas: if including perf counters in CTR records, how does that factor in?

    - Counters are in CPU, would have to expand TE ingress to get them
- Ved: Confidential Compute or a partitioned system, easier to partition a CSR than MMIO when one hart is in confidential mode
    - Same issue for trace
- Agreement that CSR easier for accessing state inside the CPU
- **Out of time. Will take up offline and aim to resolve by next meeting.**

Action items

- □