Oct 24, 2024 | 🗓 RV Performance Event Sampling TG

Attendees: Beeman Strong   Snehasish Kumar   tech.meetings@riscv.org

Notes
- **Attendees**: Beeman, Bruce, Snehasish, Chun
- **Slides/video** here
- Recap freeze discussion:
    - Freeze on overflow preferred with precise attribution (Sspesa), freeze on interrupt preferred otherwise (using epc for attribution)
        - There are some other usages, like RR, that always prefer freeze on overflow
    - Counter freeze value:
        - Avoid counting events in handler, for kernel sampling (where priv mode filtering isn't sufficient)
        - Easy way to get precise attribution for arch events, but not uarch events
        - So moderate utility
- For counter freeze, propose that:
    - Can select whether freeze is on interrupt or overflow
    - Can select which counters freeze on a per-counter basis. Select a group of counters where all freeze if any in the group get interrupt/overflow
    - So one group of counters involved in freeze
- Counter freeze is not on by default, only enabled when extension is implemented and user opts in by selecting some counters to freeze
- CTR freeze
    - Same considerations apply (prefer overflow freeze with precise attribution, otherwise interrupt freeze)
    - LCOFIFRZ is defined to freeze on interrupt (since no precise attribution today)
    - Propose adding OFFRZ bit to support freeze on overflow
- Sspesa access
    - shpms* accessible to S-mode when counter delegation enabled, and then only delegated counters update them
        - So M-mode can use Sspesa when CDE=0
    - VS-mode requires SBI to access, just like counters
- Does counter/CTR freeze happen for non-delegated counters, when CDE=1?
    - Otherwise differs from Sspesa
    - *Will discuss next time*
- Do multiple overflows really happen?
    - We don't know of any real usages, save for multi-user (e.g., hypervisor and guest both profiling in guest)
    - Opted not to add HW (more shpms* CSRs) to support a rare occurrence (multiple overflows in small window) in a rare usage (multiple sampling users)
- Output to memory for Sspesa

- ○ Doable, but would require more ISA (counter reload mechanism, OF clearing, etc). Propose not adding that now.
- ○ Could allow reload only on subset of counters
  - ■ Agreed. ISA would probably specify reload for all, but an implementation could support it only for a subset.
  - ■ We haven't discussed whether Ssplcofi or Sspesa require precise sampling/attribution for all Zihpm counters
    - ● Each counter adds incremental cost. E.g., bit(s) in the ROB
    - ● Lean towards letting implementations make value judgment on added cost vs the consistency that SW prefers
  - ■ Haven't we already added HW backoffs with Ssplcofi vs Sspesa
    - ● Yes, but those are different HW tradeoffs
- ○ If known usages have only 1 counter sampling, maybe we only need precise sampling/attribution support for 1 counter?
  - ■ *Interesting question, let's discuss more next time*
- ● SiFive has PC sampling in trace. Could leverage shpms* HW?
- ● **Out of time**, continue this discussion next time

Action items
☐