Jul 11, 2024 | RV Perf Analysis SIG

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

Notes

- **Attendees**: Beeman, BruceA, Robert, Snehasish, JimC, MattT, JohnS, Ruinland, Atish, PaulS
- **Slides/video** here
- **Bruce presenting SiFive's PIPES**
    - I'm only capturing the discussion here, see the slides for the presented content
- Does it count number of events while the tagged inst is in the pipeline, or the number of events the tagged inst incurs?
    - The latter, e.g., num insts retired during the tagged inst's life
- PIPES units can be shared by harts within a core, will include hart ID
- Does random inst selection change fetch order?
    - No, it randomly selects an inst from within the fetch bundle
- Sampling based on cycle counter, instret counter, or new programmable counters
    - The latter are like HPM counters, but a separate
- When filtering samples, simply throw away samples that don't match
- Collects TMA information associated with the sampled inst
    - Busy means waiting to be oldest, or still executing (but not stalled)
    - Backend means stalled in backend (e.g., load awaiting data), or stalled in FE due to backend (e.g., ROB full)
    - INSTRET counts instructions that retire during the tagged inst's life
- Counters count down when no inst tagged, tag an inst (really fetch) once overflowed, then switch to count up, counting events during the inst's life
- Tag a fetch line, select an inst in it
    - With 64B cacheline means need to select at random from any of up to 32 insts in the line
    - Allows capture of ITLB miss or I$ miss latency in FE stalled counter
- So have 5 fixed counters + any number of HPM counter analogs for sampling
- Much more elaborate sampling mechanism than SPE or IBS for how samples are chosen, why?
    - Wanted to be able to randomly sample (e.g., use cycles), or match PEBS for event sampling
        - But PEBS samples an inst that incurs the event, this will choose a random inst that comes after the event
- Could we just use HPM counters?
    - No way to reload them
    - Don't want to have to share them
- Why include mvendorid, mimpid, etc, in packet header? Will output be dumped into a buffer shared by multiple vendors?
    - May be optional to include that stuff in implementation

- Just one header before samples, assumes PIPES is not used in a circular buffer
- Can collect call-stack by using interrupt mechanism
- Are counters accessible by SW or a debugger?
    - Yes
- Is there priv mode filtering?
    - Yes, in spec, not in slides
- Main usage is to trace, don't need any interrupts
    - But also have interrupt mode
- Need some mechanism to stop sample selection once interrupt is pended, so SW can collect the sample before it is overwritten
- Can PIPES be used in parallel with instruction trace?
    - Yes
    - Won't that tax the decoder to handle worst case of PIPES packet + some large IT packet simultaneously?
        - May have to have prioritization

Action items

- ☐ Beeman Strong - Jul 28, 2022 - Reach out about proprietary performance analysis tools
- ☐ Beeman Strong - Jul 28, 2022 - Reach out to VMware about PMU enabling