Sep 8, 2022 | RISC-V Perf Analysis SIG Meeting

Attendees: Beeman Strong tech.meetings@riscv.org Marc Casas

Notes

- **Attendees**: Beeman, Marc, Robert C, John S, Jeff S, Philipp, Jojo, Bruce, Ved, Daniel, Chris T, Anup, Aaron
- **Slides/video** here
    - I started the video late, missed the slide review of x86/ARM/Power solutions
- Branch history used to enrich instruction profiles, heavily used for PGO (like autofdo and BOLT)
- Existing architectures have very similar solutions: x86 LBRs, AMR BRBEs, Power9 BHRB
    - Reviewed the details of these, see slides
- Philipp: Power tries to have read out not have impact, and be fast
- Beeman: could mimic existing architectures, or consider leveraging trace
    - Trace-based approach would be to route E/N-trace to a local/internal buffer only (so no overhead), then extract it only upon collecting a sample
    - Pros: use existing trace HW, smaller storage per branch thanks to compression
    - Cons: requires decode (with binary images), new enabling since fundamentally different from existing approaches
- Robert: Don't have to reinvent the wheel, maybe just use trace to shuffle history out of the SoC
    - But for PGO would want to keep it in the SoC/CPU for in-target use
- Robert: Trace can save storage, but either way storage is small so not much advantage
- Ved: need to context switch branch history records, so need to make that fast
- All agree that this gap needs to be filled, and that we should do something similar to what other architectures do
- Bruce: SiFive implementing Path Profiler, very similar to LBR but memory mapped. Donating to RVI
    - Aiming for autofdo, perf use
    - Want fast decoding using branch addresses, unlike compressed trace
    - Includes a sampling mechanism with some randomness, based on cycles, captured automatically
        - Can push PP contents to memory via N-trace (to memory), or read it out explicitly via SW
        - Not intermingled with N-trace, but not how they did it
    - Branch type filtering, cycle counts, mispredict bit
    - 2 MMIO regs, source pc and target pc, with option for up to 2 "info" regs
    - Also have call-stack mode, enabling flame graphs without full call-stack walk
    - PP clears the buffer once it's read out, in auto-export mode
- Beeman: don't think autofdo uses metadata today, just PCs
    - Ved: cycle time info can be used to time loops

    - Beeman: seen some use of LBR cycle time for anomaly detection, but nothing automated
- Beeman: **propose we start a TG** to work out a spec for branch history. May opt to start with SiFive proposal, can decide that later.
    - **Will need acting chairs**, please consider helping
- Atish: do we need separate TGs for each perf related features?
    - May be able to combine some, but branch history will likely stand alone

Action items

- [ ] Beeman Strong - Sep 8, 2022 - work on TG for branch history
- [ ] Atish Kumar Patra - Aug 25, 2022 - check on how to read multiple counters in perf when taking an interrupt on one
- [ ] Beeman Strong - Jul 28, 2022 - Reach out about proprietary performance analysis tools
- [ ] Beeman Strong - Jul 28, 2022 - Reach out to VMware about PMU enabling
- [ ] Beeman Strong - Jul 28, 2022 - Talk to security HC about counter delegation