Oct 10, 2024 | 📅 RV Performance Events TG

Attendees: Beeman Strong   Dmitriy Ryabtsev   tech.meetings@riscv.org

Notes
- **Attendees**: Beeman, Chun, Daniel, Matt, Snehasish, Bruce, Greg, CayB
- **Slides/video** here
- Reviewed missed events summarized in required/recommended events (riscv.org), see details in Required Perf Events - Google Sheets
- Have L[123] events, but not LLC currently
    - LLC is confusing bc which level is LLC varies by implementation
    - But any implementation could alias LLC events to the analogous L2/L3 events
    - LLC events used as a proxy for BW
    - Perf tool cache-misses aliases to LLC misses
    - LLC might be outside of core, shared. We're only defining core events.
    - Core may be able to count local accesses/hits/misses to LLC, if have some data source info
    - Don't require LLC, but recommended
- Add event for all cache accesses/misses, currently just load/store
- No value seen for programmable event to cout bus cycles or ref cycles
    - Can always read time for wall-clock time
- Add single event to count all predicted branches/jumps, and spec control flow events
- Do we have prefetch events?
    - Yes, from the perspective of the cache access by a prefetch, not from the prefetcher source
- Should have code fetch counts for L2/L3
- What do L2/L3 cache events mean?
    - As defined, count instructions that access/hit/miss caches
    - Filed https://github.com/riscv/riscv-performance-events/issues/7, which suggests also counting all accesses
    - Stores don't write to outer level cache, so what does CACHE.L3.STORE mean?
        - Means data was found in L3, though will fill L1 and write there
    - Does UC access count?
        - Would still snoop the L2/L3
        - No, those would fall in SNOOP events, not LOAD/STORE
    - Make sure all this is clear in event definitions
- How are WC stores counted?
    - Need STORE.MERGE event?
- CTRUPD.RET: Autofdo uses branch retired near taken, no LBR filtering. So not needed now, but could be in the future.
    - Probably good to add it
- Include I-side retire events (for CACHE, TLB, etc)
    - Complex but useful, ala Intel's FRONTEND_RETIRED

- ○ Not required
- ○ Partially addressed in the linked issue above
- Add HITM events that are non-spec, for attribution
  - ○ Google uses such events
- Do we have ITLB hit and miss?
  - ○ We have ACCESS and MISS, seems sufficient
- Need to look at which user insts/extensions need events, for assessing instruction mix and provisioning HW
  - ○ E.g., count of compressed insts seems useful
  - ○ Vector crypto? What else? Need user ISA experts to help

Action items
- ☐ rdb197@gmail.com - Aug 8, 2024 - Include Bfloat16
- ☐ rdb197@gmail.com - Aug 8, 2024 - Check on single/double vs 32/64 terminology for FP events
- ☐ rdb197@gmail.com - Aug 8, 2024 - Consider adding events for VSETVL, div/sqrt, etc
- ☐ Beeman Strong - May 23, 2024 - check on idea to count remote HITMs locally