Seal5 Development & User Meeting

11.07.2024

Agenda (11.07.2024)

- Sync on ongoing work
- Discuss Core-V Tablegen freeze
- RISC-V Summit Europe 2024
- Next meeting dates
  - ***Internal:*** *01.08.2024, 11:00 (Skype)*
  - **Public:** 08.08.2024, 11:00 (Zoom)

Ongoing work

- **DLR**
  - Intrinsics
    - Milestone: https://github.com/tum-ei-eda/seal5/milestone/7
  - CI/CD Infrastructure (GitLab → GitHub)
    - Milestone: https://github.com/tum-ei-eda/seal5/milestone/6
- **TUM**
  - Register Pairs (RV32)
    - Related issue: https://github.com/tum-ei-eda/seal5/issues/43
  - Looking into porting support for MEM/BRANCH instructions
    - Assembly level support should be easy
    - Patterns unlikely to work with CDSL2LLVM
      - GlobalISel does not support MEM patterns
      - Branches in CDSL can not be mapped to LLVM-IR
    - Further: Tablegen limitations
      - Single-output patterns (does not work for incrementing loads)
    - Alternative: detect using Python (less generic and error-prone)
      - Hardcoded handling of addressing mode
      - Difficult to fuse with ALU-only head/tail instructions (ALU → Branch, Load → ALU, ALU → Store)
    - Related issues:
      - https://github.com/tum-ei-eda/seal5/issues/23
      - https://github.com/tum-ei-eda/seal5/issues/25

Core-V Tablegen Pattern Freeze

- **See issue:** https://github.com/tum-ei-eda/seal5/issues/106
- **Affected Instructions**
  - `CV_MACUN` (16 x 16 bit multiply + 32bit acc + right shift) → 33 bit intermediate result
  - `CV_MULURN` (16 x 16 bit multiply + add(`1 << (n-1)`) + right shift) → 33 bit intermediate result
  - `CV_MULSRN`
  - `CV_MACHHSN`
  - `CV_MULHHURN`
  - `CV_MACHHUN`
  - `CV_MACSN`
  - `CV_MULHHSRN`
  - `CV_MACHHSRN`
  - `CV_MACSRN`
  - Most dot products…
  - Not affected: `CV_ADDN` (32 x 32 bit addition + trunc to 32 bits + right shift) → 32 bit intermediate result
    - Mismatch between MUL and ALU on RTL
    - Core-V LLVM compiler uses truncating patterns for MAC/MUL
    - Needs more experiments
- **Next steps**
  - Test on newest version of LLVM
  - Build minimal example to reproduce bug (MACUN? ADDN with carry? → Map to pseudo instr to drop dependency on Core-V LLVM)
  - Find the reason for the freeze (endless loop?)
  - Report bug to LLVM community
  - Submit bugfix

RISC-V Summit

- **Seal5 at RV Summit**
  - Slides: https://riscv-europe.org/summit/2024/media/proceedings/plenary/Wed-12-15-Philipp-van-Kempen.pdf
  - Recording of Talk: https://www.youtube.com/watch?v=6NGjMO90RVM
  - Poster: https://riscv-europe.org/summit/2024/media/proceedings/posters/105_poster.pdf
  - Extended Abstract: N/A

Other notes/questions?

- …