SW Overlay

May 19th 2021

Antitrust Policy Notice

RISC-V International meetings involve participation by industry competitors, and it is the intention of RISC-V International to conduct all its activities in accordance with applicable antitrust and competition laws. It is therefore extremely important that attendees adhere to meeting agendas, and be aware of, and not participate in, any activities that are prohibited under applicable US state, federal or foreign antitrust and competition laws.

Examples of types of actions that are prohibited at RISC-V International meetings and in connection with RISC-V International activities are described in the RISC-V International Regulations Article 7 available here: https://riscv.org/regulations/

If you have questions about these matters, please contact your company counsel.

RISC-V®

RISC-V International

RISC-V is a free and open ISA enabling a new era of processor innovation through open standard collaboration. Born in academia and research, RISC-V ISA delivers a new level of free, extensible software and hardware freedom on architecture, paving the way for the next 50 years of computing design and innovation.

We are a transparent, collaborative community where all are welcomed, and all members are encouraged to participate. We are a continuous improvement organization. If you see something that can be improved, please tell us. help@riscv.org

We as members, contributors, and leaders pledge to make participation in our community a harassment-free experience for everyone.

https://riscv.org/risc-v-international-community-code-of-conduct/

Agenda

1. Github Status
2. New ABI
3. Action items from code-size meeting 11/May
4. Road map – next milestones
5. Round table – open topic/issue

Github status

1. Pending PR
   - overlay engine implementation (comrv PR)
2. Pending issues
   - Issue #13 - Behavior of overlay data

New ABI - options

Option 1

Create overlay ABIs - the set of reserved registers would be predefined, so this is not flexible.

- Will we support any ABI or a few selected once (e.g. ilp32, ilp32e,…)
- New ABI will be, for example ilp32o

Option 2

Propose a generic mechanism for specifying reserved registers

Need a way to specify this in the ABI string e.g. -mabi=ilp32_x31_x30_x29_x28 to reserve x31-x28

Action items from code-size

- tech-j-ext
  - Can we leverage their work?
  - Are there any overlapping points?
- PLT - Procedure linkage table
  - Usage and overhead over the current register allocation solution
  - Did we leverage parts of it on our current implementation?
- iCache/dCache
  - Should be part of the application hooks
- Virtual memory
  - Overlay storage and cache area cannot be handled by MMU
  - Need to state that in the HLD
- Overlay to support only 32bit arch
  - Need to state that in the HLD
- Support on interrupts
  - Need to state that we are not supporting that in the HLD

Road map

- Getting approvals
  - We need to get psABI approval for our relocations.
    There was no rejection, so we should close it.
  - Closing ABI issue
- FOSSI as our new home for overlay standard
  - The RISCV porting/implementation is attended to stay on riscv/Github
  - Overlay submissions to llvm/gnu will be referenced to the official Overlay-standard

Round table

- Open issues ?


Thank You
RISC-V®