IOPMP Task Group Meeting

January 19, 2023

Video link

Agenda

- Ratification Plan Update
    - Plan Target Date: Q1’23
    - Freeze Target Date: Q3’23
    - Ratification Target Date: Q4’23
- Frozen entries in a memory domain, aka *MDCFG.F*.
    - Nvidia’s revision.
- Static IOPMP entries:
    - Andes’ proposal.
- Any more atomicity issue?
    - SID-stall-bypass
    - Agree with the per-MD scheme to stall SID

Only RISC-V Members May Attend

- Non-members are asked to please leave except for Joint Working Groups (JWG).
- Members share IP protection by virtue of their common membership agreement. Non-members being present jeopardizes that protection. Joint working groups (JWG) agree that any IP discussed or worked on is fully open source and unencumbered as per the policy.
- It is easy to become a member. Check out riscv.org/membership
- If you need work done between non-members or or other orgs and RISC-V, please use a joint working group (JWG).
  - used to allow non-members in SIGs but the SIGs purpose has changed.
- Please put your name and company (in parens after your name) as your zoom name. If you are an individual member just use the word “individual” instead of company name.
- Non-member guests may present to the group but should only stay for the presentation. Guests should leave for any follow on discussions.

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.

Collaborative & Welcoming Community

RISC-V is an 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 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/community/community-code-of-conduct/

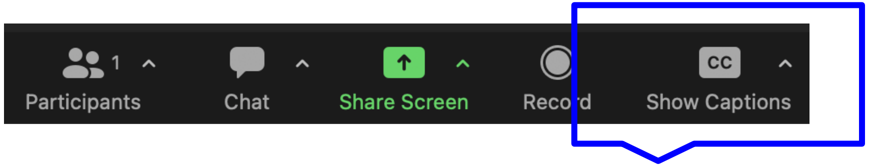
Conventions

- **For one hour meetings, please start at 5 after the start time** in order to allow people going to other meetings have time for a short break between meetings. 30 minute meetings start on time.
- Unless it is a scheduled agenda topic, we don’t solve problems or detailed topics in most meetings unless specified in the agenda because we don’t often have enough time to do so and it is more efficient to do so offline and/or in email. We identify items and send folks off to do the work and come back with solutions or proposals.
- If some policy, org, extension, etc. can be doing things in a better way, help us make it better. Do not change or not abide by the item unilaterally. Instead let's work together to make it better.
- Please conduct meetings that accommodates the virtual and broad geographical nature of our teams. This includes meeting times, repeating questions before you answer, at appropriate times polling attendees, guide people to interact in a way that has attendees taking turns speaking, …
- Where appropriate and possible, meeting minutes will be added as speaker notes within the slides for the Agenda

Zoom Translated Captions

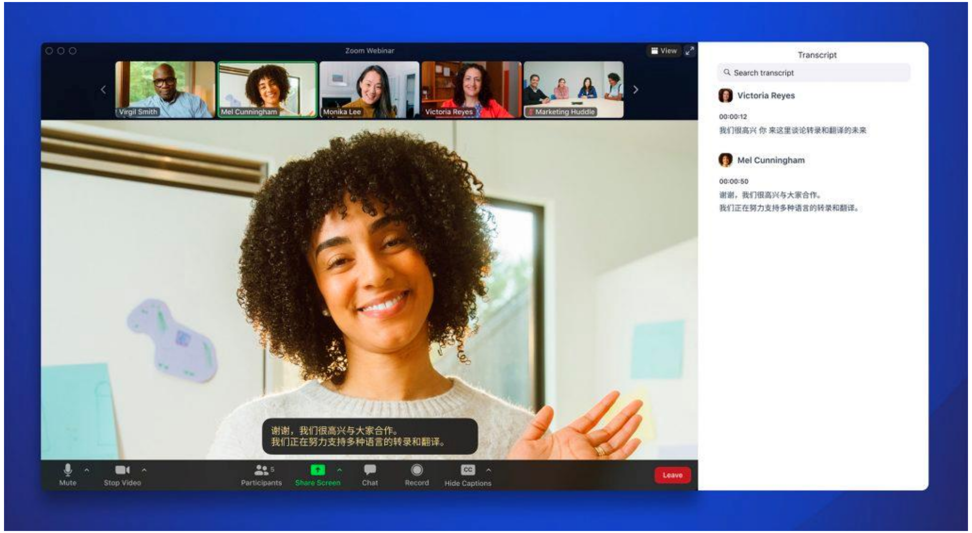
Step 1: Select **Show Captions** in the Zoom Meeting Window

- Tip: For smaller windows, this may be under More > Captions



Step 2: Ensure Speaking language is set to English. Under **Translate To**, select your desired output language.

Step 3 (optional): Under the Captions menu, select **View Full Transcript** (this will enable to you see captions in their full context)



谢谢，我们很高兴与大家合作。
我们正在努力支持多种语言的转录和翻译。

Transcript

Search transcript

Victoria Reyes

00:00:12

我们很高兴 你 来这里谈论转录和翻译的未来

Mel Cunningham

00:00:50

谢谢，我们很高兴与大家合作。
我们正在努力支持多种语言的转录和翻译。

Questions? Please email help@riscv.org

Zoom currently supports caption translation for 12 languages: Chinese Mandarin – Simplified (beta), Dutch, English, French, German,

Agenda

- Ratification Plan Update
  - Plan Target Date: Q1’23
  - Freeze Target Date: Q3’23
  - Ratification Target Date: Q4’23
- Frozen entries in a memory domain, aka *MDCFG.F*.
  - Nvidia’s revision.
- Static IOPMP entries:
  - Andes’ proposal.
- Any more atomicity issue?
  - SID-stall-bypass
  - Agree with the per-MD scheme to stall SID

Agenda

- Ratification Plan Update
  - Plan Target Date: Q1’23
  - Freeze Target Date: Q3’23
  - Ratification Target Date: Q4’23
- Frozen entries in a memory domain, aka *MDCFG.F*.
  - Nvidia’s revision.
- Static IOPMP entries:
  - Andes’ proposal.
- Any more atomicity issue?
  - SID-stall-bypass
  - Agree with the per-MD scheme to stall SID

Agenda

- Ratification Plan Update
  - Plan Target Date: Q1’23
  - Freeze Target Date: Q3’23
  - Ratification Target Date: Q4’23
- Frozen entries in a memory domain, aka *MDCFG.F*.
  - Nvidia’s revision.
- Static IOPMP entries:
  - Andes’ proposal: Permission Programmable bit
- Any more atomicity issue?
  - SID-stall-bypass
  - Agree with the per-MD scheme to stall SID

Why Static IOPMP Entries?

- Reduce the gate-count of the address matcher of IOPMP
  - A fully programmable address matcher could consume many times of gate-counts than a static one.
- Bus bridge/arbitrator already built-in address decoder
  - For a platform, the address decoders have the same function as the address matcher of IOPMPs. For some cases, we can utilize the address decoders as the address matcher, e.g., MMIO-spaced registers. The address decoders are typically unprogrammable.
- Reduce the programming mistakes
  - Prevent from coding mismatch/mistake
  - IOPMP entries are enabled at the same moment that IOPMP is enabled. ➔ No ordering issue

Why not Static PMP Entry?

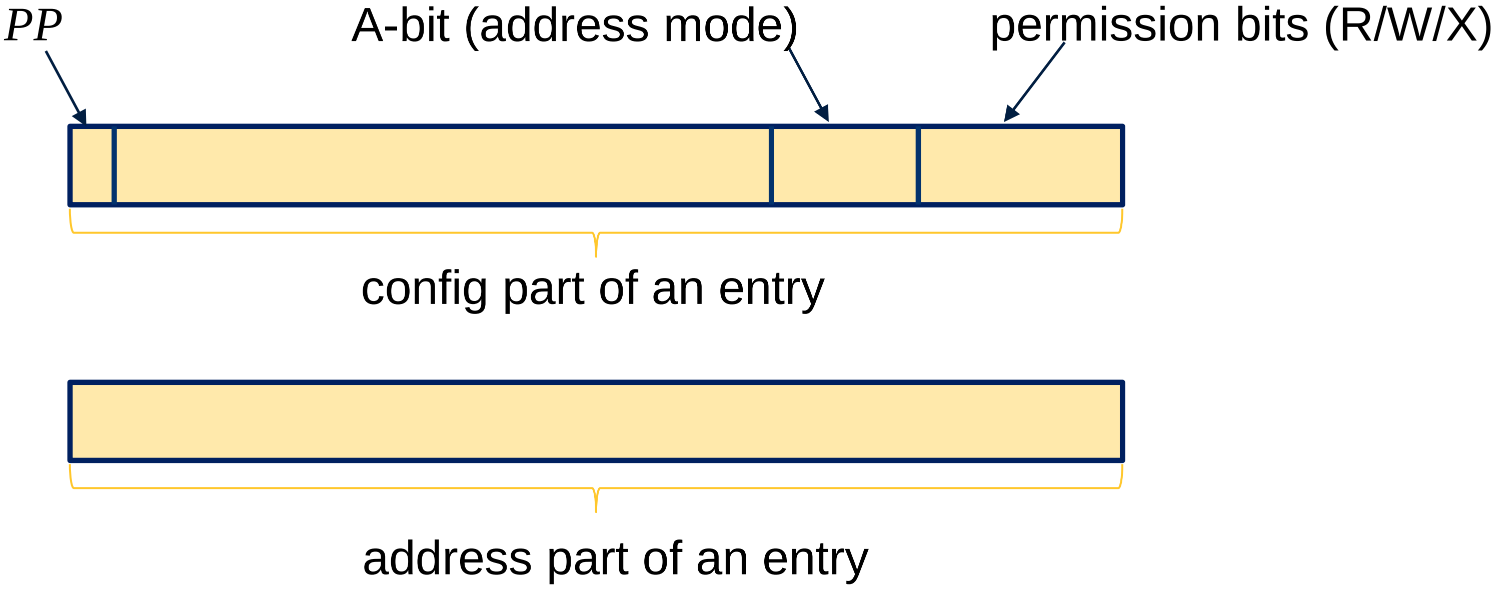
- Static PMP Entry? ➔ Possible, but Rare!
- PMP is a part of CPU IP. IP vendor usually does not know the MMIO layout.
- PMP entry would be swapped between different security contexts.
- On checking transaction, IOPMP checks source-ID.

➢ IOPMP entry has more chance of being static.

Static IOPMP Entries (by Andes)

- Apply all the locked-entry suggestions to static IOPMP entries.
- But, in some cases, before an entry gets locked, its permission could be altered over time.
- Example: secured storage in a Root-of-trust
  - Root-of-trust has a secured storage: device ID, private keys, random seed, etc.
  - The secured storage is placed at a fixed address in the MMIO space
  - At boot time, the secured storage can be fully accessed
  - Before entering REE, such an access should be controlled or denied.

An IOPMP Entry: 2 Words



PP bit

- **P**ermission **P**rogrammable:
  - ➢ Allow permission bits programmable in a fixed IOPMP entry.
  - ➢ Optional, per entry, and sticky to 0.
- *PP*=1 (on reset if implemented)
  - ➢ Permission bits (R/W/X) are still programmable even when address field in an entry is locked (by *MDCFG.F*)
  - ➢ Give users a chance to update the permission bits before PP is clean.
- *PP*=0 (also if not implemented)
  - ➢ Programmability of permission bits depend on *MDCFG.F*. (The same as the case of no PP)

Use Case: A Root-of-Trust

- The access permission to RoT need to be changed over time:
  - ➢ Its secured storage is in MMIO space and protected by an IOPMP
  - ➢ Corresponding IOPMP entry for the RoT: *entry rot*
- At boot time:
  - ➢ Access sensitive data from the secured storage.
  - ➢ Clean the permission in *entry rot* (set *R*/*W*/*X*=0/0/0)
  - ➢ Clean *PP* to lock the permission bits of *entry rot* ➔ The permission of *entry e* is NOT programmable, and also secured storage is NOT accessible.

Remarks

- When *PP* is not implemented:
  - It should be wired to zero, which means the permission bits and the address field of an IOPMP entry use the same lock mechanism (*MDCFG.F*).
- When *PP* is implemented
  - Address field: defined by *MDCFG.F*.
  - Permission bits:
    - ✓ *PP*=1: programmable
    - ✓ *PP*=0: defined by *MDCFG.F* (will be the same as “not implemented”)
- *PP* can be implemented only in the desired entry:
  - Minimize the cost
  - Test if implemented by reading back it
  - Good for the system without MD-permission mechanism

Agenda

- Ratification Plan Update
  - Plan Target Date: Q1’23
  - Freeze Target Date: Q3’23
  - Ratification Target Date: Q4’23
- Frozen entries in a memory domain, aka $MDCFG.F$.
  - Nvidia’s revision.
- Static IOPMP entries:
  - Andes’ proposal.
- Any more atomicity issue?
  - SID-stall-bypass
  - per-MD scheme

Thank You

RISC-V®