Date: 2021/02/02
Task Group: Fast Interrupts
Chair: Krste Asanovic
Co-Chair: Kevin Chen
Number of Attendees: ~10
Current issues on github: https://github.com/riscv/riscv-fast-interrupt/issues
Previous meeting minutes:
https://github.com/riscv/riscv-fast-interrupt/tree/master/minutes

Issue discussed:

#31 Does Interrupt Level Affect WFI?

We revisited and discussed the definition of interrupt threshold upon Wait for Interrupt (WFI). This is because several members argued that we should not create new counter-intuitive behavior that a lower privilege interrupt can wake up WFI but an interrupt from current higher privilege mode cannot.

After deeper discussion, we found out there are only two possible modes if we do not add extra hardware (while only use existing single reduction-sum-tree to select one global maximum):

1. [No-filtering mode] Ignore all thresholds and always wake up from WFI if there are any enabled interrupts in any privileged modes
2. [Filtering mode] Only wake up from WFI if there are interrupts above threshold (just like the case of preemption in normal usage, which reuses the same qualifying circuit).

We then discussed the usefulness of both modes and concluded that they are both useful/desirable:

1. [No-filtering mode] By dropping all thresholds from consideration for determining "locally enabled", this provides a more consistent view of WFI wakeup, but does remove ability to use threshold to remove events from set that will wake up WFI. A typical use case would be: a M-mode monitor issues a WFI after initial setup, but M-mode does not want to touch U-mode code at all (such as enabling or disabling U-mode interrupts).

2. [Filtering mode] The other mode is to filter lower privilege and possibly same privilege (below-threshold) interrupts from causing WFIs (this would be advisory as WFI can always wake at any point in time). This can be very useful because it can be very difficult to turn off each lower privilege mode interrupt individually as (unlike older CLINT) there can be 1000s of interrupts now. So it is convenient to use the threshold as a "global switch" to disable lower privilege mode interrupts from waking up WFI.

To facilitate both modes, a member proposed:

- If uie=0, don't wake WFI on user interrupts? Change from priv spec for CLINT.
- If all lower-privilege xie=0, then can use threshold value for current privilege mode to filter wakeup events?

However, this proposal redefines the behavior of *xie* in existing spec and thus breaks the backward-compatibility. To maintain backward-compatibility, another member proposed (after the TG meeting) to add extra sign bit to the threshold field:

1. [No-filtering mode] If the current privilege's threshold is -1, then WFI must wake up if any interrupt whose individual enable is set occurs at any privilege.
2. [Filtering mode] If the current privilege's threshold is 0-255, then WFI must wake up if any interrupt occurs at a higher privilege or at the current privilege and above the threshold. Interrupts at lower privilege or at the current privilege and at or below threshold are permitted to be ignored.