

Date: 2021/03/16

Task Group: Fast Interrupts

Chair: Krste Asanovic

Co-Chair: Kevin Chen

Number of Attendees: ~8

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>

Issues discussed:

#### #91 - Prototype DTS Entry?

Should spec give an example DTS entry? Discussed that adding DTS to the spec appendix would be a good idea.

#### #45 - xtvec behavior when base and original mode both supported

There is a more general discussion of delegation of feature mode bits happening in the hypervisor group. For first rev 1.0 of CLIC, we can require that all privilege modes can only run in the same mode. Spec to be updated with a comment indicating we can possibly support CLIC/CLINT mode switching in later versions.

#### #47 - CLIC reset behavior

There are a few places where CLIC reset behavior is defined. Spec to be updated with a general reset section with the reset information in one place.

#### #86 - xnxti and clearing pending for edge-triggered interrupts - closed

xnxti behavior of clearing of edge-triggered interrupts was listed in clicintip section. Updated spec to mention xnxti in clicintip section but move details to xnxti section.

#### #49 - Clarification of memory map required

Issues asks for clarification of memory map for M/U, M/S/U cases. Discussed this is generally a platform issue. The spec itself focuses on the structure within each aperture not the location of the apertures in the global physical memory space. Each platform needs to define a discovery mechanism to determine the memory map locations. We should remove or downgrade to commentary the recommendation on the layout of multiple harts and multiple privilege modes, as different platforms will want to handle these differently.

#### #77 - (was issue 57 item 1) additional detail on CLIC M/S/U memory mapped regions

Since there is a mclicbase, should there also be sclicbase/uclicbase? Discussed that lower mode base can be affected by address translation of lower modes. may not even need

mclicbase since most embedded systems will compile in the CLIC memory map as a constant instead of querying the CSR and using as a variable.

#78 - (was issue 57 item 2) - machine mode alignment - closed

Should spec specify m-mode clic spaces can be mapped with one PMP? Discussed that if PMPs are implemented, unless M-mode aperture is explicitly enabled for S and U mode, they cannot access it, so no PMP entry is required to isolate M-mode aperture.

#79 - (was issue 57 item 3) - supervisor/user/hypervisor mode alignment

Should spec mention that s/U mode clic regions should be aligned on VM page (4k) alignment so they can be mapped through TLBs. Discussed that it is a platform issue but can be listed as a recommendation in commentary.

#80 - (was issue 57 item 4) - version fields in clicinfo

Discussed that clicinfo falls into the category of platform discovery info. possible this info will be removed from CSR space.

#81 - (was issue 57 item 5) - programming clics in s and u modes

This falls into general category of delegating configurable features to lower privilege modes. There is discussions in hypervisor group about a more general pattern for these kinds of settings.

#82 - xcause registers.

original question: should xistatus registers be created instead of adding new fields to xstatus? most recent thread asked about what if machine mode in clint mode takes a trap while accessing a clic vectored interrupt. In the meeting we discussed that as with #45, for now we will mandate that all privilege modes see the same core-local interrupt controller (CLIC or CLINT), with only M-mode able to support configuring the interrupt controller.

#29 - interrupt trigger in the trigger module

Spec has been updated to support existing debug task group standard triggering behavior. Need to fix the reserved comments in memory map table.

#120 - Incorrect spec of behavior on WFI

Spec has been updated with clarification. Discussed that we still need to specify what happens with WFI inside a handler.