Date: 2022/07/19

Task Group: Fast Interrupts

Chair: Dan Smathers

Co-Chair: Kevin Chen

Number of Attendees: 6

Meetings Disclaimers Video :
https://drive.google.com/file/d/1y_XWJus8M5ZwSQ2cvEOzCjlOmsmXOnN4/view

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

Fast Interrupt DoD (Definition of Done) Status: (new location)
https://wiki.riscv.org/display/HOME/Fast+Interrupts+TG

Next meeting agenda (08/2/22)

Meeting minutes

Possible to freeze this year? Different base and levels on top of that base. figure out what the options are. simple thing. Too many variants. lots of options buried in the spec. separate section at the end what would be needed. software into appendix so not part of description. to freeze need a lot of people to read it. to get thru freeze, make it easy to read. a couple technical issues still to resolve. resolve tech issues. clear up spec. base and options. review by the horizontal committees. approval before freezing. internal review that chairs have to sign off. arch review. way to make it go smooth is clear spec. some of software is useful for designing but not really part of spec.

#191 – what is the type of access vs. Permissions. hw vector now treated as execute. so don't think there is a problem more than advisement. if you have hw vector fetch. (no different than code that can be read and executed.) if doing sw vectoring, needs to be readable. if hw vectoring, needs to be executed. Should add a note. if only doing hw vectoring only need execute. If doing sw vectoring need read.

#228 – breakpoints check pc values. so may or may not just work depending on hw. sw loads just work how loads work. so sw vectoring should just be regular. same distinction as #191. if hw based can't set.

For explicit loads and stores, watchpoint. All systems do that the same. for implicit read treated as instruction fetch. different micro arch may or may not do that fetch. so where is comparator. for hw vectoring, those implicit fetches are optional whether they work or not. add to list of architectural options. note, does not work on Zc.

#244 – debug trigger behavior. Same comment as #228? 1. execute address yes. note, optional hw feacture (see #228) load address. 2. No. note: if sw vectoring need a load address type trigger. issue closed with comment.

#230 – similar to #228. closed with comments.

Github updates since last minutes:

Specification updates since last minutes:

Open issue status:

Issues that can be closed? (Simpler **issues to discuss in red)**

#49 – reopened: prev spec had single control for multiple harts but current spec implies control per hart and implications. Closed by #231? Dependent on issue #96/226 discussion?

#160/200 - can rewrite to allow for but not assume n-extension is available. probably combine #160 with #200. closed by pull #201?

#202/pull #238 –writes/reads to clic mem mapped registers. (discuss if 64b store ordering is required. Pull is different from TG discussion)

#235 – xcause.X updated on exceptions wording.

Need spec updates:

#75 – move hw vectoring to separate section in spec - waiting until other issue spec updates before making this large text change.

#100 – reserve usage of immediate bits (0-UIE?),2,4 in xnxti imm usage since those mstatus bits are WPRI. Related to #205/211

#158 – change CLICCTRLBITS to CLICMLPBITS. (resolve #226 first?)

#221 – compressed clicintip/clicintie status registers (1-bit per interrupt) similar to AIA eip0-eip63? Add justification and create a pull with a proposal.

#228 – dpc treatment - can breakpoints be set in hw vector table? (related to #230). Code-reduction TG chose to not allow setting breakpoints on hw vector table. https://github.com/riscv/riscv-code-size-reduction/issues/161

#239 – Dret should honor inhv

#240/pull #241 – change mscratchcsw to use mstatus.mpp in pseudocode? does it matter if not visible in non-clic mode? Need to update text to clarify.

#248 – xtvec.mode clarification. current text reuses field name mode as bits 5:0 vs 1:0. better to create a new field name.

#249 – xtvec.mode - instead of redefining mode as bits 5:0, should keep previous bits 1:0 and define new clic 4-bit field bits 5:2.

Need more discussion:

#96 – proposed reformat of cliccfg. bit spec change, small hw change. Tried to close with Pull #183. more discussion about xcliccfg.nmbits. change to a programmable boundary value? discuss more. created a new pull #231 to go thru change implications. Separated xcliccfg.mlvl proposal to issue #226.

#102 - preemptible interrupt handler code (for section 7.2)

#108 – pushint/popint? defer to broader discussion on providing hardware stacking of interrupt contexts (possibly post-1.0).

#171 – CLICCFGLBITS parameter - related to #80, #158. (resolve #226 first)

#191 – reopened. question about sw vectoring.

#205 – xnxti side-effect question. `csrrsi rd, mnxti, uimm[4:0].` Pseudo-code shows uimm needs to be non-zero to cause side-effects (updating mintstatus, mcause.exccode/interupt). Is this restriction necessary? Related to #100/#211?

#219 – Interrupt ID ordering recommendations

#222 – legacy s-mode/u-mode timer interrupt setup

#226 – replace clicintattr.mode with xcliccfg.xlvl proposal (formerly #96)

#229 – clicintattr.mode WARL behavior (moot if #226 accepted)

#247 – behavior when MPP is two-bits and hypervisor not implemented

#248 – CLIC hypervisor mode (related to #92).

#250 – uedeleg/uideleg CSRs addresses were taken by float-point. not really clic anyway? need to add text to clic spec in xideleg section?

Issues need to be worked:

#91 – DTS entry – have linux group review DTS example.

#107 - heritage of features. keep researching and adding references to bibliography.

#185 – SAIL model implementation of CLIC (probably waiver)

#186 – CLIC architecture tests

#187 – QEMU CLIC implementation update

#242 – spike required if sail is waived?

Issues waiting on ratification (encoding/opcode consistency review needed)

#88 – CSR address mapping

Issues punted for rev1, keep open for future enhancements:

#92/Pull #181 – hypervisor compatibility. Pull #181 with Initial hypervisor extension proposal for CLIC. Punted for rev1.

#99 – horizontal interrupt window. punted for rev1

#101 - xnxti to trigger on equal level. punted for rev1

#106 – allow level change. Punted for rev1

#192 – allow mix of CLIC/CLINT at different priv modes punted for rev1