Date: 2021/08/03

Task Group: Fast Interrupts

Chair: Dan Smathers

Co-Chair: Kevin Chen

Number of Attendees: ~6

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:
https://wiki.riscv.org/display/TECH/Fast+Interrupts+TG

Next meeting agenda (08/17/21)

More open issue discussion?

Meeting minutes

RNMI is still in Krste's queue. No current update.

WMFI – (solution for enabling wakeup of interrupts in pending and higher modes, but not lower modes – see issue #31)github riscv/riscv-isa-manual https://github.com/riscv/riscv-isa-manual/issues/700

#86 – when are edge-triggered ints cleared? add more detail to section 5.8. add to pseudo-code. xnxti – write is affected by whether there is a pending interrupt or not. pseudo code only looks at csrrsi, change to make more general for the other kinds of csr instructions. update both pseudo-code and text. would setting any of the other bits in mstatus do anything weird? hole in spec that it doesn't discuss csr rw. Is the pseudo-code wrong? if immediate is 0, it doesn't do anything? shouldn't have side-effects in that case.

#48 – relates to #86. fix details about hardware clearing of pending bits. more xnxti discussion. xnxti can change mstatus.xie. at what point does that happen compared to clearing pending interrupt.

Create white paper on how to use clic?

#158 – clicintctlbits parameter. how many mode bits and how many clicintctl bits. can it be dynamic or not?

Sum of all bits – read-only parameter. optional if chose to freeze how many are mode and how many are ctrl. different impl. May choose to freeze the implementation. that's the only way to have a parameter that is read-only. sum of all the bits. CLICINTMLPBITS

Github updates since last minutes:

#86 re-opened.

#158 – additional comments added

Specification updates since last minutes:

Open issue status:

Issues that can be closed?

#29 – Interrupt trigger in the trigger module – closed with pull #126?

Need spec updates:

#48 – logic/state diagram for clicintip signals (related to #149?), add more clarification about hardware clearing of IP bits. only xnxti writes can clear IP bits.

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

#80 – version fields in clicinfo – remove clicinfo and make it a separate platform discovery mechanism? remove mclicbase too?

#86 – update xnxti pseudo-code and text

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

#148 – Elaborate address map possibilities. Platform should be doing the layout.

#154 – inhv clarifications

#158 – change CLICCTRLBITS to CLICMLPBITS.

#160 - can rewrite to allow for but not assume n-extension is available.

Need more discussion:

#45 – all priv modes in clic or all in clint. Need spec update that only m-mode can select. Pull #145 tries to close. need to clarify stvec.mode, utvec.mode register behavior when mtvec.mode switches?

#49 – reopened: prev spec had single control for multiple harts but current spec implies control per hart and implications.

#50 – xcause save privilege modification bits

#86 – auto clearing of edge interrupts

#96 – proposed reformat of cliccfg

#97 – proposed reformat of xcause CSRs

#98 – name of CLIC (related to #109?)

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

#31, #120 – WFI behavior – need to clarify text in pull #135. pull text has typos. Pull #144 tries to fix typos and clarify text. Discuss in TG. Tried to update pull text to clarify when interrupt is set to 0.

#155 – clarify inttrig details. Entire meeting discussed CLIC trigger details with Tim from debug group. CLIC, trigger module, and hart need to be tightly integrated since these features need to be in the core to get sync behavior. discussion on where the details are written down. E.g. CLIC spec describes set of interrupts selected for interface to debug, debug specifies what happens on that event (enter debug mode, start trace, stop trace). For now, keeping abstract action in CLIC, debug takes concrete action. try do discuss more over email reflector.

#158 – add option to reuse unused clicintattr.mode bits with clicintctrl to determine level value when 3 modes are supported but only 1 or 2 modes are selected by clicctrl.nmbits?

Issues related to work in other TGs:

#81 – programming clic in s and u modes – discussions in hypervisor group about a more general pattern of delegating configurable features

#91 – DTS entry – closed with pull #130? have linux group review DTS example.

#92 – hypervisor compatibility – goal is not to virtualize CLIC but to allow virtualization of device interrupts. Follow up with hypervisor group.

#109 – add arch string for CLIC – need appropriate named and versioned sub-extensions of clic.

#155 – clarify inttrig details. discuss with Tim from debug group. Invite Tim to fast-interrupt TG meeting to discuss.

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

#88 – CSR address mapping

Issues punted for rev1, keep open for future enhancements:

#82 – xcause register behavior with some modes in clic and some in clint.

#99 – horizontal interrupt window

#100 – reserve immediate bits

#101 - xnxti to trigger on equal level

#106 – allow level change

#108 – pushint/popint?