Date: 2021/11/09

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 (11/23/21) (note 12/7/21 will be cancelled due to summit).

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

#92 – hypervisor compatibility

#97 – proposed reformat of xcause CSRs (bigger discussion).

More open issue discussion?

Meeting minutes

issue #179 discussion – should mcause.interrupt be set when mcause.exccode = clic.id. seems correct. Only informational (doesn't change hw) but helps sw know. seems like sw needs to know. Always going to a vector so handler wouldn't look at bit. but probably want to set. Added to spec. issue closed.

Pull #178 discussion – merged #178 and fixed text clarifying when the edge-triggered pending interrupt is cleared. it is cleared the same time as mstatus.xie.

Discussed discovery, related to issue #80 – ACT yaml file to determine claimed support. off-line. for all CSRs, this bit is read-only, read-write, value between this and that. arch options. what lives in rom. has to be somewhere. In config structure. handlers aren't going to read clicinfo, mclicbase. embedded will compile into code. primarily debug. discovery csr that points to something in memory. platform issue. discovery method not mature yet however.

Issue #109 – put smclic in end of spec so can group discussion in the end of which ones have which features. refer to vector spec format. what are the clic extenstions. what do they include. If we want to have an s-mode clic, may need to define a ssclic.

Github updates since last minutes:

Issue #179 opened: nxti pseudo-code question. should mcause.interrupt be added to pseudo-code. closed.

Issue #48/PR #178: inhv interrupt clearing question. closed.

Issue #80 – clicinfo/parameters. added text that these might go away. closed.

Specification updates since last minutes:

Issue #179 – added setting mcause.interrupt to nxti pseudo-code. closed.

Issue #48/pull #178 – clarify only writes to nxti cause state change, clarified when edge-triggered pending interrupt is cleared. updated spec. closed.

Issue #80/pull #175 – updated text mentioning clicinfo/mclicbase might go away with discovery mechanism TG.

Open issue status:

Issues that can be closed?

Need spec updates:

#45 – all priv modes in clic or all in clint. Need spec update that only m-mode can select. Pull #173 tries to close.

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

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

#96 – proposed reformat of cliccfg. bit spec change, small hw change.

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

#109 – add arch string for CLIC – need appropriate named and versioned sub-extensions of clic. Smclic? smclicx if we break into pieces.

#158 – change CLICCTRLBITS to CLICMLPBITS.

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

#180 – look for references to processor in spec and change to hart

Need more discussion:

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

#97 – proposed reformat of xcause CSRs (bigger discussion).

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

#29/#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.

#171 – CLICCFGLBITS parameter - related to #80.

Issues related to work in other TGs:

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

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?