NOTE: updated zoom info for next meeting and beyond.

Meeting link: https://zoom.us/j/95666422067

Passcode: 174425

Join link: https://zoom.us/j/95666422067?pwd=aHM1ZEJDcFJ3cGE0UWdTRmtaUXhsQT09

Date: 2023/09/26

Task Group: Fast Interrupts

Chair: Dan Smathers

Co-Chair: Jean-Baptiste Brelot

Number of Attendees: 7

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

Next meeting agenda (10/10/23)

Issue #358/357/347 - mret inhv pseudo-code

Issue #356 – xinhv still cleared after xRET with new spec wording?

Issue #355 – mnstatus fields referenced seem wrong. location of latest rnmi spec?

Issue #349 – discuss moving CLIC memory mapped registers to indirect CSRs

Issue #348 – discuss location of inhv for future if enabling CLIC per mode

Issue #347/pull #354 – exception handler pseudocode needs checking

Issue #314 – shv==0 requirement for xnxti. more discussion needed. New CSR?

Issue #308/pull #325/pull #344– xnxti service loop – fix of pull #343 based on discussion.

Issue #303 – How are CLIC-only CSRs and fields handled in CLINT mode

Issue #307/pull #331 – Parameter cleanup (compare with pull #297?)

Meeting minutes:

SAIL status – almost done with approving vector extension PR into SAIL.

#349 - Fixed csr so pipeline can figure out conflicts. indirect csrs, don't know select register. value may not be ready yet. so things in ind csrs can't cause pipeline events. so means ind csrs aren't critical timing. so flexible but more constrained on what they can do. just like AIA, it is a good candidate. so ind csrs is like a local memory map. only hart can access so cleaner than memory map. ind csrs handles priv mode permissions. memory map can do it with pmps. one issue, remote writes to interrupt pending. in AIA they have the msi stores that can cause interrupts. so IP bits are the ones that have to be visible. think about how to align the ip/ie with AIA definition so PCIE cores can do things in the same way.

Would be a big change. usually try to avoid overlap. preemption is main difference between AIA and fast interrupts. work on proposal and what it would work on. still need memory arrays for interrupt pending. so maybe config would move to csrs? in AIA, hart doesn't need to know where the files are in memory. for virtualization it just looks at csrs. so can migrate the context around the machine and doesn't need the memory addresses. indir csrs would almost like adding a layer on top of memory map. faster to initialize because grouped into bit vectors instead of byte vectors. memory map and csr map layout can be different. keep both as an option. probably just need the IP? maybe specify they don't appear in the memory map? so lets work on a proposal. broader implications. main advantage with multiple cores. having the fw on each core understand where it's interrupts are vs seeing it thru csrs. so boot time software can config setup and cores just run using csrs.

#355 – rnmi location. location pointed to in #355 seems correct. 0.4. rnmi has spike support. rnmi waiting on SAIL/ACT. act – need way to generate non-maskable interrupts. act needs some infrastructure. last work in march/april 23 timeframe. RAS is assuming it is there.

RNMI – add 4 csrs to hold state. Mncause would have to have additional fields and mnret needs to do the right thing to resume the fast interrupt. NMI means software can't mask interrupts so can be in the middle of anything, vector off. might log error or shut system down. then may need to return. so off to the side. so almost no interaction. won't interact with clic. will update spec.

So two sections. non-resumable and resumable. no interaction with the fast interrupts section with rnmi.

#358 – can only do mret in mode so change xinhv to minvh. put comment that mstatus.MPP =cur_priv (ends up in m-mode). also fix ~1 when inhv is set.

Github updates since last minutes:

Specification updates since last minutes:

Open issue status:

Issues that can be closed?

#307/pull #331 – NUM_INTERRUPT listed in two sections (text cleanup)

#355 – clic interaction with rnmi

Need spec updates:

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

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

Pull #286 – text update fix for #96/158/171/226/49 - allow multiple implementations for priv mode/level.

Need more discussion:

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

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

#303 - How are CLIC-only CSRs and fields handled in CLINT mode?

#308/pull #325 – xnxti service loop behavior

#314 – shv==0 requirement for xnxti

#355 – mnstatus fields referenced seem wrong. location of latest rnmi spec?

#349 – discuss moving CLIC memory mapped registers to indirect CSRs

#348 – discuss location of inhv for future if enabling CLIC per mode

#347/pull #354 – exception handler pseudocode needs checking

#356 – is xinhv still cleared after xRET or was that changed.

#357/358 – mret/inhv pseudo-code question

#360 – which smstateen bit will clic use

Issues need to be worked:

#91 – DTS entry – have linux group review DTS example. If add CLINT compatibility option, just use same DTS entry as clint?

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

#185 – SAIL model implementation of CLIC

#186 – CLIC architecture tests

#187 – QEMU CLIC implementation update

#221 – compressed clicintip/clicintie status registers (1-bit per interrupt) similar to AIA eip0-eip63? Add justification and create a pull with a proposal. Pull #270. add prior art reference.

#242 – SPIKE model implementation of CLIC

Issues punted for rev1, keep open for future enhancements:

#92/Pull #281 – hypervisor compatibility. still 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

#108 – pushint/popint? defer to broader discussion on providing hardware stacking of interrupt contexts. Punted for rev1

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

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

#**329 -** support for hw register save.