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: 2023/06/20

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 (07/18/23) (07/04/23 cancelled due to holiday)

Issue #339/pull #340 – clarification of xcause/xret in clic spec

Issue #333 – xcause.xinhv - should x be faulting priv or handling priv.

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

Issue #308/pull #325 – xnxti service loop – more discussion needed.

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

Meeting minutes:

Barcelona summit – ACT plans and gaps and challenges. pull to add new CLIC CSRs. There are ACT regressions.

SAIL regressions. SAIL has a ci – create branch. anytime do a checkin to my branch, the ci is run. when it merges to main, it gets run. and there is a set of compilation of sail model and running basic tests. old set of riscv tests that is run. eventually the act tests.

Saravana on PLIC code. send code to bill for peer review. Do first pass look. have simulation that simulates interrupt. nothing to compare against. no spike model. manual verification of values. no qemu model. how to send an interrupt to reference model. saravana building on external model. need an external model interrupt simulator. Allen will send Saravana a link.

Order – spike first, then sail, then qemu? eventually sail first as it is adopted over time.

Sail chicken egg problem. if goes into sail first. have acts that basic functionality is there. then sometime in future, spike is cross-check. configuration problems. have to understand spike and sail can be configured in similar manor. Spike is not really configurable. low priority item for them. have been trying to have configuration in sail with regards to csrs and warl. how complete will sail be? won't do vendor specific things. not possible to do all permutations of csrs and warl. defined a syntax for the common ones we expect. people are encouraged to build what is defined like what is legal and what is not. if illegal mapping easy recommendation is don't change.

ACT test coverage. coverage model. yaml formatted file with coverage you want. tool runs and did you get the coverage you want. more for test writers. yaml has all the things you are trying to test for. does union of all the coverage you asked for.

Acts run test and store state into memory. read and compare. if doesn't store inhv state into signature. coverage looks at trace state. anything that changes state are part of sail log file and that's how coverage knows if it was covered or not. register change/csr change, pc changes, arch visible should be seen in the log file. completion of stores was one hole. some concern when it is flagged as done. non-isa part of clic (clicintctrl/ip/ie/attr) are to mem-mapped registers. if sail model provided it as part of the log file then it could be done. isac (isa coverage tool). tool would need to be updated to look for those memory mapped coverage. coverage – make sure that this interrupt pending bit got set when this bit in clic bit was set or cleared. how to write json coverage – isac tool is in different repository. in software tools? riscv/software that has full documentation. coverage reports of acts run against sail, should be in github? not in archtest directory, in admin. everyone has to get to it because everyone has to file test reports. maybe those are in google docs. artifacts aren't supposed to be in github. so that's why it is a google drive. drive.google.com

Discussed Issue #339 – Pull looks correct. issue closed.

Discussed Issue #333 – can't see inhv unless software wrote it or an exception happened.

In u-mode. get s-mode interrupt try to read table. table read causes m-mode exception. interrupt handler and exception handler. sinhv is set in interrupt handler. interrupt handler and exception handler. the mode of the handler is what text says. the thing that handles the table read exception, just sees a table read exception. it will see sinhv is set so reason for exception was a table read in s-mode. when ret from m-mode exception handler, normally don't touch sinhv. what may be wrong is exception will create mepc which points at the table entry.

Some times page tables. s-mode running, set to s-mode handler handles page fault. at end, just does a ret. we should walk thru the scenario of a page handler. idea is paging a table. even if m-mode does nothing else, want it to report it. but to resume assumes s-mode/vs-mode to bring table back in and resume. xinhv only useful to the exception handler and interrupt handler shouldn't really see it.

Needs a bit more thought. inhv only usable by the exception handler. doesn't need to be visible to interrupt handler mode. good one to do in email to discuss a bit.

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)

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

#333 – xcause.xinhv - should x be faulting priv or handling priv.

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.