

**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: 2023/05/23

Task Group: Fast Interrupts

Chair: Dan Smathers

Co-Chair: Jean-Baptiste Brelot

Number of Attendees: 5

Meetings Disclaimers Video :

[https://drive.google.com/file/d/1y\\_XWJus8M5ZwSQ2cvEOzCjlOmsmXOnN4/view](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 (05/06/23)**

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:**

Sail discussion: For adding new extensions to Sail, look at sail pull 246. Compare with Zicond, smepmp. Csrs are more difficult because have to touch more files. otherwise have one file with the extension. with csrs it probably could have been done that way but it wasn't.

Discussing #329 – support for hw save. Split into 2 conceptual pieces: 1) Hardware save restore instruction like push pop. 2) Instruction triggered by interrupt. automatically do that in ABI compatible way. have shadow registers to switch context more quickly. lots of pieces to put together. We will focus on current clic spec and think there is no impediment to doing this later. labeled issue as post 1.0.

#314 – smclicshv xnxti only works with clicintattr.shv=0. more discussion.

Sounds like on a given interrupt handler, they would like to dynamically switch between software and hardware vectoring. so have 2 different hw vectoring handlers, one where it needs

to save/restore, and one where save/restore has already been done. what is the best way to tell the hw that? all the vectoring in clic is done by xvtv. would need a mode bit? can't just double size of table because don't know number of interrupts. use even/odd?

### **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)

#322/pull #330 – xcliccfg field text cleaning. text needs to be cleaned up due to newly added fields.

### **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

#329 – support for HW register save?

### **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 (probably waiver)

#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 required if sail is waived?

**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.