Inconsistency between Privileged spec and Unprivileged spec related to counters #769

Open **Silabs-ArjanB** opened this issue on Nov 12, 2021 · 4 comments

Silabs-ArjanB commented on Nov 12, 2021 Contributor

In the “Counters” chapter of Unprivileged ISA specification the following is mentioned:

“For a low-end implementation, the upper 32 bits of each counter can be implemented using software counters incremented by a trap handler triggered by overflow of the lower 32 bits.”

This seems to conflict with the fact that mcycleh and minstreth are mandatory in the Privileged Specification.

I have two questions related to this:

- It would be great indeed for low-end implementation if cycleh and instreth do not need to be implemented. This however only makes sense if the implementation is free to not implement mcycleh and minstreth either while still being compliant to the Privileged specification and this seems not to be allowed (as nowhere it is mentioned that an implementation is allowed to trap on accesses to mcycleh and minstreth). Is implementation of mcycleh and minstreth optional (while still claiming compliancy to the Privileged ISA specification)? Even better, would it be allowed to trap on all of mcycle, mcycleh, minstret and minstreth CSRs while still claiming compliancy to the Privileged specification (as low-end implementations can easily do without any of these CSRs)?
- If we implement the mentioned ‘trap upon overflow’ for each of the counters, then it would be nice if the cause of such trap would get standardized in mcause. Are we supposed to just use some of the ‘designated for custom use’ values in mcause or are there plans to standardize these overflow bits. (I am aware of the proposed Sscofpmf extension, but that seems more related to the Zihpm extension as opposed to the Zicntr extension).

aswaterman commented on Nov 12, 2021 Member

The text you quoted is non-normative so shouldn't be taken as Gospel.

You're certainly right that this text is in conflict with the M-mode architecture, but note that a correct implementation of the unprivileged architecture technically doesn't require M-mode at all.

allenjbaum commented on Nov 13, 2021

The conflict I see here is there is no architectural defined way to trap on those counter overflows, and that trap isn't invisible to Mmode - is that the conflict you see?
I interpreted the "software counter" to mean a CSR that didn't have counter hardware, as opposed to saving it in SRAM somewhere or something, ao mcycleh, etc., would still exist.
I also realize that is silly, since they're objecting to more than just the counter HW, but the flops s well.
...

aswaterman commented on Nov 13, 2021 Member

@allenjbaum I was thinking along those lines: the visibility of the trap to M-mode is the most problematic aspect, IMO.

Silabs-ArjanB commented on Nov 15, 2021 Contributor Author

Thank you for your answers. Agreed that 'that a correct implementation of the unprivileged architecture technically doesn't require M-mode at all'. This is why I added the phrase 'while still claiming compliancy to the Privileged ISA specification' in my question. The reason behind my question is indeed that it would be nice if there would be a way to eliminate the cost of the associated 128 flip-flops (mcycle(h) and minstret(h)) and 2 64-bit counters in small embedded cores, while still being compliant to both the Privileged and Unprivileged specifications. It is clear that this is not possible.

2

Assignees

No one assigned

Labels

None yet

Projects

None vet

No milestone

Development

No branches or pull requests

3 participants