AIA User Priority Mask Extension
Minimizing Critical Sections Side-Effects on Real-Time Automotive Systems

Sandro Pinto, José Martins, Manuel Rodriguez, Kajetan Nuernberger*, and Thomas Roecker*

Centro ALGORITMI/LASI - Universidade do Minho, OSYX Technologies, *Infineon AG

Abstract

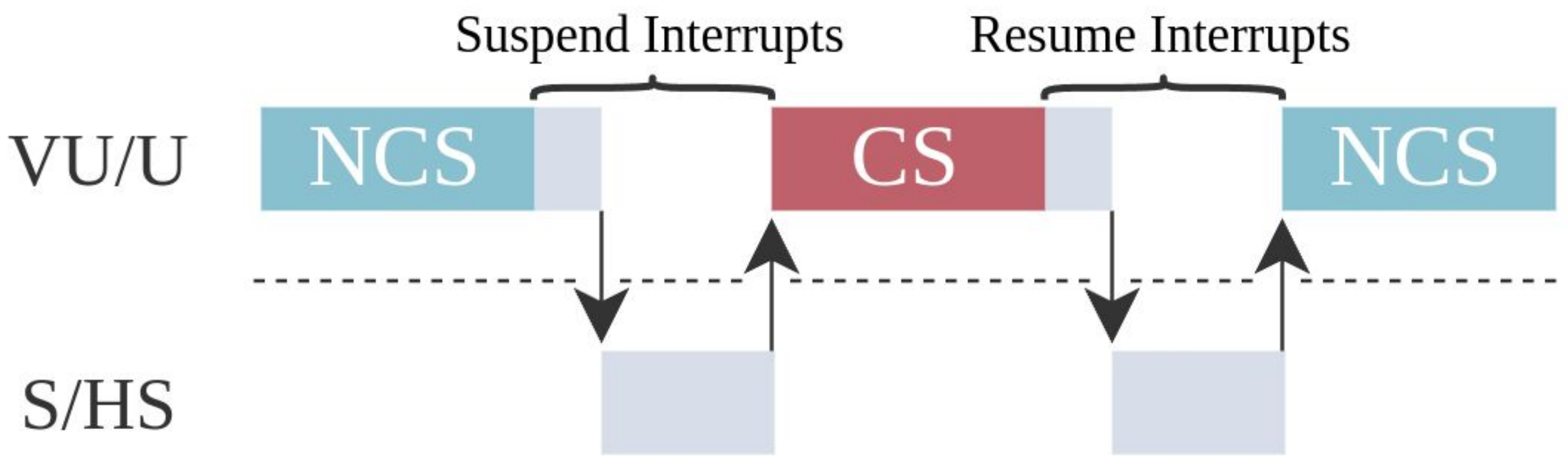
Critical sections are widely used in many real-time automotive scenarios; however, if not adequately supported at the ISA level, it can lead to unintended performance impact. Studies within the AUTOSAR community point up to 30 % CPU performance lost in handling of critical sections. In this work, we advocate that the RISC-V ISA (and related specifications) cannot efficiently support critical sections for real-time automotive systems. To address that, we propose a novel AIA extension. We implemented the proposed extension on a CVA6-based SoC endowed with an Advanced Interrupt Architecture (AIA) IP and functionally validated the intended behavior. We are now deriving quantitative evidence/statistics to support the discussion of the current proposal at RISC-V International. We are going to open-source all artifacts to promote collaboration within the RISC-V community.

RISC-V AIA

- The Advanced Interrupt Architecture (AIA) is the state-of-the-art reference architecture for interrupt handling.
- Each IRQ source is assigned a minor identity per hart, which defines its priority within the preemption scheme.
- For each privilege level and virtual hardware hart capable of receiving MSIs, the IMSIC hart contains an interrupt file.
- Each interrupt file within the IMSIC has a priority-based masking mechanism via a configurable threshold for the IRQ presentation.
- A nonzero threshold value P blocks interrupts with identities ≥ P, filtering lower-priority signals to the hart.
- **However, User-mode software lacks a direct interface to the IMSIC for disabling/suppressing interrupts.**

Critical Sections: Automotive Use case

- Critical sections are required in various real-time automotive scenarios.
- Every time a critical section is enforced, a pair of software routines for enabling/disabling interrupts is invoked.
- Although these operations are short (few nano to microseconds), they are frequently invoked, introducing non-negligible overhead.



NCS - Non-Critical Sections; CS - Critical Sections

AIA User Priority Mask

- The AIA User Priority Mask Extension is aimed at giving restrained control to user mode over interrupt priority.
- This will allow user mode applications to disable interrupts up to a certain priority level in, for example, time-sensitive critical sections.
- We propose 3 new CSRs:
  - User Threshold (ueithreshold). Determines the minimum interrupt priority (i.e., maximum interrupt identity) for an interrupt to be signaled from the attached S-mode interrupt file.
  - Supervisor-defined Threshold (sminueithreshold). Determines the maximum interrupt priority (i.e., minimum interrupt identity) that can be written into ***ueithreshold***.
  - Threshold Computation (ufueithreshold). RO register with the final threshold value used to determine if a pending-and-enabled S-mode interrupt can be signaled to the attached hart.
- **With the AIA User Priority Mask Extension, user applications can mask interrupts up to a certain level defined by the OS.**
- **The OS controls whether user-level code can mask interrupts;** either preventing any masking (default behavior) or allowing all interrupts to be masked by user code.

ueithreshold	sminueithreshold	ufueithreshold
> 0	> 0	min(ueithreshold, eithreshold)
> 0	0	ueithreshold
‘dont care’	N	eithreshold

Final threshold computation based on ueithreshold, min_eithreshold and eithreshold values.

Status, Roadmap, and Conclusion

- Implemented the proposed extension in a CVA6-based SoC endowed with an AIA IP.
- Functionally validated the intended behavior by running a set of low-level benchmarks and micro-operations
- Proceeding towards the quantitative evaluation:
  - Quantify PPA-impact on a reference implementation. From initial estimates, this is considered negligible.
  - Quantify the benefit for real-time performance on software stack based on the Bao hypervisor and application benchmarks.
- **We intend to open-source all artifacts to promote collaboration within the RISC-V community.**

Universidade do Minho