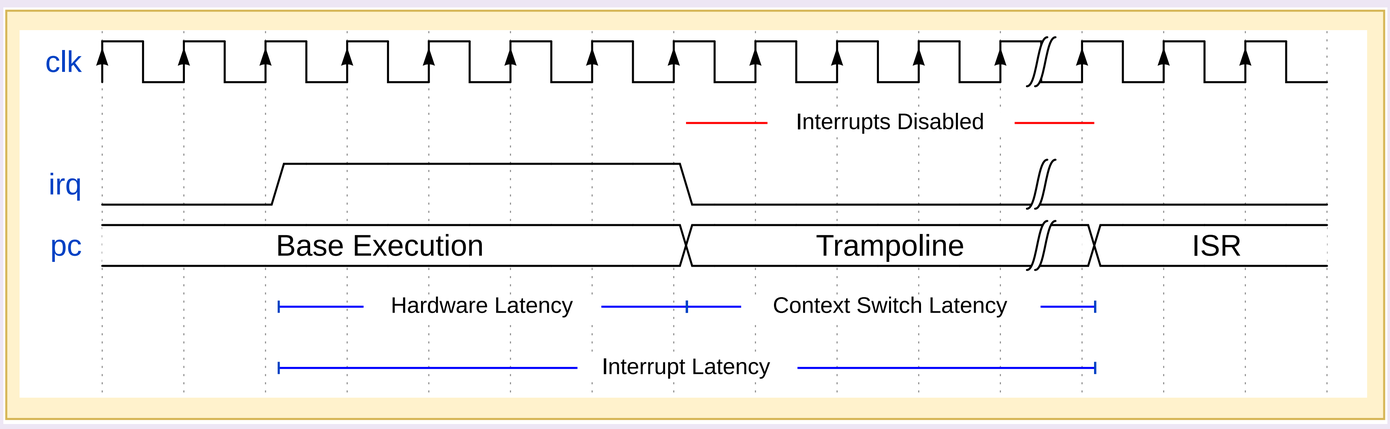
Atalanta: Open-Source RISC-V Microcontroller for Rust-Based Hard Real-Time Systems

Antti Nurmi Abdesattar Kalache Henri Lunnikivi

Tampere University, Tampere, Finland
RISC-V Summit Europe 2024

Interrupt Latency

- The interrupt latency of a processor is a fundamental characteristic of real-time performance and is formed by two components:
  - **Hardware Latency** – Initial reaction latency of processor to interrupt request.
  - **Context Switch Latency** – Latency of storing/restoring processor state (register file, CSRs) from/to the stack.



RTIC Framework

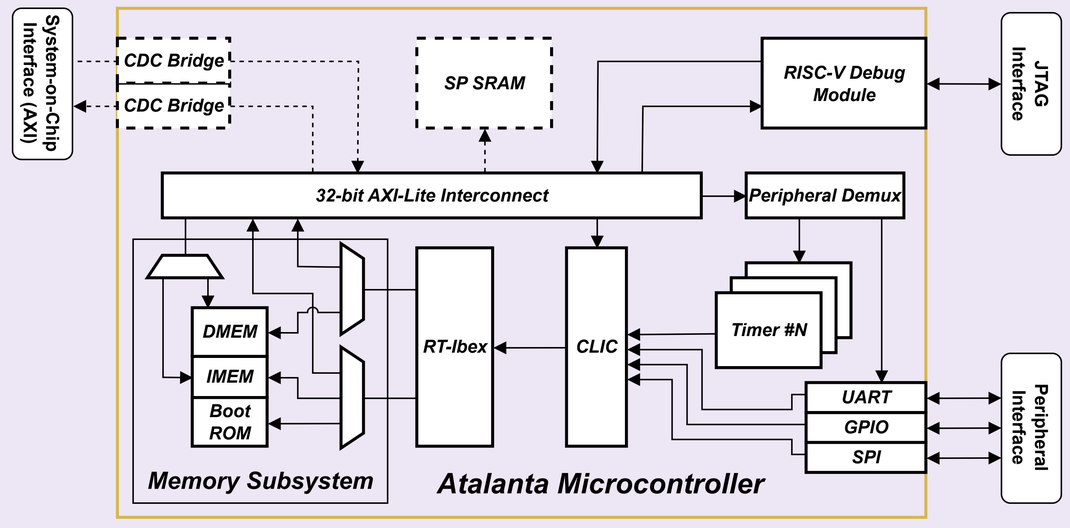
- The *Real-Time Interrupt-Driven Concurrency* Framework (**RTIC**) [1] is a task-based alternative to thread-based frameworks with formal guarantees for
  - *memory safety, absence of data races and defined behavior* due to the Rust language and
  - *deadlock-free execution, single shared stack execution, single dispatch and bounded priority inversion* due to the framework itself.
- **Tasks** are implemented as **interrupt handlers**, thus task switching is as efficient as the interrupt handling mechanism of the hardware without *any* additional overhead.

RT-Ibex

- The open-source *Ibex* processor was extended with a configurable interrupt interface and interrupt-level-based preemption to create the *core-local interrupt controller* (**CLIC**)-compliant *RT-Ibex*.

Microcontroller Architecture

- A minimal, extendable architecture designed for low-latency memory access to accelerate context switching.



Comparison with State-of-the-Art

Microcontroller	ISA	Interrupt Controller	Interrupt Context-Save	ABI	instruction Count	T_a	T_s	Interrupt Latency
Cortex®-M0	ARM	NVIC	Hardware Stacking	AAPCS	9	6	n.a	16
Cortex®-M0+	ARM	NVIC	Hardware Stacking	AAPCS	9	6	n.a	15
Cortex®-M3/4	ARM	NVIC	Hardware Stacking	AAPCS	9	6	9	12
Cortex®-R5	ARM	VIC/GIC	Register Banking	AAPCS	9	n.a	n.a	20
SiFive E21	RISC-V	CLIC	Software Stacking	n.a	n.a	n.a	n.a	20
Balas et al.[2]	RISC-V	CLINT	Software Stacking	EABI	12	6[a]	18	24
Balas et al.[3]	RISC-V	CLIC	HW Stacking + Register Banking	EABI	12	6[a]	18	6[b]
This Work	**RISC-V**	**CLIC**	**Software Stacking**	**EABI**	**12**	**5**	**16**	**21**

[a] Susceptible to jitter from multi-cycle instructions prior to vector table entry fetch.
[b] Interrupt latency can increase in the case of nested interrupts.

Future Work

- **Architectural Optimizations** – clock domain split, improved hierarchy, hardware stacking.
- **Case Studies** – Analysis of real-time performance with formal and empirical methods.
- **ASIC Implementation** – PPA analysis with 22 nm technology node, memory models.

References

[1] RTIC Contributors. *RTIC: The Hardware-Accelerated Rust RTOS*. https://rtic.rs/2/book/en/. 2024.
[2] R. Balas and L. Benini. "RISC-V for Real-time MCUs - Software Optimization and Microarchitectural Gap Analysis". In: *2021 Design, Automation & Test in Europe Conference & Exhibition (DATE)*. 2021, pp. 874–877.
[3] R. Balas, A. Ottaviano, and L. Benini. "CV32RT: Enabling Fast Interrupt and Context Switching for RISC-V Microcontrollers". In: *IEEE Transactions on Very Large Scale Integration (VLSI) Systems* (2024), pp. 1–13.

Tampere University

github.com/soc-hub-fi/atalanta

sochub.fi