A Model-driven development framework for highly Parallel and EneRgy-Efficient computation supporting multi-criteria optimisation

Technology behind the AMPERE SW framework

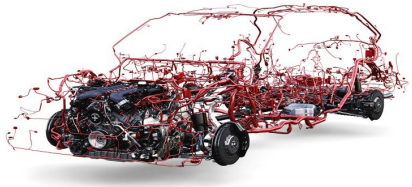
AMPERE Final Event Webinar

Claudio Scordino— Evidence Srl
27 June 2023

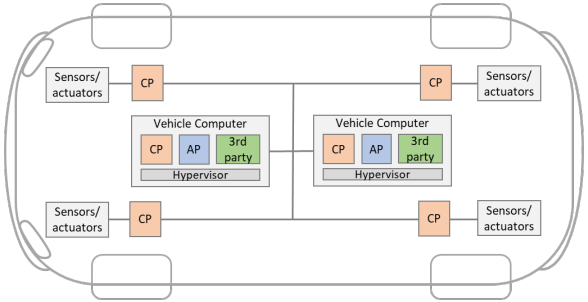
The AMPERE project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 871669

Problem definition

- **The complexity of systems is increasing due to a growing number of functionalities**
  - Example: ADAS and autonomous driving in automotive
  - Other domains face (or expected to face) a similar trend
- **Greater complexity requires more hardware (ECUs)**
  - Modern luxury cars contain more than 100 ECUs
  - Increasing space, weight, power and cost
  - Lack of semiconductors
  - Network cabling inside a vehicle is not tenable
- Automotive is moving towards a zonal architecture with a central vehicle computer



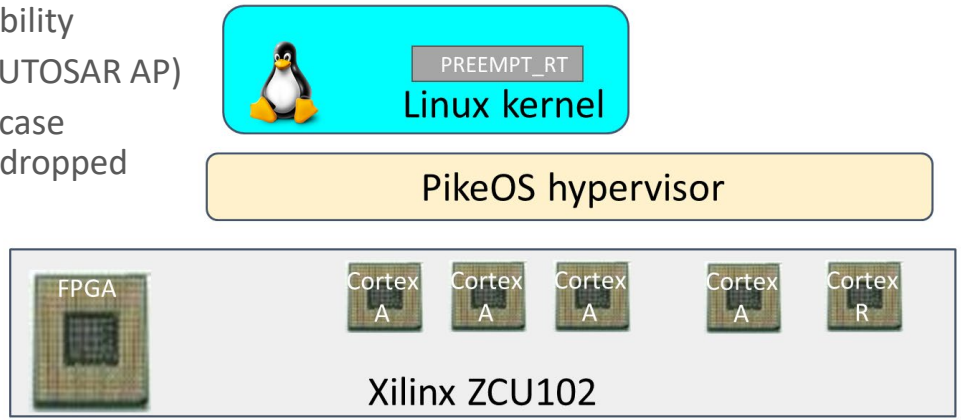
https://www.whichcar.com.au/car-advice/how-much-wiring-is-in-your-car



In AMPERE we have tackled this challenge by integrating functionalities with different levels of criticality on a single software architecture (i.e. mixed-criticality)

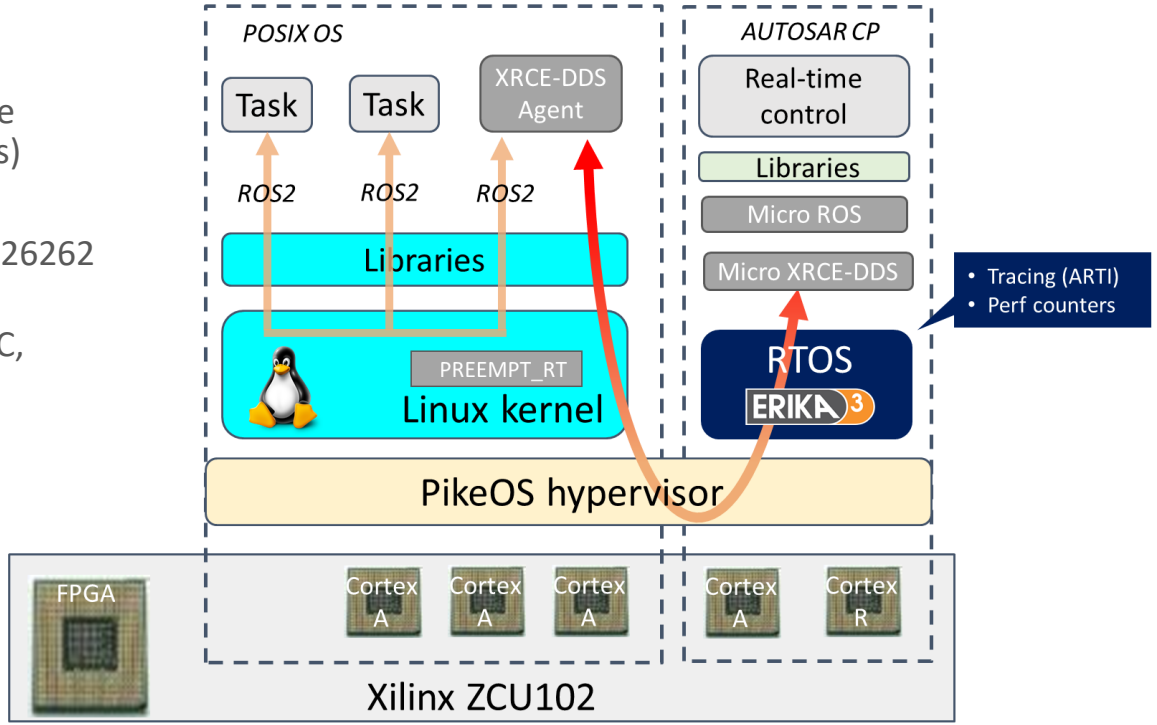
Overall Architecture

- **Flexibile hardware selected: Xilinx ZCU102**
  - Quad-core Cortex-A53
  - Dual-core Cortex-R5F
  - FPGA
- **PikeOS hypervisor:**
  - Safety through resource partitioning and isolation
- **Linux OS:**
  - Performance, scalability, reliability
  - POSIX OS (used by ROS and AUTOSAR AP)
  - PREEMPT_RT to lower worst-case latencies (worst-case latency dropped to about **1%** of original value)



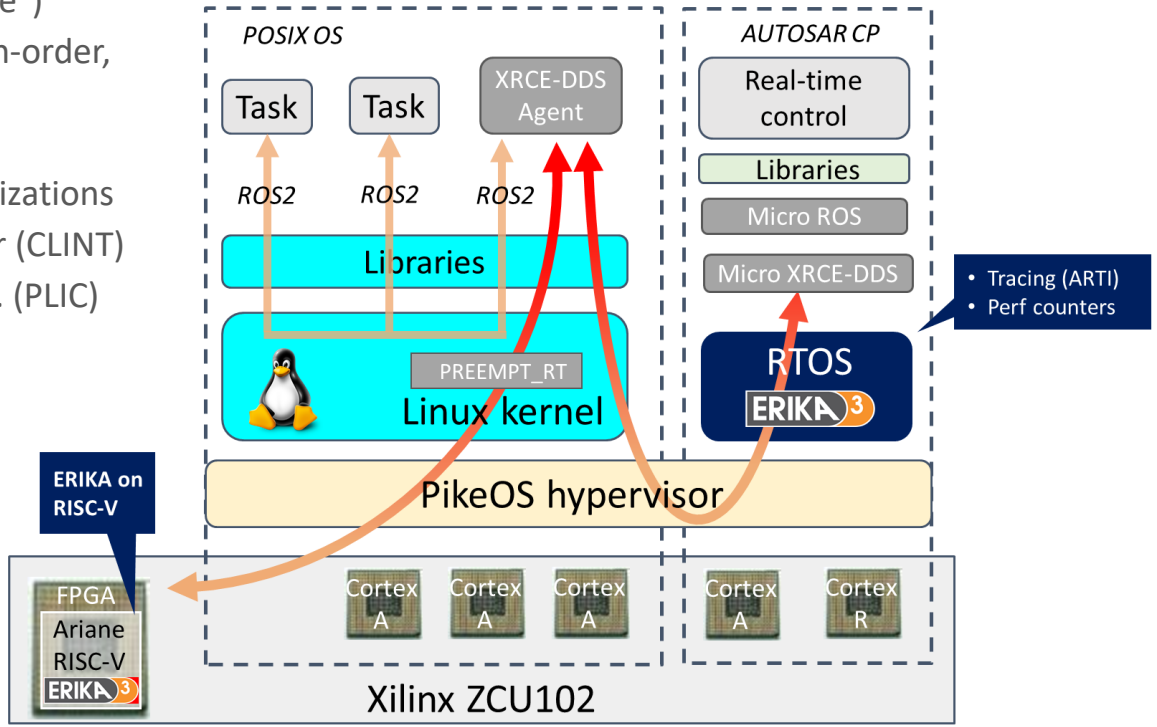
Overall architecture

- **Mixed-criticality** architecture
- **AUTOSAR Classic RTOS:**
  - ERIKA Enterprise RTOS
  - RTOS designed for automotive Electronic Control Units (ECUs)
  - Minimal footprint (few KBs)
  - Certifications: OSEK/VDX, ISO26262 ASIL-D
  - Reference standards: MISRA-C, AUTOSAR Classic OS
  - More information later on
- **Inter-domain communication:**
  - Micro-ROS and DDS-XRCE



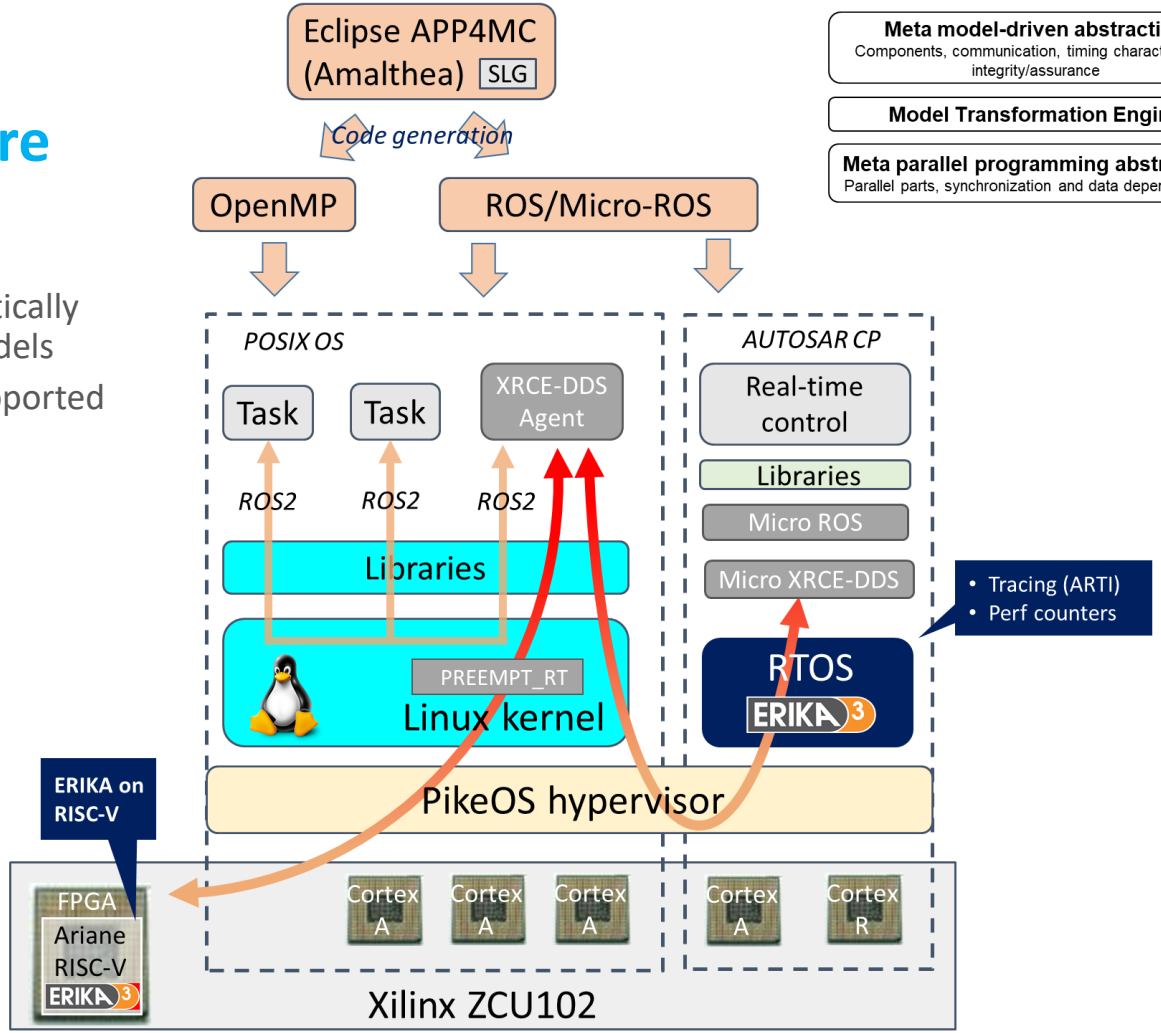
Overall architecture

- **Soft-core RISC-V:**
  - Open-source CVA6 ISA ("Ariane")
  - 64 bits, 6-stage, single issue, in-order, branch predictor
  - Synthesized on FPGA
  - Hardware and software optimizations
  - Core Local Interrupt Controller (CLINT)
  - Platform Local Interrupt Contr. (PLIC)
- **Ported and optimized ERIKA RTOS**
- **Micro-ROS communication**



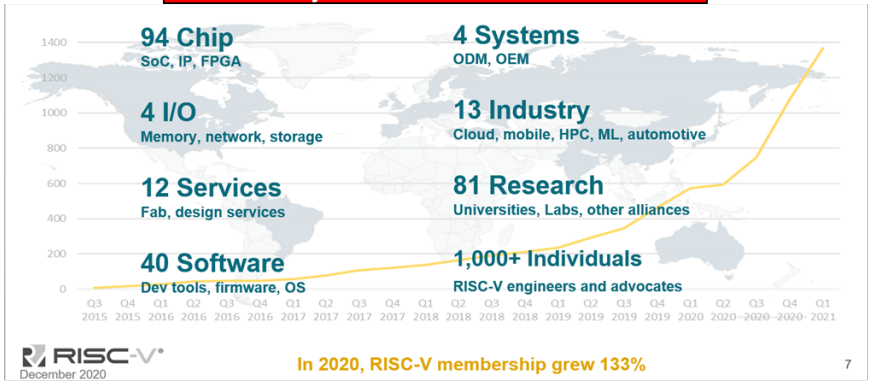
Overall architecture

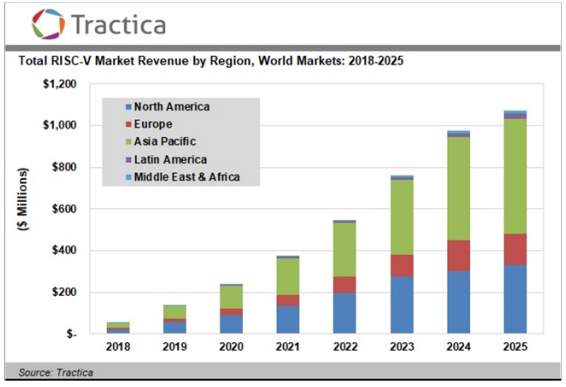
- **Model-based design:**
  - Communication code automatically generated from Amalthea models
  - Both ROS2 and Micro-ROS supported

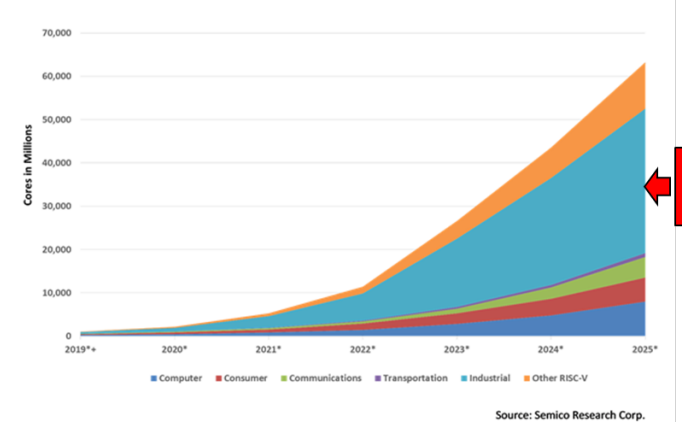


Why we also considered RISC-V

Trend: already 2000+ members across 70 countries



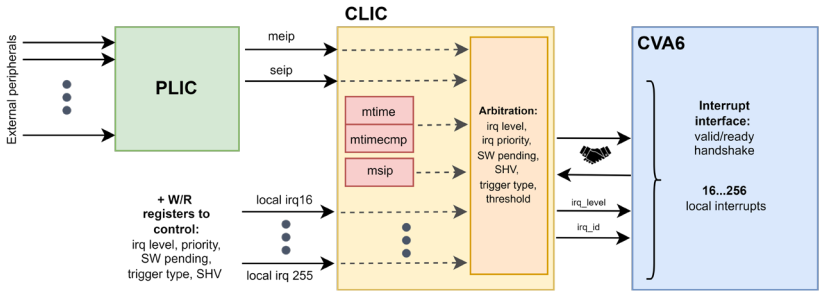


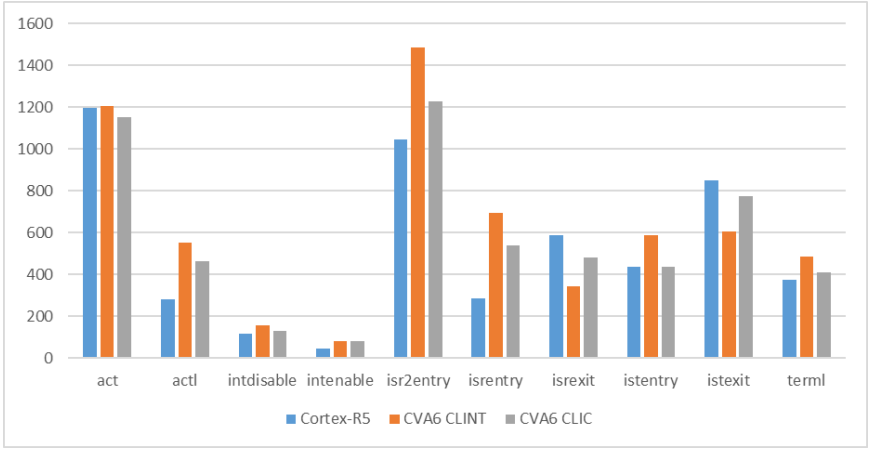


Large adoption expected for trasportation

Hardware and software optimizations on RISC-V

- **Software optimizations:**
  - ISR type 2 is not activated as a task, reducing activation overhead
  - Use of supported hardware interrupt priority level
  - Support for CLIC interrupt controller
- **Hardware optimizations:**
  - ETHZ added Core Local Interrupt Controller (CLIC)

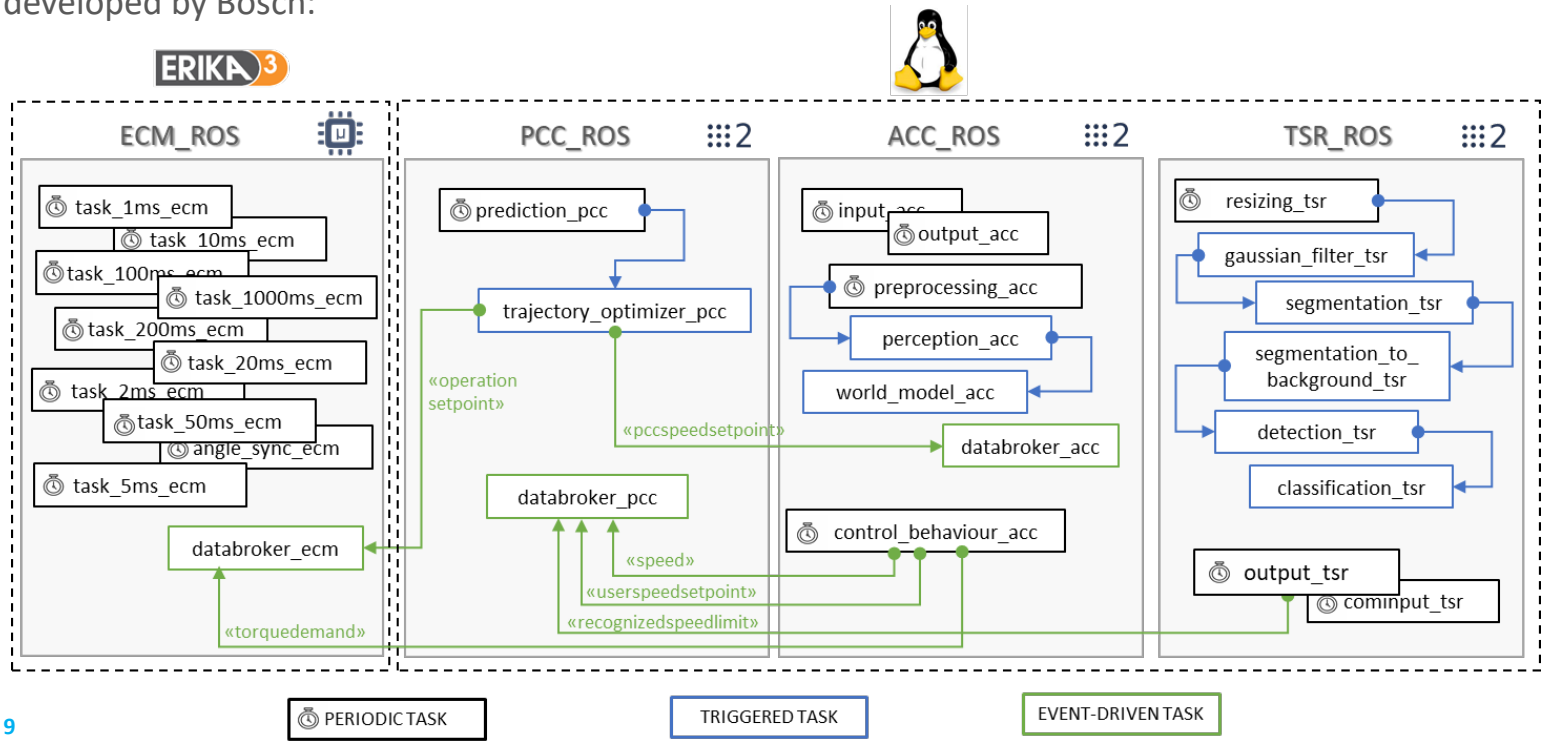




Full details in the following paper (soon on IEEE):
L. Cuomo, C. Scordino, A. Ottaviano, N. Wistoff, R. Balas, L. Benini, E. Guidieri, I.M. Savino, ***Towards a RISC-V Open Platform for Next-Generation Automotive ECUs***, 11th International Conference on Cyber-Phyisical Systems (CPSIoT), Budva, Montenegro, June 2023.

Automotive use-case: predictive cruise control

- From Amalthea model we generated and executed the Predictive Cruise Control developed by Bosch:



www.ampere-euproject.eu

The AMPERE project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 871669