ISTITUTO
DI TECNOLOGIE DELLA
COMUNICAZIONE,
DELL'INFORMAZIONE
E DELLA
PERCEZIONE

Scuola Superiore
Sant'Anna

Activity on embedded systems
Scuola Superiore Sant'Anna

Giorgio Buttazzo, Marco Di Natale

Retis
Real-Time Systems Laboratory

Retis
Real-Time Systems Laboratory

The RETIS Group Since 1996

It includes 30 people:

- 2 Full professors
- 2 Associate professor
- 3 Assistant professors
- 5 Post Docs
- 5 Research associates
- 13 PhD students

G. Buttazzo
RETIS Coordinator

M. Di Natale

G. Lipari

T. Cucinotta

E. Bini

M. Marinoni

P. Pagano

G. Franchino

A. Giantomassi

A. Parri

M. Falcitelli

M. Petracca

A. Ruscelli

G. Cecchetti

C. Salvadori

F. Aderohunmu

I. Barsanti

Mission of the RETIS Lab

- Increase software predictability through suitable
  - Operating systems mechanisms
  - Design methodologies and tools
  - Timing and performance analysis
- Provide real-time support for new computing platforms (multi-core, distributed, cloud, FPGA, heterogeneous…)
- Make embedded systems resource efficient (w.r.t. time, memory, bandwidth, energy, …)
- Prevent and manage overload conditions through adaptive behavior.

Research Topics

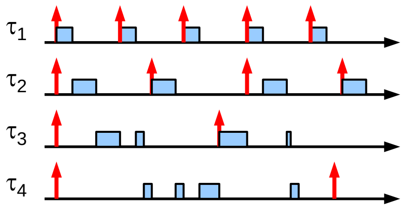
Real-Time Systems Laboratory

Resource management

Efficiency

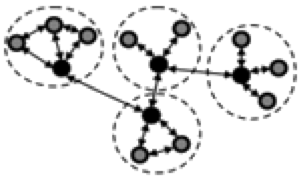
Real-Time scheduling

Timing Analysis



Real-Time Communication

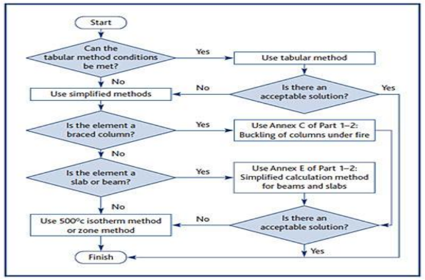
Protocols



Real-Time Systems Laboratory

Design methods and Tools

Optimization

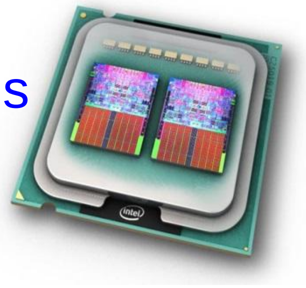

Start
Can the tabular method conditions be met?
Yes
Use tabular method
No
Use simplified methods
No
Is there an acceptable solution?
Yes
Is the element a braced column?
Yes
Use Annex C of Part 1–2: Buckling of columns under fire
No
Is the element a slab or beam?
Yes
Use Annex E of Part 1–2: Simplified calculation method for beams and slabs
No
Use 500°c isotherm method or zone method
No
Is there an acceptable solution?
Yes
Finish

Adaptive behavior

Real-time Control

Real-Time Operating Systems

Multicore & heterogeneous platforms

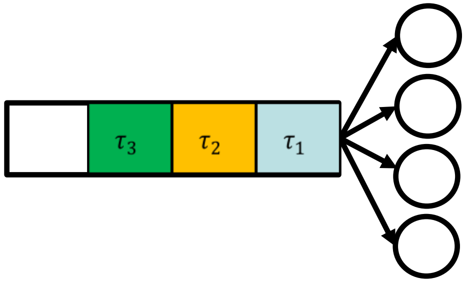


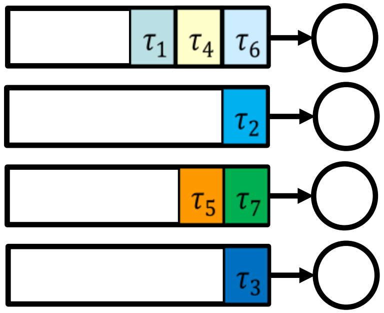
Research on Multiprocessor scheduling

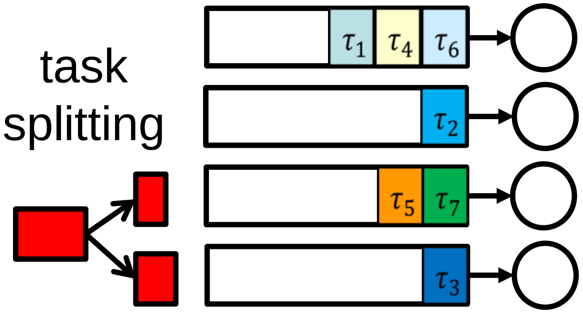
Retis

Real-Time Systems Laboratory

Semipartitioned scheduling







Global Scheduling

Full migration

Partitioned Scheduling

No migration

Semi-Partitioned Scheduling

Only some tasks migrate

Efficient, but too costly and complex to analyze

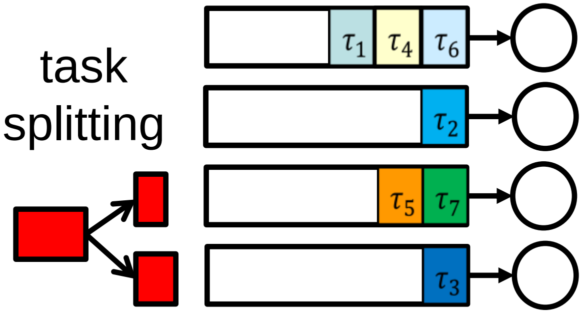
Low overhead and easier to analyze, but too inefficient

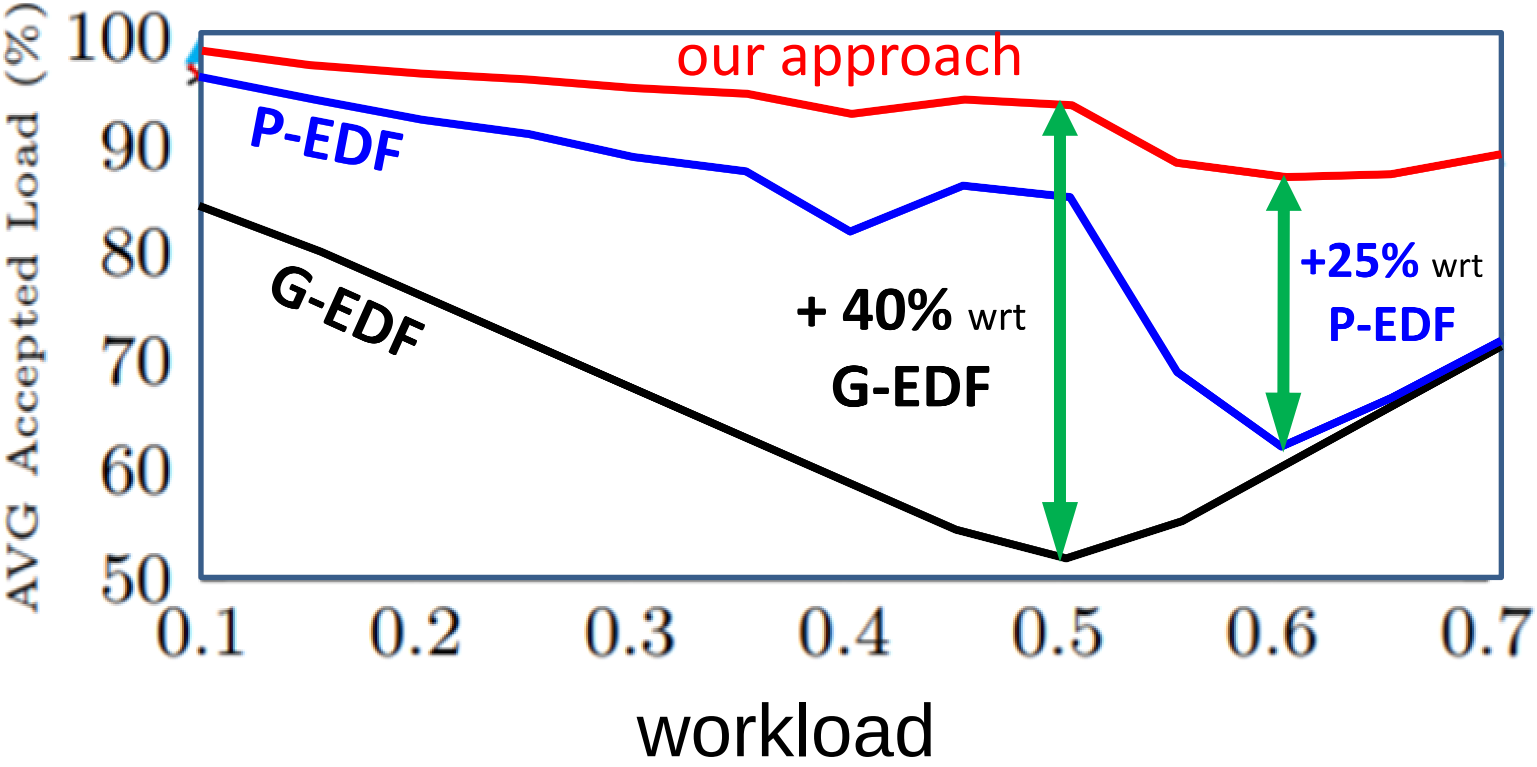
Combines all the advantages, but

- Assuming a-priori knowledge of the workload
- The analysis is highly complex

Semipartitioned scheduling

- Use approximate methods to simplify the analysis
- Assuming dynamic workload and no a-priori knowledge



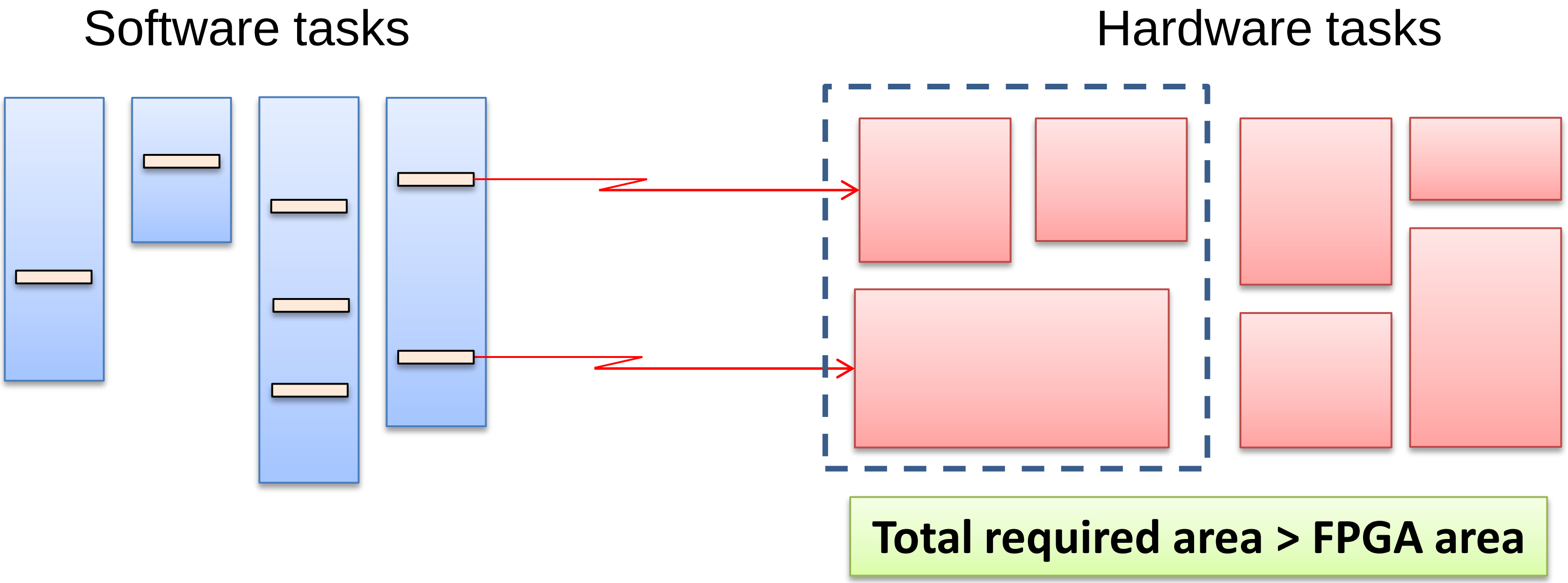


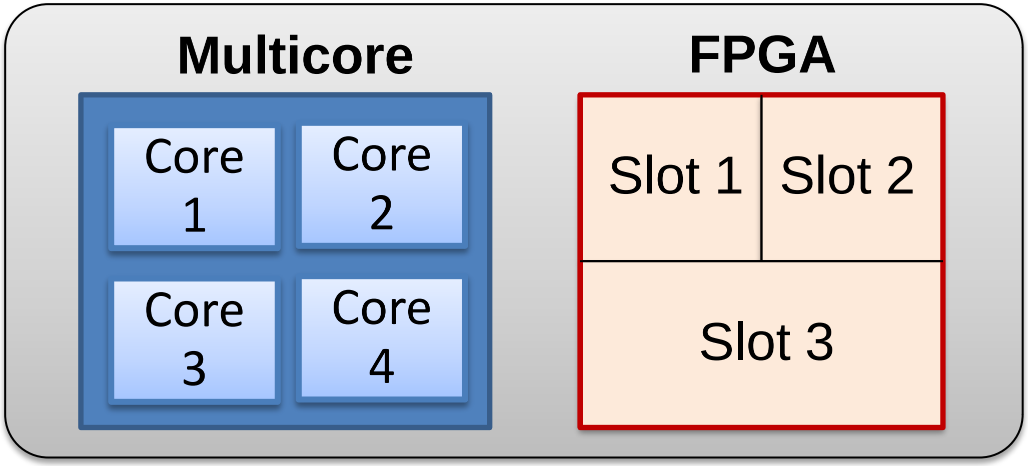
Research on Heterogeneous platforms

Retis

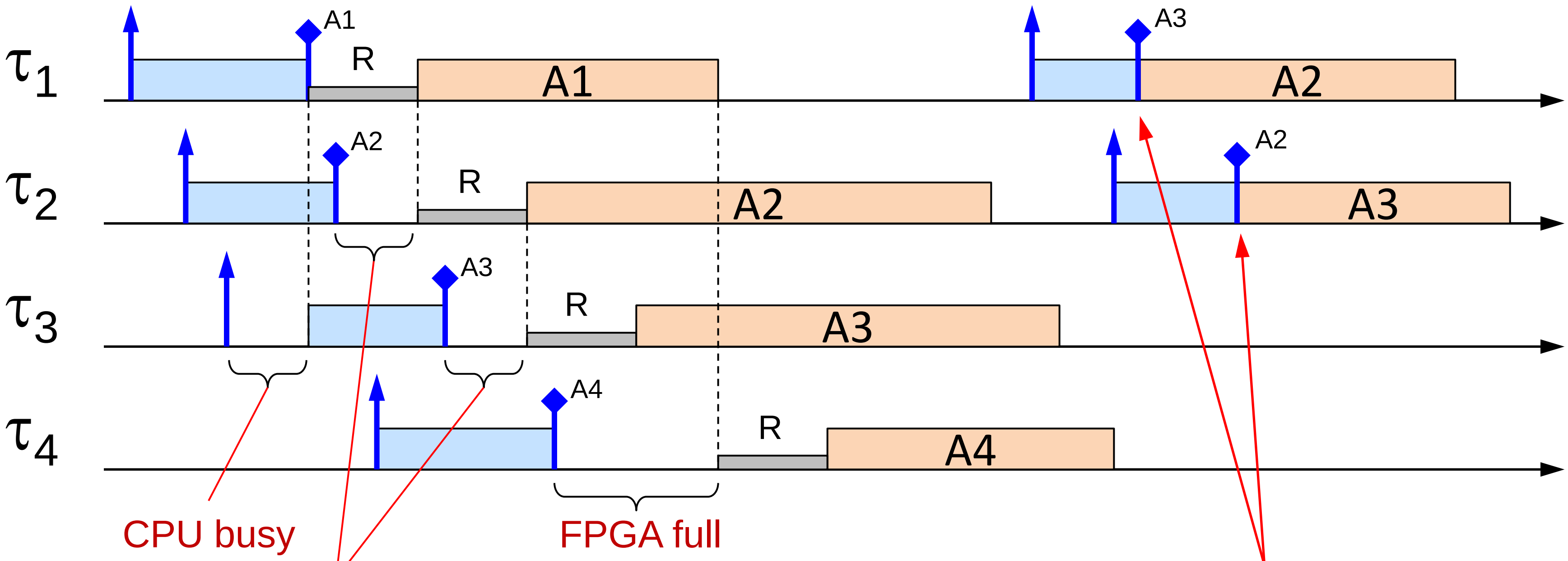
Real-Time Systems Laboratory

Sharing FPGA by DPR





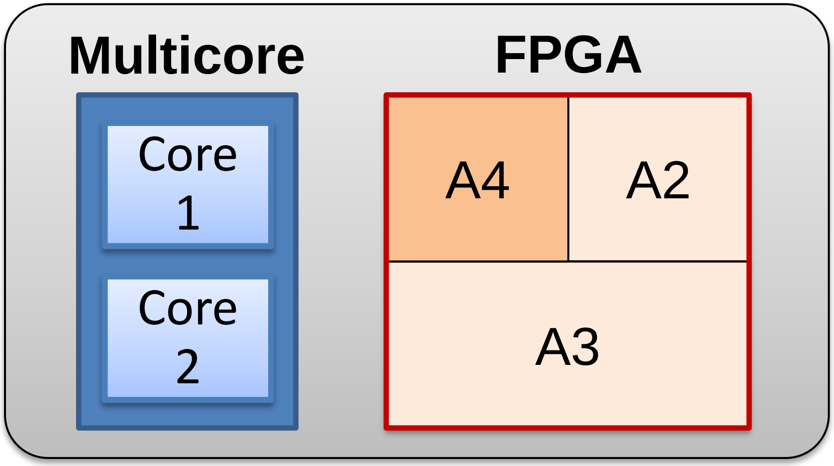
Sharing FPGA by DPR



CPU busy

FPGA full

FRI busy



No delay since the slots are already programmed

Contributions

- A scheduling framework for hardware tasks the guarantees a predictable behavior (bounded delays);
- Anaysis of worst-case response-time bounds;
- Design a feasible partition of the FGPA into slots as a function of the task set;
- Implement a preemptive reconfiguration interface.
- Provide a kernel support of the framework on Linux.

Research on Hypervisors

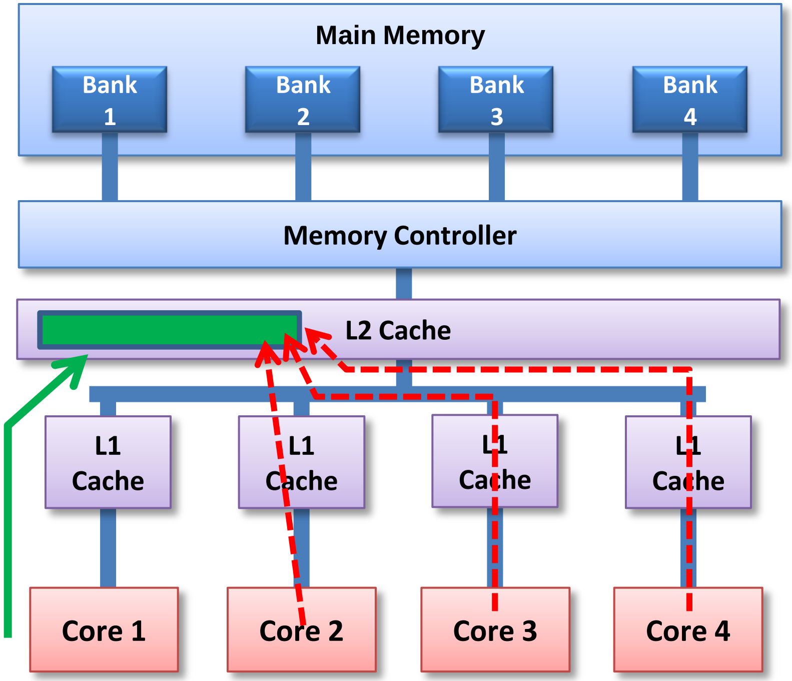
Retis

Real-Time Systems Laboratory

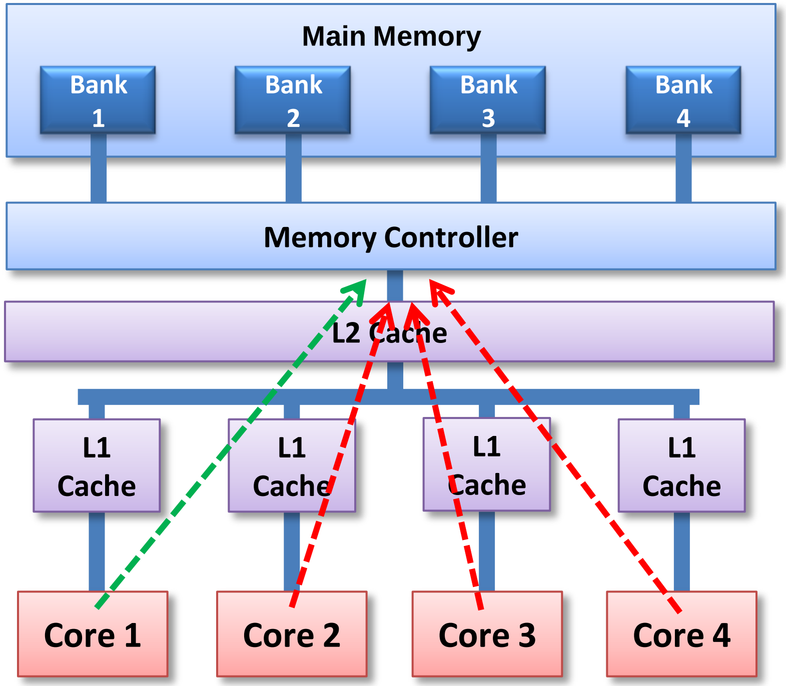
Cache and memory contention

Applications running in parallel on different cores can incur in highly-unpredictable interference due to cache and memory bandwidth contention

Interference in last level cache

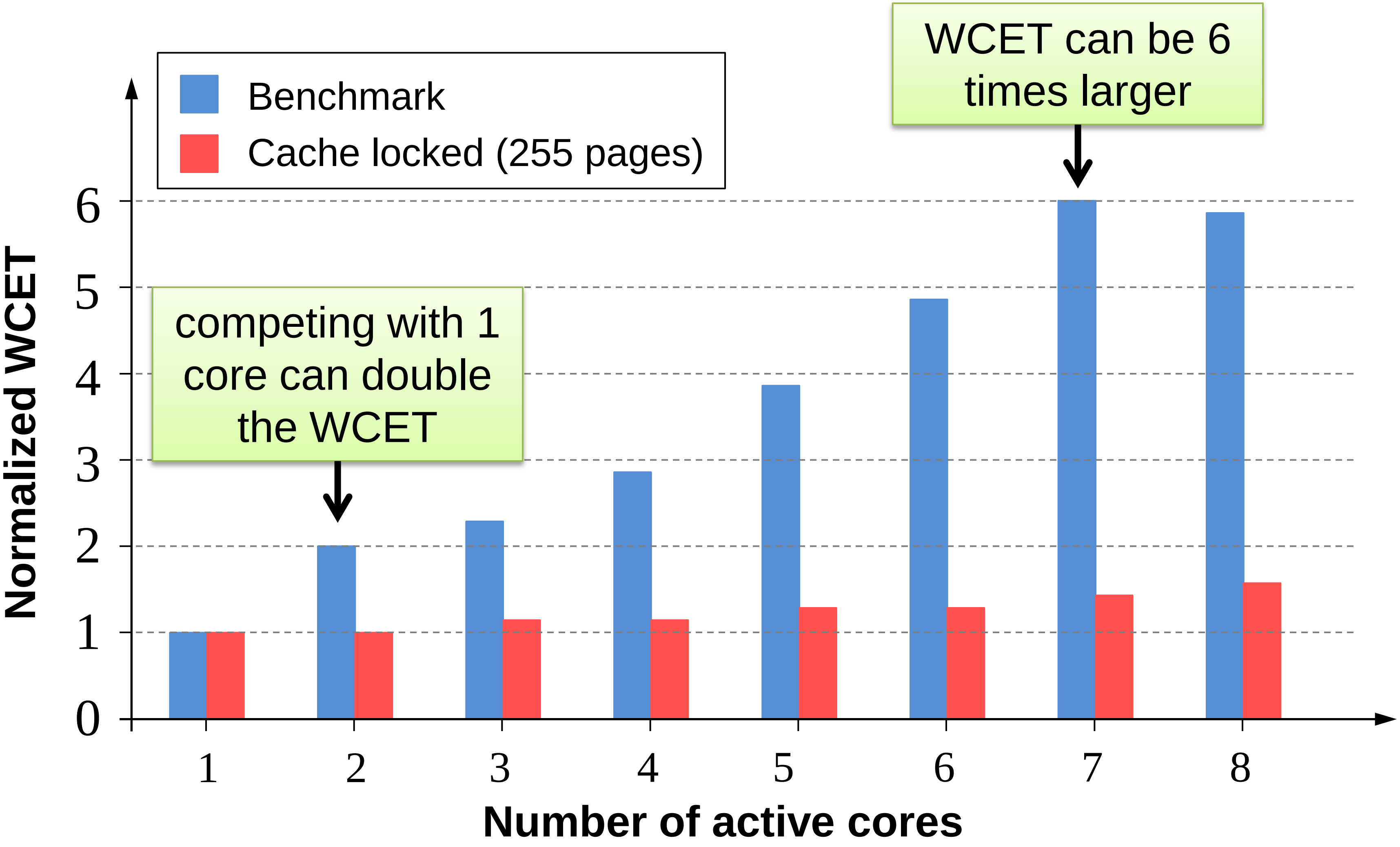


Interference at memory controller



The WCET issue

Test by Lockheed Martin Space Systems on 8-core platform

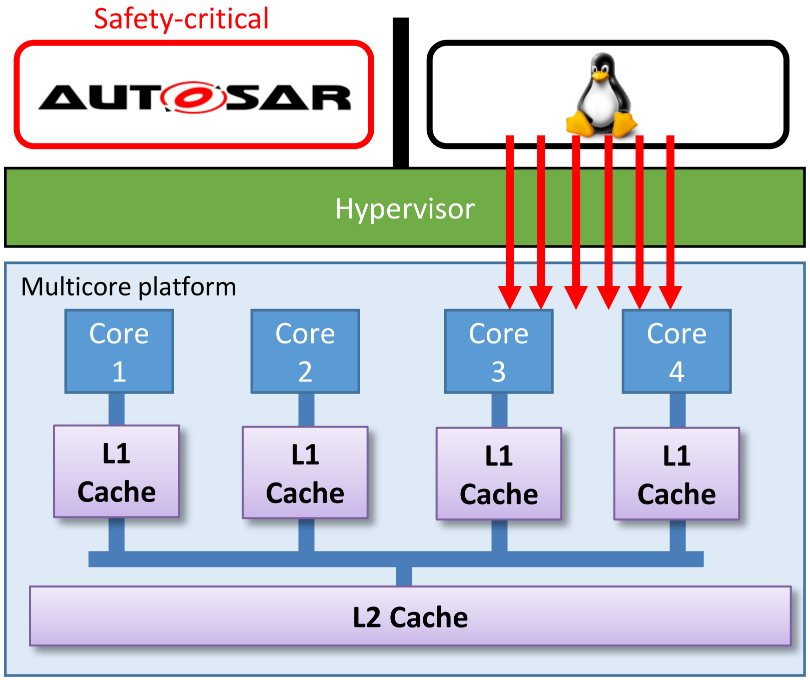


competing with 1 core can double the WCET

WCET can be 6 times larger

Need for isolation

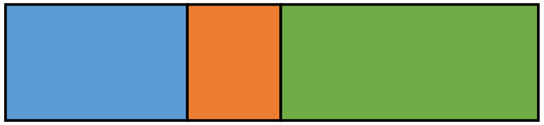
- Due to resource contention, non-critical applications (e.g., multimedia) can delay safety-critical tasks.
- What if the Linux host starts flooding the system with memory transactions (e.g., due to an attack or a malfunctioning)?



Proposed solution

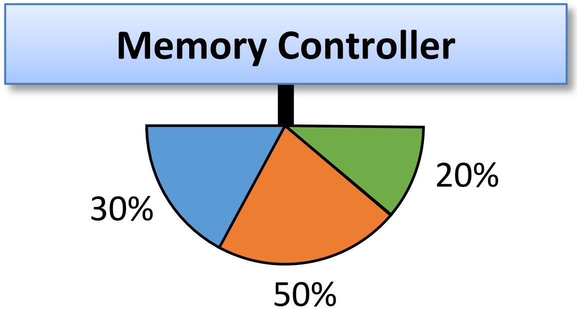
- Hypervisor with strong isolation capabilities

Cache coloring



- Strict cache partitioning to avoid interference
- Integrated with virtualization of the address spaces realized by the hypervisor

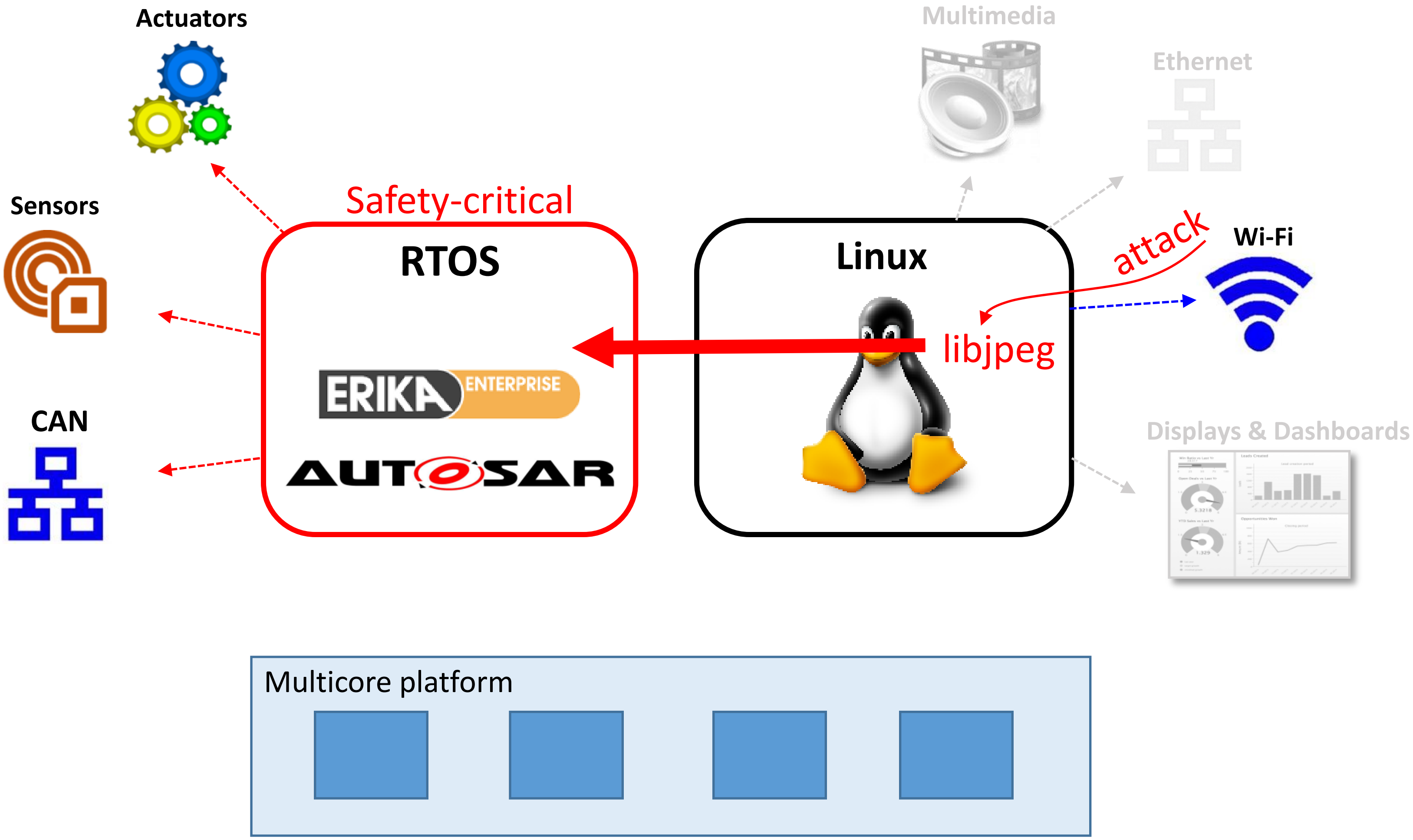
Memory bandwidth reservation



- Guarantees predictable delays in accessing the main memory
- Isolation is implemented for each virtual processor managed by the hypervisor

Security issues

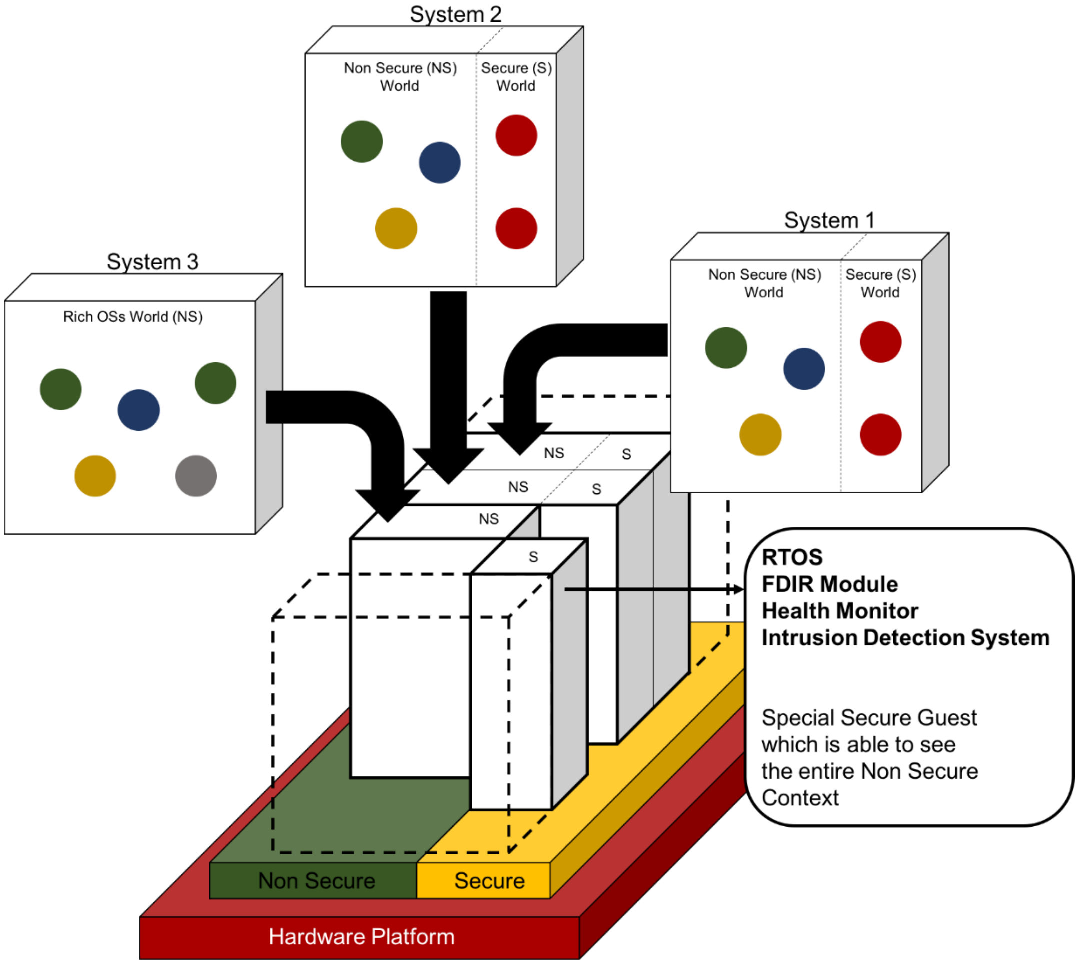
- Multi OS solutions are prone to cyber attacks



Dual-Hypervisor

Dual-hypervisor solution based on ARM TrustZone and ARM Virtualization Extensions

- Allows integrating multiple systems domains with both **secure** and **non-secure** domains
- Full virtualization of trusted execution environment
- By-design isolation of the domains with separated virtualization engines

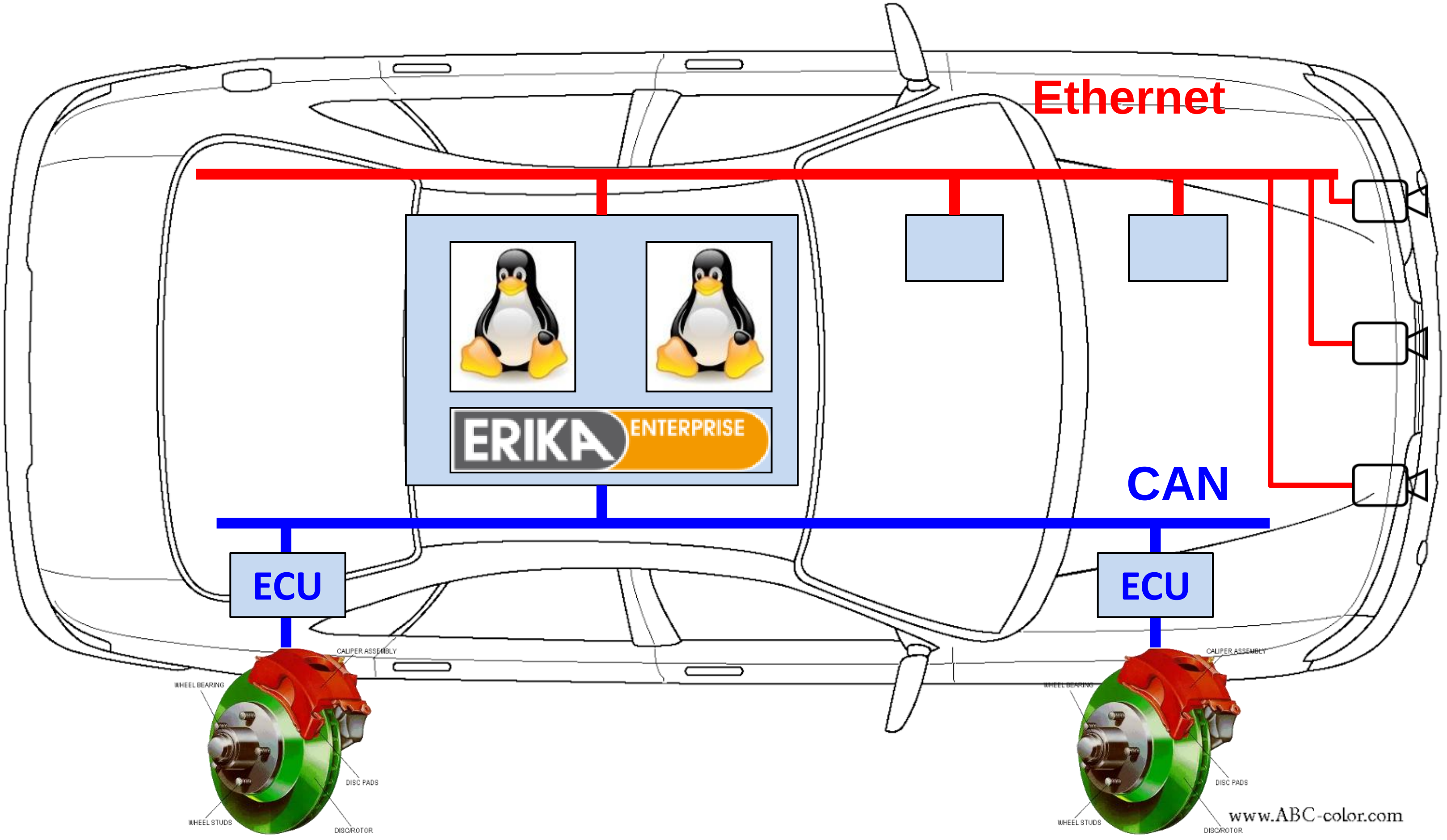


Research on Heterogeneous Networks

Retis

Real-Time Systems Laboratory

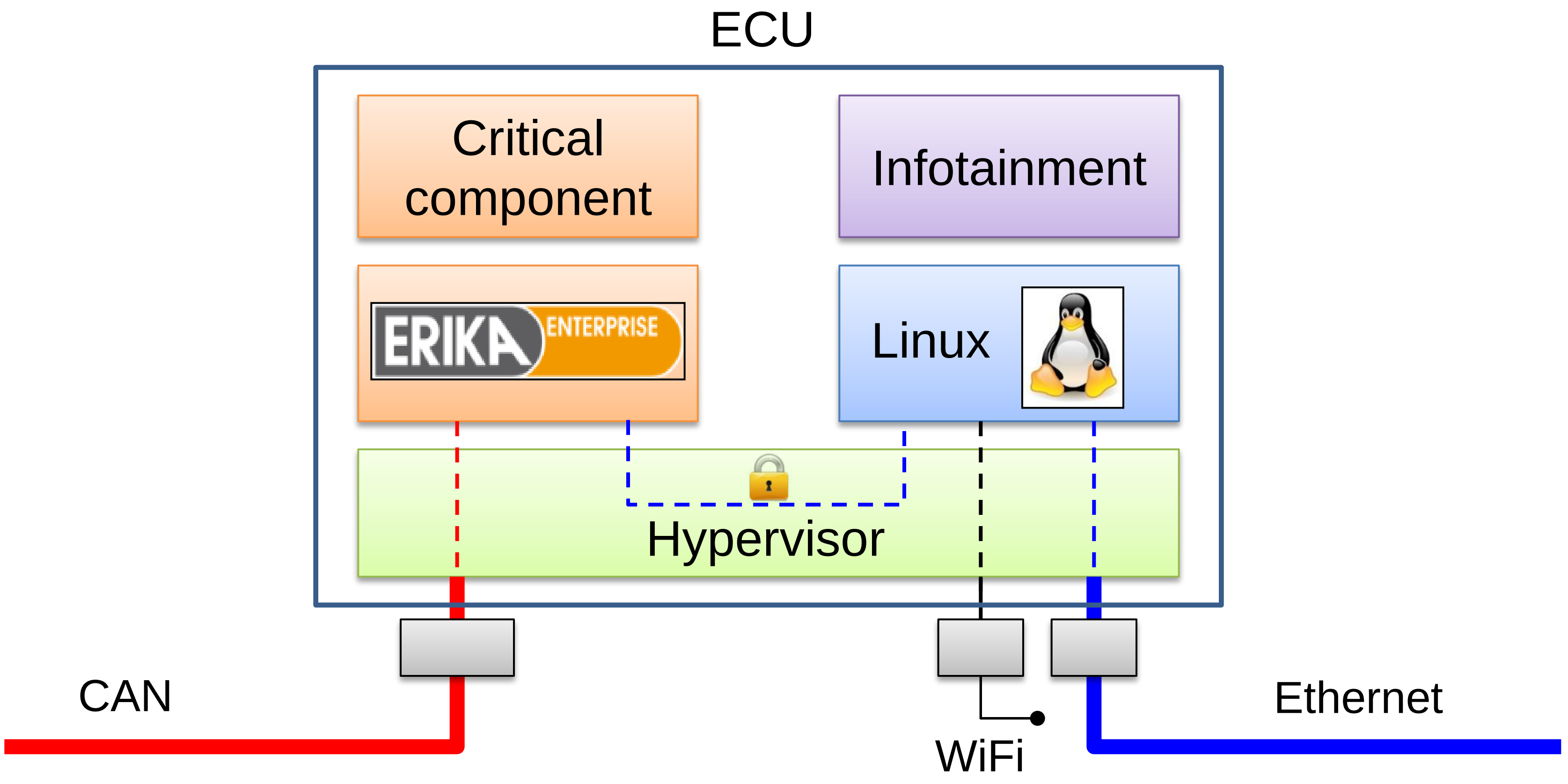
It provides time-critical, predictable and reliable communication for automotive systes in heterogeneous systems and networks.



www.ABC-color.com

Cyber Security

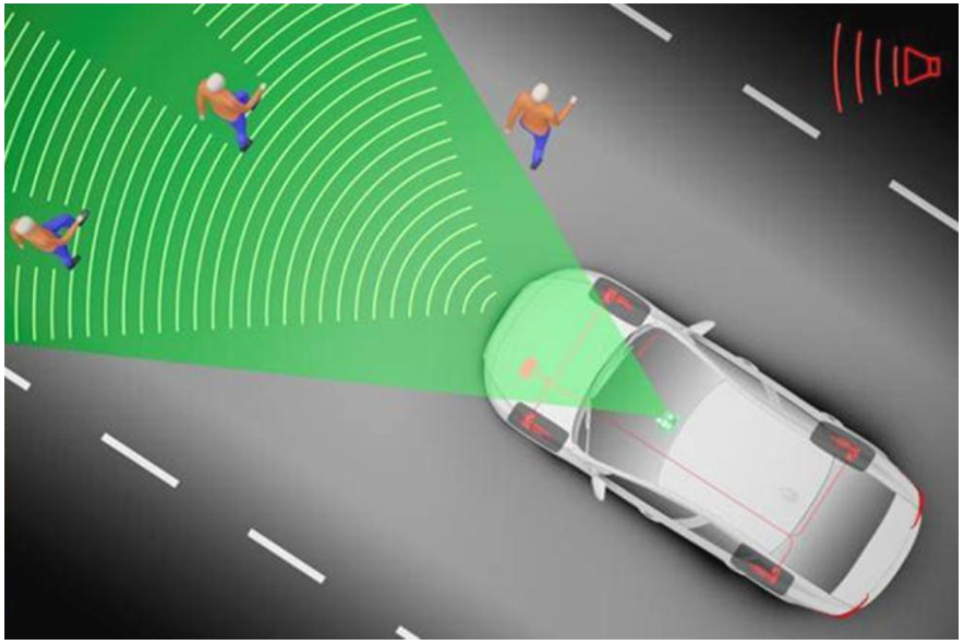
Using a **hypervisor** to **isolate software components** running in the same ECU or Central Gateway:



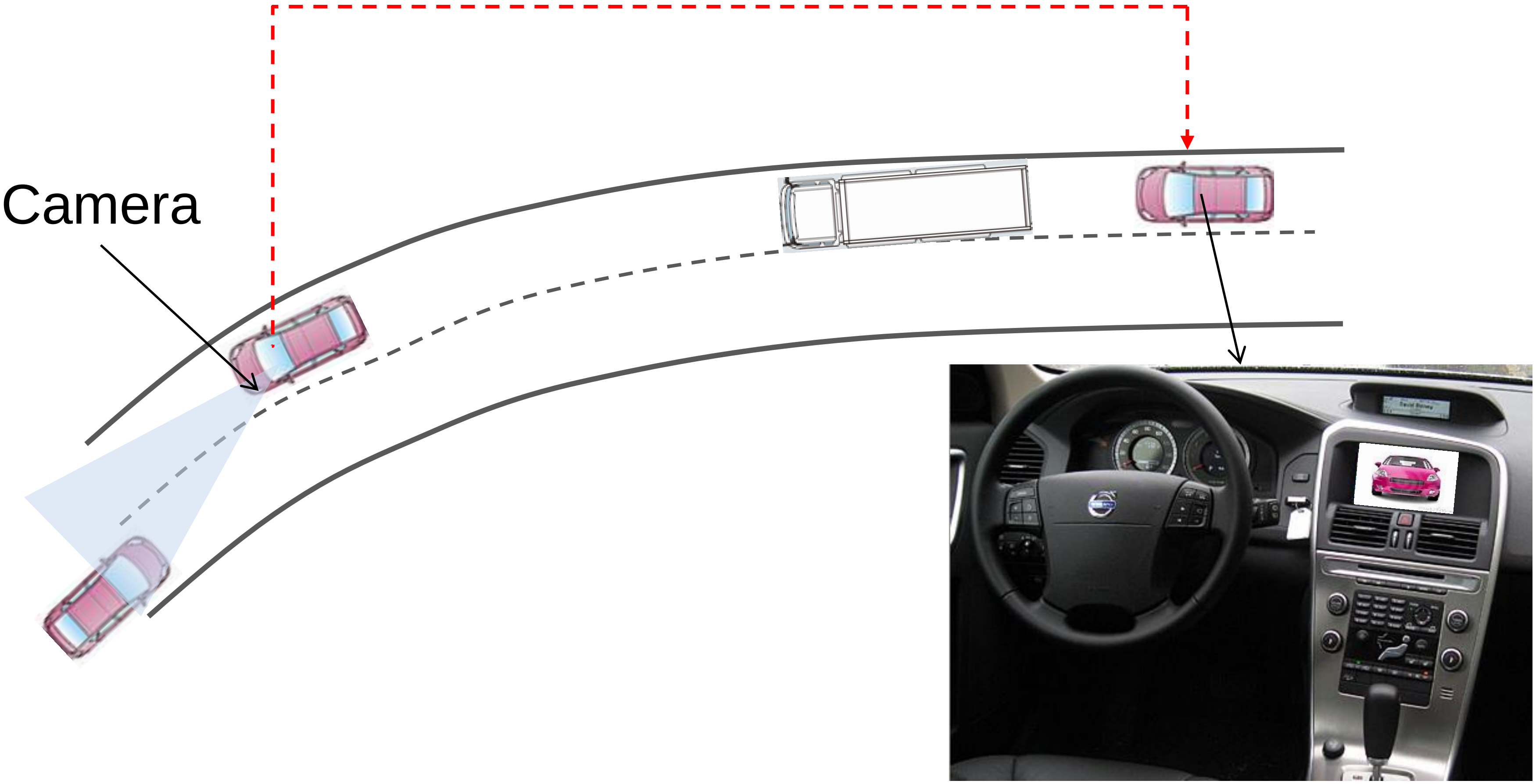
Provide support for

Autonomous driving

Active safety



Active safety: real-time V2V video transmission



live video stream

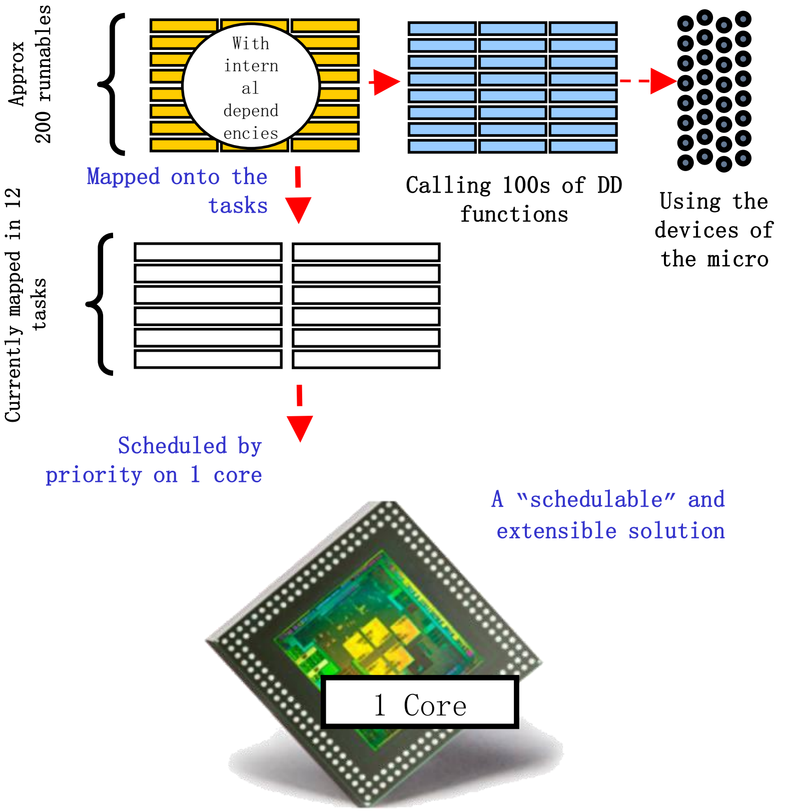
Camera

IHU live video display

Thank you

Design Optimization and Synthesis

- **Optimal deployment on multicores**
  - development of OSEK and then AUTOSAR-compliant **RTOS for multicores** with time predictability
  - Partitioning of functionality on multicore platforms (design with 500 AUTOSAR runnables approx)



Approx 200 runnables

With internal dependencies

Mapped onto the tasks

Calling 100s of DD functions

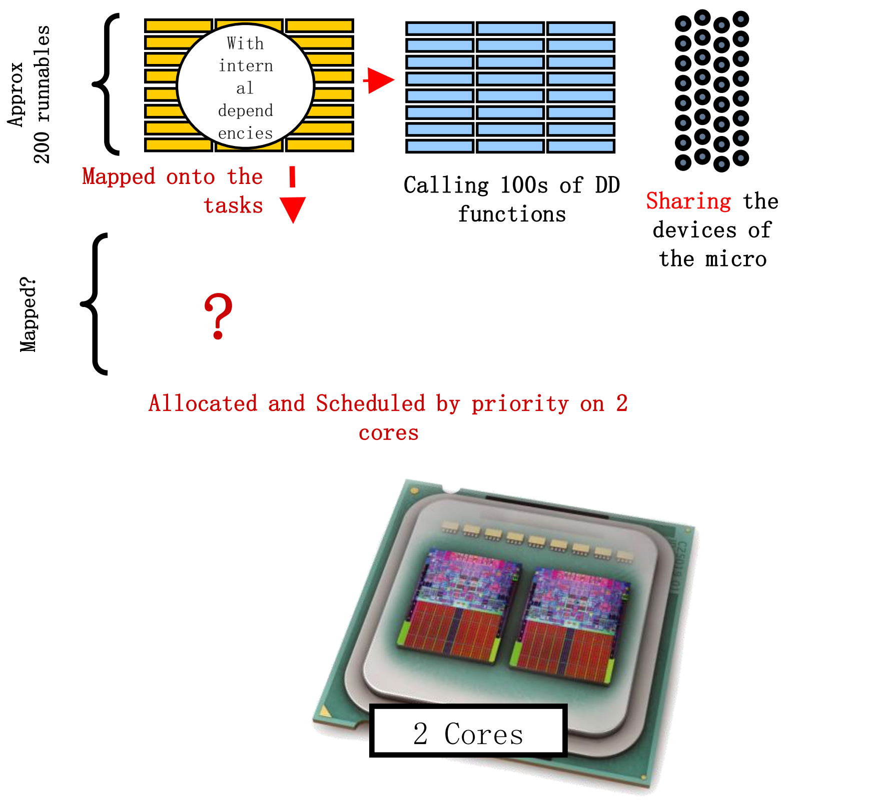
Using the devices of the micro

Currently mapped in 12 tasks

Scheduled by priority on 1 core

A "schedulable" and extensible solution

1 Core



Approx 200 runnables

With internal dependencies

Mapped onto the tasks

Calling 100s of DD functions

Sharing the devices of the micro

Mapped?

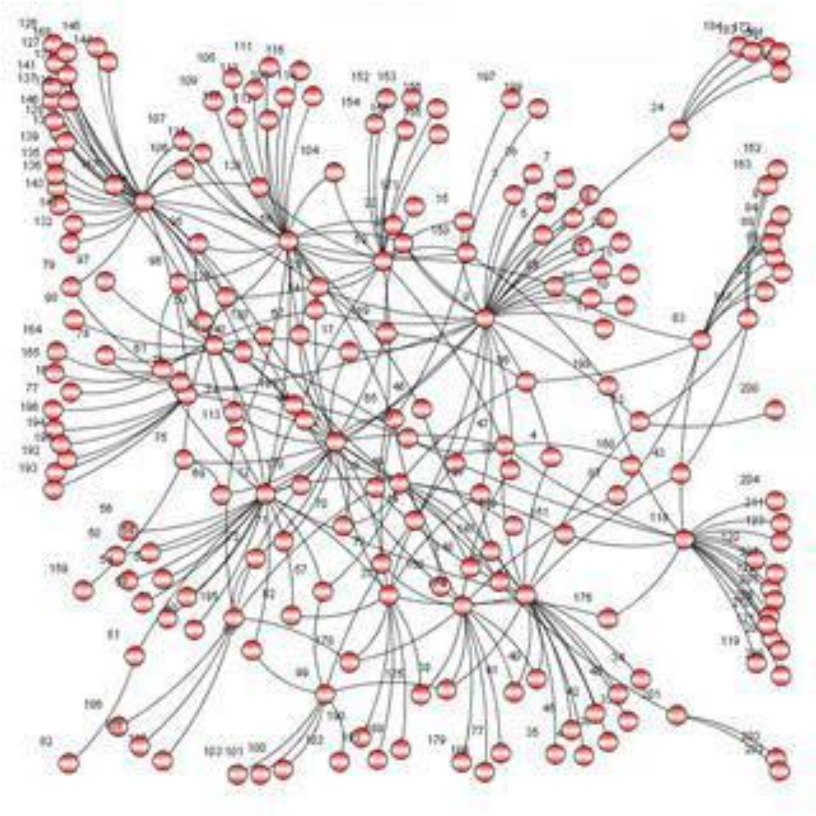
?

Allocated and Scheduled by priority on 2 cores

2 Cores

The problem

Engine control applications are tightly coupled, with a huge number of data and functional dependencies

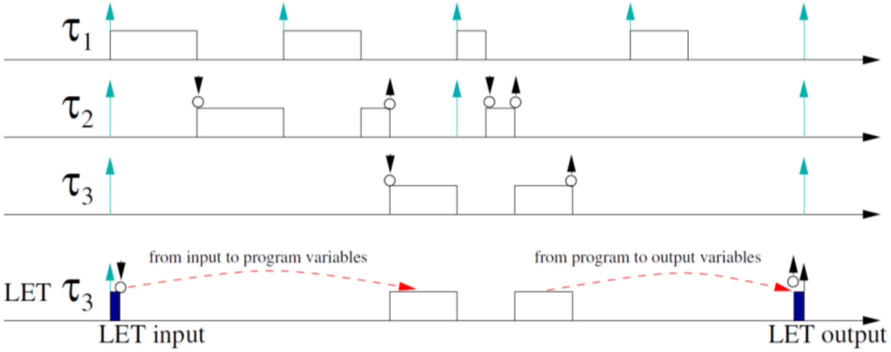


What is the right model (code, Simulink, AUTOSAR?) for extracting the units of allocation (tasks, runnables?)

What are the metrics for allocation (time, extensibility, robustness)

LET on multicores

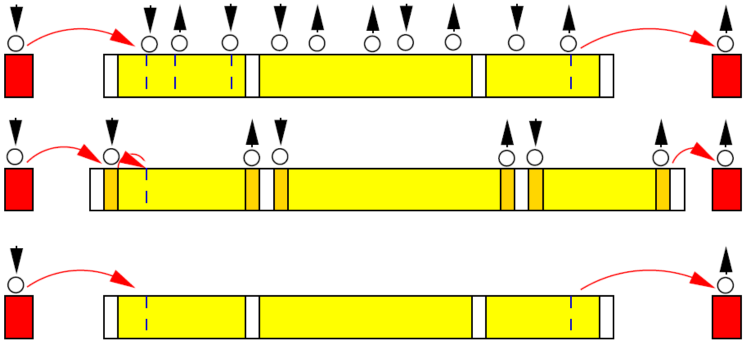
WATERS challenge 2017 …


τ_1
τ_2
τ_3
LET τ_3
from input to program variables
from program to output variables
LET input
LET output

(Kirsch et al ?) Tasks input data at the beginning of their period and output is delayed until the end of the period (trade output jitter for delay)

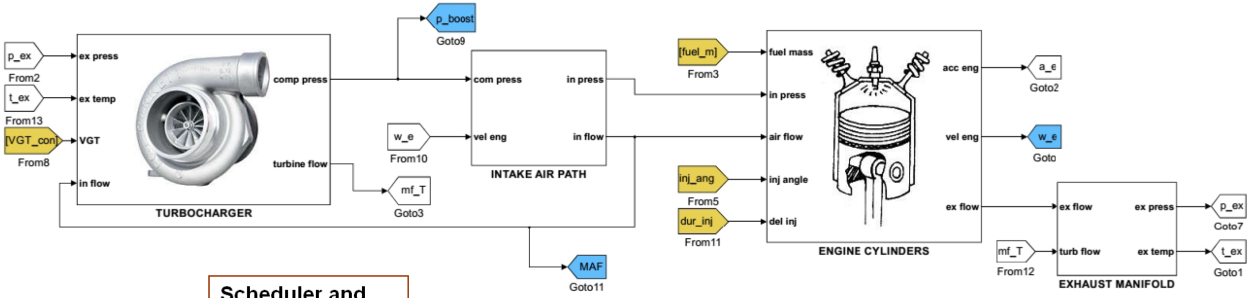
Also improves determinism in the access to memory !

LET also brings similarity with the AUTOSAR RTE immediate communication model

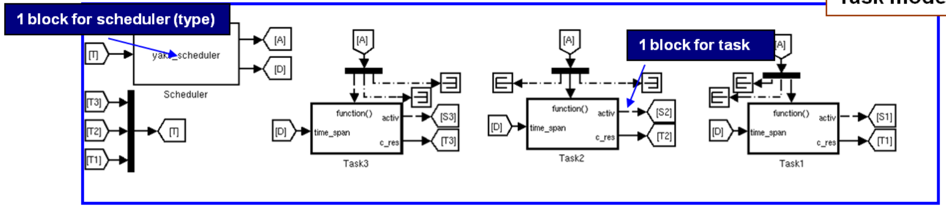


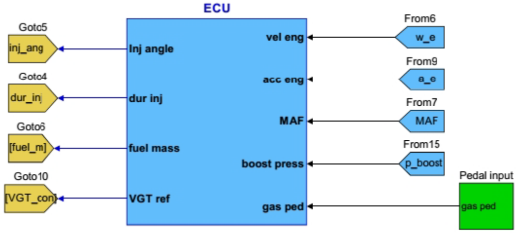
Functional impact of scheduling

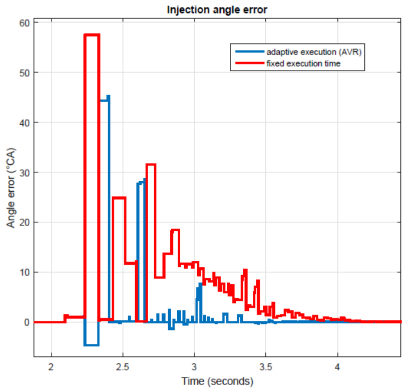
The T-Res project w. Engine model



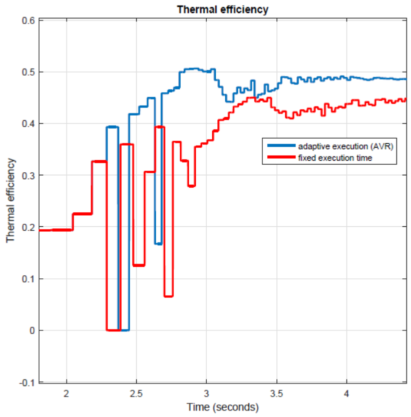
Scheduler and Task model



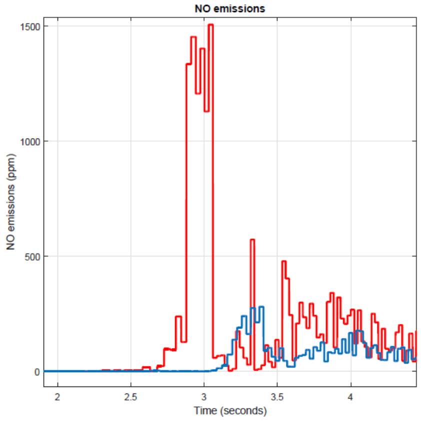




(a) Injection errors



(b) Thermodynamic efficiency

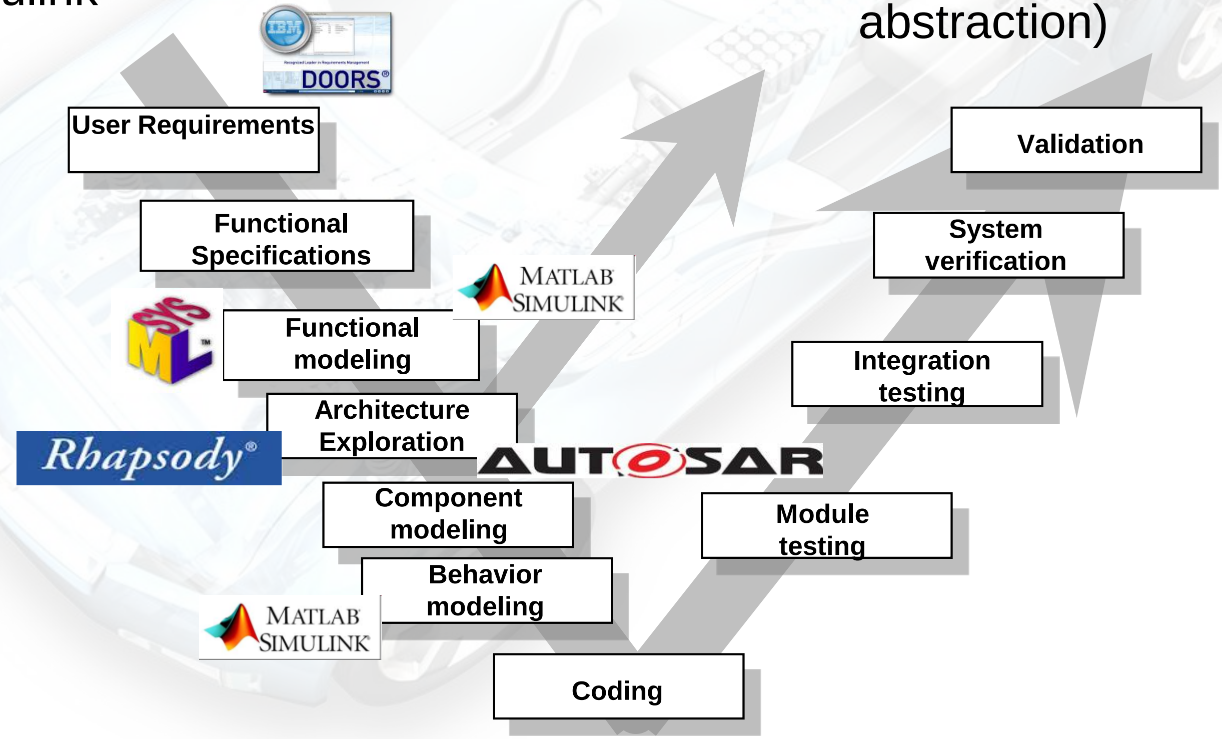


(c) NOx emissions

Other (industrial) projects

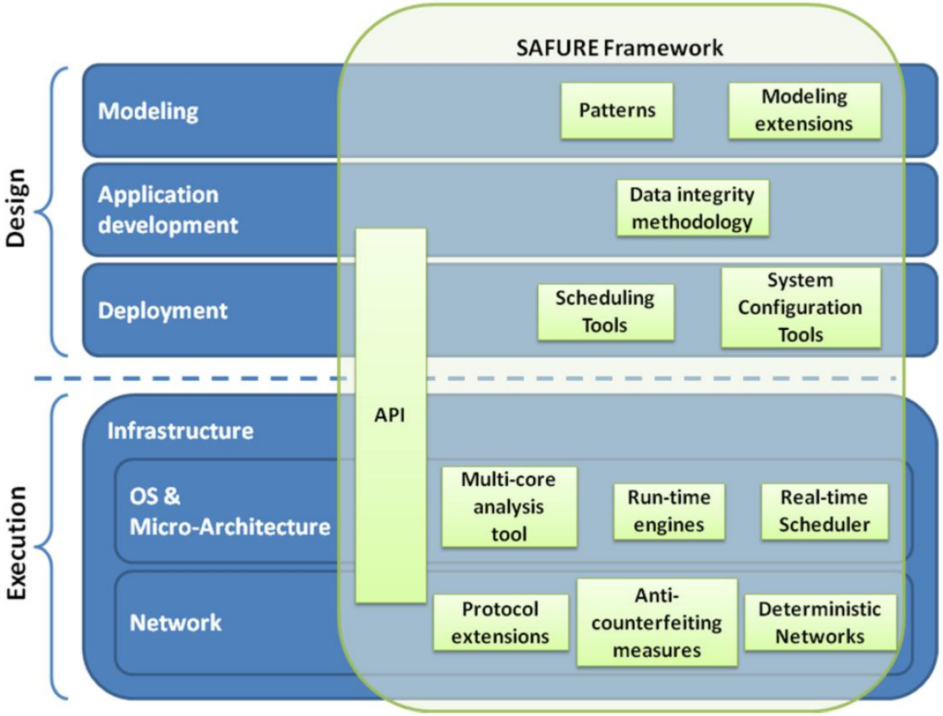
- **Magneti Marelli – Mathworks**
  - **Model-based development,**
  - **Integration of heterogeneous models**
    - UML/SysML – Rhapsody and Papyrus
    - AUTOSAR – Artop
    - Simulink

- **Automatic code generation**
  - Artop/Rhapsody for RTE / mixed criticality / security
  - Simulink for multicores, customized code gen for OS and I/O (customization and abstraction)



Horizon2020 SAFURE – Safety and Security by design for interconnected mixed-critical CPS

- engineering methods for cyber-physical systems using a holistic approach to **safety** and **security by design**.
- tools and capabilities to **prevent attacks in real-time**, keeping critical subsystems in their safety and security boundaries.



- **guidelines** to assist designers and developers during the whole engineering process, addressing safety and security “by design” across all levels.

Automotive Security - Horizon2020 SAFURE

- **guidelines** to assist designers and developers during the whole engineering process, addressing safety and security “by design” across all levels.
- Industrial partners (automotive) Magneti Marelli, Escrypt (Bosch), TTTech
- Objectives and Work at SSSA:
  - Provide AUTOSAR and UML extensions for **modeling Security and Safety Requirements**
  - Provide Methods for the **automatic generation of code using the AUTOSAR CSM (Crypto Service Module) functions** for message encryption at the RTE level
  - Study impact of MAC (Message Authentication Codes) on timing performance and possible **policies for optimized MAC truncation**

More issues !!

- **Seeking partnership for the definition of embedded systems courses**
  - Topics
  - Examples
  - Tools and Methods
  - Projects…