Real-time Safety-Critical applications in MPSoCs

Certification Challenges

Photo by Reuben Wu.

Raytheon Technologies

Juan Valverde

Staff Research Scientist
Embedded Systems (Ireland)

May 11th 2020

Our businesses

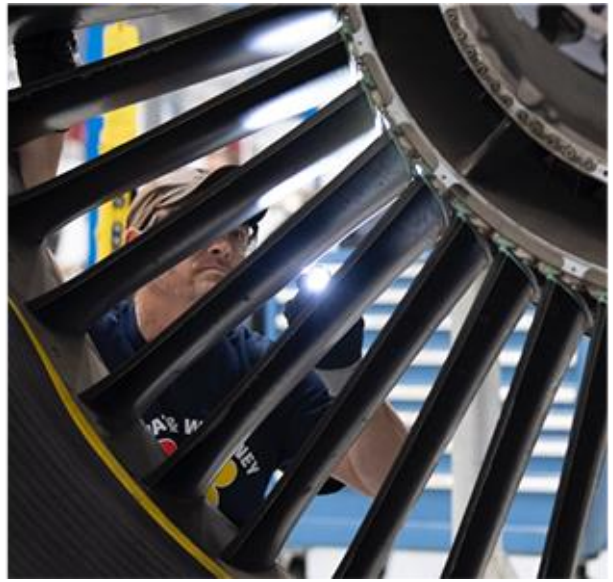
Raytheon Technologies

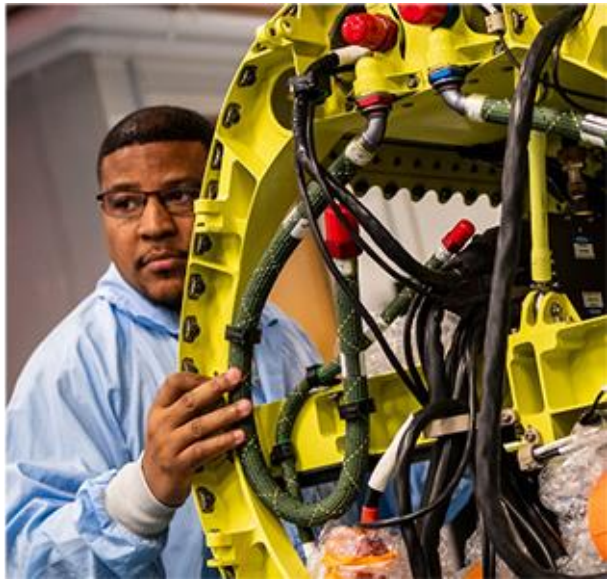
Collins Aerospace

Raytheon Intelligence & Space

Pratt & Whitney

Raytheon Missiles & Defense





Our key capabilities

Actuation, Propeller & Landing Systems

Aerostructures

Aircraft Engines & Auxiliary Power Systems

Avionics

Cybersecurity

Data Analytics

Interiors

Missile Defense

Precision Weapons

Systems Integration & Sensors

Raytheon Technologies Research Centre

Cork, Ireland

Established in 2010, focuses on energy, security and aerospace systems

Berkeley, CA

Established in 2009, focuses on cyber physical systems and embedded intelligence

East Hartford, CT

Founded in 1929, focuses on a broad range of system engineering, thermal, fluid, material, and informational sciences

Rome, Italy

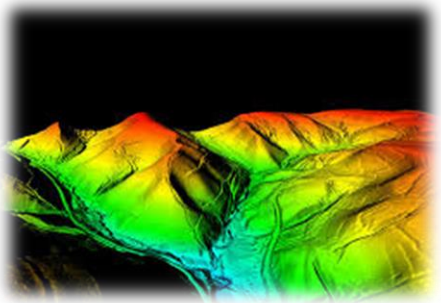
Joined UTC in 2012, focuses on model-based design and embedded systems engineering

Safety-Critical Applications at RTX

RTX is one of the largest suppliers of aerospace systems

- Safety-of-life operation is a critical technology differentiator in RTX
- From Avionics to Engine PHM, **Embedded Systems are a critical part of our products**

e.g. Vehicle Management Computer for rotorcraft, fixed-wing and UAS



- Will feature triple multi-core processors, high-speed communications and open architecture for use in high-redundancy flight critical applications
- Higher processing capability will enable fly-by-wire and autonomous flight

e.g. Situational awareness for autonomous operations

- Heavy use of image processing and sensor fusion for 3D environment reconstruction, obstacle detection, etc.
- More autonomy, more criticality!

e.g. Run-Time PHM of Engines

- Monitoring is critical
- Instrumentation limited by physical constraints: space, temperature, etc.

Most applications are safety-critical and have hard real-time requirements

Challenges Imposed by Certification

THERE ARE MANY SOLUTIONS IN THE STATE OF THE ART FOR THESE PROBLEMS…BUT, CAN THEY BE CERTIFIED?

Integration, Performance and SWAP-C

Mostly forced to use COTS

- **Lack of information**
- **Engagement with manufacturer**
- **Difficult guaranteeing isolation**
- **Almost impossible WCET estimation**

COTS + SW Layers

- Platform Usage Domain, performance limitation
- Hypervisors: virtualization, security layers
- Asymmetric Redundancies

Domain Specific Architectures

- ASIC or static FPGA design
- Very good performance…but difficult and expensive
- Improved verifiability but poor adaptability

COTS IPs + Domain Specific Architectures

- Better balance performance vs. dependability
- Rely on well-known architectures with expansions
- Enhanced visibility: Security and Safety

You can do many things today, certifying them for aerospace… is a different story

Current Solutions for Safety-Critical ES

MOST SOLUTIONS ARE EITHER COTS-BASED OR DOMAIN SPECIFIC

- E.g. Vehicle Management Computer for rotorcraft, fixed-wing and UAS: 3 asymmetric commercial multicores, with different HALs.
- E.g. Motor Control Systems for actuation very frequently use dedicated Flash-based FPGAs with dedicated control architectures, redundant or not.
- E.g. Display controls include commercial GPUs and SoCs but the level of criticality is not maximum, if so, they are supported by co-processors like FPGAs.

but…

… for instance a standalone **GPU** performing a critical task is difficult… kernel co-scheduling?

… how do you ensure time determinism in a **COTS multicore**? How do you enforce it?

… how long does it take to fully design your system in an **FPGA**? Who does that?

… which is the best programing model for **heterogeneous** solutions?

… how can we decrease **V&V** overhead as complexity increases?

You can do many things today, certifying them for aerospace… is a different story

MASTECS

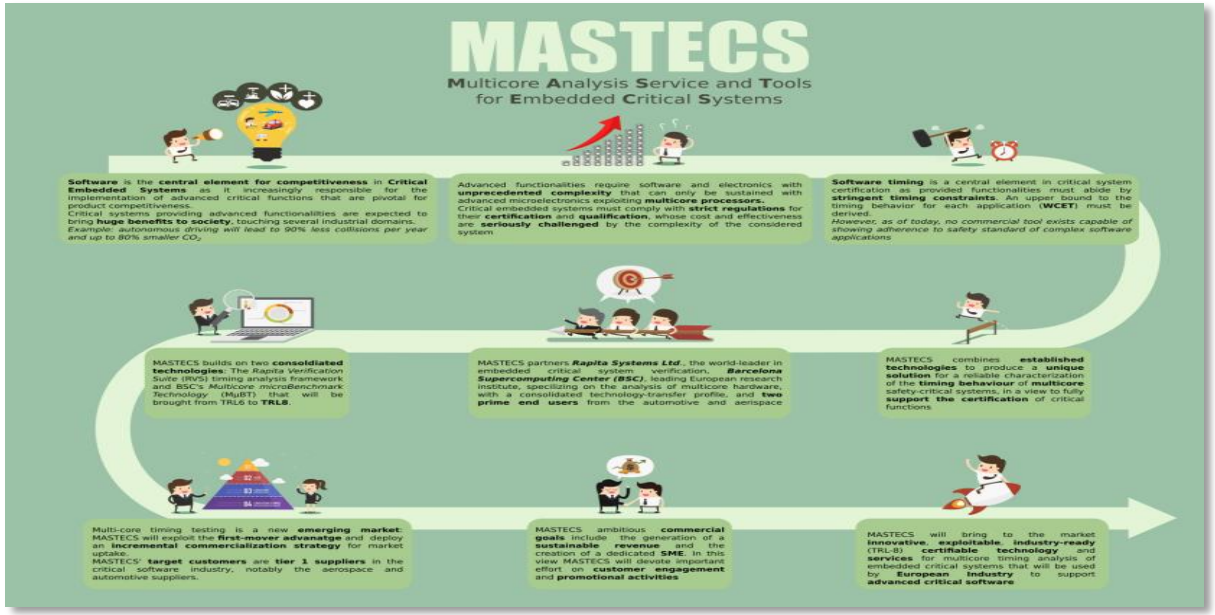
*Multicore **A**nalysis **S**ervice and **T**ools for **E**mbedded **C**ritical **S**ystems*

About the Project

Call: Fast Track to Innovation (FTI) H2020-EIC-FTI-2018-2020 ***Coordinator:*** Barcelona Super Computing Centre - **Consortium**: 4 Partners
Duration: 2 years ***Start date:*** Jan 2019

Objectives

1. Reduce time to market for avionics and automotive HW/SW providing more information and in a faster way to the certification authorities.
2. Support the introduction of more computing intensive tasks (e.g. to increase autonomy) in safety critical applications.
3. Create a methodology and set of tools to address timing non-determinism from the initial stages of the design process.



MASTECS Consortium

Barcelona Supercomputing Center
Centro Nacional de Supercomputación

Raytheon Technologies

RAPITA SYSTEMS LTD

MAGNETI MARELLI

Raytheon Role & Value

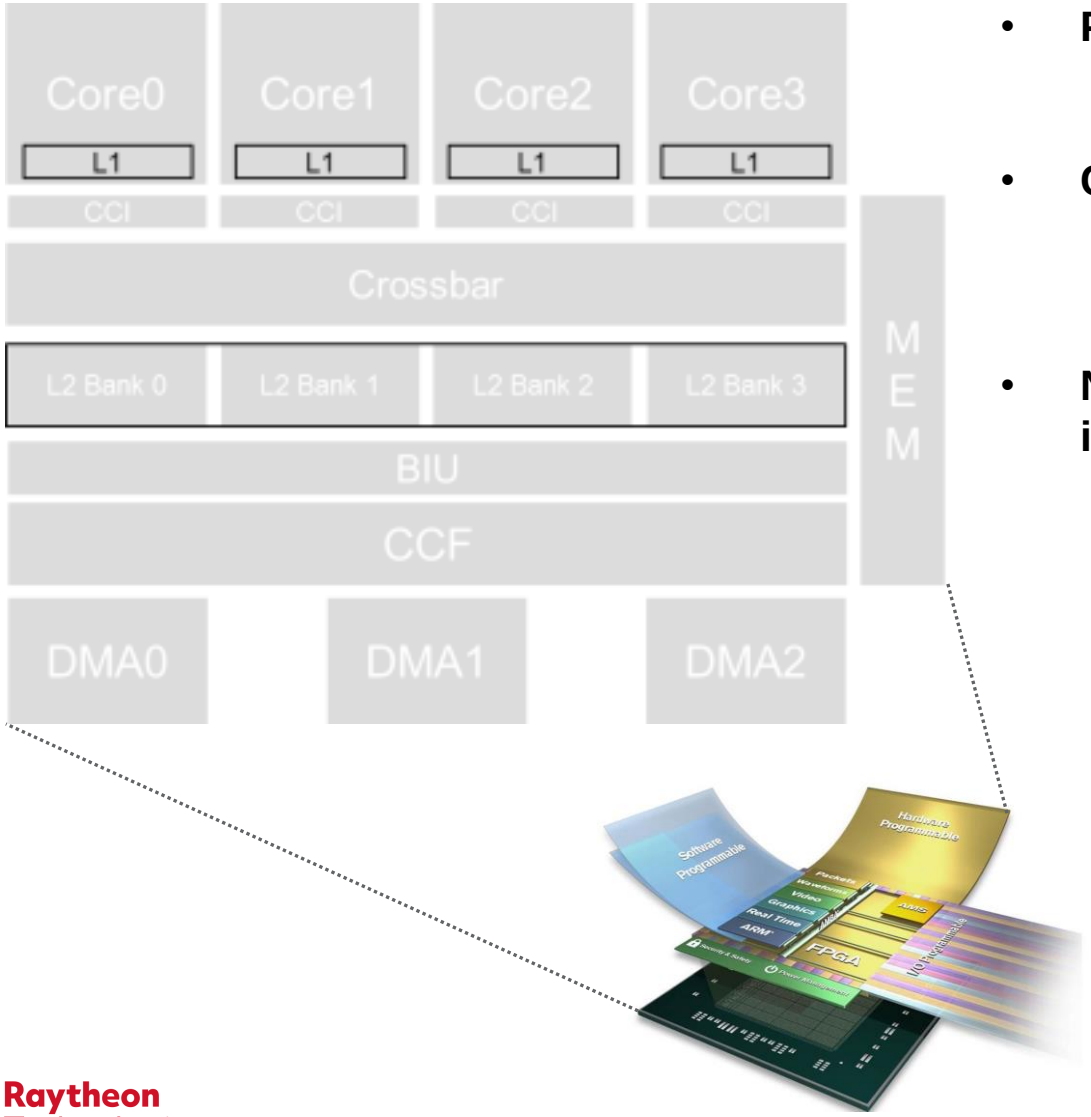
General Value

- Make Multicore Certification possible and much faster
- Being able to introduce more complex processors and hence more embedded intelligence to:
- Enabling more computing intense applications: more autonomy, better monitoring
- Give support to applications: Situation Awareness for Auto-Taxiing (SIS) or Civil Certified Vehicle Management Computer (EES)

Raytheon Role

- Use-case provider: Civil Certified Vehicle Management Computer.
- Enhance design methodology using accurate timing analysis not only for certification but in early stages of the design workflow through optimization models.

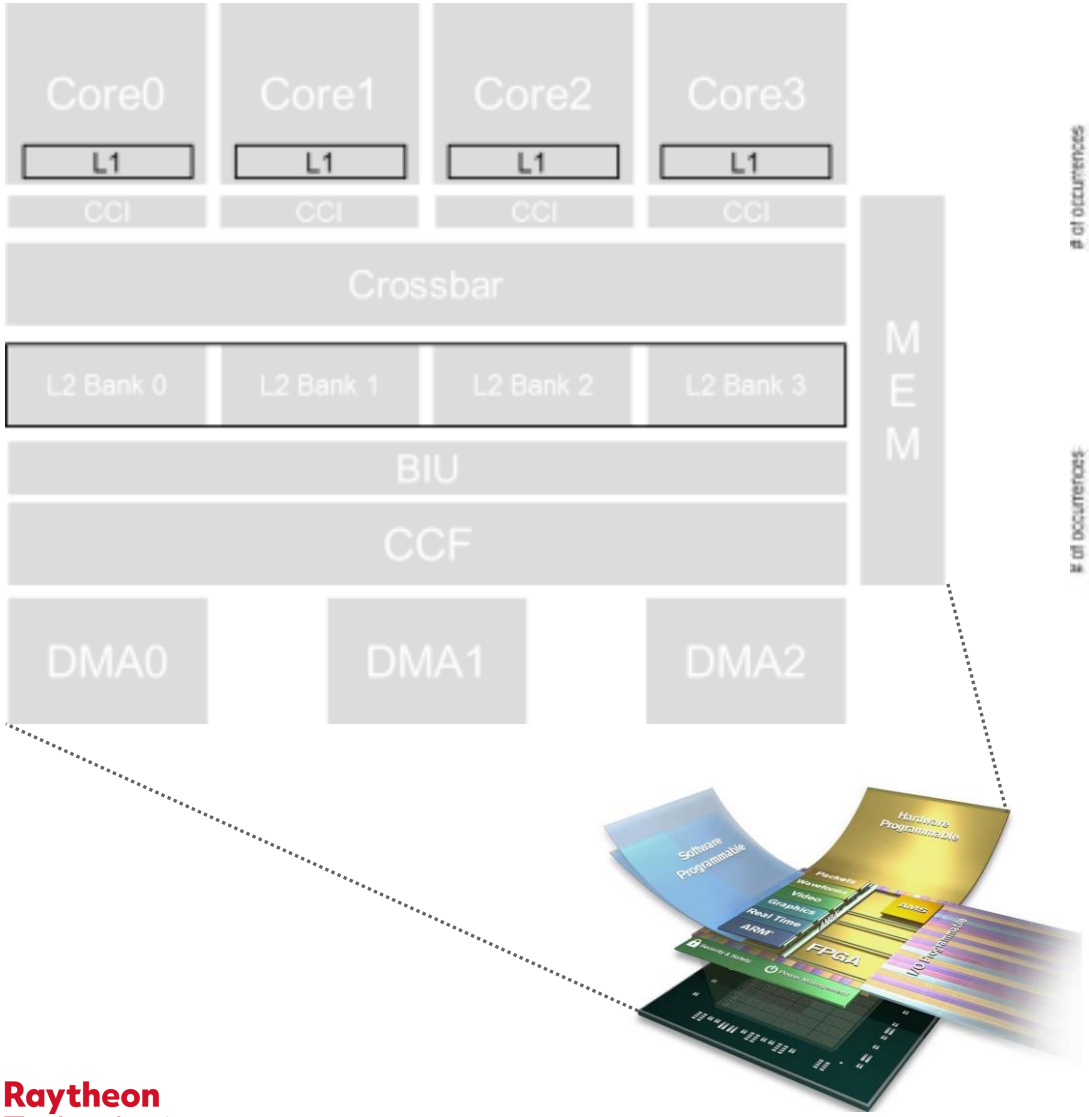
Problem: Multiprocessing systems

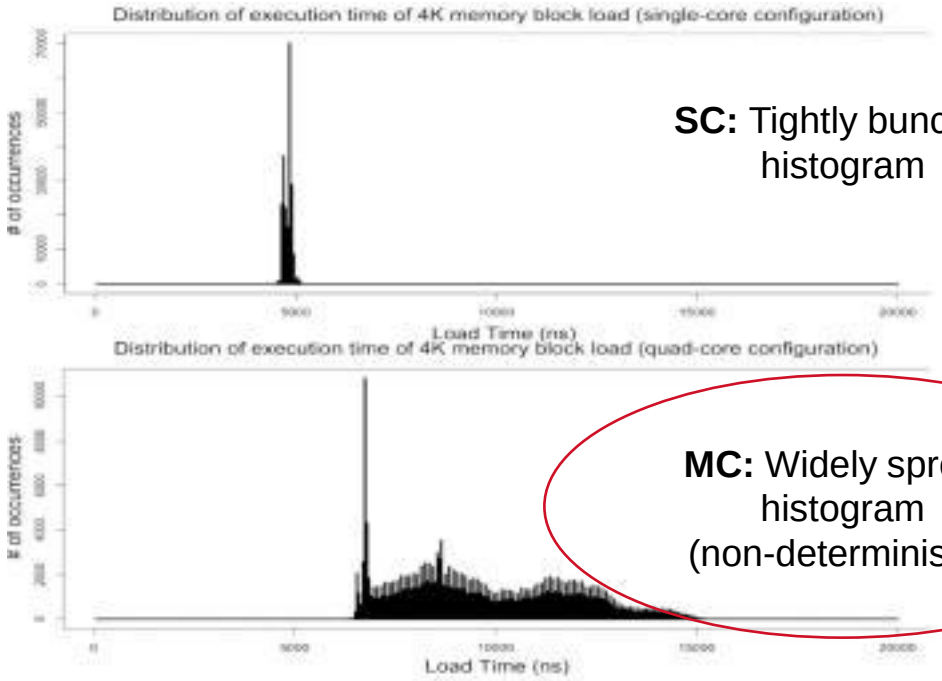


- **Problem: shared resources**
  - Space partitioning: mostly solved
  - Time partitioning: Not solved!! **<u>WCET Calculation!</u>**
- **Certification**:
  - DO-254 for HW
  - DO-178C for SW… NO MULTICORE!
  - CAST-32A to capture concerns
- **Not only MC!! But complex MPSoC with complex interconnects:**
  - Coherency protocols
  - Arbitration
  - Queues
  - Cache architectures
  - IO

DESIGNED FOR PERFORMANCE NOT SAFETY!!!!

Problem: Multiprocessing systems





SC: Tightly bunched histogram

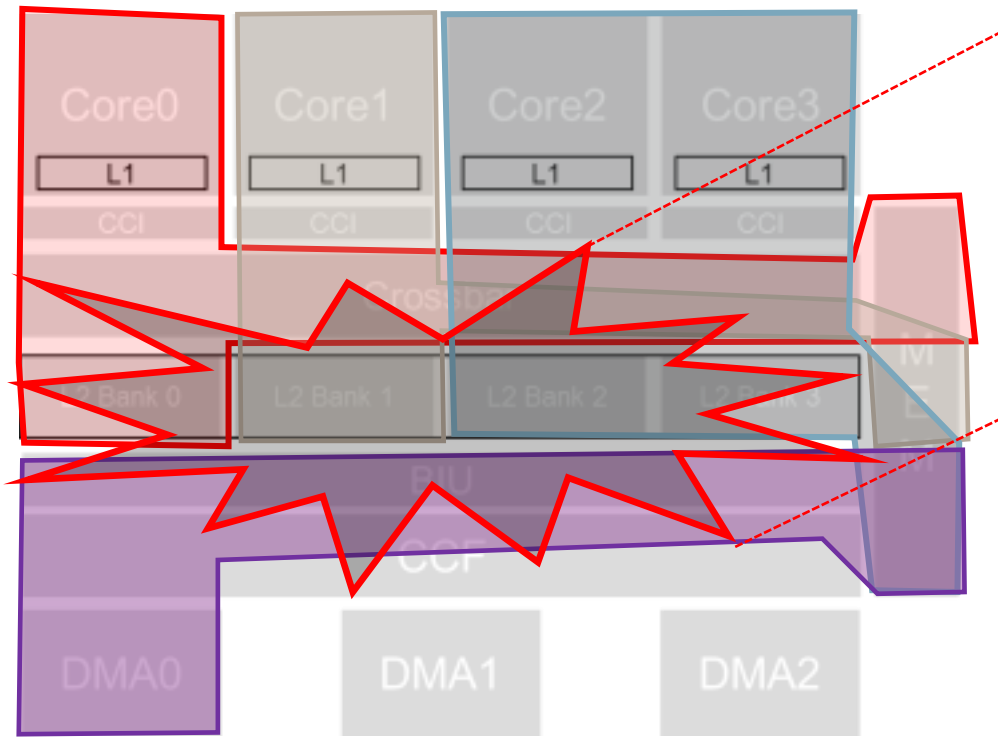
MC: Widely spread histogram (non-deterministic)

Figure 15. Example of impact of interferences on 4K memory load operations (1 vs. 4 cores)

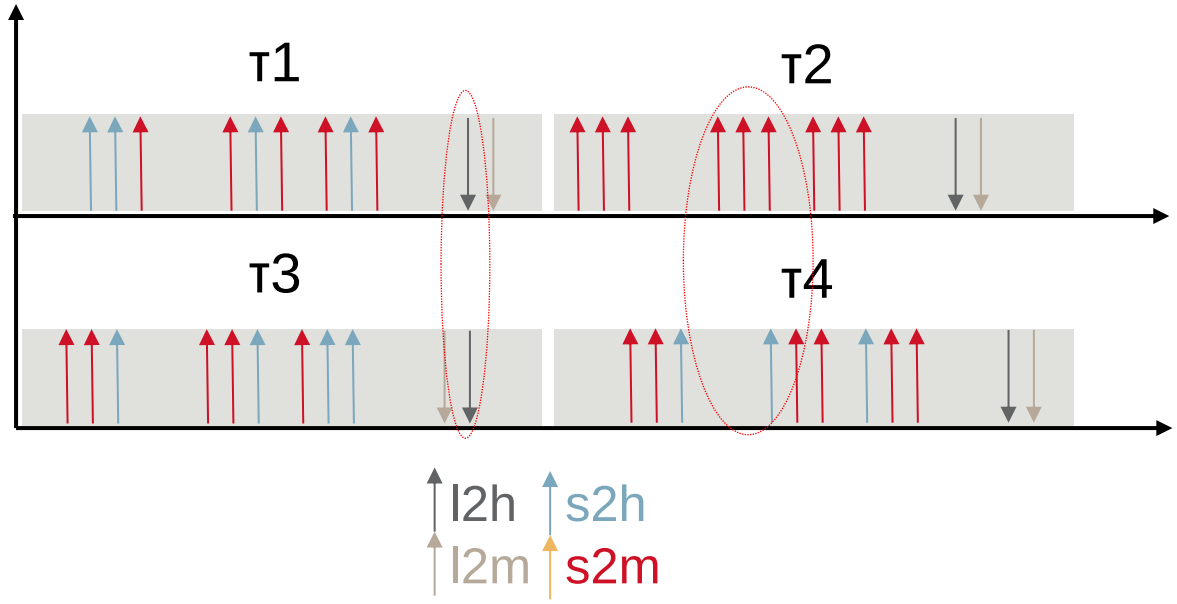
"For both cases, we collect memory access time and application's execution time ***while one core executes the benchmark and others execute a stressing benchmark over the same DRAM controller."***
(FAA TC-16/51, p. 48)

Source: Lynx Software

Partitioning: Solution?



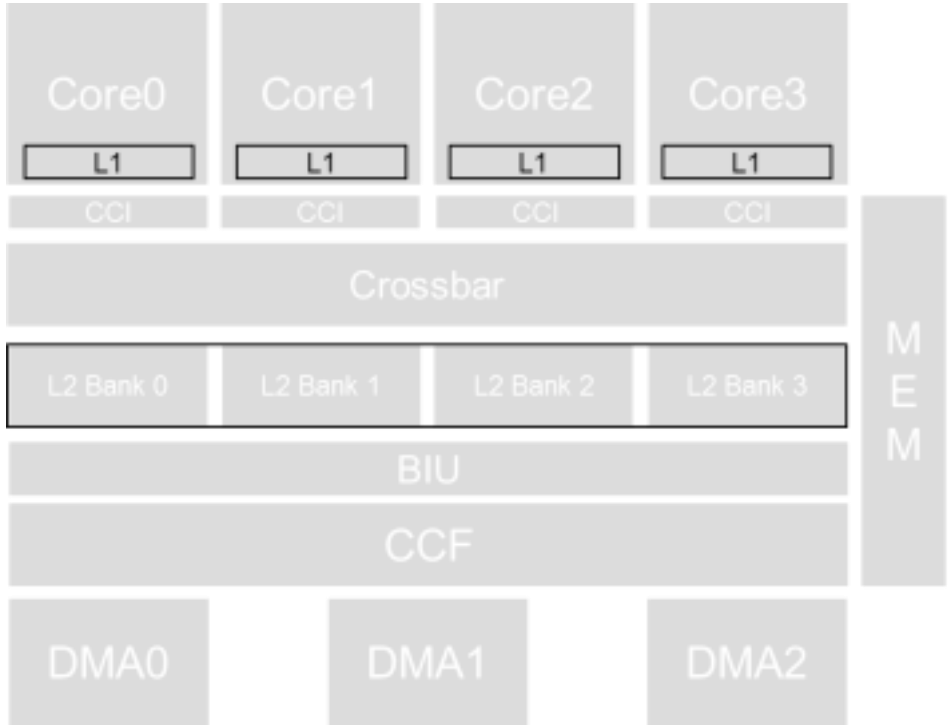
Hidden Contentions in interconnect

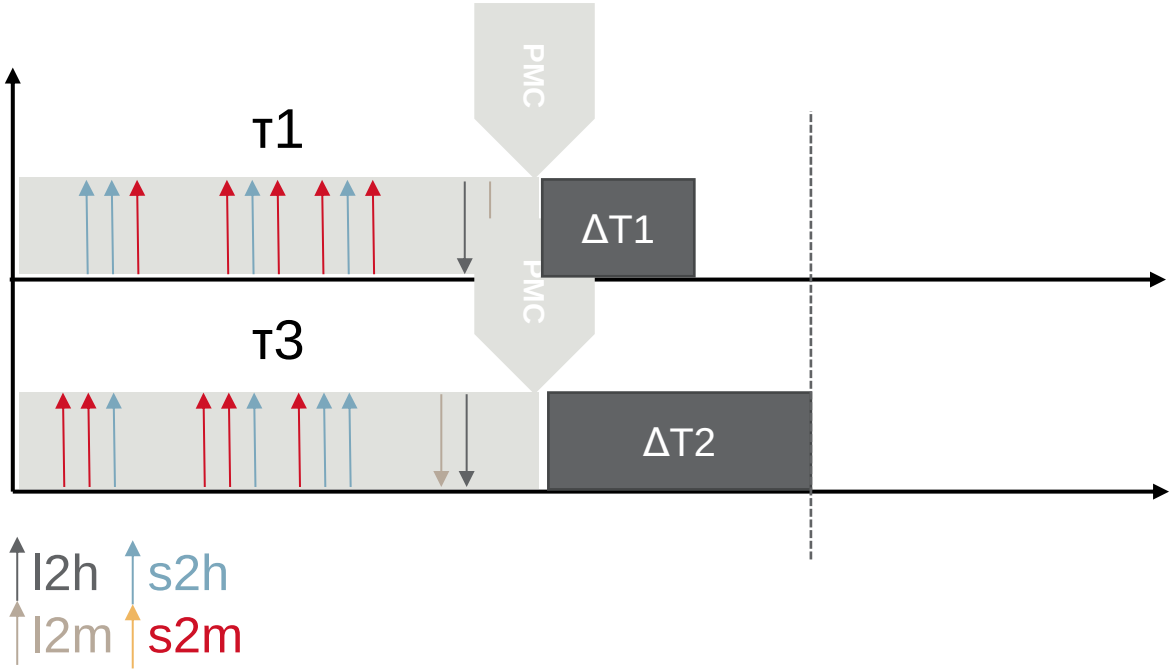


Part of the solution…

Task Characterization: Solution?

Part of the solution…

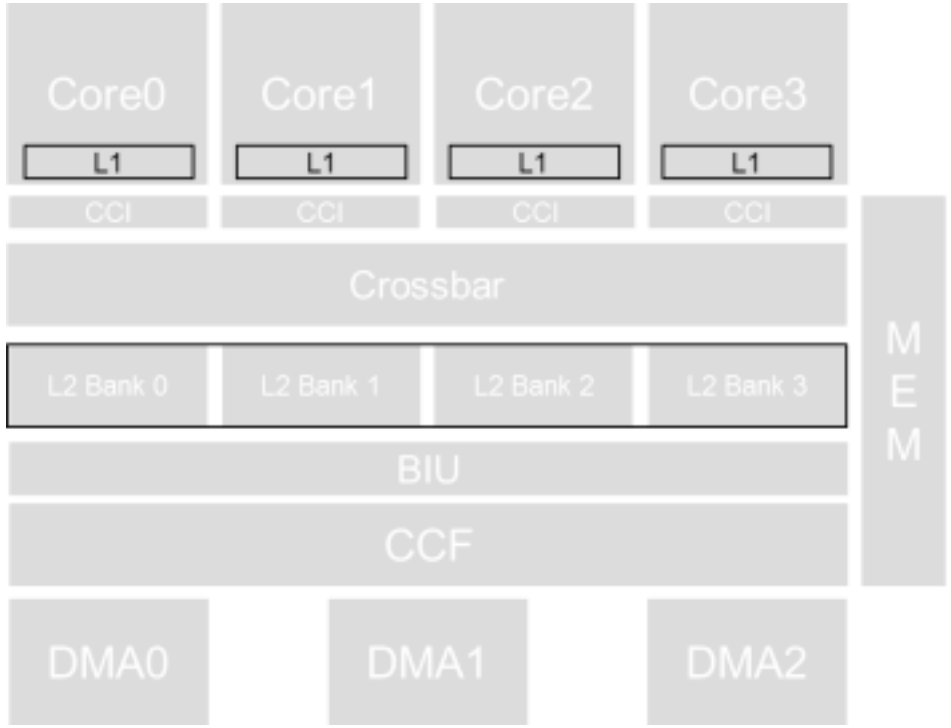




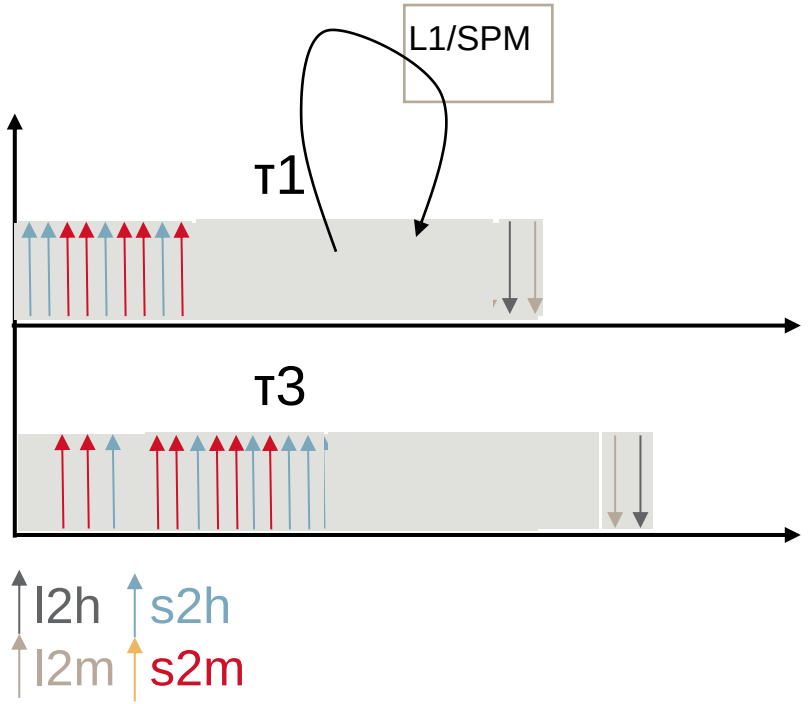
- Footprints need to include tracing: time dependencies
- If not, we need to find worst case pairing, which is too pessimistic
- Extract information for the Task Contention Model is **DIFFICULT**
  - Like effects of queues in burst transactions

TCM → ΔT1 & ΔT2

Force MoE to decrease contention: solution?

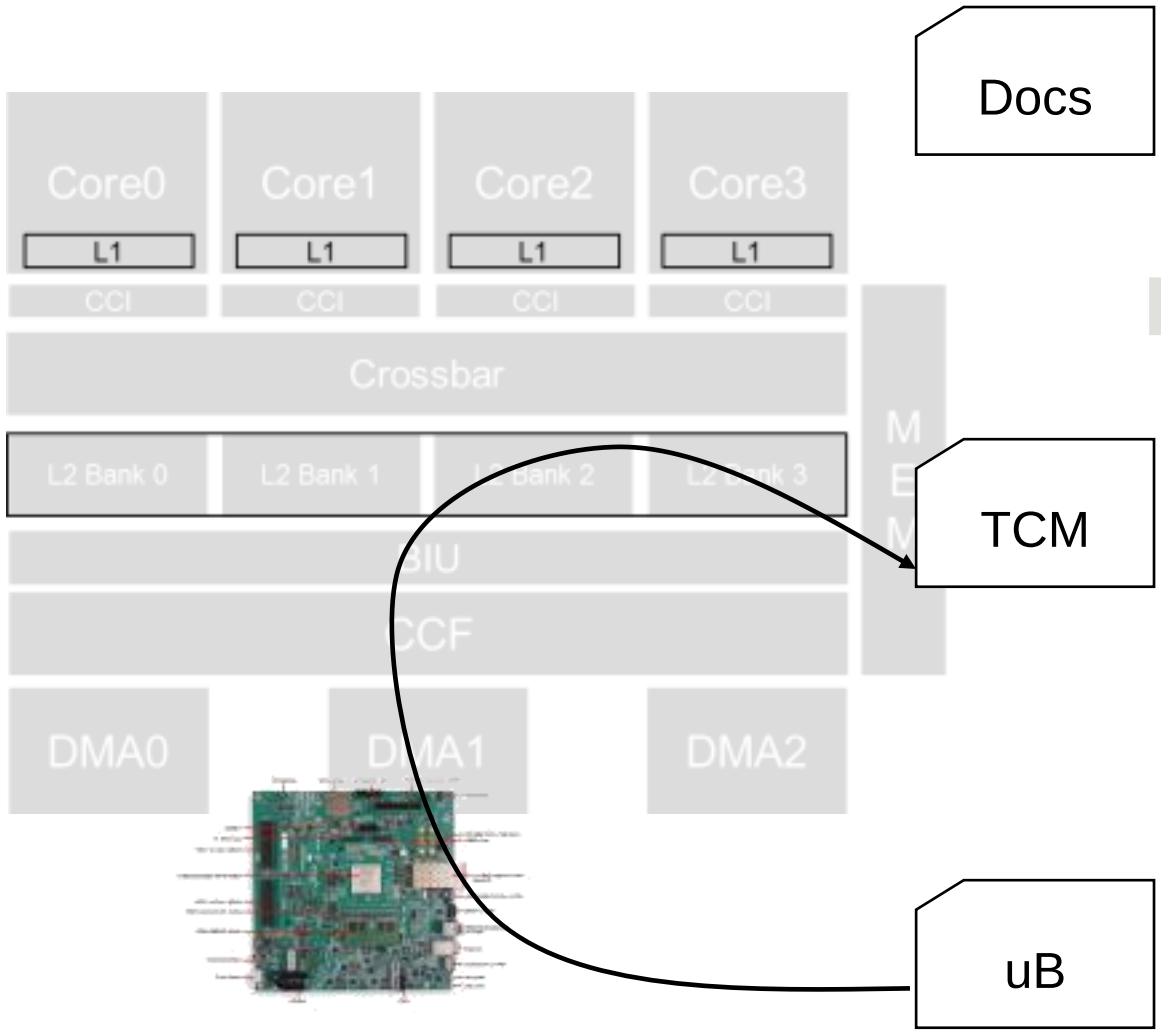


Part of the solution…

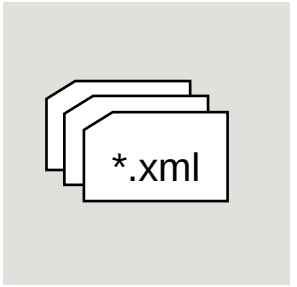


- PREM, 3-Phase, AER, MemGuard, etc…
- Controlling overlap
- Controlling that the execution phase is confined within the core
- REQUIRED TO HAVE GLOBAL SCHEDULER!!
- 3rd party apps?

Is HW Modelling Possible?



Expand
SHIM, AMALTHEA,…?



- We can use an **existing format and expand it with information about contention.**
- Some information will come directly from the manufacturer, **BUT SOME WILL HAVE TO BE EXTRACTED, this is extremely difficult without the manufacturer's help.**
- We can use **Performance Monitor Counters to learn** contention latencies in cache, DMAs, etc.
- We have to **limit the state space** proposing architectures.

This is only for multi-cores… adding GPUs and FPGAs to the game complicates things a lot!!!

If not using COTS… what can we do?

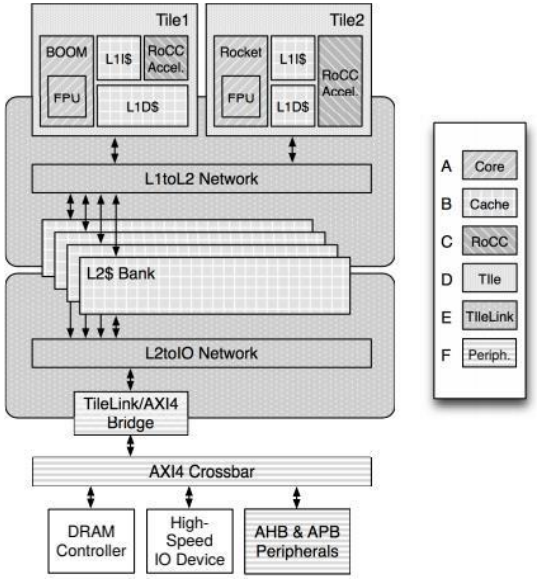
CUSTOM ARCHITECTURES &

BEYOND MOORE SOLUTIONS

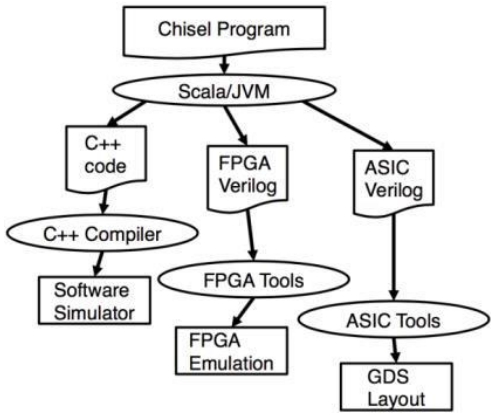
RISC-V based architectures

ARCHITECTURES FOR SAFETY-CRITICAL DOMAINS

- **Collins Aerospace** is a **Silver** Member of the **RISC-V Foundation**.
- **Verification** from the very beginning: **Formal** specs for RISC-V (**Kami**, **Sail**, etc.)
- Specifically tailored **instruction extensions**: IO, crypto, monitors, etc.
- **Safety** and **Security** enhancements: redundancies, anomaly detectors, SCA protection, etc.
- **Reusable** building blocks.
- **Customizable Performance Counters** for full observability.



Rocket Chip Generator (Berkeley)

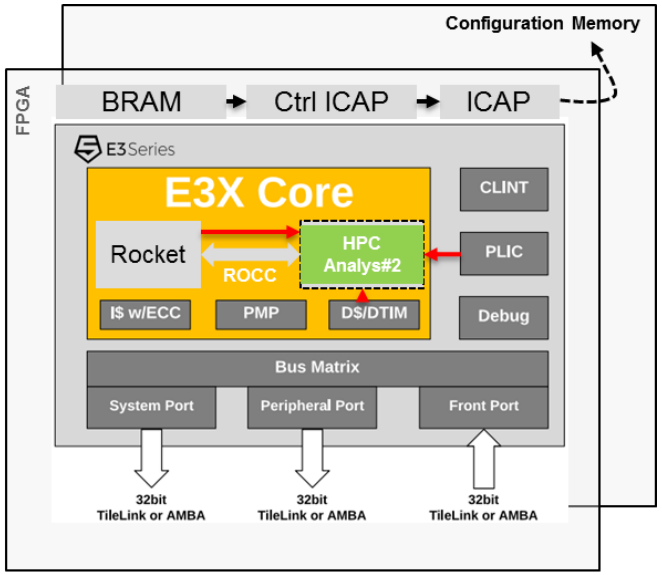


Toolchain Rocket Chip Generator and Chipyard

```
// app.c
Instruction #1
Instruction #2
Instruction #3
Instruction #4
Instruction #5
CUSTOMX() // returns HPC analysis and launch reconfiguration
Instruction #6
Instruction #7
Instruction #8
CUSTOMX()
Instruction #9
Instruction #10
CUSTOMX()
```

Machine Hardware Performance Monitor Event Register	
Instruction Commit Events, mhpeventX[7:0] = 0	
Bits	**Meaning**
8	Exception taken
9	Integer load instruction retired
10	Integer store instruction retired
11	Atomic memory operation retired
12	System instruction retired
13	Integer arithmetic instruction retired
14	Conditional branch retired
15	JAL instruction retired
16	JALR instruction retired
17	Integer multiplication instruction retired
18	Integer division instruction retired
Microarchitectural Events , mhpeventX[7:0] = 1	
Bits	**Meaning**
8	Load-use interlock
9	Long-latency interlock
10	CSR read interlock
11	Instruction cache/ITIM busy
12	Data cache/DTIM busy
13	Branch direction misprediction
14	Branch/jump target misprediction
15	Pipeline flush from CSR write
16	Pipeline flush from other event
17	Integer multiplication interlock
Memory System Events, mhpeventX[7:0] = 2	
Bits	**Meaning**
8	Instruction cache miss
9	Memory-mapped I/O access

Hardware Performance Counter Registers

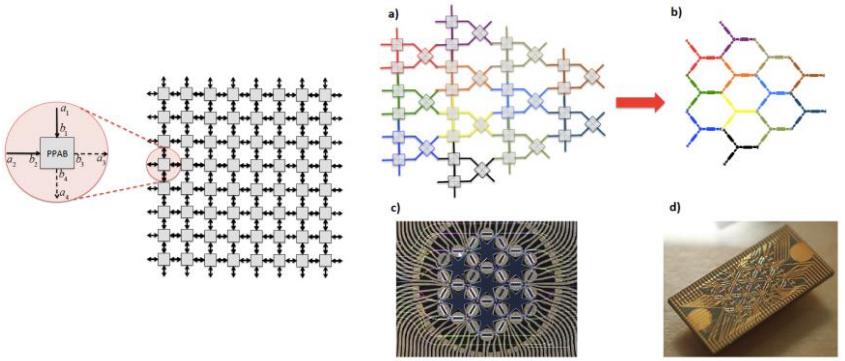


System Schematic

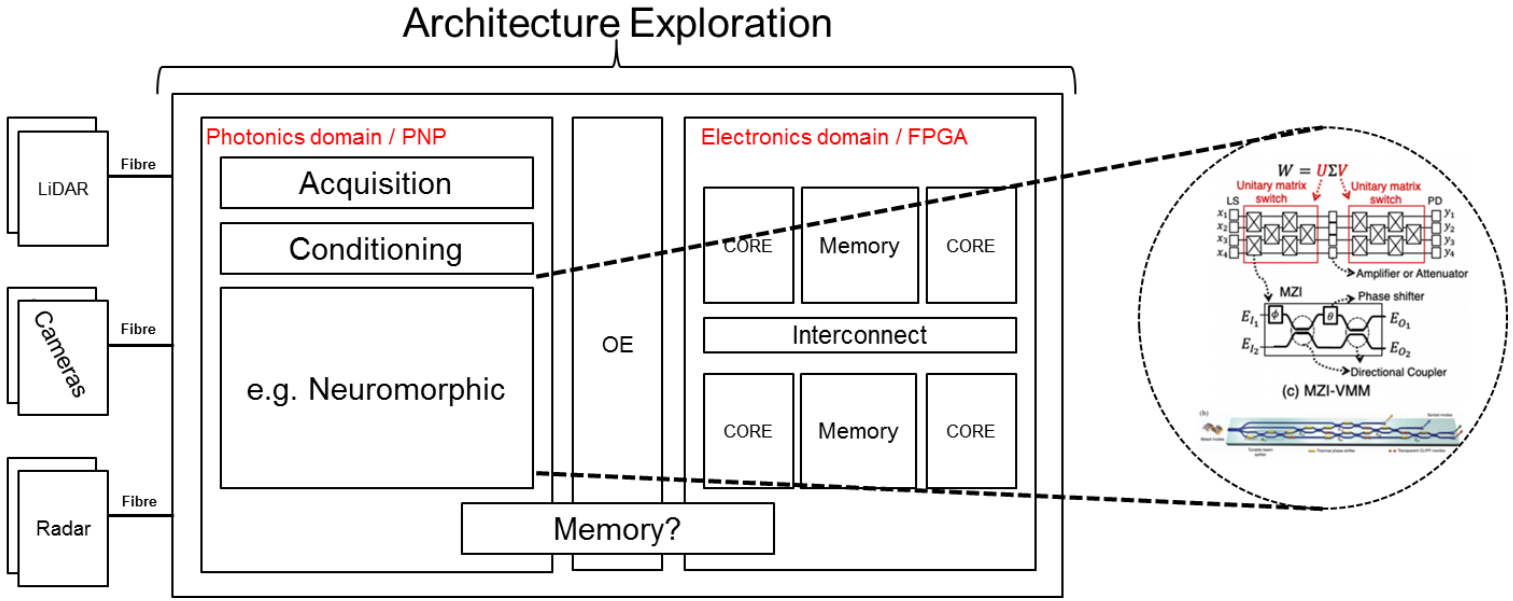
Photonic Computing

TOWARDS MORE HYBRID PROCESSING ARCHITECTURES

- Prepare for **limitations in conventional electronics**. Death of Moore's Law, etc.
- Optical **sensing** and **communications** are **already used** in commercial products, a **more computing** approach is necessary.
- Programmable Nano-Photonic Processors (**PNPs**) and Field Programmable Photonic Arrays (**FPPAs**) are already a reality.
- It is proven that photonic circuits can improve **speed** and **energy** consumption although they currently have **limitations** in terms of **scalability**.
- Execution paradigms can change if **photonic memories** or photonic links can be used to alter the **memory hierarchy limitations**.



Field-programmable photonic arrays DANIEL PÉREZ, IVANA GASULLA, AND JOSE CAPMANY



Conclusions

- **Certifying** multi-processing systems is difficult. Possible, but expensive.
- **Partitioning is** required…enforced by Separation Kernels or Hypervisors like Lynx Secure, pikeOS, VxWorks, etc. supported by HW features.
- **Task Characterization** is required to control access to shared resources.
- **Modify the Execution Model** to control contention is required to minimise overlapping and assign quotas to partitions.
- **HW Models** are very helpful to be able to flow up requirements for scheduling and partitioning… but how accurate are they?
- Alternative solutions will decrease time to market in the long term.

Characterization to allow modelling for such critical analysis… is a real challenge

Raytheon Technologies

Raytheon Technologies

Thank You!!!

Juan Valverde
valverj@rtx.com