THALES

MAKS 2017

Zhukovsky | july 18 - 23

Multi-core Management – A new Approach

Dr Marc GATTI, Thales Avionics
Marc-j.gatti@fr.thalesgroup.com

MAKS – IMA Conference
20th July, Moscow

Abstract – Multi-core Management – A new Approach

Certification of a mono or multicore processor is going to request to demonstrate that we are capable of mastering the determinism of the execution of all the applications which are going to be executed. Regarding the multicore we introduce a level of complexity to be managed regarding the execution of the application in parallel on each of the cores of the multicore processor whatever is the internal architecture of the processor.

In an IMA context, in a mono-core processor:

- This determinism is insured by the control of the WCET allowing defining a maximal boundary for all the accesses to all the services offered by the Operating System.
- The Platform Provider has no information about the applications which are going to be executed. In this condition the computation of a WCET on a multi-core, like it is done currently, will be realized by introducing constraints at the level of the internal functioning of the multi-core processor.

Our approach is to combine both WCET and MAF (and or MIF) spare time in order to manage the execution of all the application, in parallel, on a multi-core, safely. It is what we propose to address in our paper and present during the conference.

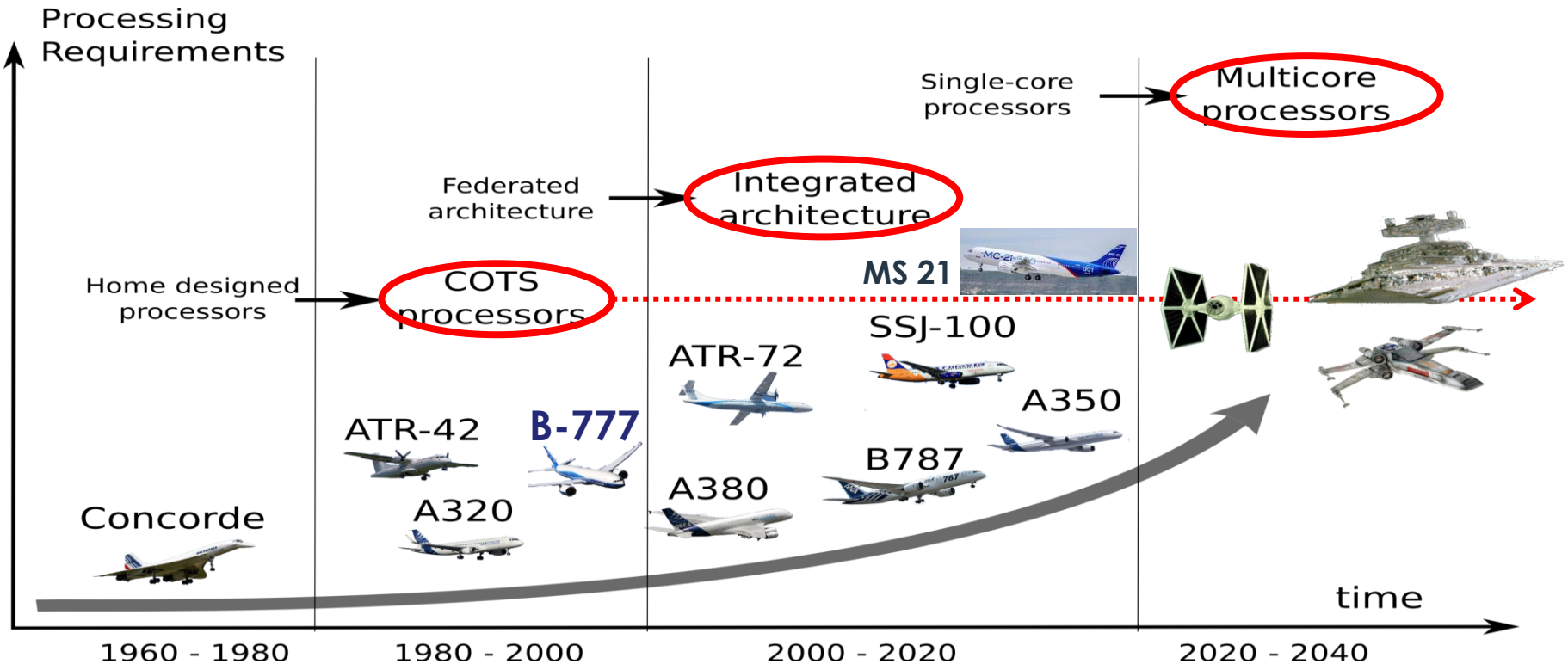
SUMMARY

- Context
  - Multicore Introduction
    - Problem Statement
    - Current Studies for IMA
  - Overview of Potential SW sol.
- Conclusion / future works

SUMMARY

- Context
  - Multicore Introduction
    - Problem Statement
    - Current Studies for IMA
  - Overview of Potential SW sol.
- Conclusion / future works

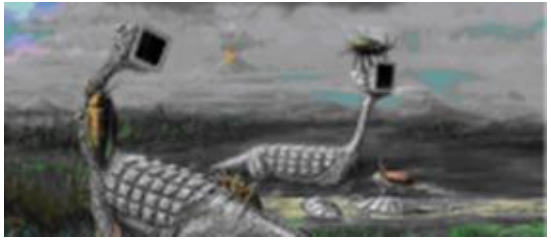
DIGITAL AVIONIC SYSTEMS EVOLUTION



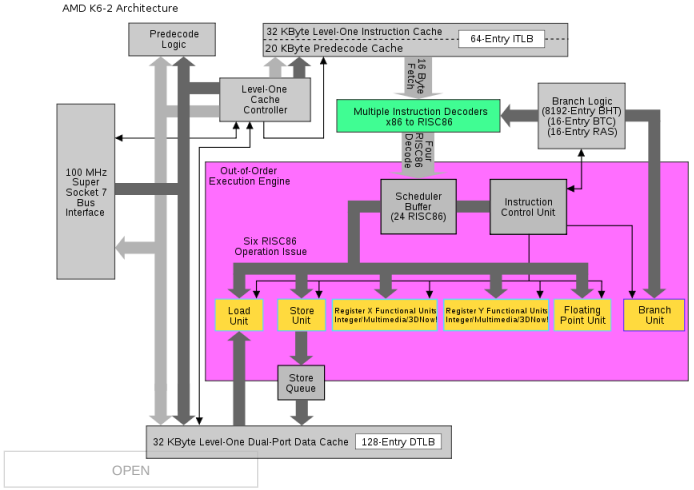
SoC and Semiconductor Technology Evolution

Semiconductor fabrication and continuing miniaturization

- „Semiconductor Archeology“: Last “real“ standalone CPUs cores has been designed 20 years ago (250nm)
  - Mono-Core processor Era is finished



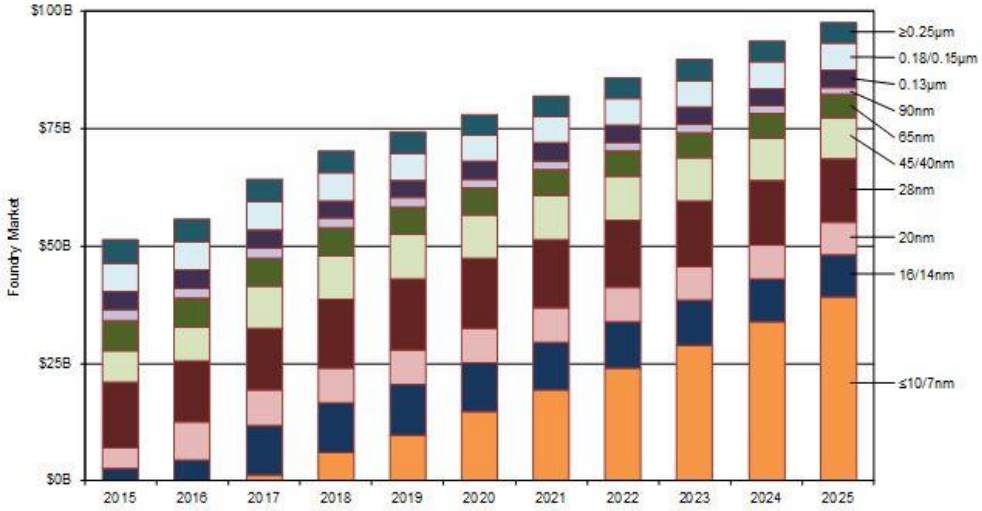
  - In the meantime the integration density has increased and will double again by 2018



Source : TTTech by Courtesy

2017-2025: Key Fabrication Technologies in 5-7 years?

- Embedded SoC with HPC (1-8+ cores) and on-chip networking/switching/IO can be designed in 28nm to 65nm
- HPC/Embedded Supercomputing will target 7-14nm, hundreds of cores, safety/security, cryptos, peripherals, I/O's…
- 5nm in work (2020+)
  Skyrocketing costs?
- 1nm-Gate demonstrated in 2016 (Berkley Lab)



Source : TTTech by Courtesy

AVIONICS STAKES & PARADIGM

Improving the SWaP (Size, Weight and Power) of the IMA embedded platform

- Reduce the Size, the Weight and the environmental Footprint (Power Consumption)

While Increasing

- Availability, Safety, Reliability
- Security
- And the performances per a significant factor compare to the current generation

While continuing to Master Certification Issue

- Aircraft Embedded Systems have to be certified following certification requirements of Federal Aviation Administration (FAA), European Aviation Safety Agency (EASA) and other agencies as required.
- Multicore platforms despite of advantages introduce significant certification challenges.

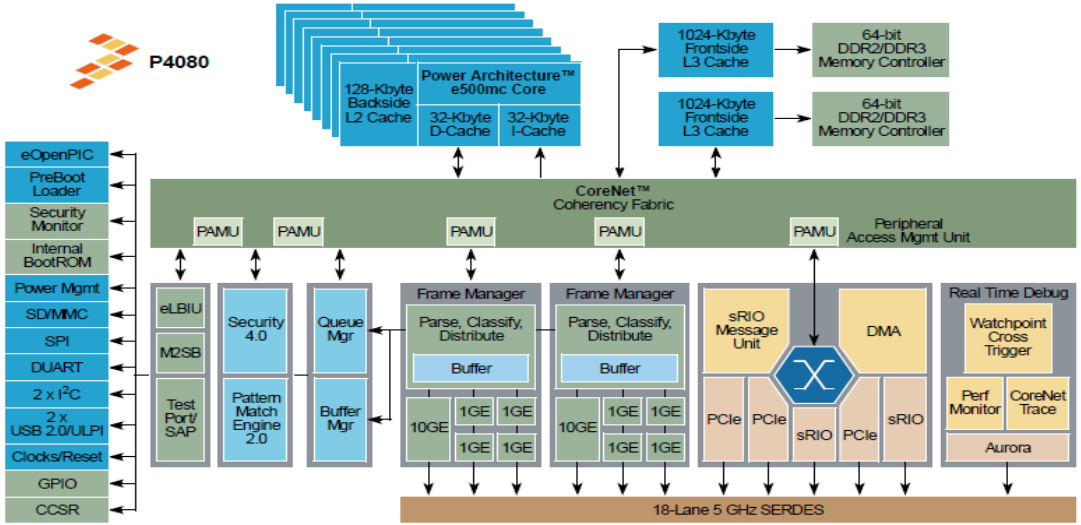
SUMMARY

- Context
- **Multicore Introduction**
  - Problem Statement
  - Current Studies for IMA
  - Overview of Potential SW sol.
- Conclusion / future works

MULTI-CORE: INTRODUCTION

What is a multicore processor?

- Multicore processor is characterized by N (N ≥ 2) processing cores
- A set of shared interconnected resources (Memories, PCIe, Ethernet, Cache, Registers, etc.)
- An interconnect to manage the accesses that can be
  - An arbitrated bus
  - A switched network like



Multicore management in certified embedded platform can be summarized to Interferences management

THALES

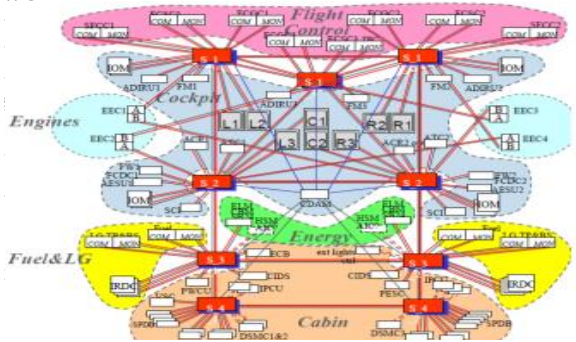
Multi-Core Processors in tomorrow's real-time equipment

Cockpit Display

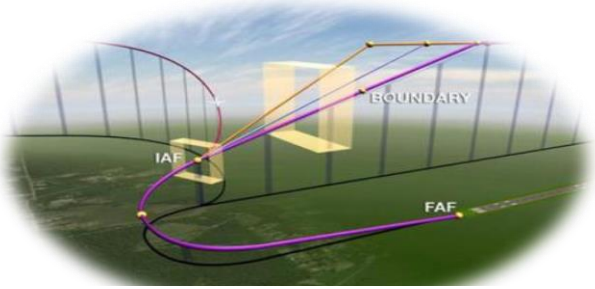
Navigation

Ground Systems

- More processing power
- More heterogeneous functions
- Agility w.r.t evolving technologies
- Certification requirements
- **Performances and timing guarantees**



Integrated Modular Avionics



Trajectory Based Optimization

Control&Payload Computers

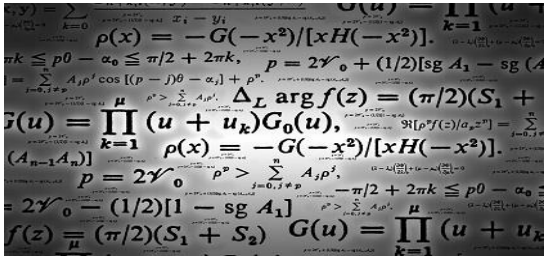
SUMMARY

- Context
- Multicore Introduction
- **Problem Statement**
- Current Studies for IMA
- Overview of Potential SW sol.
- Conclusion / future works

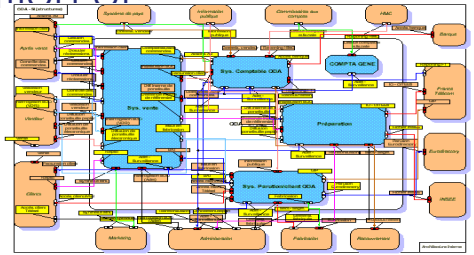
Assurance Methods in Avionics

Finding the right balance to achieve both Airworthiness certification authorities expectations and Technical needs of industrial actors will:

- Certainly require a combination of!



Scientific approaches



Systems Architecture & Design

Engineering Sense & Judgment

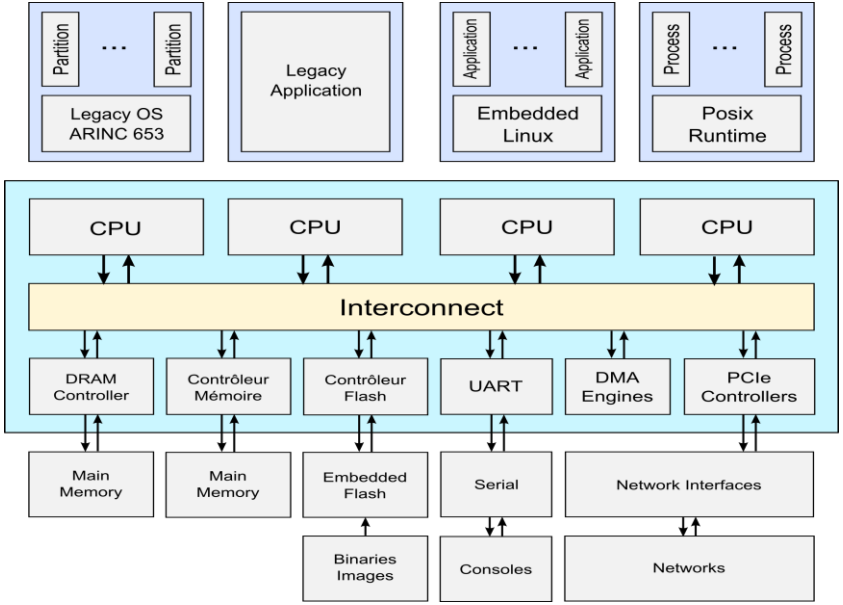
Certification Expertise
Systems Safety Expertise
System / HW / SW Architects
System / HW / SW Engineers

Agreements with Certification Experts

Development & Process Assurance

Interferences issues (problem statement 1/2)

Resources sharing in MCPs

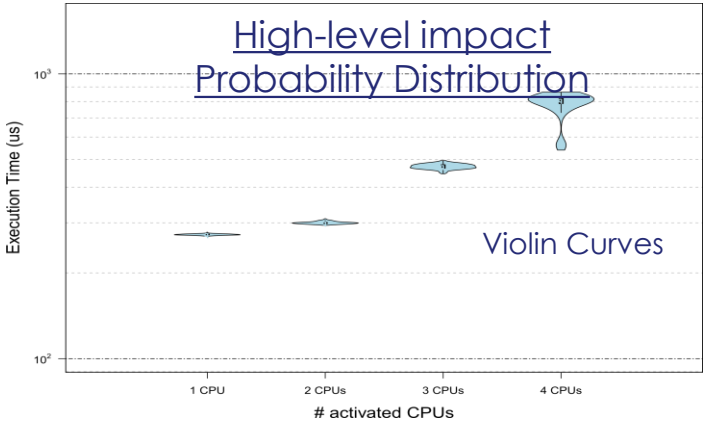


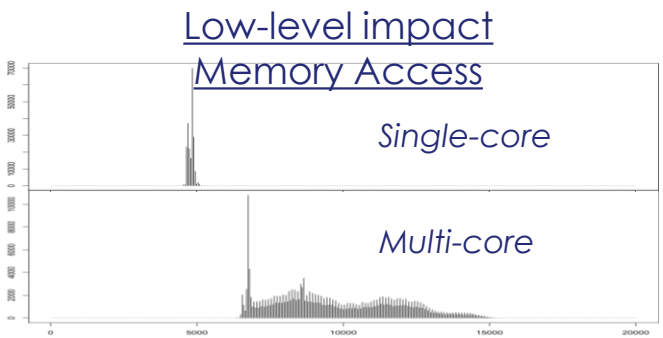
Network Topology

Arbitration rules

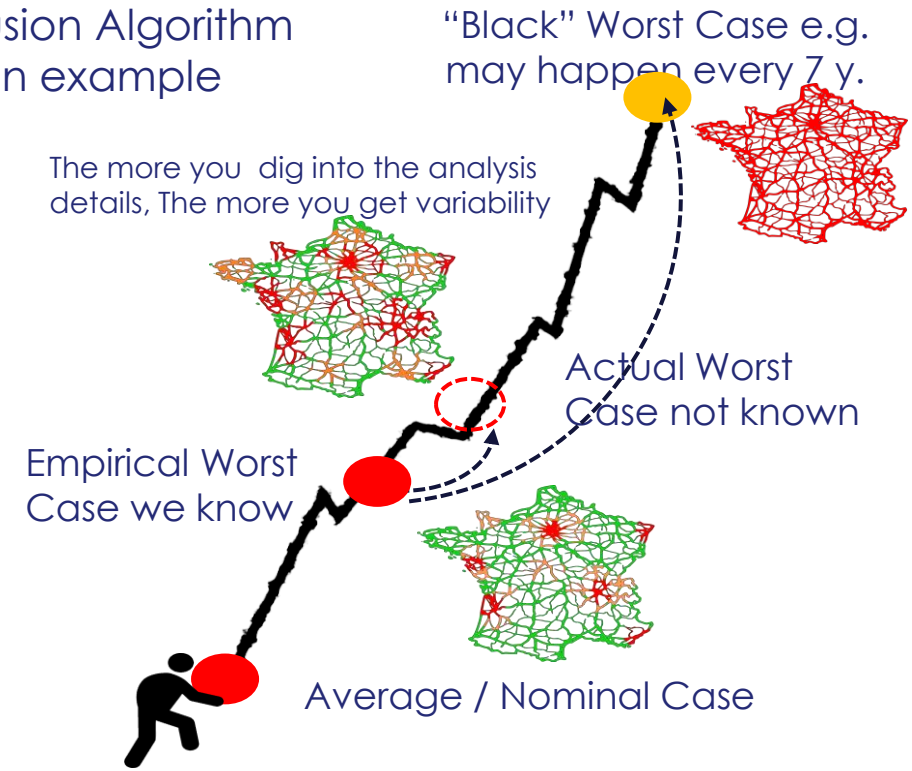
Inevitable Interferences…

Interferences issues (problem statement 2/2)





Data Fusion Algorithm
As an example



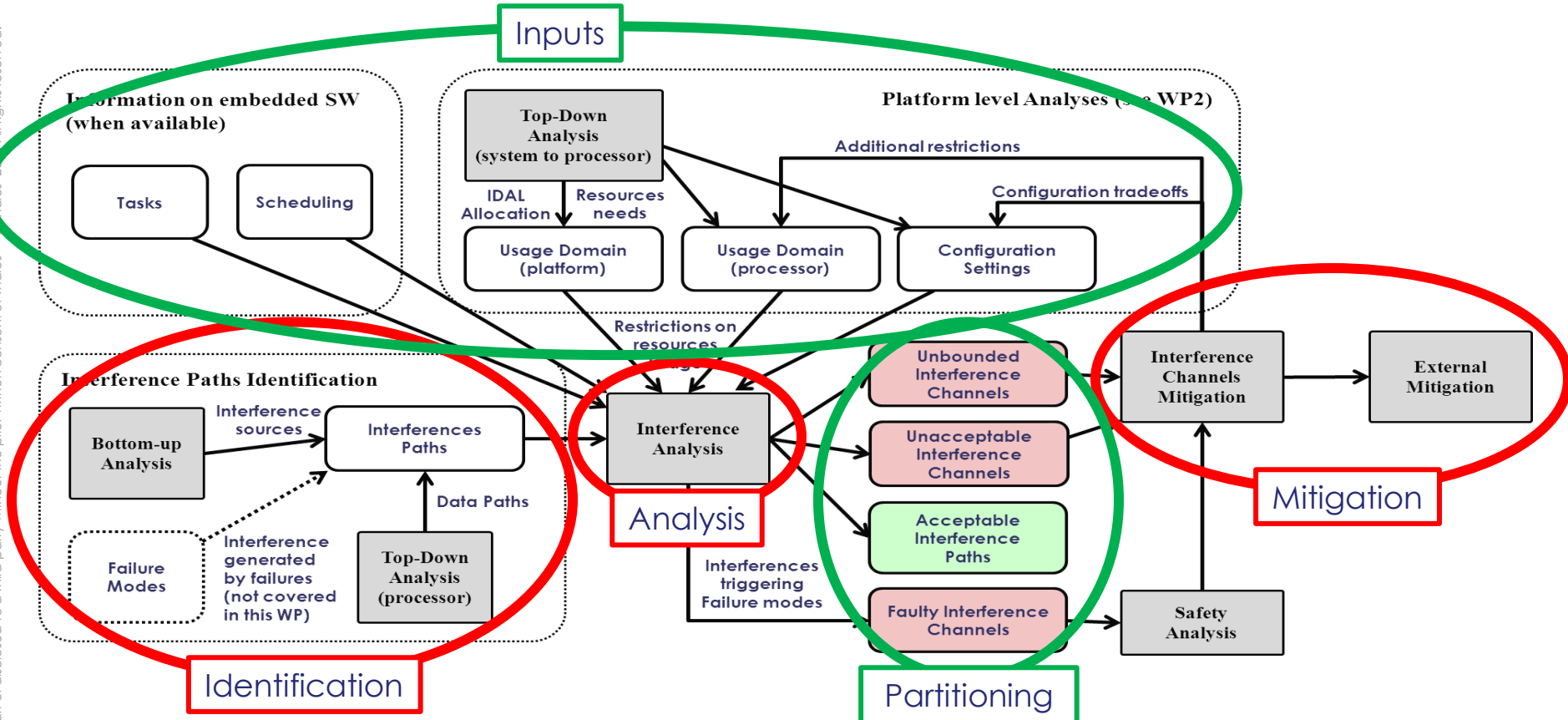
➔ **How to tackle these issues via some assurance method ?**

Assurance Method tackling interferences issues in Avionics

Combine several approaches including:

- Process
  - Consider a multi-core processor as a system, and apply system level methods (see ARP4754)
  - Follow MCP CRI (EASA) recommendations : identify, analyze and mitigate interference channels
- Scientific / technological approaches
  - Stop criteria for test campaigns
  - Deterministic Platform Software (DPS)
  - Global interference analysis methods (e.g. isWCET)
  - WCET Evaluation methods (static, dynamic, probabilistic…)
- Negotiation
  - Threshold on stop criteria
  - Synergy with endurance tests
- Engineering sense
  - Experts review
  - Knowledge base maintenance

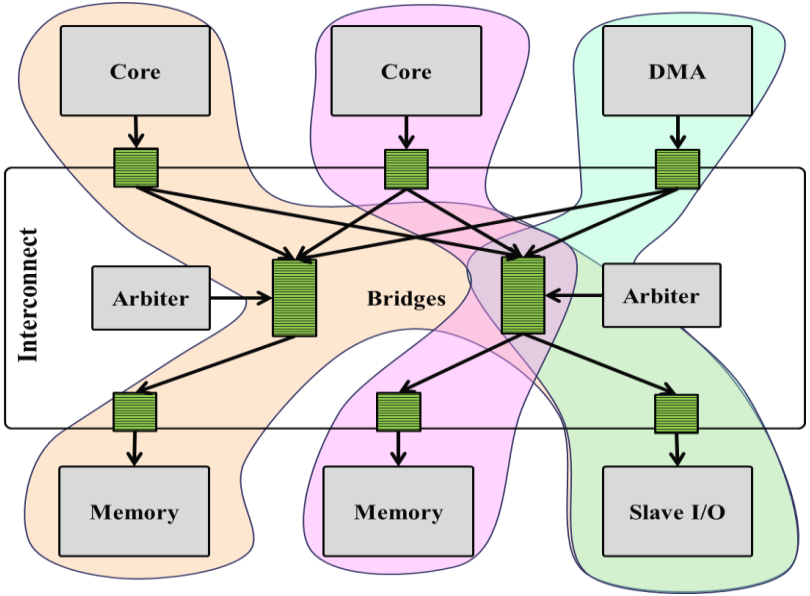
Interferences aware safety process



Focus on interferences analysis

Consider *interference paths* and mark them as

- Unbounded, or bounded but unacceptable
  → *Interference channels* mitigation
- Bounded and acceptable
- Trigger failure modes (e.g. transaction loss)
  → Tackled by specific analyses



Feared events

- Discontinuity in processor behavior
  - silent mode change
- Singularity ("*close*" discontinuities)
  - Notion of "closeness" to be defined…
- Boundary conditions

Expected rationale

- No singularity has been observed given the following restrictions…

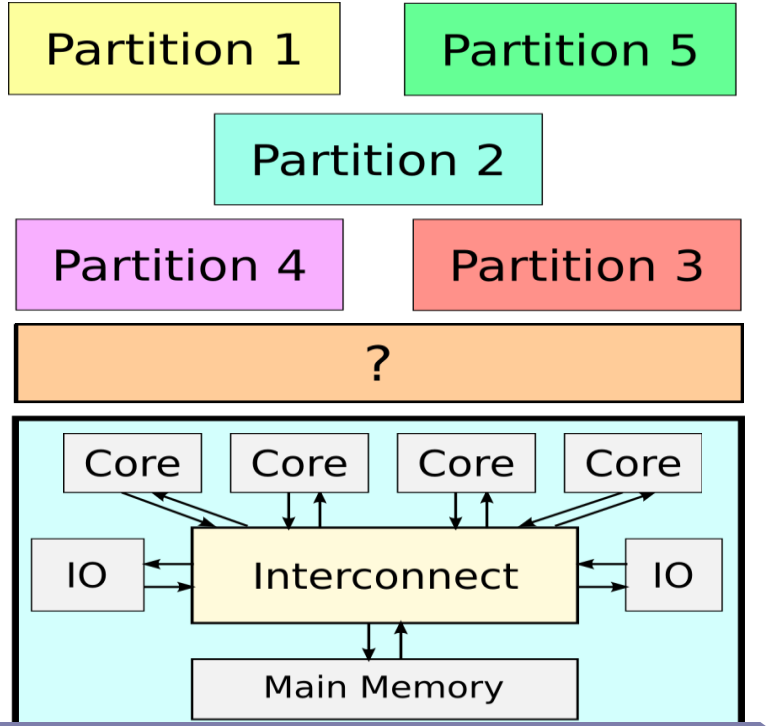
SUMMARY

- Context
- Multicore Introduction
- Problem Statement
- **Current Studies for IMA**
- Overview of Potential SW sol.
- Conclusion / future works

MULTICORE FOR IMA, "GOOD PROPERTIES"

How could Avionics Platforms take benefit of multicore processors ?

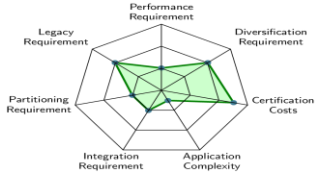
- Allow all cores to be used whatever the level of criticality
- Minimize porting and re-certification efforts of legacy applications
- Compatibility with ARINC 653 and ARINC 664 guidelines for APEX and Network partitioning
- Incremental certification



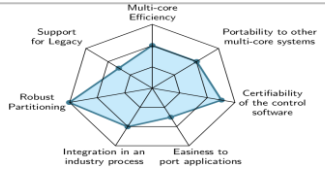
Digital avionic systems confidence have never regressed during technological steps

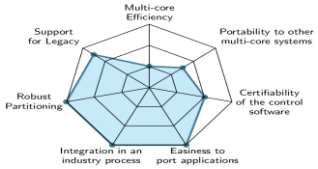
Evaluating Deterministic Platform Software against Avionic Case Studies

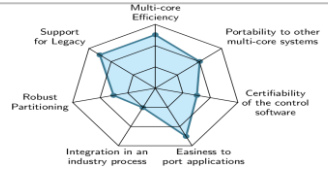
Evaluation Principles

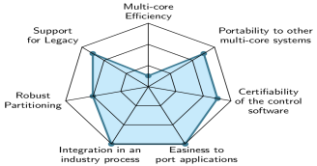


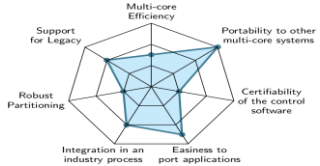
VS











	Good	Average	Bad
Deterministic Execution Model **(Decoupling tasks into execution and com phases)**	FADEC IMA	Data Services	IFE
Deterministic Adaptive Scheduling **(Monitor deadline miss, and provide snapshots of HW/SW state for further investigation)**	FADEC IFE	IMA	DS
Marthy **(Catch silently attempts to access shared resources & Re-emit accesses during given time windows**	IMA	FADEC DS	IFE
Memguard **(Allocate budget on accesses)**	IFE	DS	FADEC IMA
Runtime WCET Controller **(Check correct advance of execution at runtime → Suspend some non-critical tasks =**	DS IFE	IMA	FADEC

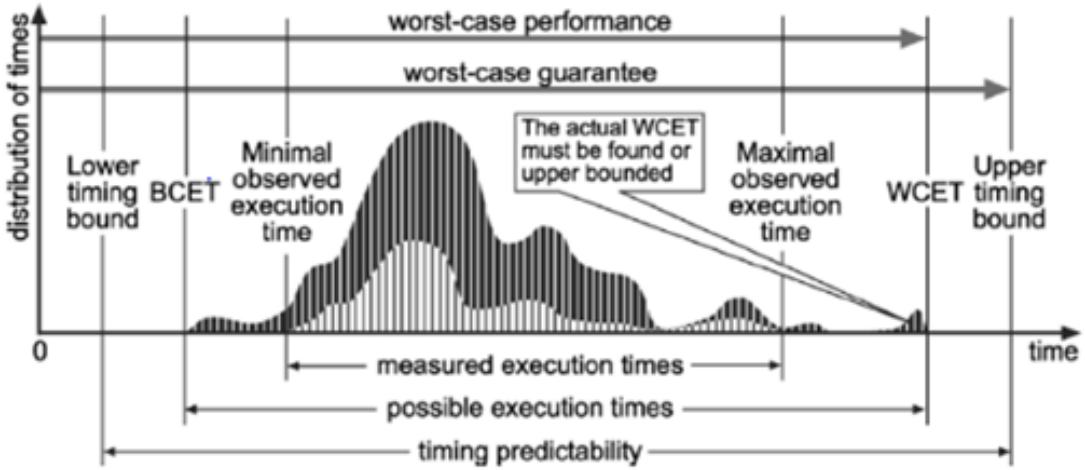
SUMMARY

- Context
- Multicore Introduction
- Problem Statement
- Current Studies for IMA
- **Overview of Potential SW sol.**
- Conclusion / future works

How to master a Multi-core processor based architecture

WCET Computation

- WCET takes into account the maximal time to access a resource of the platform. This access time is maximized of course to be sure not to exceed.



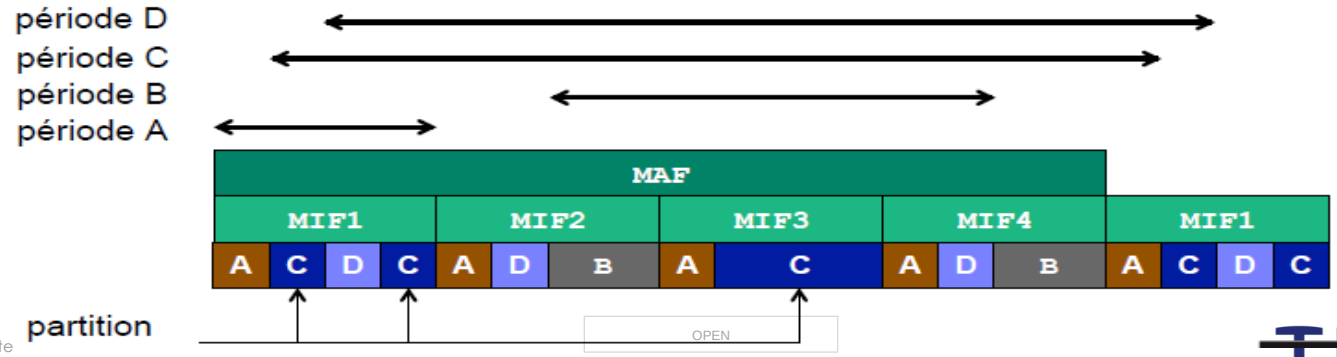
Temporal partitioning

Partition and Frames

- All the application execution one the platform are periodic
- The period is defined, by the ARINC 653, as a Major Frame (MAF).
- This periodic execution MUST be respected.

Partition	A	B	C	D
Période	10ms	20ms	40ms	40ms

MAF	MIF
40ms	10ms

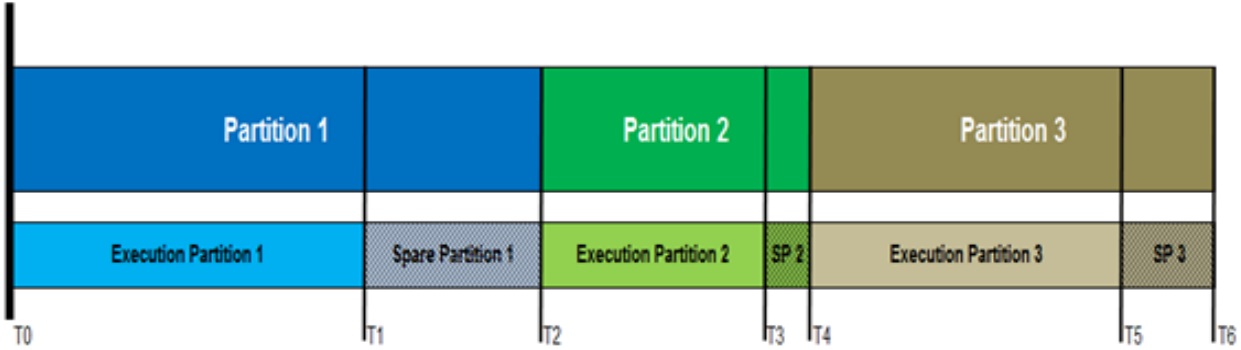


Our approach

Combine at the same time WCET and MAF.

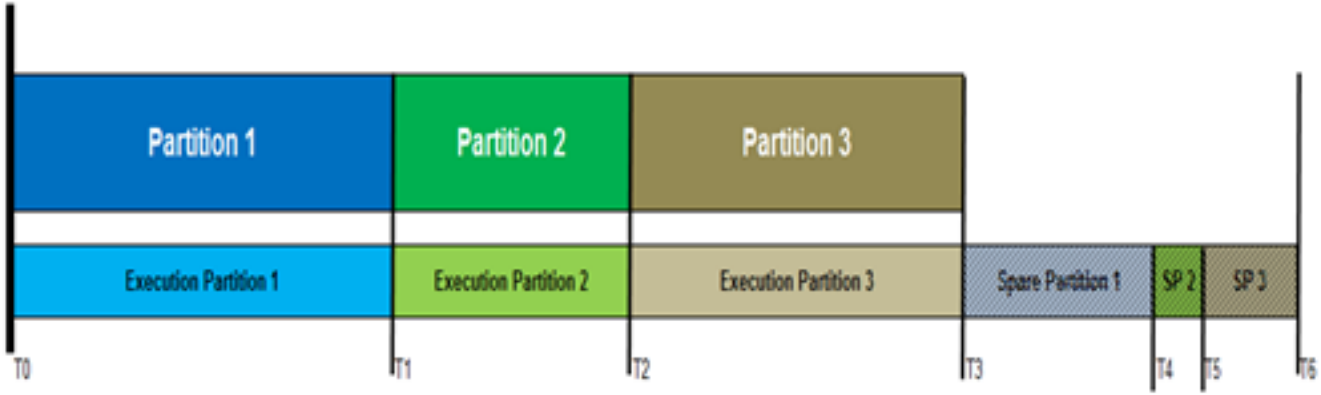
- Current MAF computation considers the computation of all WCET to allow computing the elementary time of every partition thus that of the MIF (Minor Frame) and thus that of the MAF.

WCET computation is over estimated

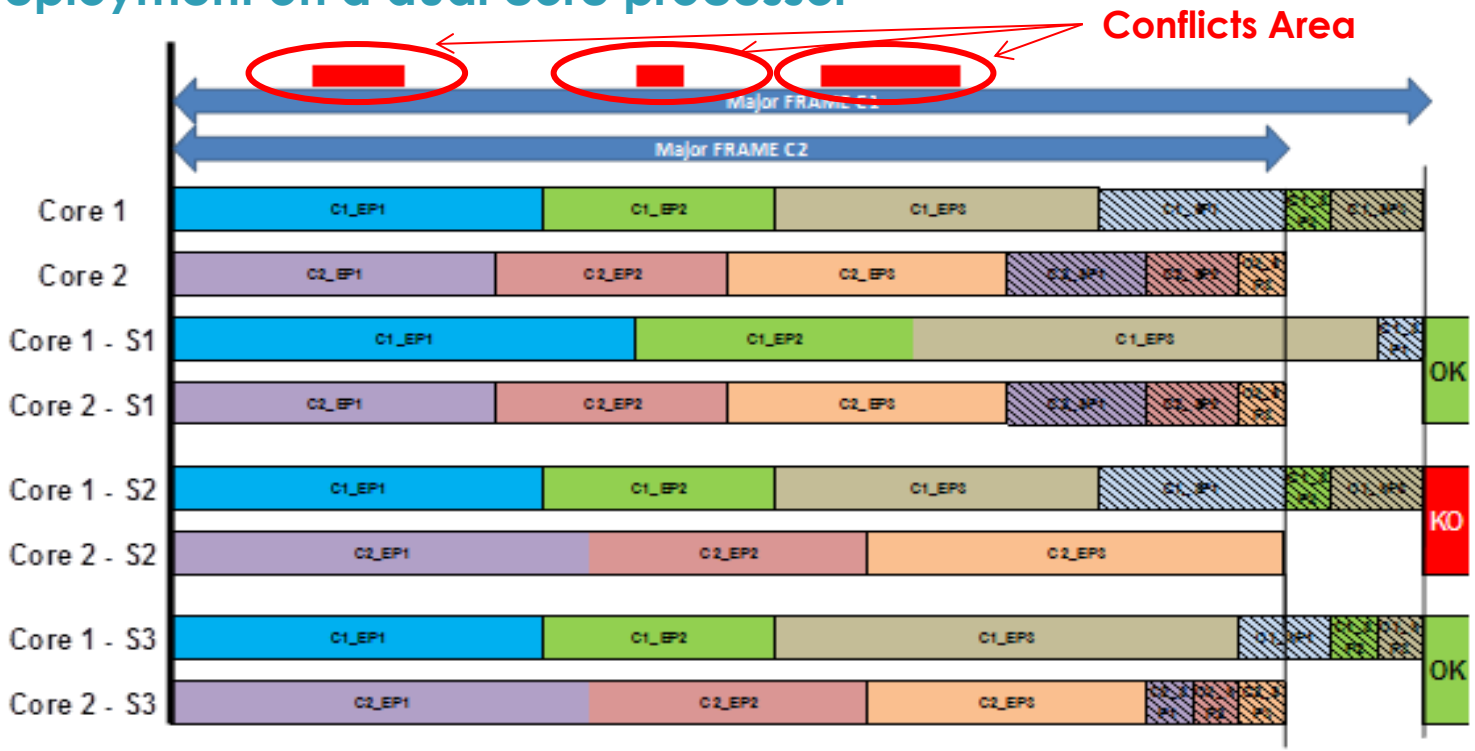


Manage Spare Time

- Our approach is based on the management of the MAF and on the "Spare Time" available at the end of every execution of a partition given by Operating System.
- OS monitor application execution and in case of Spare, identifies it and keeps it to be able to re-assign it to another application which would need it to solve conflicts.



Deployment on a dual core processor



Four ways of functioning

- Mode « 1 » corresponding to lines « Core1 and Core2 »
  - Theoretical case without conflict allowing to define
    - The partitions run on Core1 with their Spare and their MAF C1
    - The partitions run on the Core2 with their Spare and their MAF C2

- "Solution 1" = lines «Core1–S1 and Core2–S1», first approach of conflict resolution:
  - On Core 1 or 2, OS allows systematically the execution of the next partition once the first one is ended.
  - OS is able to compute "spare-time" given by the PART1 to "n" on core "x".
  - Solution continue to generate margins on Core1 and Core2 ➔ **acceptable**.

Four ways of functioning

- "Solution 2" = lines "Core1–S2 and Core2–S2 ", 2nd approach of conflict processing:
  - On Core 1 or 2, OS allows systematically the execution of the next partition once the first one is ended
  - During a conflict, OS uses available spare time, per MAF, to solve the conflict
  - In our example, Execution on C1 not impacted, only C2 is affected by conflict.
  - Solution generate a suppression of the margins on C2 ➔ **unacceptable**

- "Solution 3" = lines "Core1–S3 and Core2–S3", 3rd approach of conflict processing:
  - On Core 1 or 2, OS allows systematically the execution of the next partition once the first one is ended.
  - OS is able to compute "spare-time" given by the PART1 to "n" on core "x"
  - In our example, OS distributes the impact of the conflicts on C1 and C2
  - Impact allows keeping margins on both cores.
  - Solution minimizing conflict impact & continue to generate margins on Core1 and Core2 ➔ **acceptable**.

Certification Issues

Multi-core certification is complex in an IMA context

Thank to our approach

- Cores behavior is completely managed at OS level
- OS determines and manages all the available times or " spare-time ».
- In case of conflicts, their resolution is assured by the OS by assigning a part of the global spare-time available at MAF level.
- Resolution will be done by assigning this spare-time on all the partitions affected by the conflict to be able to keep systematically a margin for the resolution of the following conflict.
- We guarantee the respect for the MAF on each Core while solving elementary conflicts at the level of each Core.
- Based on an upgrade of the Operating System ARINC653 standard authorizing the execution of the next partition without waiting for the maximal time computed from the use of the WCET of every service offered by the OS.
- Computation of the Spare-Time can be realized at – DESIGN TIME ➔ Under system Integrator responsibility
- Computation of Spare-Time can be realized at – Run TIME – ➔ Auto-learning, according to applications allocated to each Core;

Our approach can be deployed at the same time on a multi-core processor functioning in SMP mode (or BMP) in the case of the execution of a multi-threaded application or on a multi-core processor functioning in AMP mode.

SUMMARY

- Context
- Multicore Introduction
- Problem Statement
- Current Studies for IMA
- Overview of Potential SW sol.
- **Conclusion / future works**

CONCLUSION

In our approach we don't allow taking benefit from all the computation power offered by the processor, what is not the main objective.

Our approach allows offering the best ratio between Performance, Power dissipation, Size (SWAP)

- Promotes an approach which, based on the standard ARINC653 allows to respect the execution time at MAF I
- We adjust the execution time of each elementary partitions of each Core according to the encountered conflicts.

Our approach should be able to be favorably accepted by certification authorities

- It resumes their request of having a demonstration of independence between applications.
- Demonstration being supported by the Operating System and demonstrable at "DESIGN TIME" and/or at "Run TIME".


via
čas
time
PARIS
BRUSSELS
NAPOLI
LONDON
FRANKFURT
WIENNA
THANK YOU
BRUSSELS
MUNICH
AMSTERDAM
PARIS

QUESTIONS ?