D7.8 Roadmap of further research on the implications of parallel execution on the highest safety-critical systems

Version 1.0

Documentation Information

Contract Number	871669
Project Website	https://ampere.bsc.es/
Contractual Deadline	30.06.2022
Dissemination Level	PU
Nature	R
Author	Holger Blasum (SYS), Arne Hamann (BOS), Marco Merlini (THALIT), Massimilano Polito (THALIT), Michael Pressler (BOS), Ida Marina Savino (EVI), Claudio Scordino (EVI), Darshak Sheladiya (SYS), Thomas Vergnaud (TRT), Dirk Ziegenbein (BOS)
Reviewer	Tommaso Cucinotta (SSSA)
Keywords	Parallel execution, safety-critical systems

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

Change Log

Version	Description Change
V0.1	Drafted first version
V0.2	Version with all partner inputs
V1.0	Final version after consortium review

Table of Contents

Executive Summary

The European project AMPERE realized a full ecosystem supporting and easing the development of future high-performance and real-time embedded applications that require the non-functional requirements (such as time predictability, energy-efficiency, safety and security) inherited from the cyber-physical interactions, on heterogeneous architectures including multi-core, GPU and FPGA acceleration.

The objective of the project is to use the most advanced energy-efficient and highly-parallel heterogeneous platforms, to fully exploit the benefits of performance demanding emerging technologies, such as artificial intelligence or big data analytics. To reach such goal, AMPERE uses a combination of model-driven engineering (MDE) and parallel execution techniques, tackling important technical challenges in the two fields of system design and the computing software stack of CPSes, among others.

This deliverable describes experiences on how to map parallel execution to safety, as specified in standards, as well as a first collection / vision of future tasks to be addressed.

1. Introduction

We give a view of safety relevance of the parallel execution on the highest safety-critical systems by AMPERE industrial partners. We start with a general view of parallel execution and functional safety in general (Section 2). We then look closer at experiences from the work done by the industrial partner working with ERIKA OS (Section 3), in particular experiences from the ISO 26262 certification on the AURIX architecture (Section 3.1), the research work done on the AMPERE ARM platform (Section 3.2) and a positioning of open source in the AUTOSAR context (Section 3.3). This is followed by a view railway standards (Section 4), starting with continuous integration (Section 4.1) and a discussion of the implications for heterogeneous computing on railway standards (Section 4.2). A conclusion (Section 5) summarizes the experience and recommendations.

2. Parallel execution and functional safety in general

In this chapter we discuss the impact of the AMPERE toolchain and its capabilities with respect to the following aspects:

- How does the AMPERE toolchain simplify and speed-up the certification of safety critical applications?
- How does the AMPERE toolchain increase the design efficiency, and, thus, reduce the costs for complex system design?
- What are the learnings on how standards need to be adapted to support the AMPERE ecosystem?

These aspects are discussed in the subsequent sections.

2.1. Support for certification of safety critical applications

The following requirements were formulated in the beginning of the project.

> **[SYS-PCC-REQ-102]**: The HW/SW platform shall support the dynamic addition of applications (also during operation time) without jeopardizing the correct functioning of the existing applications deployed on the platform (performance compositionality).
>
> **[SYS-PCC-REQ-108]**: The HW/SW platform shall separate applications from each other spatially and temporally such that they do not interfere with each other, and such that cascading failures are prevented.
>
> **[SYS-PCC-REQ-110]**: The HW/SW platform shall warrant deployment-independent behaviour such that a distributed application exhibits the same deterministic input-output behaviour regardless of how it is deployed onto the platform.
>
> **[SYS-PCC-REQ-113]**: The AMALTHEA model shall be extended to cover the key characteristics of the CPSoS (software parallelism, accelerator offloading, and publish-subscribe middleware communication paradigms).

All these requirements target into the direction of simplifying the so-called “Safety of the intended functionality” (SOTIF) [ISO21448]. SOTIF is defined as the absence of unreasonable risk due to hazards resulting from functional insufficiencies of the intended functionality or by reasonably foreseeable misuse by persons.

The AMPERE toolchain does not explicitly aim at guaranteeing functional correctness and the avoidance of "functional deficiencies". However, the SW artifacts running on the supported hardware platforms adhere to the above requirements. This means that a large number of undesirable functional behaviours that may occur dynamically during system runtime, e.g., due to timing variations, can be excluded by design. This severely restricts the behavioural space of complex and highly parallel systems, which is the prerequisite for having any chance at all of certifying the system at a reasonable time and cost.

Examples of concrete technical approaches that are described in more details in the deliverables of the other work packages are for instance:

- Prevention of timing interference through reservation-based scheduling (SCHED_DEADLINE), that can also be used to isolate multiple OpenMP run-times among each other, and dynamic memory bandwidth regulation
- The data-flow analysis that supports the exploitation of intra-Task parallelism ensures data consistency even when moving data to accelerators via OpenMP features.
- Guaranteed freedom from race conditions in automatically parallelized code.

2.2. Increase of design efficiency

Programming highly parallel platforms with different heterogeneous computing units is a very complex and time-consuming endeavour, which is also very error-prone. Estimating the impact of decisions with respect to several criteria such as time and energy is practically impossible to be handled manually by a developer. Moreover, parallel programming of such systems requires a broad know-how of a number of different implementation technologies that hardly any developer can combine.

This is exactly where the AMPERE toolchain comes handy. It enables applications that are enriched with DSL descriptions to be deployed automatically and optimized with regard to multiple criteria on supported target platforms. This has the potential to close the productivity gap we currently see in the engineering of modern CPS and automotive applications. In this project this was demonstrated with the PCC use-case, where existing CPS systems are enhanced with a new component implementing a new advanced functionality.

2.3. Unification of tool interfaces

Dynamic execution behavior has to satisfy explicit constraints coming directly from functional system requirements, as well as indirect constraints coming from non-functional requirements (such as e.g. in a system with multiple applications to keep interference/resource conflicts from applications limited). Hence there is typically a need not only to analyze (1) explicit resource allocation (e.g. logical memory, communication devices) but also (2) indirect resource use (e.g. physical memory across the memory hierarchy, CPUs, system buses, including potential resource conflicts. Many languages for system specification and explicit resource constraints exist (to name a few relevant for the AMPERE project, these are Amalthea and Capella, but also AUTOSAR XML formats and PikeOS in AMPERE using its own XML format for assigning resources to partitions), and this information can also be automatically processed to further derive assurances of avoiding resource conflicts, e.g. by appropriate scheduling. Some of the tools try to avoid resource conflicts by exclusive resource allocation by default, such as PikeOS partitions, although controlled information flow can be set up (D5.4). Nonetheless, overall the tool landscape is still fragmented: At a low level, there would be a common foundation in the form of e.g. XML and EMF (Eclipse Modeling Framework).

2.4. Impact on standardization

The main relevant standards in the automotive domain are the AUTOSAR Classic and Adaptive platforms. From the point of view of the goals that AMPERE pursues, it is not necessary to extend these standards since they come along with all relevant features that are needed to build and deploy automotive systems. Rather, it is necessary to appropriately couple the AMPERE ecosystem into the tool landscape for the development of AUTOSAR systems to fully exploit the benefits offered in terms of productivity and simplification of certification. This is done in AMPERE mainly by extending the open-source framework APP4MC that is already well established in the tooling landscape of Bosch.

In the context of AMPERE, APP4MC has been extended to include the following concepts in order to be able to use the full scope of AMPERE capabilities in automotive systems development:

- Accelerator offloading
- Publish-subscribe middleware communication paradigms
- Intra-task software parallelism exploitation
- Energy awareness

In order to benefit from the advantages of the AMPERE Ecosystem in other application fields, a similar domain integration as achieved by APP4MC in the automotive sector must be undertaken. In the robotic field, for instance, the tooling landscape surrounding ROS2 is a promising starting point.

3. Parallel execution and functional safety in automotive ISO 26262

3.1. About ERIKA Enterprise ISO26262 qualification

ERIKA Enterprise is a RTOS designed and developed by Evidence Srl for the automotive domain. It was initially designed according to the OSEK/VDX specification, then according to the AUTOSAR Classic standard.

When the AMPERE project started, Evidence Srl did not have any experience on ISO26262 qualification of its own RTOS. The reason is that the RTOS used to be qualified by Evidence's customers (e.g. Magneti Marelli, Vodafone Automotive, Ferrari, etc.) who were paying the company for development and consulting activities. Thus, the AMPERE project was seen as a mean to cover such lack of know-how and an opportunity of exploitation, helping Evidence to get closer to the market needs.

Just after the project start, and after the acquisition of the partner Evidence by Huawei Technologies, the qualification of the RTOS was performed outside the AMPERE project thanks to the effort of the Evidence team and of the Huawei team in China. In particular, ERIKA has reached the highest level of safety qualification (i.e. ASIL-D) on the Infineon Tricore AURIX architecture. Although the qualification has been released by TUV in China, the involvement allowed the company to get a better understanding of the overall process and the needed documentation and tools. Indeed, Evidence has been involved in all the major activities:

- Implementation of the missing parts needed for ASIL-D (such as Memory, Timing and Service protection as specified in the AUTOSAR Classic OS Scalability Class 4);
- Documents and diagrams for high-level and low-level architecture design;
- Requirements tracking and mapping;
- MISRA-C checking: MISRA C is a set of software development guidelines for the C programming language developed by The MISRA Consortium. Its aims are to facilitate code safety, security,

portability and reliability in the context of embedded systems, specifically those systems programmed using the C programming language [Misra];
- Unit testing with coverage analysis;
- Integration testing;
- Failure Mode and Effects Analysis (FMEA) process, i.e. a systematic approach to identify failures in a process, product or service, etc.

3.2. AMPERE paving the way for further certification

The Infineon Tricore AURIX chip used for ASIL-D qualification of ERIKA is suitable for making an automotive Electronic Control Unit (ECU) based on AUTOSAR Classic standard, but is not suitable for creating the innovative platform envisioned in the AMPERE project. Indeed, it does not support a hypervisor, nor can it run a general-purpose OS like Linux.

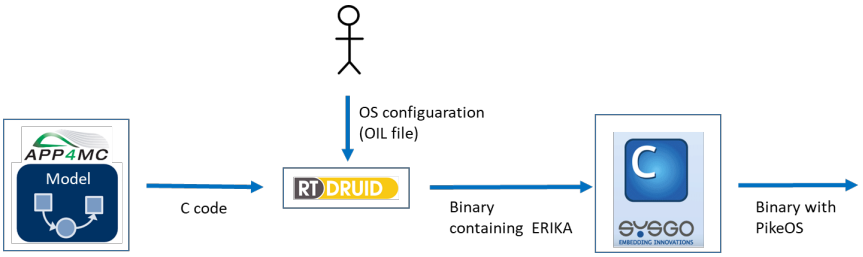
The reference platform selected for the AMPERE project (i.e. Xilinx ZCU102) is very flexible and contains different types of processing cores: quad-core Cortex-A, Cortex-R5 and a FPGA that can be used to synthesize soft-cores (e.g. RISC-V). Such platform, however, lacks some hardware supports (e.g. lockstep) to reach the highest automotive safety-level (i.e. ASIL-D). Nevertheless, we believe that it is an excellent playground to investigate qualification aspects of Multi-OS software architectures even at a lower SIL level (e.g. ASIL-A/B).

In AMPERE, we have designed a mixed-criticality run-time software architecture consisting of the following components:

- The **PikeOS** hypervisor, developed by partner SYSGO;
- **ELinOS**, a distribution of the general-purpose OS Linux, provided by partner SYSGO and running as guest VM of PikeOS;
- **ERIKA Enterprise** RTOS, running as guest VM of PikeOS;
- The **Micro-ROS** open-source library (https://micro.ros.org), based on eProsima's XRCE-DDS library (https://github.com/eProsima/Micro-XRCE-DDS), for handling the communications between Linux and ERIKA Enterprise.

As shown in the following picture, the off-line toolchain, instead, consists of the following components:

- Eclipse **APP4MC** (https://www.eclipse.org/app4mc/), containing the Amalthea tool, for application modeling and generation of the C code. This tool has been modified to also generate code for the ERIKA RTOS.
- The **RT-Druid** tool, for the generation of the binary code of ERIKA Enterprise from the C code generated by Amalthea. The RTOS configuration is manually provided by the user in the form of an OIL file. It would be interesting to enhance the APP4MC tool to automatically generate this configuration file.
- The **Codeo** tool, developed by partner SYSGO, for the generation of the binary tool including PikeOS.



The activity on qualification of ERIKA Enterprise on such complex Multi-OS software stack should follow a ***compositional* approach**, taking into account the overall architecture, listing assumptions and constraints of the various components and setting the safety goals for the RTOS.

In particular, given the isolation properties provided by the PikeOS hypervisor with static partitioning, the ERIKA Enterprise RTOS could be qualified at the highest safety level allowed by the specific hardware. Such qualification **should also take into account the library used for communication towards Linux** (i.e. Micro-ROS). Finally, the existing documents for the qualification of the RT-Druid tool could be properly extended to take into account this more complex architecture and the toolchain of AMPERE.

3.3. Open-source and ISO26262 qualification: opportunities vs issues

ERIKA Enterprise used to be provided on GitHub as open-source software (https://github.com/evidence/erika3) until the AUTOSAR Consortium in 2019 argued that such licensing was conflicting with the exploitation clause of the consortium. Prevented from opening the source code of the RTOS, Evidence has then worked for releasing the source code at least to the AUTOSAR members. This activity has led to the set-up of the "**Open-ERIKA**" project [ETFA2023], which aims at releasing the RTOS and the needed tools open to the needing partners, for free usage for educational and commercial purposes.

The project plans to have a first working release around Q1 2024, including the same codebase of the GitHub code, with a novel code generator based on ARTOP and XText. Further developments as well as support for additional architectures will be planned afterwards. From the first contact with the old ERIKA ecosystem, Evidence has recorded a growing interest for an “AUTOSAR Classic Demonstrator” which could be used as a reference implementation for further standard development, as a “demo space” for Tier2 (ERIKA has been already used to demonstrate Compilers, Debuggers, Validation Tools, Hypervisors in the past). In particular, the Open-ERIKA project represents an excellent opportunity of growth for all academic and research institutions (including the ones of the AMPERE project) that can join the AUTOSAR Consortium for free and thus get immediate access to a state-of-the-art AUTOSAR Classic stack, and as well for startups that, through the AUTOSAR Partnership, can get access to an AUTOSAR Classic OS codebase for their projects.

The code released on the AUTOSAR repositories of course will not include ISO26262 qualification. However, we believe that such disclosure will create business opportunities for Evidence also in terms of functional safety. AUTOSAR partners will be free to evaluate and test the RTOS for free, and then ask for a qualified commercial version of the software. Moreover, such disclosure will represent an excellent frame for setting up joint "win-win" collaborations between Evidence and other AUTOSAR partners, to split the effort and costs for qualification. In the end, the open-source strategy is expected to enlarge not only the number of users, but also the number of qualified versions of the RTOS.

4. Parallel execution and functional safety in railway

4.1. Continuous integration

The starting point of the AMPERE workflow is typically an Amalthea model representing the system. Such a model is rather low-level, as it contains the details of the execution flows and the interactions points

between them. Because it is low-level, an Amalthea model is not convenient as a starting point of a complete engineering process: engineers would not create Amalthea models by hand.

The combination of Capella with Amalthea was a first experiment to study how to produce a low-level model (Amalthea) from a high-level system architecture model (made with Capella). Capella is widely used in the Thales system engineering workflow.

A Capella model typically lacks some information that is necessary at the Amalthea level. Typically, Amalthea requires the specification of CPU ticks for each *runnable*; Capella does not natively handle such an information. In the context of AMPERE, TRT made a lightweight extension to Capella to enable the specification of CPU ticks as annotations within a Capella model. In order to fully integrate the AMPERE works within a Thales system design workflow, it would be necessary to develop a convenient Capella *point of view* to enable the proper specification of CPU ticks.

Enhancing Capella to enable a more seamless integration with the AMPERE tools would enable the use of the AMPERE workflow in a continuous system engineering integration chain at Thales.

4.2. Implications of execution on heterogeneous computing processors for the functional safety aspects defined in the IEC61508, EN 5012X standards

A thorough analysis of safety standards has been reported in section 2.1.1 of deliverable “D1.4 Analysis of functional safety aspects on multi-criteria optimization and final release of the test bench suite”. In the following, only some conclusions adapted to this section context have been reported.

International Safety standards don’t address the topic of SW functional safety in relation to the number of processors/cores SW is executed on. Standards focus instead on three main aspects to ensure delivered SW can meet its safety requirements:

- Company processes that have been set up to design, implement and validate SW and HW implementing safety functions must be compliant to guidelines contained in safety standards;
- Company organization must adhere to guidelines contained in safety standards to ensure independence between people involved in design/implementation and people involved in validation of any artifact implementing safety functions;
- SW implementing safety functions must be designed, implemented and validated according to provisions contained in EN50128.

This being said, environments with heterogeneous computing processors are already used nowadays in railway systems with the goal of implementing safety functions.

As an example, the railway SIL3 certified Interlocking System produced by THALIT (named “UCS”) uses a 2oo2 (two-out-of-two) redundant architecture to reduce the rate of failures that could cause accidents and ensure proper safety level is met on the hardware side.

Basically, UCS uses two CPUs boards running two different software that implement the same functions. A module checks that both boards produce the same output (i.e. both boards send the same commands to field equipment). If a failure happens in any of the two boards, the module checking those outputs is able to trigger a UCS transition to a safe state and the UCS will stop UCS and move it to a stay there, that is a safe state waiting for maintenance intervention.

Another example is Thales Axle Counter. The Axle Counter is a SIL4 certified equipment that uses a 2oo3 (two-out-of-three) redundant architecture. In this case, three CPUs run the same SW in parallel and two of them must give the same result for the output to be asserted.

Equipment using this kind of parallelism can be SIL certified by a safety assessor based on current safety standards.

An analysis could be useful to assess if safety improvements could also be met by using parallel computing instead of redundant architectures.

However, when parallel computing is used (whether to improve performance or with the goal of improving safety) some new problems arise that should be managed from a safety perspective (among others, race conditions and deadlocks). These issues at the moment don't seem to be fully covered by software standard EN 50128. A road map should be traced to update these standards so that those issues could also be addressed.

5. Conclusion

5.1. State of the art

AMPERE partners have used lock-step / redundancy approaches redundancy to reach high ASIL (Section 3.2) and SIL levels (Section 4), however on relatively simple systems.

AMPERE partners have pointed out that special and temporal non-interference (in particular [SYS-PCC-REQ-108] in Section 2) are important requirements of safety-critical systems, and are required in many standards [NB17]. Some more recent safety standards such as the avionic multicore guidance [EASA22] explicitly demand for analysis of absence/mitigation of interference even for statically configured systems.

5.2. Potential of AMPERE approach for verification of determinism of complex systems

AMPERE has worked on feature-rich ARM multicore SoCs (e.g. Xilinx ZCU 102), that expose more resources, provide multiple computation cores, and potentially also have a good infrastructure for dynamic update [AbCu20, ACM22, TECS23, CPBD22, CuAb21, Cuci20, ERAB20, FMKM20, KQTZ21, Lcte00, MBFB22, MCMA21, MiRQ21, MQPHZR22, MRFPQ, QRHZ00, QRSG20, SCAO21, SMRH21, SPBB21, YuRQ20a-c, YuRQ21]. AMPERE aims at establishing determinism even for such feature-rich systems, including prevention of timing interference through the use of a separation kernel with controlled communication channels [D5.4] reservation-based scheduling (SCHED_DEADLINE) and dynamic memory bandwidth regulation, ensured data consistency when moving data to accelerators and guaranteed freedom from race conditions in automatically parallelized code [D3.3].

That is, the AMPERE toolchain shows technical potential to make feature-rich multicore SoCs more controllable. This is also applicable across domains (e.g. automotive and railway).

AMPERE partners consider the existing non-interference requirements in the automotive and railway standards such as AUTOSAR, ISO 26262, EN 50128 to be sufficient to justify the use of the AMPERE toolchain, which makes the determinism of more simple sequential systems available to richer and more parallel systems.

Our recommendations are the following:

- It has been observed, that for complex systems, a compositional approach is needed for certification, taking into account the overall architecture, listing assumptions and constraints of the various components and setting the safety goals of the composed architecture (Section 3.2).
- It had been observed (Section 4.2) that new problems arise that should be managed from a safety perspective (among others, race conditions and deadlocks). These issues at the moment don't seem to be fully covered by railway software standard EN 50128. It also has been pointed out that

this not necessarily needs to be done by expanding the normative sections of the safety standards (Section 2.4). However, the non-normative sections (guidance) of safety standards could be improved.

- A good way to achieve extensions and supplements to existing standards could be a new (e.g. Horizon Europe) research call (possibly also community support action) for precisely collecting state-of-the-art and proposing such amendments to standards. This in particular, as the technologies are cross-domain, could be done in a cross-domain fashion.
- As described in Section 2.3, beyond the semantics of component-and-connector models, it is needed to create a common semantics to express invariants of appropriately constrained dynamic execution behavior, including temporal guarantees, non-functional properties (e.g. separation and independence from unintended interference) that might be result from resource allocation based on non-interference analysis. A common language could be developed specifically for specifying dynamic execution behavior and elements of assurance cases (safety cases/security cases) across different usage domains as both design and verification tools.

6. Acronyms and Abbreviations

AI	Artificial Intelligence
AMPERE	A Model-driven development framework for highly Parallel and EneRgy-Efficient computation supporting multi-criteria optimization
API	Application Program Interface
AUTOSAR	Automotive Open System Architecture
COTS	Commercial Off-The-Shelf
CPS	Cyber Physical System
CPSoS	Cyber Physical System of Systems
CPU	Central Processing Unit
DMP	Data Management Plan
DPR	Dynamic Partial Reconfiguration
DSML	Domain-Specific Modeling Language
EMF	Eclipse Modeling Framework
FMEA	Failure Modes and Effects Analysis
FPGA	Field Programmable Gate Array
GPU	Graphics Processing Unit
HAL	Hardware Abstraction Layer
HERO	Heterogenous Research Platform
IPR	Intellectual Property Rights
MDE	Model-Driven Engineering
MISRA	Motor Industry Software Reliability Association
MPSoC	Multiprocessor SoC
OEM	Original Equipment Manufacturer
OS	Operating System
RTOS	Real-Time Operating System
SoC	System-on-Chip
SOTIF	Safety of the Intended Functionality
XML	Extensible Markup Language

7. References

[AbCu20] Abeni, Luca; Cucinotta, Tommaso: Adaptive partitioning of real-time tasks on multiple processors. In: Proceedings of the 35th Annual ACM Symposium on Applied Computing. Brno Czech Republic : ACM, 2020 — ISBN 978-1-4503-6866-7, p. 572–579

[ACM22] Ara, Gabriele; Cucinotta, Tommaso ; Mascitti, Agostino: Simulating execution time and power consumption of real-time tasks on embedded platforms. In: Proceedings of the 37th ACM/SIGAPP Symposium on Applied Computing. Virtual Event : ACM, 2022 — ISBN 978-1-4503-8713-2, p. 491–500

[TECS23] T. Cucinotta, A. Amory, G. Ara, F. Paladino, M. Di Natale. "Multi-Criteria Optimization of Real-Time DAGs on Heterogeneous Platforms under P-EDF," ACM Transactions on Embedded Computing Systems, 2023

[AUTOSAR] AUTOSAR “AUTomotive Open System ARchitecture” consortium, www.autosar.org

[CPBD22] Casini, Daniel; Pazzaglia, Paolo ; Biondi, Alessandro ; Di Natale, Marco: Optimized Partitioning and Priority Assignment of Real-Time Applications on Heterogeneous Platforms with Hardware Acceleration. In: Journal of Systems Architecture Bd. 124 (2022), — arXiv:2205.06773 [cs]

[CuAb21] Cucinotta, Tommaso ; Abeni, Luca: Migrating Constant Bandwidth Servers on Multi-Cores. In: 29th International Conference on Real-Time Networks and Systems. NANTES France : ACM, 2021 — ISBN 978-1-4503-9001-9, p. 155–164

[Cuci20] Cucinotta, Tommaso: Model-based engineering of high-performance embedded applications on heterogeneous hardware with real-time constraints and energy efficiency (2020), p. 20

[EASA22] EASA. (2022). AMC 20-193 Use of multi-core processors. https://www.easa.europa.eu/sites/default/files/dfu/annex_i_to_ed_decision_2022-001-r_amc_20-193_use_of_multi-core_processors_mcps.pdf

[EN50126] EN 50126 - Railway Applications, https://standards.globalspec.com/std/260302/EN%2050126

[EN50128] EN 50128 - Railway Applications, https://standards.globalspec.com/std/14317747/EN%2050128

[EN50128] EN 50129 - Railway Applications, https://standards.globalspec.com/std/13113133/EN%2050129

[ERAB20] Economo, Simone ; Royuela, Sara ; Ayguadé, Eduard ; Beltran, Vicenç: A Toolchain to Verify the Parallelization of OmpSs-2 Applications. In: Malawski, M. ; Rzadca, K. (Hrsg.): Euro-Par 2020: Parallel Processing, Lecture Notes in Computer Science. Bd. 12247. Cham : Springer International Publishing, 2020 — ISBN 978-3-030-57674-5, p. 18–33

[FMKM20] Forsberg, Björn ; Mattheeuws, Maxim ; Kurth, Andreas ; Marongiu, Andrea ; Benini, Luca: A Synergistic Approach to Predictable Compilation and Scheduling on Commodity Multi-Cores. In: The 21st ACM SIGPLAN/SIGBED Conference on Languages, Compilers, and Tools for Embedded Systems. London United Kingdom : ACM, 2020 — ISBN 978-1-4503-7094-3, p. 108–118

[IEC61508] IEC 61508 - Functional Safety, http://www.iec.ch/functionalsafety/

[ISO21448] ISO/PAS 21448:2019, Road vehicles — Safety of the intended functionality

[ISO26262] ISO 26262-1:2011, Road vehicles - Functional safety

[KQTZ21] The OpenMP API for high integrity systems: Moving responsibility from users to vendors. M Klemm, E Quiñones, T Taft, D Ziegenbein, S Royuela. ACM SIGAda Ada Letters 40 (2), 48-50.

[Lcte00] LCTES 2020 A Synergistic Approach to Predicatable Compilation and Scheduling on Commodity Multi Core - YouTube. URL https://www.youtube.com/watch?reload=9&v=n4pyvDcbCe4&list=PLyrlk8Xaylp4KF3J_svUstmH-l9RERLDV&index=3&t=0s.

[MBFB22] Mazzola, Sergio; Benz, Thomas; Forsberg, Björn; Benini, Luca: „A Data-Driven Approach to Lightweight DVFS-Aware Counter-Based Power Modeling for Heterogeneous Platforms“, SAMOS 2022

[MCMA21] Mascitti, Agostino ; Cucinotta, Tommaso ; Marinoni, Mauro ; Abeni, Luca: Dynamic partitioned scheduling of real-time tasks on ARM big.LITTLE architectures. In: Journal of Systems and Software Bd. 173 (2021)

[MiRQ21] Miguel Pinho, Luis ; Royuela, Sara ; Quiñones, Eduardo: Real-time Issues in the Ada Parallel Model with OpenMP. In: ACM SIGAda Ada Letters Bd. 40 (2021), Nr. 2, p. 96–102

[Misra] Misra consortium, https://www.misra.org.uk/

[MQPHZR22] A. Munera, E. Quiñones, M. Pressler, A. Hamann, D. Ziegenbein, S. Royuela. Increasing CPS Productivity and Resiliency through Accelerated Software Replication. International Conference on Reliable Software Technologies (AEiC 2022)

[MRFPQ] A Munera, S Royuela, R Ferrer, R Peñacoba, E Quiñones. International Conference on High Performance Computing, 19-33. 2020.

[NB17] Nordhoff, S., & Blasum, H. (2017). Ease standard compliance by technical means via MILS. In S. Tverdyshev (Ed.), International workshop on MILS: architecture and assurance for secure systems, MILS 2017, Nürnberg, Germany, March 14, 2017. Zenodo. https://doi.org/10.5281/zenodo.571175

[QRHZ00] Quiñones, Eduardo ; Royuela, Sara ; Hamann, Arne ; Ziegenbein, Dirk ; Fosberg, Björn ; Benini, Luca ; Scordino, Claudio ; Gai, Paolo ; u. a.: A Model-driven development framework for highly Parallel and EneRgy-Efficient computation supporting multi-criteria optimisation

[QRSG20] Quinones, Eduardo ; Royuela, Sara ; Scordino, Claudio ; Gai, Paolo ; Pinho, Luis Miguel ; Nogueira, Luis ; Rollo, Jan ; Cucinotta, Tommaso ; u. a.: The AMPERE Project: : A Model-driven development framework for highly Parallel and EneRgy-Efficient computation supporting multi-criteria optimization. In: 2020 IEEE 23rd International Symposium on Real-Time Distributed Computing (ISORC). Nashville, TN, USA : IEEE, 2020 — ISBN 978-1-72816-958-3, p. 201–206

[SCAO21] Stevanato, Andrea ; Cucinotta, Tommaso ; Abeni, Luca ; de Oliveira, Daniel Bristot: An Evaluation of Adaptive Partitioning of Real-Time Workloads on Linux. In: 2021 IEEE 24th International Symposium on Real-Time Distributed Computing (ISORC). Daegu, Korea (South) : IEEE, 2021 — ISBN 978-1-66540-414-3, p. 53–61

[SMRH21] Saeed, Ahsan ; Mueller-Gritschneder, Daniel ; Rehm, Falk ; Hamann, Arne ; Ziegenbein, Dirk ; Schlichtmann, Ulf ; Gerstlauer, Andreas: Learning based Memory Interference Prediction for Co-running Applications on Multi-Cores. In: 2021 ACM/IEEE 3rd Workshop on Machine Learning for CAD (MLCAD). Raleigh, NC, USA : IEEE, 2021 — ISBN 978-1-66543-166-8, p. 1–6

[SPBB21] Seyoum, Biruk ; Pagani, Marco ; Biondi, Alessandro ; Buttazzo, Giorgio: Automating the design flow under dynamic partial reconfiguration for hardware-software co-design in FPGA SoC. In: Proceedings of the 36th Annual ACM Symposium on Applied Computing. Virtual Event Republic of Korea : ACM, 2021 — ISBN 978-1-4503-8104-8, p. 481–490

[YuRQ20a] Yu, Chenle ; Royuela, Sara ; Quinones, Eduardo: OpenMP static TDG runtime implementation and its usage in Heterogeneous Computing

[YuRQ20b] Yu, Chenle ; Royuela, Sara ; Quiñones, Eduardo: OpenMP to CUDA graphs: a compiler-based transformation to enhance the programmability of NVIDIA devices. In: Proceedings of the 23th

International Workshop on Software and Compilers for Embedded Systems. St. Goar Germany : ACM, 2020 — ISBN 978-1-4503-7131-5, p. 42–47

[YyRQ20c] Static analysis to enhance programmability and performance in OmpSs-

[YuRQ21] Yu, Chenle ; Royuela, Sara ; Quiñones, Eduardo: Enhancing OpenMP Tasking Model: Performance and Portability. In: McIntosh-Smith, S. ; de Supinski, B. R. ; Klinkenberg, J. (Hrsg.): OpenMP: Enabling Massive Node-Level Parallelism, Lecture Notes in Computer Science. Vol. 12870. Cham : Springer International Publishing, 2021 — ISBN 978-3-030-85261-0, p. 35–49