Analyses and Architectures for Mixed-Critical Systems: Industry Trends and Research Perspective

Special Session Extended Abstract

Lars Bauer[1], Marvin Damschen[1], Dirk Ziegenbein[2], Arne Hamann[2], Alessandro Biondi[3], Giorgio Buttazzo[3], Jörg Henkel[1]
[1]Karlsruhe Institute of Technology, Karlsruhe, Germany
[2]Robert Bosch GmbH, Renningen, Germany
[3]Scuola Superiore Sant'Anna, Pisa, Italy
{lars.bauer,henkel}@kit.edu,dirk.ziegenbein@de.bosch.com,giorgio@sssup.it

ABSTRACT

The ongoing Cyber-Physical Systems (CPS) and Internet of Things (IoT) revolution is leading to the ubiquity of applications with real-time requirements. In traditional real-time applications, dedicated and isolated systems were built for each safety-critical functionality, but this paradigm no longer suits emerging applications. The automotive industry is a prime example of a domain that is currently undergoing the radical shift to integrated mixed-criticality systems. A heterogeneous mix of applications with diverse requirements in terms of timing and safety criticality is run on hardware platforms that are increasingly heterogeneous, including specialized hardware accelerators. Examples of such systems are provided in this special session along with reviews of challenges and current research directions in industry. Ways to utilize the performance of application-specific accelerators while maintaining the system's flexibility using runtime reconfiguration are presented. It will benefit university researchers/professors, students, industry professionals, and computing system designers who are interested in working on mixed-critical systems.

CCS CONCEPTS

• **Computer systems organization → Real-time system architecture**; • **Hardware → Hardware accelerators**.

KEYWORDS

Mixed Critical Systems, Automotive, Reconfigurable Systems

ACM Reference Format:
Lars Bauer[1], Marvin Damschen[1], Dirk Ziegenbein[2], Arne Hamann[2], Alessandro Biondi[3], Giorgio Buttazzo[3], Jörg Henkel[1]. 2019. Analyses and Architectures for Mixed-Critical Systems: Industry Trends and Research Perspective: Special Session Extended Abstract. In *2019 International Conference on Embedded Software Companion (EMSOFT'19 Companion), October 13–18, 2019, New York, NY, USA.* ACM, New York, NY, USA, 2 pages. https://doi.org/10.1145/3349568.3356997

EMSOFT'19 Companion, October 13–18, 2019, New York, NY, USA

ACM ISBN 978-1-4503-6924-4/19/10…$15.00
https://doi.org/10.1145/3349568.3356997

1 INTRODUCTION

Examples of the ubiquity of emerging IoT applications with real-time requirements include infrastructure (e.g., smart grids), health care (e.g., seizure prediction), transportation (e.g., automated driving), industrial automation (e.g., robotics) and many more that touch all aspects of our everyday lives. While in traditional real-time applications dedicated systems were built for each safety-critical functionality, this paradigm no longer suits emerging applications. The reason is that apart from traditional real-time requirements like a guaranteed worst-case execution time (WCET), tremendous performance requirements need to be fulfilled under additional non-functional constraints like power consumption, cost, space, and weight. To fulfill these requirements, integrated *mixed-criticality systems* are designed instead of a dedicated system for each functionality. This integration entails that safety-critical tasks of one functionality are no longer run in isolation, but they rather share the same computing platform with low-critical or even non-critical software.

A prime example of a domain that is currently undergoing the radical shift from the design of numerous isolated safety-critical systems to integrated mixed-criticality systems is the automotive industry. Where traditional automotive architectures consist of more than 100 single-function electronic control units, the industry is now moving to more centralized computing platforms. Thus, a heterogeneous mix of applications with different models of computation and with diverse requirements in terms of timing and safety criticality is run on hardware platforms that are increasingly heterogeneous, including specialized hardware accelerators, e.g., for deep learning.

When falling for old habits and designing mixed-critical systems with numerous fixed and single-function accelerators, however, the system becomes inflexible while the non-functional constraints are not fulfilled effectively. Instead, we will present ways to utilize the performance of application-specific accelerators in this special session, while maintaining the system's flexibility using runtime reconfiguration. The crux of runtime reconfiguration in mixed-critical system design is to unite a changing set of specialized hardware accelerators, which are shared between tasks of different criticality, with WCET and schedulability guarantees without compromising performance benefits.

We will target this problem from two directions in this special session:

(i) real-time analysis and predictable runtime mechanisms, and **(ii)** hardware architectures.

We will further highlight how advances in one direction lead to opportunities and challenges in the other. From this context, the vast opportunities and challenges of major architectural trends like heterogeneous multi-processor system-on-chips in the design of mixed-critical systems will be addressed in this special session. This special session will benefit university researchers/professors, students, industry professionals, and computing system designers who are interested in working on mixed-critical systems. And clearly, in the light of the increasing industrial demand and academic advances, this is an excellent time to revisit mixed-critical systems design in a special session.

2 MIXED-CRITICALITY SYSTEMS IN PRACTICE: STATUS & CHALLENGES

The dominant architectural trend in the automotive industry is the integration of formerly separated function domains onto centralized computing platforms. This leads to a heterogeneous mix of applications with diverse extra-functional requirements regarding safety, security and timing being implemented on heterogeneous, specialized hardware platforms (comprising, e.g., application cores, safety cores, GPUs, deep learning accelerators). We will provide examples of such systems in this special session and give insights in current methods on how to integrate applications with different levels of criticality safely on a single platform. Furthermore, challenges and current research directions pursued in industry are presented.

3 DEVELOPMENT AND ANALYSIS OF REAL-TIME APPLICATIONS ON HETEROGENEOUS FPGA-BASED SOC

FPGA-based System-on-Chips (SoC) are promising heterogeneous platforms to develop next-generation CPS that require both high-performance computation and execution predictability. Nevertheless, considerable challenges have to be addressed to properly exploit their computing capabilities with the end of developing real-time applications, ranging from software modeling, scheduling, and analysis methods to resource management of programmable logic deployed on FPGA fabrics. We present analysis techniques and runtime resource management mechanisms for FPGA-based SoC in this special session that have been conceived to address a number of such challenges.

A programming model for heterogeneous software running on FPGA-based SoC is illustrated, including both software tasks running on CPUs and hardware accelerators deployed on the FPGA fabric. A set of scheduling mechanisms are covered in the special session, conceived to ensure predictable delays when exploiting hardware acceleration on reconfigurable FPGAs, and their implementation and integration in Linux. Then, an approach is presented to address the problem of performing timing- and resource-aware automated floorplanning of hardware accelerators to optimize a set of performance metrics. The integration of the floorplanning approach within commercial tools by Xilinx is also discussed.

We also illustrate budgeting techniques for isolating mixed-criticality hardware accelerators when contending the system bus in the special session, also revealing potential denial-of-service attacks that are possible of modern reconfigurable FPGAs. Timing analysis of the delays introduced by bus contention and the automated synthesis of the bus hierarchy to match timing constraints are also addressed. And finally, the design and the implementation of a hypervisor layer is presented in the special session to realize mixed-criticality applications on the Xilinx Zynq UltraScale+ platform.

4 RUNTIME-RECONFIGURABLE ARCHITECTURES FOR WCET GUARANTEES AND MIXED CRITICALITY

Real-time systems require tasks to be analyzable for worst-case execution time (WCET) guarantees. However, WCET analysis lags years behind current microarchitectures with out-of-order scheduling pipelines, several hardware threads and multiple (shared) cache layers. To satisfy the increasing performance demands of emerging real-time applications, microarchitectures with analyzable performance features are required [2, 3]. We will first introduce processors with a runtime-reconfigurable instruction set that employs hardware accelerators via an integrated FPGA as one way to achieve high performance that is amenable for WCET guarantees [1, 3].

We will then discuss that scheduling real-time tasks of a single criticality generally result in underutilization of accelerators. This provides an opportunity to design hardware architectures that enable resource sharing between safety-critical real-time tasks and non-critical best-effort tasks: while real-time tasks have guaranteed access to the accelerators, best-effort tasks can utilize the accelerators for speedups on a best-effort basis. This is the governing idea of our hardware architecture that enables mixed-criticality accelerator sharing on commercial-of-the-shelf platforms like Xilinx Zynq UltraScale+. It further enables tasks to submit data-flow graphs of hardware accelerators that are executed out-of-order while fulfilling data dependencies using a *token system* that we will detail in this special session. Finally, we will show that out-of-order execution of accelerators not only benefits best-effort tasks, but can also lower the guaranteed WCET of real-time tasks.

ACKNOWLEDGMENTS

Partially funded by the Deutsche Forschungsgemeinschaft (DFG, German Research Foundation) – Project Number 146371743 – TRR 89 Invasive Computing.

REFERENCES

[1] Marvin Damschen, Lars Bauer, and Jörg Henkel. 2016. Extending the WCET Problem to Optimize for Runtime-Reconfigurable Processors. *ACM Transactions on Architecture and Code Optimization (TACO)* 13, 4, Article 45 (Dec. 2016), 24 pages. https://doi.org/10.1145/3014059

[2] Marvin Damschen, Lars Bauer, and Jörg Henkel. 2017. CoRQ: Enabling Runtime Reconfiguration Under WCET Guarantees for Real-Time Systems. *IEEE Embedded Systems Letters (ESL)* 9, 3 (Sept. 2017), 77–80. https://doi.org/10.1109/LES.2017.2714844

[3] Marvin Damschen, Lars Bauer, and Jörg Henkel. 2017. Timing Analysis of Tasks on Runtime Reconfigurable Processors. *IEEE Transactions on Very Large Scale Integration (VLSI) Systems* 25, 1 (Jan. 2017), 294–307. https://doi.org/10.1109/TVLSI.2016.2572304