Design Challenges in Compute-intensive Mixed-criticality Systems: System, Platform and Application Perspectives

Date: **4th Oct, 2015**
Place**: Amsterdam, the Netherlands**
Forum**: Embedded Systems Week 2015**

Abstract: Criticality is a designation of the level of assurance against failure needed for a system component. Complex embedded systems in domains like automotive and healthcare are evolving into mixed-criticality systems (MCS) in order to meet stringent non-functional requirements relating to cost, quality, safety etc. The criticality levels classify various functionalities as safety-critical, time-critical or non-critical. This tutorial focuses on a class of MCS that involves functionalities requiring compute-intensive processing as well as low-latency time-/safety-critical applications (e.g., feedback control loops). Common examples include advanced driver assistance systems (ADAS) in the automotive domain or interventional x-ray (iXR) systems in the healthcare domain where compute-intensive image and video processing execute together with or as part of safety-critical feedback control tasks. The platforms are migrating from single core to multi-core and, more recently, to many-core architectures with various forms of hardware accelerators (e.g., CPUs, GPUs, DSPs and FPGAs). At the system-level, an important design decision is the right choice of the platform architecture which further involves translating the application-level requirements into the platform-level requirements. Resource sharing among applications is important from a cost perspective. All this requires a clear view on application-level, platform-level as well as system-level aspects.

This tutorial is composed of three parts: System perspective (Part 1), Platform perspective (Part 2) and Application perspective (Part 3). At the system-level, we will consider self-driving vehicles and advanced driver assistance systems as the driving example to explore challenges for next-generation compute-intensive mixed-criticality application and system design. Architectural design and deployment challenges will be illustrated based on an industrial use-case stemming from the collaboration between TNO (www.TNO.nl) and NXP (www.nxp.com), and extended with results of TNO's research in the European Artemis project EMC2 (artemis-emc2.eu, grant no. 621429). Next, at the platform-level, we will address the theoretical and practical trade-offs in computer system organization and the latest developments and trends in computer architecture. This part will particularly draw its motivation from the healthcare domain, iXR in particular, and illustrate results from the ongoing European Artemis project ALMARVI (almarvi.eu, grant no. 621439). This includes a demonstration and a hands-on tutorial, where the attendants will be able to carry out a couple of exercises on a heterogeneous multicore infrastructure. Finally, at the application-level, we will illustrate a number of design aspects for predictable implementation of feedback control applications on multi-core architectures. Further, we will discuss various performance and trade-offs analysis methods for embedded control systems while considering shared implementation platforms. Results will be shown from the ongoing activities under both the EMC2 and ALMARVI projects.

Acknowledgement: This tutorial and the presented work is supported by ARTEMIS projects 621429 EMC2 and 621439 ALMARVI.

Part 1 (Speaker: Teun Hendriks, TNO):

- How to translate user requirements to system requirements
- Methods to perform Design Space Exploration for mixed-criticality application on shared computational platforms with the Y-chart paradigm.
- Tool support for Design Space Exploration

Part 2 (Speaker: Zaid Al-Ars, Delft University of Technology):

- Identify available computational hardware alternatives and their advantages and disadvantages, bottlenecks in the target applications in terms of throughput, latency, communication, etc.
- Select most appropriate computational hardware platform to address the limitations in the target application
- Use advanced characteristics of the computational platform to optimize the performance of the target application

Part 3 (Speaker: Dip Goswami, Eindhoven University of Technology):

- Analysis, synthesis and design challenges: control and streaming applications implemented on shared multi-core platforms.
- Predictable implementation of feedback control loops on multi-core platforms
- Performance and trade-off analysis: quality of control, resource usage and allocation

Target

This tutorial is targeted towards an audience with a background in embedded systems design and interested in compute-intensive mixed-criticality applications and system design challenges. The level of the tutorial will be from introductory to intermediate. No background in advanced system-level design techniques and control theory will be assumed. The material to be presented will be useful to researchers from the embedded systems domain, doctoral students, and software developers who are interested in various system-, platform- and application-level methods and design questions in similar contexts.

Speakers

Teun Hendriks is currently a senior research fellow at TNO Embedded Systems Innovation (www.TNO.nl, www.esi.nl) with a large industrial experience in the development of compute-intensive embedded systems and products. His current research focuses on (compute-intensive) systems in context, and associated information-centric architectures and embedded intelligence to assure operational performance with necessary quality, reliability, and robustness characteristics. Prior to joining TNO and the Embedded Systems Institute, he lead the integration of Traffic and Travel Information into vehicle navigation systems for Philips and later Siemens VDO. Teun Hendriks is also an independent consultant for Traffic and Travel Information services, and member of the steering board of TISA (www.tisa.org): home of the TMC and TPEG ISO standards.

Zaid Al-Ars is an assistant professor at the Computer Engineering Lab of the Delft University of Technology (TUDelft), where he leads the research activities of the multi/many-core research theme. His work focuses on addressing the bottlenecks in heterogeneous multicore architectures and proposing optimized solution alternatives for system performance, memory, power, reliability, etc. Prior to joining the TUDelft, he spent a number of years in the Product Engineering Group of Infineon Technologies and Siemens Semiconductors in Munich, Germany. He is a co-founder of two startups in the field HPC for compute-intensive applications and computer technology.

Dip Goswami is currently an assistant professor in Electronic Systems (ES) group of Electrical Engineering in Eindhoven University of Technology (TU/e). He obtained his Ph.D. in Electrical and Computer Engineering from the National University of Singapore (NUS) in 2009. During 2010-2012, he was an Alexander von Humboldt Postdoctoral Fellow at TU Munich, Germany. He research interests are in the direction of embedded control systems, cyber-physical systems and robotics. Over the last few years, he is particularly working on efficient implementation of embedded control systems in resource-constraint domains like automotive systems and robotics.