Guest Editorial Special Section on Real-Time and (Networked) Embedded Systems

EMBEDDED systems are increasingly taking advantage of the opportunities afforded by ubiquitous connectivity and networks to offer new and original functions and solutions. More importantly, embedded systems are reaching far into unconventional areas of application, promising new solutions to problems such as energy conservation, transportation, environment preservation, and health care. Instrumenting the world with intelligent sensors, and the ability to process the data in real time, could well provide some answers to many of the problems facing our society today. The design of this kind of systems, especially when they are integrated in safety-critical applications is, however, made complex by the distributed nature of many applications, the intricacies of the interactions between the components, and the stringent requirements in terms of hard real-time constraints and resource utilization often associated with these applications. Therefore, the design of embedded systems requires the development of efficient platforms supported by innovative technologies, methodologies and algorithms.

This Special Section presents some of the most significant research work representing the state-of-the-art in the area of the embedded and real-time systems for automation systems. The section builds on top of and extends the contributions to the track on "Real-Time and (Networked) Embedded Systems," which was part of the IEEE International Conference on Emerging Technology and Factory Automation (ETFA), held in Hamburg, Germany, in September 2008. The section is divided into three groups of related papers. The first group contains papers dealing with scheduling, optimization, and analysis of embedded real-time systems, the second group deals with the integration of components into distributed systems, analyzing aspects related to composition, and quality-of-service; finally, the third group of papers investigates, in particular, the problems related to real-time networks, in both wired and wireless environments.

A. Embedded Real-Time Systems

There is a trend towards integrating more and more embedded software applications on the same CPU. These software applications can be both newly developed applications, as well as applications that are reused, i.e., legacy applications. At the same time, there are still stringent requirements on the embedded systems to be resource efficient, as the amount of available resources (e.g., CPU and memory) is often limited due to economical reasons.

The papers presented in this first group deal with several problems related to the analysis and optimization of embedded system platforms. The first paper presents a framework for dealing with open environments where applications are allowed to be initially developed independently, and later on integrated with predictable interference among applications without the need for a detailed system level analysis revealing all application internal details such as task properties. The second paper targets legacy applications, where systems are constructed out of software that is already developed possibly without conforming to any particular real-time model. Hence, some system parameters are typically unknown, thus complicating any kind of traditional real-time analysis to be performed. In running such systems, the paper presents two feedback-based schedulers tailored for dealing with legacy applications. Resource constrained systems is the topic of the third paper, presenting a technique for deriving tight bounds on buffer sizes of embedded applications consisting of multirate tasks exchanging messages. The fourth paper presents an integrated, cross-abstraction framework for the development of efficient platforms, focusing on the case of multiprocessor chip design. Finally, the fifth paper proposes techniques for enhancing the reproducibility of interrupts, an essential condition for effectively debugging applications embedded in nondeterministic environments.

The paper "*Resource-Sharing Servers for Open Environments*" presents an open environment tailored for independently developed real-time applications. Inherent in the problem of integration, the development of embedded systems consisting of multiple independently developed applications typically requires a system level analysis possibly identifying and pinpointing interference across applications. For example, a task inherent in one application may interfere with another task in another application and *vice versa*. This interference can be in the temporal domain, when the time spent executing one task in one application directly impacts when another task in another application will be scheduled for execution. Interference may also stem from tasks sharing some resource that requires mutual exclusive access, i.e., when one task has access to the resource no other tasks may get access to this resource and are thereby blocked from such access. Embedded systems development would be greatly simplified if some of these dependencies among applications were removed.

This paper presents a technique that, by using abstract interfaces of applications instead of a traditional task model, hides internal details not needed at the point of integration, i.e., system level analysis is simplified. Moreover, given this setting, techniques allowing for interapplication resource sharing is discussed in detail, i.e., tasks resident in different applications may share mutually exclusive resources.

The paper "*Legacy Real-Time Applications in a Reservation-Based System*" presents a technique to schedule applications with partly unknown properties, e.g., applications

Digital Object Identifier 10.1109/TII.2009.2027059

with unknown task worst-case execution times. The work is motivated by the existence of legacy applications, applications already developed not conforming to the traditional models assumed by most real-time analysis techniques. A lot of time, money, and effort have been invested in the development of these applications and they cannot easily be replaced by completely new software.

The paper presents two feedback based schedulers for reservation based systems built with the Constant Bandwidth Server (CBS). The CBS create a predictable interference from tasks running inside the server on the rest of the system, still providing quality-of-service to its internal tasks. The two proposed controllers, one for general legacy tasks and another for periodic legacy tasks, are evaluated through a set of experiments performed in a Linux-based implementation.

The paper "*Improving the Size of Communication Buffers in Synchronous Models With Time Constraints*" presents a buffer size optimization technique for embedded applications constructed from multirate software components, i.e., the components communicate through message buffers and they are periodically executing at different rates.

In many embedded applications the amount of resources is limited. In particular, memory is often scarce, and its usage should be minimized in all ways possible. Having a clear semantics of software component behavior allows for a predictable validation and verification of various functional and extra-functional properties using simulation and formal methods. One such property is the buffer size required to allow for communication among embedded applications constructed from software components executing at multiple rates.

The paper presents an approach to derive the minimum buffer size in embedded applications where the software components are executing at different rates. The approach has been applied in an automotive case study, and it indicates an improvement over existing techniques.

The paper "*Cross-Abstraction Functional Verification and Performance Analysis of Chip Multiprocessor Designs*" introduces a framework for the verification and performance estimation of chip multiprocessors. Because of the large design space for this kind of architectures, optimization and verification requires extensive, and at the same time efficient, analysis to strike the right balance between performance and resource utilization.

The paper focuses in particular on multiprocessor designs based on a bus matrix interconnection infrastructure. The methodology proposed in this work follows the model-based design principles, with the development of domain specific modeling languages and the use of heterogeneous models (transaction-level, cycle accurate FSM, timed automata) to conduct the analysis at different levels of abstraction. Of particular interest is the integration of several techniques for functional verification, that allow the designer to choose the appropriate level of abstraction, improving scalability and accuracy at the same time. The methodology is supported by a comprehensive set of freely available tools for functional and real-time verification and for performance estimation, and provides bridges between the various models in the framework to integrate the results of the analysis.

The paper "*Pinpointing Interrupts in Embedded Real-Time Systems Using Hashed Execution Contexts*" proposes a method for reproducing the exact sequence of interrupts to drive a system into a specified state, often representing a faulty software execution. Reproducibility is essential for debugging systems immersed in a nondeterministic environment: when a faulty behavior is observed in the field, the offline debugging and analysis of the software implementation requires that the timing of the external events, which influence the application through the interrupt subsystem, be reproduced to trace the fault back to its origin.

This work proposes a simple method that does not perturb the execution of the application significantly, it is platform-independent, and at the same time does not resort to expensive and often inexact dedicated hardware. Instead, the proposed technique records the interrupt location by encoding the state of the processor through a set of hash functions that can account for the contents of the registers, the stack, and the heap. The hashed state results in markers that are used to pinpoint the interrupts. The method is not exact, due to the potentially limited context used for the state of the processor, and the possible conflicts generated by the hash functions. However, the paper evaluates different solutions for the context and the functions, showing that accuracy over 99% is possible in the majority of cases by considering registers and a partial view of the stack only.

B. Distributed Real-Time Systems

As modern applications become more and more complex, there is an increasing need for adaptivity and flexibility of their underlying architectures. Inherent in this trend is the shift from a centralized to a distributed implementation, which can more easily provide alternatives for task execution and can natively support fault tolerance for the most demanding safety-critical applications. Distributed systems, however, introduce an extra degree of complexity which has to do with managing the interaction between the different parts. The papers in this group provide different answers to the problems of analyzing or implementing a composition. The first paper uses the service-oriented architecture to simplify the communication among components in a network. The second paper proposes different algorithms for the composition of services, while preserving quality-of-service in a dynamic environment. Finally, the third paper proposes and evaluates various techniques for the asynchronous communication in a Java environment.

The paper "*A Real-Time Service-Oriented Architecture for Industrial Automation*" presents an Ethernet-based distributed architecture for industrial automation. Using service oriented architectures, the problem of identification, discovery, and communications among networked components is greatly simplified.

The paper presents a framework, developed in the context of the RI-MACS European project, enabling properties such as real-time and quality-of-service to be guaranteed. This allows for runtime configuration of the system, and an implementation done in Linux is used to evaluate the approach. In particular, the paper shows significant improvement over existing technologies both in terms of absolute response time, and especially in terms

of the predictability of the behaviors, which is obtained through a technique of temporal isolation.

The paper "*QoS-Aware Real-Time Composition Algorithms for Service-Based Applications*" presents a model for quality-of-service-aware composition of services in distributed systems. Applications are constructed by the composition of services located throughout the distributed system, hence, enhancing reuse and potentially decreasing the development time.

The paper builds on the service oriented paradigm and on distributed real-time systems, and presents two algorithms for quality-of-service-aware composition of service-based applications. The first algorithm is exhaustive, to be used for offline composition of applications, and the second algorithm is based on heuristics making it applicable for execution online during system runtime. The problem is an optimization problem, and the experiments show that the heuristic approach is capable of finding solutions very close to the exhaustive one, yet only exploring much fewer configurations. The performance of the second algorithm makes it suitable for deploying this technique in a dynamic environment, where services and their composition can be changed at runtime to match the current real-time requirements and resource availability.

The paper "*Simple Asynchronous Remote Invocations for Distributed Real-Time Java*" presents an extension to the Java remote method invocation (RMI) such that it supports asynchronous remote invocations. A special mechanism to carry out asynchronous communications within remote invocations is presented.

The paper presents the proposed mechanism along with its integration in DREQUIEMI, a distributed real-time framework based on the real-time specification for Java (RTSJ) and Java's RMI technologies. Of particular interest in this work is the evaluation of the asynchronous mechanism against the current synchronous model of distributed real-time Java. The experiments analyze traffic, bandwidth, memory consumption and invocation rates under different conditions, and show that significant improvements in response time are possible for simple communications.

C. Real-Time Networks

Communication networks are at the basis of distributed systems, providing the needed infrastructure to implement the desired form of interaction between components. In the context of real-time systems, the properties and performance of the network become critical factors which may significantly impact the overall functionality of the system. The papers in this group address several aspects related to communication networks. The first paper presents a protocol for wireless sensor networks (WSNs) to avoid the problem of "hidden node collisions," thereby increasing quality-of-service in terms of reliability, timeliness, and system lifetime. The second paper presents a dynamic topology management protocol, which increases network lifetime and routing performance of WSNs. Finally, the third paper targets the problem of real-time communication on top of wired Ethernet for predictability.

The paper "*Improving Quality-of-Service in Wireless Sensor Networks by Mitigating "Hidden-Node Collisions"*" presents a new simple and efficient mechanism to increase quality-of-service in terms of reliability, timeliness, and system lifetime of WSNs suffering from the "hidden node problem." The hidden node problem is inherent when two nodes, not visible to each other, simultaneously transmit a message to a third node visible to both transmitting nodes, hence causing a collision. As most WSNs have scarce resources in terms of energy and computing power, retransmission of messages must be avoided to not impact throughput, message transfer delays and energy-efficiency of the WSN.

The paper presents the H-NAMe collision avoidance mechanism, a simple yet efficient mechanism, along with its implementation in a IEEE 802.15.4/ZigBee WSN. The experiments performed show increased throughput and a significant reduction in number of collisions. Finally, a small test application of mobile robots is used to show the potential of H-NAMe in a real application.

The paper "*An Adaptive Approach to Topology Management in Large And Dense Real-Time Wireless Sensor Networks*" presents a dynamic topology management protocol for WSNs. A topology management protocol is used to control sleep transitions of nodes in WSNs, hence playing an important role in terms of energy consumption and temporal performance.

The paper presents a dynamic topology management protocol developed for WSNs with real-time requirements. Classical topology management protocols have not stressed the importance of bounded delays, hence making them unsuitable for real-time applications. In addition, they are typically limited to static configurations which are not appropriate for event-driven data transmission which requires network reconfiguration, and are unable to take advantage of changes in the system due to the addition or removal of nodes. Along with the extension to dynamic reconfiguration, the paper also introduces a novel energy balancing technique, which exploits the redundancy in the network to run the nodes at low duty cycles through a node exchange policy. The presented protocol is evaluated using simulation, and is shown to have good performance both in terms of network lifetime (energy consumption) and of routing performance (timeliness).

The paper "*Hardware Acceleration for Conditional State-Based Communication Scheduling on Real-Time Ethernet*" presents a hardware implementation of a scheduler for timing predictable transmission of messages over Ethernet. As standard Ethernet provides no bound on communication delays due to message collisions and its binary backoff mechanism, a number of solutions have been presented over the years. Typically, these solutions implement some kind of arbitration scheme that provide bounded delays on message transmissions, and thereby enable real-time communication. A desire in the research and development of various real-time Ethernet solutions is to use standard commercial off-the-shelf (COTS) Ethernet components, due to higher cost associated with custom hardware, and a number of solutions with software-based message arbitrators have been proposed. A drawback of these software solutions running on top of standard hardware is their

associated computation overhead in the processor running the message arbitrators.

The paper presents the network code processor (NCP), which is a hardware implementation of the network code framework enabling the use of conditional state-based communication schedules, to be compared with running a software implementation on top of COTS Ethernet controllers. These conditional state-based communication schedules provide expressive means for specification and execution with choice points while staying verifiable. The evaluations of the approach show throughput comparable with standard Ethernet.

A Special Section like this one relies on the active support of several people, and we would like to take the opportunity to thank them all: the authors for their contributions and their cooperation in promptly replying to the reviewers' comments; the reviewers, for their thorough reviews and comprehensive comments, which contributed significantly to the quality level of the papers that were accepted for publication; and finally, Prof. Richard Zurawski, Editor in Chief, for his guidance in preparing and finalizing this Special Section, from the very first steps until its publication.

THOMAS NOLTE, *Guest Editor*
Mälardalen University
MRTC
Västerås, Sweden
(thomas.nolte@mdh.se)

ROBERTO PASSERONE, *Guest Editor*
University of Trento
Trento, Italy
(roberto.passerone@unitn.it)

Thomas Nolte (S'01–M'09) received the B.Eng., M.Sc., Licentiate, and Ph.D. degrees in computer engineering from Mälardalen University (MDH), Västerås, Sweden, in 2001, 2002, 2003, and 2006, respectively.

He was a Visiting Researcher at the University of California, Irvine (UCI), Los Angeles, in 2002, and a Visiting Researcher at the University of Catania, Catania, Italy, in 2005. He was a Postdoctoral Researcher at the University of Catania in 2006, and at MDH in 2006–2007. He became an Assistant Professor at MDH in 2008. He is now an Associate Professor of Computer Science at MDH since 2009. His research interests include predictable execution of embedded systems, design, modeling and analysis of real-time systems, multicore systems, distributed embedded real-time systems. He is Program Leader of the PROGRESS National Strategic Research Centre, President of The Swedish National Real-Time Association (SNART), and Co-Chair of the IEEE IES TCFA Real-Time Fault Tolerant Systems Subcommittee.

Dr. Nolte is a Member of the Editorial Board of Elsevier's *Journal of Systems Architecture: Embedded Software Design*, since 2009; Guest Editor of the IEEE TRANSACTIONS ON INDUSTRIAL INFORMATICS, 2008 and 2009; Co-Organizer of international events including the International Workshop on Compositional Theory and Technology for Real-Time Embedded Systems (CRTS), 2008 and 2009 [co-located with the IEEE International Real-Time Systems Symposium (RTSS), 2008 and 2009], and Program Co-Chair of IEEE International Conference on Emerging Technologies and Factory Automation (ETFA), the Real-Time and (Networked) Embedded Systems Track, 2008 and 2009, and IEEE International Workshop on Factory Communication Systems (WFCS), the Work-in-Progress (WIP) Track, 2008.

Roberto Passerone (S'96–M'05) received the Laurea degree (*summa cum laude*) in electrical engineering from the Politecnico di Torino, Torino, Italy, in 1994, and the M.S. and Ph.D. degrees in electrical engineering and computer sciences from the University of California, Berkeley, in 1997 and 2004, respectively.

From 1998 to 2005, he was with Cadence Design Systems, Berkeley, CA, where he held various positions from Senior Member of Technical Staff in the System Level Design Product Group, to Research Scientist in the Cadence Berkeley Laboratories. Since 2006, he has been an Assistant Professor with the Department of Information Engineering and Computer Science, University of Trento, Trento, Italy. At the University of Trento, he is the Leader of the COMBEST European Project, and of the Artist Design Network of Excellence. His research interests include the design and implementation of high-performance microprocessors, system level design, communication design and formal methods.

Dr. Passerone has been Co-Chair of the Semantics Working Group and the Rosetta Working Group in the Accellera Standardization Committee and is now secretary of the IEEE P1699 Rosetta Standard. He is also Co-Chair of the Embedded Systems Subcommittee of the IEEE IES TCFA. He has been Guest Editor for the IEEE TRANSACTIONS ON INDUSTRIAL INFORMATICS in 2008 and 2009 and co-organizer of several events, including the Real-Time and (Networked) Embedded Systems Track at the IEEE International Conference on Emerging Technologies and Factory Automation in 2008 and 2009, and the Work-in-Progress section of the IEEE Symposium on Industrial Embedded Systems in 2009.