HiPEAC 2018 – HEPSYCODE Tutorial: Program

Title:	HEPSYCODE: HW/SW CO-DEsign of HEterogeneous Parallel dedicated SYstems
Organizers & Speakers	Luigi Pomante, DISIM & DEWS Center of Excellence, Università degli Studi dell'Aquila, Italy luigi.pomante@univaq.it (contact person) Vittoriano Muttillo, DISIM, Università degli Studi dell'Aquila, Italy vittoriano.muttillo@graduate.univaq.it Giacomo Valente, DISIM, Università degli Studi dell'Aquila, Italy giacomo.valente@univaq.it
Motivation:	Design automation is the main topic of this tutorial, and it is one of the main topics of the HiPEAC conference. The tutorial presents a methodology that could be very valuable to exploit emerging system architectures for dedicated/embedded systems. In fact, it works considering multiple NF constraints, and it considers heterogeneous platforms that are composed of both fixed and reconfigurable elements. Moreover, the framework that support this methodology is extensible, in the sense that new elements can be simply added.
Intended audience:	As intended audience: - embedded systems designers that are interested to know more about hw/sw co-design - embedded systems designers that need to cope with heterogeneous architectures - embedded systems designers that need to cope with more NF requirements - researchers involved in EDA, in particular ESL synthesis
Objectives:	Main objectives: - to present the state of the art about the most used commercial and academic design tools in the field of hw/sw co-design (with particular attention to design space exploration considering F and NF requirements) - to present a methodology, called HepsyCode, able to support the development of parallel systems in different application domains - to show a live demo related to the use of HEPSYCODE with one or more case studies
Abstract:	In the last years, the spread and importance of embedded systems are even more increasing, but it is still not yet possible to completely engineer their system-level design flow. The main design problems are to model functional (F) and non-functional (NF) requirements and to validate the system before implementation. Designers commonly use one or more system-level models (e.g. block diagrams, UML, SystemC, etc.) to have a complete problem view and to perform a check on HW/SW resources allocation by simulating the system behavior. In this scenario, SW tools able to support designers to reduce cost and overall complexity of systems development are even more of fundamental importance. Co-existence of functional and non-functional requirements is the most relevant challenge. Unfortunately, there are no general methodologies defined for this purpose and, often, the only option is to refer to experienced designer indications with respect to empirical criteria and qualitative assessments. In such a context, this tutorial faces the problem of the HW/SW co-design of dedicated (possibly embedded and real-time) systems based on heterogeneous parallel architectures and presents a framework (with related methodology and prototypal tools), called HEPSYCODE

	(http://www.hepsycode.com/), able to support the development of such systems in different application domains. First of all, the tutorial presents the state of the art about the most used commercial and academic design tools in the field, with respect to methodology, design flow, system models and design space exploration (DSE) techniques. Next, it illustrates the reference HW/SW co-design flow, starting from the adopted model of computation (i.e. CSP-like), and then describing the different methodology steps, focusing on the system-level design space exploration approach that allows the related co-design methodology to suggest an HW/SW partitioning of the application specification and a mapping of the partitioned entities onto an automatically defined heterogeneous multi-processor architecture. Finally, a live demo will show the use of HEPSYCODE toolchain with some reference examples and case studies.
Necessary background:	- Knowledge of C language - Knowledge of basic concepts about embedded system architectures
References:	[1] C. Brandolese, W. Fornaciari, L. Pomante, F. Salice and D. Sciuto, "Affinity-driven system design exploration for heterogeneous multiprocessor SoC," in IEEE Transactions on Computers, vol. 55, no. 5, pp. 508-519, May 2006. [2] L. Pomante, "System-level design space exploration for dedicated heterogeneous multi-processor systems", 22nd IEEE International Conference on Application-specific Systems, Architectures and Processors (ASAP), Santa Monica, CA, 2011, pp. 79-86. [3] L. Pomante, "HW/SW co-design of dedicated heterogeneous parallel systems: an extended design space exploration approach," in IET Computers & Digital Techniques, vol. 7, no. 6, pp. 246-254, November 2013. [4] L. Pomante. "Electronic System-Level HW/SW Co-Design of Heterogeneous Multi-Processor Embedded Systems", The River Publishers Series in Circuits and Systems, June 2016. [5] http://www.pomante.net/sito_gg/hepsycode.htm
Tutorial material:	Slides and access to online material
Tutorial plan:	14.00 - 15.30 Topic 1 A System-Level Methodology for HW/SW Co-Design of Heterogeneous Parallel Dedicated Systems - speaker Luigi Pomante This talk presents the state of the art about the most used commercial and academic embedded design tools, with respect to methodology, design flow, system models and design space exploration (DSE) techniques. Next, it illustrates the reference HW/SW co-design flow, starting from the reference model of computation, based on a communicating sequential processes (CSP), and describing the different steps, including a system-level design space exploration approach that allows the related co-design methodology to suggest an HW/SW partitioning of the application specification and a mapping of the partitioned entities onto an automatically defined heterogeneous multi-processor architecture. Finally, a live demo will show the use of HepsyCode tool with some reference examples and one or more case studies. 15:30 - 16:00 Coffee Break 16.00 – 16.45 Topic 2 Real-Time and Mixed Criticality Extensions for the HepsyCode Methodology: Past, Present, and Future work – speaker Vittoriano Muttillo This presentation focuses on a framework (and related tool) for modeling, analysis and

	validation of mixed critical systems, through the exploitation of the HepsyCode methodology, improved to consider both real-time (RT) and mixed criticality (MC) requirements. Starting from different HW based, OS-based, and Hypervisor-based solutions (both in the research and industrial domains), a classification of different works in the mixed criticality system scenarios is provided, in order to help both researchers and industries to find resources as close as possible to their needs. Next, a HW/SW co-simulator to be integrated into an ESL HW/SW co-design methodology targeting embedded heterogeneous parallel systems with MC and RT constraints is presented. The final result will be a methodology able to support mixed-criticality systems developments by suggesting both the platform and mapping solutions for the specific mixed-criticality application. 16:45 - 17:30 Topic 3 A HW/SW Unified approach for embedded system monitoring - speaker Giacomo Valente This talk focuses on the necessity of a monitoring action on embedded systems, and provides a unified view of this: ● by analyzing why the monitoring is done in academic and industry (debugging, profiling, verification) ● how the infrastructures are realized to provide the monitoring information ● the impact of the choices, focusing on software and hardware overhead Then, it shows a classification that allows a unified vision of the monitoring action, with tools and their pros/cons, and presents a framework that allows the inclusion of monitoring in HepsyCode methodology. The final goal is to offer Hepsycode with the “Monitorability”, i.e. the possibility to support the selection of a suitable monitoring solution for the final system, by customizing an infrastructure among the existing ones. The choice will be keeping in consideration the final impact of the solution respect to NF constraints.