COMPONENT-BASED TECHNOLOGIES FOR HW/SW CO-DESIGN

[(1)]Elena Alaña, [(1)] Francisco Ferrero, [(1)] Ana-Isabel Rodríguez, [(1)] Raúl Valencia

[(2)]Eric Conquet

[(3)]Juan Antonio de la Puente, [(4)] Juan Zamorano,

[(5)]Fernando Herrera, [(5)] Roberto Varona

[(1)]GMV, Isaac Newton, 11, Tres Cantos, Madrid, 28760, Spain, Email:

{ealana, fferrero, airodriguez, rvalencia}@gmv.com

[(2)]European Space Agency (ESA), ESTEC Noordwijk - The Netherlands, Email:

eric.conquet@esa.int

[(3)(4)]Universidad Politécnica de Madrid (UPM), Spain, Email:

[(3)]jpuente@dit.upm.es, [(4)]jzamora@datsi.fi.upm.es

[(5)]Universidad de Cantabria (UC), Spain, Email:

fherrera@teisa.unican.es, roberto@teisa.unican.es

KEYWORDS

HW/SW co-design, model-based system engineering, component-based engineering, AADL, UML-MARTE

ABSTRACT

Hardware/software (HW/SW) Co-development techniques exploits the trade-offs between hardware and software in a system through their parallel design, taking advantage of the flexibility of the system design to develop architectures that can meet stringent timing and performance requirements with a shorter design cycle. Advances in the automation of the design of HW and SW, the adoption of the Model Driven Architecture (MDA) paradigm and Component-Based techniques make possible the definition of a proper integration substrate that facilitates the continuous interaction of the HW and SW development paths.

Following a component-based approach allows system engineers to visualise the system as a composition of application components interconnected each other, independently of the underlying execution platform. This approach not only provides potentially high benefits for reuse and better organization of the system, but enhances the HWSW partitioning process as system functions are allocated to modular, functional units that communicates each other via well-defined interfaces.

In the particular area of Component-based technologies, this paper presents the work and research carried out on the use of Component-based technologies for HW/SW Co-Design within the scope of two research programs: ESA HWSWCO (Hardware-Software Co-design) TRP (Technology Research Program) study, contract ESTEC 22810/09/JK, and the European Commission (EC) COMPLEX project (FP7/247999).

The paper presents the major outcomes of those studies in those technological areas.

The HWSWCO study has investigated the HW/SW co-design phase to integrate the co-design phase as part of a system process and methodology previously developed for the European Commission, the ASSERT (Automated proof-based System and Software Engineering for Real-Time applications) process (refer to [1]) and related TASTE toolset (The ASSERT Set of Tools for Engineering [2]).

Besides, COMPLEX project has aimed to develop a design methodology and framework for the iterative exploration of the design space for embedded HW/SW systems.

The co-design methodology follows the Model Driven Architecture [5] paradigm in which the system model is composed of different model viewpoints that describe the system from different perspectives: the system Platform Independent Model (PIM), Platform Description Model (PDM) and Platform Specific Model (PSM).

The HWSWCO process used the model viewpoints inherited from the ASSERT process [1] that describe the application by means of different model views. These model views provide the data modelling, functional and interface definition in different AADL [8] models that form part of the PIM. The ASSERT Model Transformer (AMT) tool processes those model viewpoints, combines them with the description of the platform in the PDM, and generates the PSM input to the ASSERT toolset [1].

The COMPLEX project also defined model viewpoints in the UML [7] model to manage the complexity of the system defining several model views that focus the system designer attention on specific aspects of the system. It uses the UML/MARTE profile [7] to support the description of the functional and non-functional properties of the system, and extends UML by means of profiling to support the description of the design exploration space. The PIM model is composed of the data and functional modelling and the communications and concurrency view. They model not only the application with independence from the underlying execution platform, but also its non-functional (timing) properties. The PDM is formed of the platform view, which describes both the HW and SW components that form part of the system execution platform. Finally the PSM is represented by the architectural, which describes the hardware and software architecture, the allocation of application components onto platform resources, and the DSE parameters, rules and constraints that will guide the search of the optimal architecture. Additionally to these views, the COMPLEX methodology defines the verification view that models the interaction between the system and external sub-systems interacting with it.

An Adequate HW/SW partitioning analysis is the basis of the decision-making process for the system architecture that meets the system timing and performance requirements:

- High-level HW/SW estimation for power and performance analysis (references to [9]): the main methodologies for SW simulation are SW execution on a host computer and instruction-set simulation. The application SW can be directly executed on the host using a native compiler. The host OS can be used to emulate the target platform RTOS.
- Design Space Exploration (DSE) analysis is a key phase in HW/SW co-design process ([4]). The DSE loop actually consists of tuning parameters of the system platform and allocating the application components to the different computational resources. It also may change the memory assignment, as well as lower level design parameters such as clock frequency or bus/network width. The result of this DSE loop is a system architecture that meets performance requirements and fulfils the non-functional properties and constraints annotated in the UML model.
- Worst Case Execution Time (WCET) analysis of the system functions. Since the high-level estimation and DSE loop are based on the definition of a stimuli scenario, the real-time behaviour of the system might not be fully checked as the worst case execution path might not be reached. Therefore, it is necessary to determine the WCET of the system functions and perform a schedulability analysis to formally verify that the system meets the deadlines constraints of each system function. This analysis is therefore combined with the schedulability analysis described below.
- Finally, the schedulability analysis of the system. It is necessary the definition of analysis frameworks that take into account both the SW and HW systems (i.e. their computational models). The HW system exhibits parallelism of their tasks, while the SW system shows concurrency in the execution of the system functions. Therefore, a combined analysis should be performed that analyse the influence of both shared resources in the HW and SW systems.

Two different *use case demonstrators* are being developed to exercise both HW/SW co-design methodologies. Indeed, both are derived from the space domain and consist of simplified versions of applications that run on satellite payloads.

Ann image processing system that executes high-cost function on an FGPA was developed to exercise the methodology of the HWSCO project.

In the COMPLEX project, the selected use case consists of an object survey and hazard estimation application and software GPS developed in GMV. Their performance and timing figures are evaluated by the tools developed in the COMPLEX project. The system is based on ARM9 and ARM11 microprocessors, and the use of virtual platforms to verify both the functional and performance behaviour of the system.

REFERENCES

[1] ASSERT (Automated proof-based System and Software Engineering for Real-Time applications) – For a reliable and scientific approach in system and software engineering. Final Report

[2] E. Conquet, M. Perrotin, P. Dissaux, T. Tsiodras, and J. Hugues, "The TASTE Toolset: turning human designed heterogeneous systems into computer built homogeneous software," in Proceedings of Embedded Real Time Software and Systems (ERTS), 2010

[3] Panunzio, M., Vardanega, Tullio. (2009). "On Component-Based Development and High-Integrity Real-Time Systems", Proc. of the 15th IEEE International Conference on Embedded and Real-Time Computing Systems and Applications

[4] Chang, H., Cooke, L., Hunt, M., Martin, G., McNelly, A. J., and Todd, L. 1999 Surviving the SOC Revolution: a Guide to Platform-Based Design. Kluwer Academic Publishers

[5] OMG Model Driven Architecture (http://www.omg.org/mda/)

[6] SAE International, "Architecture Analysis and Design Language (AADL)", http://www.aadl.info/

[7] UML profile for MARTE, (http://www.omgmarte.org/)

[8] OSATE AADL editor, Webpage: http://www.aadl.info/aadl/currentsite/tool/osate-down.html

[9] SCoPE, Webpage: http://www.teisa.unican.es/scope

[10] AADST, Webpage: http://www.teisa.unican.es/AADS

[11] Impulse Co-Developer, Webpage: http://www.impulseaccelerated.com/

[12] HWSWCO ESA TRP study. Webpage: http://hwswcodesign.gmv.com/

[13] COMPLEX Project Webpage: http://complex.offis.de/

[14] MULTICUBE Project Webpage: http://www.multicube.eu/