



Object-oriented co-design and functional test techniques

Results in Brief

Co-design of embedded hardware/software systems

Support tools for the object-oriented design of hardware/software systems, based on the System C description language, were developed to provide the migration path from executable specifications to efficient hardware implementations.



The steadily increasing complexity of embedded systems comprising hardware and software components requires comprehensive methodologies that support the design process from multiple levels of abstraction. A most interesting development was the adoption of well-known and approved object-oriented modelling and synthesis techniques widely employed in software development. Nevertheless, there are major differences between the nature of software and hardware

and direct adaptation of common object-oriented programming languages would not be possible.

System C is a C++ class library and methodology that introduced the missing, but typical hardware features in C/C++. Within the ODETTE project, the transition from objective VHSIC Hardware Description Language (VHDL) to System C was achieved. This change was mainly driven by the project's industrial partners who saw

this as the best way forward. More importantly, dedicated tools for the cosimulation of hardware's and software's functional and temporal behaviour was developed at Synopsys' laboratories in France, which provide executable specifications of the complete system.

As part of the System C design and verification tool suite CoCentric® System Studio, it allows System C users to employ intellectual properties (IPs) developed in VHDL. Moreover, the cosimulation environment provides for the simulation of the complete hardware/software system, where the software parts of the model are written in C/C++ and hardware parts in System C. Modelling of the complete hardware/software system, described in different languages is enabled by the ability of the System C modelling environment to cooperate with HDL simulators.

The dynamic behavioural model of the hardware/software system, in interaction with its environment model, can subsequently provide the necessary insight for an indepth analysis of complex designs. For refinement, software-hardware partitioning, simulation and verification tasks during the design process tool support is provided from the highest abstraction levels down to the register-transfer level within one single environment.

Discover other articles in the same domain of application





This project is featured in...

RESEARCH*EU MAGAZINE









Last update: 3 March 2008

Permalink: <u>https://cordis.europa.eu/article/id/83841-codesign-of-embedded-hardwaresoftware-systems</u>

European Union, 2025