UML as a Next Generation Language for SoC Design

Luciano Lavagno
Research Scientist
Cadence Berkeley Labs

Wolfgang Mueller
Advanced Design Technologies
C-LAB/Paderborn University

The Unified Modeling Language™ (UML) has already received considerable acceptance in the domain of telecommunication and automotive systems and we see emerging activities in the field of Systems-on-Chip (SoC) design [1-3].

Although different UML diagrams are in use for various purposes such as testbench description and the specification of requirements, architectures and behavior, there has yet to be a major breakthrough of UML for SoC design. However, with the outcome of a new generation of customizable UML tools based on the principles of model-driven architecture (MDA), we expect a major impact in this area as well.

In the field of model-based verification (also called model-based testing in the software and automotive systems domain), for instance, some UML tools already support testbench specification and code generation. For example, tools like Tau (Telelogic) generate validation sequences in the TTCN-3 language from Sequence Diagrams. Other tools like AutomationDesk (dSPACE) use Activity Diagrams for test descriptions and test script generation.

Typical usage patterns of UML subsets in the field of behavioural SoC designs exploit State Machine, Activity and Sequence Diagrams in combination with Use Cases. I-Logix has been a pioneer to generate synthesizable VHDL from State Charts, the precursor of State Machine diagrams in UML. More recent work also considers the combined use of State Machine and Activity diagrams with their compilation to VHDL, SystemC, and Handel-C for simulation and synthesis. In that context, the electronics industry (e.g., Fujitsu, NEC, and ST Microelectronics) investigated several case studies and hardware/software-codesign methodologies based on UML have been explored (e.g., ACES from NEC labs).

A key strength of UML is its ability to be extended with domain-specific customizations — the so-called *profiles*. A prominent example of an existing profile relevant in the embedded system domain is the UML profile for schedulability, performance and time (SPT). This is expected to be improved and extended by the upcoming Modeling and Analysis of Real-Time and Embedded systems (MARTE) profile.

Of particular interest for SoC design is the submitted UML extension profile for SoC [4], as well as the Systems Modeling Language [5] (SysML). The SoC profile is targeting for SystemC-oriented applications. SysML extends UML for general systems engineering and deals with continuous quantities, requirements, physical assemblies, i.e., blocks, and so on.

In order for a profile to be used for real applications, it must be:

- easy to use by domain experts and preferably to come with application guidelines
- formalized as a machine-readable standard format given by a so-called metamodel.

Several tools are already available, which take such metamodels as input for their configuration where some support customizable code generation. These include tools from Accelerated Technology, Aonix, ARTiSAN Software, I-Logix, for instance..

All in all, we see emerging interest in UML's application for SoC specification and analysis, and there are several significant and ongoing efforts to customize UML towards SoC design. With the emergence of methodologies and the availability of a first set of dedicated tools, we see that UML has great potential to complement current C++-based languages for SoC design.

We also see SysML as a real opportunity for a specification language beyond transaction-level and electronic system-level modeling. An additional potential lies in UML's application for architectural, component, and interface description. Intellectual property (IP) integration and packaging — e.g. in the direction proposed by the Spirit consortium [6] — can become easier by inheriting some of UML's advanced object-oriented concepts.. To this end, we also realize that the current UML standard together with SysML is a complex framework of highly intertwined concepts and we definitely see a clear need for education before the SoC designer can benefit from UML and its rich set of concepts.

References

[1] L. Lavagno and W. Mueller. UML for SoC Design Workshop at DAC'05, Anaheim, CA, USA, June 2005, www.c-lab.de/uml-soc

[2] G. Martin and W. Mueller. UML for SoC Design Workshop at DAC'04, San Diego, CA, USA, June 2004.

[3] G. Martin and W. Mueller (eds.). UML for SoC Design, Springer, June 2005.

[4] OMG. UML Extension Profile for SoC. ptc/06-04-18, April 2006.

[5] SysML Merge Team. Systems Modeling Language Specification Version 1.0 Draft. OMG document ad/2006-03-01. March 2006.

[6] Spirit Consortium, www.spiritconsortium.org

Luciano Lavagno is a Research Scientist at Cadence Berkeley Labs, where he has worked on several projects in the System Level and digital implementation spaces.

Wolfgang Mueller is heading the advanced design technologies group at C-LAB, a joint R&D institute of Paderborn University and Siemens.