Advanced simulation tools on top of SystemC

Lecture summary for FRUCT seminar in Turku

Michel Gillet and Sergey Balandin
Nokia Research Center, CS CTC, Helsinki, Finland
Michel.Gillet@nokia.com, Sergey.Balandin@nokia.com

The lecture starts with brief overview of the current state of embedded networks research, as it was the main target area for the developed simulation tools. Than we provide description of the available instrumentation tools that are used for modeling "classical" networks, e.g. NS-2, LOTOS, SDL, etc. For each of studied tools we present an analysis of why it does not fit needs of the embedded networks, what features and functionality are missing and what internal and environmental modules are really good. This part of the lecture and provided analysis is particularly interesting for people working with different simulation platforms as it gives an idea of how to move best practices seen for some platforms to your main simulation platform.

From the first part of the lecture, it becomes clear that none of the available tools fulfill our needs. As a result, we were forced to develop our own advanced simulation platform for embedded networks. The next question was whether to develop it from scratch or use some available platform like SystemC, TLM, NS-2, or something else. After analysis we have found out that the best base platform that provides link to hardware and modeling of time sequence is SystemC. At the moment SystemC is a de facto industry standard for bus level modeling and simulation. Another strong advantage of SystemC is that it allows easy implementation of the models with different levels of abstraction. As a consequence simulations can be performed with different levels of details, which save a lot of computational power and time.

In the next module the lecture summarizes our experience from development on SystemC. The first fully developed model was implementation of SpaceWire standard, which is one of the closest technical solutions to our needs in mobile industry and it is the only stable standard for embedded networks on top of point-to-point interconnect. Then, based on our experience of SystemC model development for Spacewire we made a modeling framework for a new MIPI standard – UniPro. It is important to mention that development of SystemC model was done as official part of MIPI UniPro standardization activities and for that a special subgroup of UniPro working group was created by representatives of Nokia, Philips/NXP and ST Microelectronics.

The main body of the lecture describes basics and gives short outlook of how we build UniPro model. Here we assume that people have basic knowledge of the embedded network technologies. If you need more training on this subject we recommend to attend lecture on “Embedded networks” that is given by us on the first day of FRUCT seminar in Turku.

In the UniPro model we have paid special attention to make proper layering of the model, with clear and transparent definition of the interlayer and inter-modular interfaces, so that the resulting model would accurately reflect the specification structure and be easy readable and understandable for people without modeling background. This is especially important when one think of reviewing and verification of the model.

Another point discovered in our study was that even if SystemC provides good basis for the model, it is not enough to cover all our needs. As an extension we start to use own SystemC libraries, made a replacement of sc_module by our own module, which enables model factory, added creation of use cases in XML, developed python binding of the SystemC library and so on. In addition to SystemC direct plug-ins, a number of 3rd party tools are used for simplifying post processing and visualization of the simulation results. For example, in our simulator we have defined special simulation outputs which allow direct use of such well known tools as NAM (it is a part of NS-2 distribution), Matplotlib, SVG, etc. On the lecture you will see the screenshot examples of outputs obtained using different additional tools and will be able by yourself evaluate how valuable is such input for fast and efficient analysis of the simulation results.

In the conclusion of this lecture we will tell a bit about our future plans and “hot” topics that need to be addressed in the near future. Among such topics one of the most important is more extensive use of specification’s SDL models in our simulations and in particular co-simulations between SDL and SystemC and between SDL and TLM models. Also some interesting topics for the new FRUCT projects in this field will be proposed.