2016 DESIGN AND VERIFICATION™ DVCON CONFERENCE AND EXHIBITION EUROPE

SystemC gaps encountered in Virtual Platform development

Eyck Jentzsch, MINRES Technologies GmbH, Munich, Germany (*eyck@minres.com*)

***Abstract*—Highest simulation speed while offering accuracy and insight into details of state and communication are promises of virtual platform deployments. Using SystemC as modeling language of choice reveals a few areas of improvement to better support the fulfillment of these promises and to make the life of virtual platform developers easier.**

Keywords—SystemC, TLM, signal assigment, data visibility, tracing

For today's virtual platforms (VP) it is crucial to provide highest simulation speed while offering accuracy and insight into details of state and communication. Unfortunately a few areas in SystemC lack features to support modeling styles and pattern used to achieve this efficiently. Those are:

- Delayed signal assignments
- Efficient value tracing
- A basic and extensible TLM2 transaction recording

To get high simulation performance specific modeling styles and design pattern evolved over the past years. They aim to minimize the switching between methods and threads in the design as this incurs quite some overhead and lowers the speed of the simulation. One of these styles is the approach of tick-less clocks [1]. The clock is not described as a constantly changing signal rather a signal carrying the clock period and optionally a phase. Processes are triggered upon changing the clock period, which minimizes context switching during simulation. All behaviors of a module are described based on calculated delays derived of the clock period. Thus activation of processes is minimized and performance improves.

This is in many cases sufficient in virtual platform development esp. if the timing of the functionality is not of interest or needed. But for some blocks an RTL-accurate description of the behavior might be needed, as this might be a crucial feature for the use of the VP. And here a lack of features of SystemC is revealed: schedule the assignment of values to signals at a certain time e.g. based on calculations using the clock period. Currently the options are to fallback to event-based descriptions periodically triggering processes or to use a combination of events and processes to do the assignments. Both imply additional process invocations and delta cycles as well as additional implementation efforts. Extending the signal assignment to add a timed delay would greatly improve the situation and ease the life of a VP developer.

A promise of VPs esp. for embedded software development is to provide visibility far beyond what's possible in silicon at lower costs. In the current implementation the promise implies a (sometimes huge) performance penalty. Having a decent VP the penalty of tracing can sum-up to a slow-down of a factor of 3-5. Root cause is the implementation of tracing as visitor pattern: before advancing the time (or after a delta-cycle) each signal and variable is checked for changes. Developing the tracing into a listener pattern e.g. for the SystemC build-in types sc_signal and sc_port would improve the situation drastically. A write operation to a signal would trigger a notification that a changed value should be recorded. This way the tracing infrastructure does not need to check each and every delta cycle for a change of values. Introducing a wrapper for native data types could serve the same purpose allowing transparently the notification of the tracing listener upon writing a variable. The concept could be developed into or tied to data introspection beyond what is provided currently by the SystemC Verification Library (SCV).

The TLM library of the reference implementation can improve in another area and provide tracing facilities for transactions i.e. for the generic payload. SCV might come to the rescue so this is not a limitation in the language itself, all means to implement TLM transaction recording are available. What's missing is a reference implementation e.g. based on SCV transaction recording, which could also facilitate the development of open-source GUI-based tools to visualize and analyze the communication in a VP. The tracing could be implemented transparently as part of a TLM socket as it is done prototypically in SCV4TLM [2]. It would allow the use of tools like SCViewer [3], Impulse [4], or Eclipse Trace Compass [5] to correlate and analyze communication and behavior. The use of such tools greatly enhances the utility of a VP as they also allow correlating the communication amongst the VP blocks and the events in the embedded software running on the VP.

REFERENCES

[1] Look Ma, No Clocks! Improving Model Performance, David Black, XtremeEDA, SystemC Day 2010

[2] SCV4TLM, https://github.com/eyck/txviewer/tree/master/scv_tr_sqlite

[3] SCViewer, https://github.com/eyck/txviewer/tree/master

[4] Impulse, http://toem.de/index.php/projects/impulse

[5] Eclipse Trace Compass, http://tracecompass.org/