# AN OPEN-SOURCE TOOL FOR SYSTEMC TO VERILOG AUTOMATIC TRANSLATION

J. CASTILLO, P. HUERTA and J. I. MARTÍNEZ

*Universidad Rey Juan Carlos,*
*C/ Tulipan S/N, 28933, Mostoles, Spain,*
*{javier.castillo, pablo.huerta, joseignacio.martinez}@urjc.es,*

***Abstract*— As the complexity of electronic systems increases, new ways for describing these systems are proposed. One actual trend involves the use of system level languages that allows the description of the whole system in a higher abstraction level. This type of methodology helps a designer to obtain an appropriate Hw-Sw partition, where the Sw is compiled to the target platform and the Hw is refined to bring it down to a lower level of abstraction in order to be synthesized. This last step usually requires the use of a translation tool that from a description of the system in a system level modeling language, converts it to an equivalent one in a standard Hardware Description Language, usually Verilog or VHDL. This works presents a tool that from a SystemC RTL description generates its equivalent Verilog code ready to be synthesized by any standard Verilog Synthesis Tool.**

***Keywords*— SystemC, Verilog, Translation.**

## I. INTRODUCTION

The increasing complexity of the electronic systems has made necessary the exploration of new solutions in order to reduce its development time.

One of this solutions is to use new description languages (OSCI, 2002; Celoxica, 2005; Xilinx,2006; Pellerin and Thibault, 2002) which allow the designer to describe the system in a higher level of abstraction.

From this high level description of the system, a tool must provide a flow to reach the final silicon implementation. Following a traditional hardware-software codesign flow (Chiodo *et al*, 1994) (Fig.1), the system level description is profiled, and an appropriate hardware-software partition is proposed. Then the software has to be compiled to the targeted microprocessor. The hardware high level implementation can be directly converted into a working hardware using a proprietary synthesizer, like Handel-C or CatapultC. Another approach consist of rewriting it to obtain a lower and more detailed level of abstraction appropriate for hardware synthesis tools. This step is commonly made by hand with the problems it represents (time cost, prone to errors, etc.).

This work presents an open-source tool that taking as input a hardware module described using a high level description language, SystemC, gives as a result an equivalent description in Verilog. This Verilog description can be synthesized using any standard RT synthesis tools.

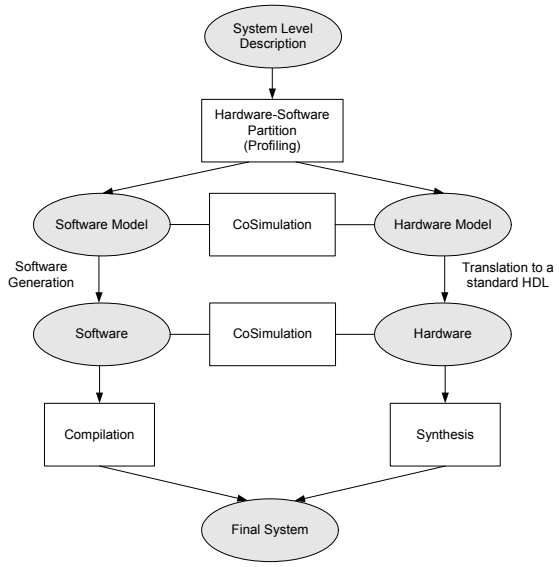


Figure 1. Traditional Hardware-Software CoDesign Flow

## II. SYSTEMC

### A. SystemC Overview

Nowadays hardware is usually described using HDLs such as VHDL and Verilog. However, software and system designers develop their code in C/C++. Hardware/software codesign becomes a very hard task due to the use of different languages at each abstraction level, or even in the same level (IP exchange). In this context, a single language that can be used in all the design stages is needed.

SystemC is a library of classes for C++ and a simulation kernel that provides all the features needed to describe a system in all its abstraction levels and a reference platform for IP exchange. SystemC also provides a library called the SystemC Verification Standard (SCV, 2002). This library provides classes and methods to build a verification methodology called Transaction Model Style (TLM) based on the use of transactors (Yuri *et al*, 2005). This verification methodology can be used with designs made with other HDLs. Many EDA vendors provide tools that allow SystemC to be mixed with different languages.