

Migrating from RTL to ESL via SystemC Training

Hassan Sohofi, Vahideh Akhlaghi, Fatemeh Asgari, Ali Shahabi, and Zainalabedin Navabi

Electrical and Computer Engineering, School of Engineering Colleges, Campus 2

University of Tehran, 1450 North Amirabad, 14395-515 Tehran, Iran

{h_sohofi, vahideh, asgari, shahabi}@cad.ece.ut.ac.ir, navabi@ece.wpi.edu

Abstract—we are presenting a training package for describing hardware with C++ and SystemC. The package contains videos, reading materials, slides, and software. After covering the basics, we start at the RT-level with a section on VHDL. This is followed by a corresponding SystemC presentation, and then moving up into SystemC channels. Finally some complete system-level examples are described using SystemC. The package readies an RTL designer for advanced ESL designs.

I. INTRODUCTION

The ever-increasing complexity of digital systems has introduced new challenges for the design community. As designs get more complex, the RT-level methodologies and tools do not suffice for the development of such complex systems. On the other hand, the ability to deliver new products within a short period of time becomes crucial for remaining in the business. To solve these problems, designers are migrating to a higher level of abstraction than RTL, called Electronic System Level (ESL) in which designs are described at system level. At that level, different hardware description languages have been developed at which SystemC is the dominant language. SystemC is a hardware description language based on C++ language.

In our presentation package, we provide an easy learning methodology which facilitates the migration of RTL design engineers toward system-level design development using SystemC. This transition contains a review of logic design followed by an introduction to RT-level VHDL language. After that, a comprehensive review of C++ which is necessary for learning SystemC is provided.

The major part of the training package describes different aspects of SystemC in three steps. In the first step, a simple

mapping from VHDL constructs into SystemC constructs for describing RTL and logic components is provided. This mapping is performed using a SystemC/HDL design environment, called *ROSTA*. In addition, this step introduces some basic constructs of SystemC. The second step discusses SystemC constructs from a linguistics view. It introduces a more general view of SystemC constructs. The third step describes SystemC channels which readies the trainee for system-level design and development.

The next part of the training package introduces some complete system examples described in SystemC with their corresponding C++ test-benches. At the end, some short videos present the use of our software package for SystemC simulation and VHDL/SystemC translation.

II. STRUCTURE OF THE TRAINING PACKAGE

The training package contains eight sections. Each section has its own video, reading material, and a set of presentation slides. Figure 1 shows the structure of the package.

Section one gives an introduction to the basics of system-level and system design.

Section two provides a preview of RTL logic design concepts. The purpose of this section is to highlight only those topics that are essential for HDL-based design.

Section three presents VHDL at the RT level. In this section we discuss how a design is described in VHDL for simulation and synthesis. For this purpose, only a subset of VHDL is needed and many complex language structures are not covered here.

Section four shows how C/C++ language can be used to

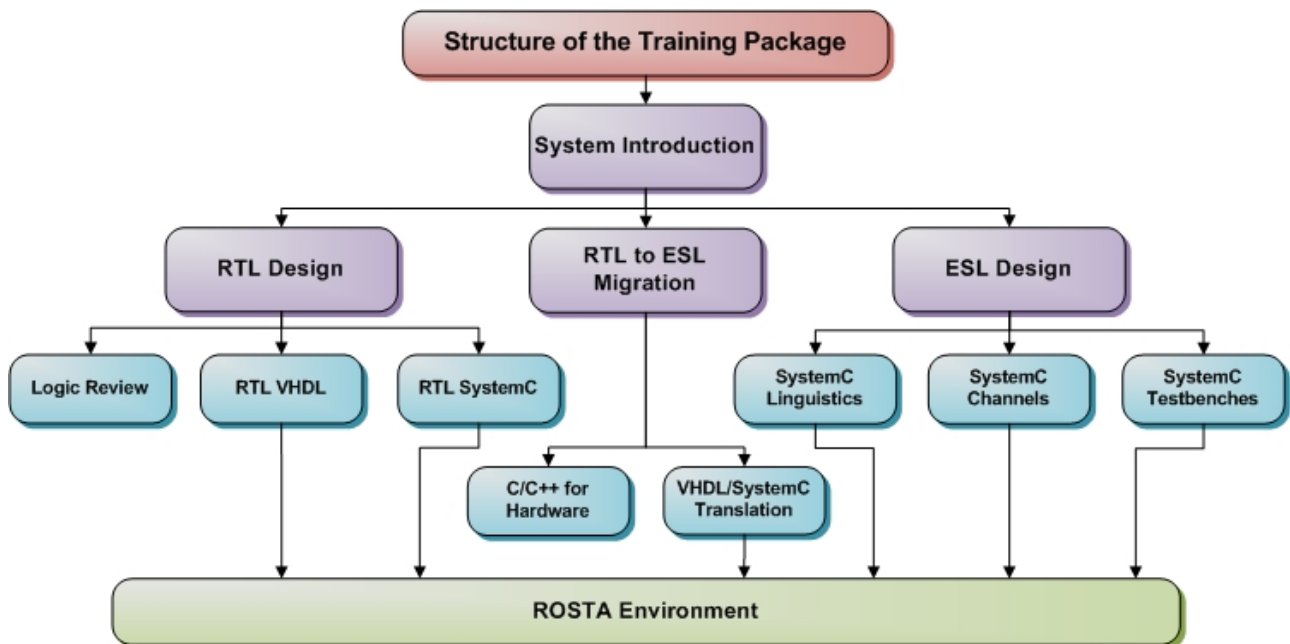


Figure 1. Structure of the Training Package

describe hardware. It prepares software familiar trainees for learning hardware description languages. Also it shows the limitations of software languages for describing hardware designs.

Section five is the main part of the package which describes the SystemC language as a hardware description language. SystemC is originally developed for system-level design. In spite of this, it has the necessary constructs for RTL and even gate-level designs. This section describes SystemC in an incremental fashion starting with simple RTL components and progressing into more complex system-level ones. For designers who are familiar with other RTL hardware description languages such as VHDL, this section provides a methodology and a tool for translation between VHDL and SystemC at the RT level. Tools for VHDL to SystemC and SystemC to VHDL translations are *VSC* and *TVS* (which are parts of the *ROSTA* environment), respectively.

Section six discusses some more advanced topics of SystemC with emphasis on SystemC channels.

Section seven presents some complex system examples described in SystemC. It also includes the related test-benches for these system descriptions. The test-benches are written in C++ language.

Finally Section eight explains the *ROSTA* environment which has been developed in CAD Research Group at the University of Tehran. It consists of a SystemC simulator and VHDL/SystemC translators (*VSC* and *TVS*).

The complete package provides enough utilities for learning advanced system-level design.