

Extracting IP-Cores of a Digital Design for New Top-Down Binding and Reuse of HDL Modules

Sarang Kazemi Nia¹, M.B. Ghaznavi-Ghouschi
Microelectronics Research Center, Urmia University, Urmia, Iran
st_s.kazeminia@urmia.ac.ir

Abstract: *In Hardware Description Languages (HDL), modular design and reuse of the previously designed components are widely used. It impacts on the Time-To-Market and life-cycle of the design components. In this paper we present a method to extract software Intellectual Properties (IP) from a given HDL design. The extracted IP cores are in Perlilog [1] templates which can be used for a new binding by Perlilog. Our proposed approach gives a systematic procedure in extraction, top-down binding and reuse of HDL modules. A new framework for declaration of modules within a top-module called here as Abstract Top Module is introduced. Experimental results of applying the proposed algorithms on a set of complete HDL designs are also presented.*

Keywords: Hardware Description Languages (HDL), Digital Design, IP cores, Top-Down binding, IP Reuse.

1 Introduction

Hardware Description Languages (HDL) are widely used in design, test and simulation of digital circuits. HDL descriptions used to describe the target designs in a unified and mostly technology and implementation independent approaches. Verilog® is a C-like, ready to learn or use HDL language. Verilog® language mostly used but not excluded for digital circuits circumstance. In Verilog® it is possible to describe hierarchical design and instantiating of modules and sub-modules. These HDL modules are called software IP-Cores. IP-cores are in turn generated from other inner sub-modules or HDL language primitives and components. This lets to use the

results of previously designed components in new designs. There are various types of complexity and rankings in IP-Core. It impacts highly on the final design. One of the most popular approaches in IP-Cores reuse is organized thru Virtual Socket Interface (VSI) Alliance VSIA [10]. The main goal in this approach is to have VSI Socketized IP-Cores for SOC and Reuse purposes [10]. Reuse of the IP-cores and reuse methodology lead to have more complex systems and systems-on-a-chip [7] in top-down approach or canonical forms. There are various tools for involving in HDL module generation and IP-Core bindings [1][5][6]. Using [5] makes automatically generate the module instantiations, top level module, and lower level module. The proposed tool in [6] takes one or more Verilog® modules and connects them together in a newly generated top level module. It can take a top level module prototype as an optional input as well. Further it supports aliases where signals with different names in the modules are connected together on the top level using a common signal name. Automatic width specification is also performed. The resulting top level will most likely still require manual editing, but 99% of the tedious copy, paste and typing work is eliminated [6]. Perlilog [1] helps the digital designers to invoke some ready blocks or IP-Cores of a large project as software packages and connect them automatically to generate “Top Module”. Also in turn Perlilog makes the test of large modules much easier [1]. In this approach there are two basic questions: How can access to the IP-Cores which are invoked in a given Top module? And how can extract these

¹ Contact person