Tutorial One

Functional Verification of System on Chips - Practices, Issues and Challenges

Subir K. Roy (Coordinating Presenter)
Synplicity, Inc., USA
subir@synplicity.com

S. Ramesh
Indian Institute of Technology Bombay, India
ramesh@cse.iitb.ernet.in

Supratik Chakraborty
Indian Institute of Technology Bombay, India
supratik@cse.iitb.ernet.in

Tsuneo Nakata
Fujitsu Laboratories Limited, Japan
nakata@flab.fujitsu.co.jp

Sreeranga P. Rajan
Fujitsu Laboratories of America, USA
sree@fia.fujitsu.com

Abstract

System on Chip (SoC) designs inherit all the well known verification and validation difficulties associated with complex ASIC designs, besides adding their own set of newer problems. These arise because SoCs are primarily implemented by re-using Intellectual Property (IP) cores. It is well known that verification today constitutes about 70% to 80% of the total design effort, thereby, making it the most expensive component in terms of cost and time, in the entire design flow. It is expected to get even worse for SoC designs.

In a complex SoC design flow functional verification is very important; any behavioral or functional bug escaping this phase will not be detected in the subsequent implementation phases and will surface only after the first silicon is integrated into the target system, resulting in costly design and silicon iterations. A number of academic and industrial research laboratories have been carrying out research on functional verification of SoCs based on different approaches. Partial success has been achieved in deploying them. Many of the issues relate to intrinsic limitations of some of the approaches taken; while others have to do with the quality of the design information, by way of, design descriptions, design documentations and design specifications, from which the overall verification objectives are derived. SoCs have brought to focus the need to carry out design and verification concurrently. For the design and verification task to proceed concurrently there is a need to capture formally, design information and implementation details at various levels of abstraction. Another reason for the need to formalize is that, as designs become more complex, functional verification will have to be carried out using the divide and conquer approach.

We discuss several approaches based on compositional verification. For these approaches to succeed, specifications of either, the individual modules, or individual IPs, if any are used, have to be stated formally. There exist several commercial offerings addressing the area of SoC functional verification.
Most of these use some form of divide and conquer approach related to compositional verification. The basis of success of some of these tools lies in the fact that the specifications of the IP cores are in essence captured in some executable form, be they formal specification languages, or commonly used HDLs suitably modified for the purpose.

This tutorial is structured to provide information on the state of the art in the area of functional verification. It will focus on existing methodologies, tools, and practical approaches based on universal simulation, emulation, formal verification, and semi-formal verification that can be employed to overcome the SoC verification problem. We discuss a number of real life verification projects, describing the various techniques used and the effectiveness of these techniques. We conclude the tutorial by presenting issues, which form the current focus for research.

Dr. Subir K. Roy is currently a Project Leader at Synplicity Inc. Prior to that he was an Assistant Professor in the Department of Electrical Engineering, IIT Kanpur from Nov. 1993 to Jan. 2001. From Feb. 1998 to Jan. 2000 he worked as a Researcher in the CAD Laboratory of Fujitsu Laboratories Limited, Kawasaki, Japan, in the area of formal and semi-formal verification of VLSI chips. His contributions in this area are in the automatic formulation of certain classes of properties from the RTL description of designs, compositional verification of SoCs, and design abstraction for model checking.

His broad areas of research interests are in the semi-formal & formal verification of SoC designs, Synthesis of Asynchronous Digital Systems, Low Power Synthesis and High Level Digital System Design. Dr. Subir K. Roy received the B.E. degree in Electronics & Telecommunication Engineering, from the University of Pune in 1982, the M.Tech. degree in Electrical Engineering from IIT Madras in 1984 and the Ph.D. degree in Electrical Engineering from IIT Bombay, in 1993. Dr. Roy has a number of publications, and has delivered several invited talks and tutorials in academic and commercial organizations.

Dr. S. Ramesh has more than 10 years of research experience in the areas of formal specification, verification and design of concurrent and reactive languages. He received the BE degree in electronic and communicaion engineering from the Indian Institute of Science, Bangalore, India in 1981, and the PhD degree in Computer Science and Engineering from Indian Institute of Technology, Bombay, India in 1987. From Aug. 1987 till Dec. 1988, he was a post-doctoral fellow at Technical University Eindhoven, Netherlands. He was then a visiting researcher at Tata Institute of Fundamental Research, Bombay for 9 months. His collaborative research with Gerard Berry's group at INRIA, Sophia-Antipolis, France, led to the development of a language for describing distributed controllers called Communicating Reactive Processes (CRP). Currently he is a Full Professor in the Dept of Computer Science and Engineering, Indian Institute of Technology, Bombay. He is also heading the Centre for Formal Design and Verification of Software recently set up in IIT Bombay to develop tools and techniques for formal verification of industrial software and hardware systems.

Dr. Supratik Chakraborty received the B.Tech. in Computer Science and Engineering from Indian Institute of Technology (IIT), Kharagpur, and subsequently received the MS and Ph.D. degrees in Electrical Engineering from Stanford University. He has served for one year as Member of Research Staff in the Advanced CAD Research group at Fujitsu Labs of America, Inc.

Currently, he is an Assistant Professor in Computer Science and Engineering at IIT Bombay, and is a Principal Investigator in the Centre for Formal Design and Verification of Software at IIT Bombay. His primary research interests are in the use of formal methods for verification and analysis of digital
systems. He is also interested in complexity and approximability issues in verification problems. He has also worked on approximation techniques for timing analysis and verification of asynchronous systems.

He has taught a post-graduate level course on asynchronous system design and verification at IIT Bombay. He has also given a tutorial lecture on BDDs and hardware verification at a workshop on formal methods held at IIT Bombay. In addition, he has given several invited presentations on formal verification methods, asynchronous systems and timing analysis at academic and commercial organizations.

Dr. T. Nakata received the B.S. degree in Electronic Engineering and the M.S. and Ph.D. degrees in Information Engineering from the University of Tokyo, Tokyo, Japan in 1981, 1983 and 1986, respectively. He joined Fujitsu Laboratories Ltd (FLL), Kawasaki, Japan in 1986 and has been engaged in research and development of computer aided design. He spent one year as a Visiting Scholar at the University of California at Berkeley from 1993 to 1994. He is a member of the IEEE and the Information Processing Society of Japan (IPSJ). His research interests include design methodology and verification of VLSIs. He presently heads the System Level Verification group at FLL.

Dr. Sreeranga P. Rajan is currently a Member of Technical Staff at Fujitsu Research Laboratories (Sunnyvale, California). Dr. Rajan's research contributions span from, developing methods for formal verification and system design, to developing compilers for embedded processors. Dr. Rajan's research in high-level verification led to the development of an automatic high-level model checking tool that has been used to debug large designs such as an ATM switch and to debug high-level synthesis tools. Dr. Rajan's early work led to the first system that integrates two powerful techniques in formal verification: theorem-proving and model-checking. The integration, which consists of efficient computation schemes within a theorem-proving framework, has led to automatic strategies for solving hard/large verification problems in software and hardware. The integrated system is distributed world-wide and has been used in industries and universities in Projects, including network protocol design, software requirements analysis, and hardware verification. Dr. Rajan has numerous publications, and has given several invited talks and tutorials.