Abstract

New trends in the Application of Formal Verification Technology in Digital Design

Eman El Mandouh & Ashraf Salem

Mentor Graphics

Cairo, Egypt

Functional verification is a critical element in the development of today’s complex digital designs. Up to 70 percent of the design development time and resources are spent on functional verification. Functional bugs are still the number one cause of silicon re-spins. They are hard-to-verify structures for which traditional simulation based verification is inefficient. Accordingly, Formal based verification becomes an emerging predominant methodology for performing hardware verification. Formal verification mathematically proves that a design property holds for all points of the search space. It comes in complement to simulation such that it needs no test bench or stimulus to operate, only the design intent (or specifications) described as properties (assertions, assumptions and cover statements) along with the design written in hardware description language are passed to the formal verification tool and assertions are examined whether they are proven to hold or found to falsify during the formal analysis session such that the given assumptions are satisfied. In general, a design property is a codified statement about a specific intended behavior of the design that must hold true under normal operating conditions. An asserted property is a property that’s expected to be always true. In practice, formulating formal design properties is usually extracted from high-level specifications that are written in natural languages. Formal verification is performed by tools such as automatic theorem provers and model checkers. Model checkers analyze a model of the design along with its properties are able to formally/mathematically prove whether or not these properties are true. Formal methods indicate a pass/fail result for the properties, and in the event of failure, counterexamples can also be generated. The advantages of formal methods are that stimuli do not need to be provided, and once a decision is reached, the result is proven correct. The disadvantage is that proving properties on complex designs can often be computationally expensive in time and processing requirements or even impractical for large designs.

Applying formal technology with block level verification in ad hoc manner can result in potential undesirable and painful results, accordingly a structured application of formal technology should started with identifying a good candidates of the design blocks for formal verification. Formal technology is excellent with design blocks that are concurrency in nature or containing multiple data streams while
they may not be that effective with sequential design blocks or design elements with a lot of data transformation. The second step should be describing the behavior of the candidate design block, its timing and its interfaces, in natural language from which the formal design properties and assumptions will be extracted. Finally the formal strategy should be picked up according to the main goal of applying formal technology which may be focus on finding proves for critical logic and design blocks, bug hunting for complex logic of the design, interface and connectivity check testing and finally coverage improvement and coverage closure.

The results of formal application are highly impacted by many factors that we will discuss in details during this session:

a) The initialization of the design under verification: Formal application requires whether initial reset state for DUV or can be used with loaded simulation states at specific time of design simulation runs.

b) Design constraint definition: DUV interfaces should be constrained so hat formal tools consider only legal stimulus, but specifying exact assumptions may be difficult and may lead to under constraint which may result in reporting of false bugs/missed proofs or over constraint the DUV which may result in detection of false proofs/missed bugs

c) Formal Budget: available processing resources and the need to set cut points to ignore irrelevant logic

d) Design Complexity: Abstraction of design for formal analysis

e) Design properties complexity: simplify complex properties down to a set of simpler properties and avoid badly written properties (vacuous proofs)

f) Designs with multiple clocks:

g) X-Semantic checks in Formal analysis: equivalence checkers will miss differences in RTL and netlist simulations caused by subtle X semantics

Applying formal verification directly to user-written properties has been proven to be very hard because users typically needs to be quite experienced to build a successful design properties and setup the formal correctly to get meaningful results. One of the major targets for formal applications is making the formal technology easier to use or by hide the formal technology from the end user and convert formal application to push bottom tools. The following list of formal technology applications will be covered in this tutorial session

1- Application of Formal Technology in SOC Connectivity Check

Formal techniques can be used to verify top level SOC integration simple electrical connection (or hookup) of the inputs and outputs of different IPs present in the SOC, and the verification of dynamic/functional integration, where besides the pure electrical connectivity a temporal and a functional dimension needs to be brought into the picture. Formal applications can help by automated generation of assertions and constraints to detect integration errors, automated generation of scripts to capture the SOC design information and invoke a formal verification tool on which to prove the validity or correctness of these assertions.
2- **Application of Formal Technology in Advanced Design Checks Analysis**

Formal Base Design Check Analyzers are much more than the traditional static linter. It uses formal analysis “under-the-hood” to analyze the sequential nature of the design logic, so it can detect problems caused by propagation of information over time. FSM Deadlock/Livelock, Register Stuck at logic and many other sequential bugs are useful examples about Formal Based Design Checks.

3- **Application of Formal Technology in Coverage Closure and Witness waveforms**

Formal property checker can be used to analyze coverage holes left by module level simulation in order to achieve early coverage closure. Formal verification can be sued to identify statically proven unreachable (i.e never satisfied design branch) design entities and hence provide feedback for simulation coverage analysis to exclude such design blocks from coverage results. Or to identify reachable design blocks and promote witness waveform for their exercises and hence boost the coverage of hardly reachable design blocks.

4- **Application of Formal Technology in X-Initialization Problems and X-Propagation Checks**

The semantics of X in RTL are extremely dangerous as RTL bugs can be masked. This state may lead to terrible failures of chips such as no clock toggling, unexpected control signal values. Formal technology has been used to detect X problems either from missed initialization of design signals or the impact of propagating X value in the design registers.

**Summary:** The Formal application tutorial will sum up how formal based verification is complementing the simulation verification techniques, industry trends about utilization of formal verification in SOC. It will cover the theory & the application for formal techniques in functional verification of design blocks; tips to get best results from the application of formal technology in design verification. It will also cover the utilization of formal techniques in formal design check analysis, SOC connectivity verification, Coverage holes detection and how formal can work with simulation to guarantee fast coverage closure. Finally we will examine the application of formal techniques in X-Init verification and X-Propagation detection.

**Biography**
Ashraf Salem is Engineering Director in Mentor Graphics Egypt. He manages a group of 110 engineers working in the development of Emulation, Simulation, Embedded Systems, and Automotive products. Dr. Salem obtained his Ph.D. from Grenoble University, France in 1992. He got his B.Sc. and M.Sc. in Computer Engineering from Ain Shams University in 1983, 1987 respectively. He was the CEO of the Technology Innovation and Entrepreneurship Center (TIEC) and professor of Computer Engineering, Faculty of Engineering, Ain Shams University. Dr. Salem participated in the establishment of ANACAD branch in Egypt in 1995 that then has been acquired by Mentor Graphics and became one of the largest multinational development centers. Dr. Salem is participated in the establishment of Software Engineering Competence Center and Information Technology Institute in Egypt.

He published more than 100 scientific articles in the fields of Computer Aided design of Digital circuits. He chaired the technical committees in a number of international conferences, and he supervised more than 20 PhD and M.Sc. thesis in digital design and Embedded systems. Also, he participated in a number of international research projects and developed one of the pioneer research products for circuit verification in the eighties.

Eman El Mandouh is Quality Assurance Manager at Mentor Graphics Egypt. She owns more than 13 years experience in digital design verification techniques namely simulation based verification, formal based verification, synthesis and equivalence checking. Eman is managing the verification and testing activities for Mentor Graphics Assertion Based Formal Verification Solutions, which covers assertion synthesis for formal and emulation flows, clock domains crossing analysis, formal based advanced design checks and model checkers formal analyzers. She is a Ph.D. candidate in Cairo University where she obtained her B.Sc. and M.Sc. in Computer Engineering in 2000, 2013 respectively. She had been certified as Green Belt Six Sigma for Information Technology and Software Engineering from the European Software Institute (ESI) and Certified Quality Manager and Organizational Excellence from American Society of Quality. She published number of articles in the field of assertion based verification and Software testing.