Abstract

In recent times, most leading chip design companies are seriously investigating the possibility of integrating property verification into their pre-silicon validation flows. Property verification allows the designer to express the key correctness requirements of a design in terms of formal properties and verify them over a given implementation. Property verification is predominantly used in two forms in pre-silicon validation, namely (a) dynamic property verification (DPV), and (b) static Formal Property Verification (FPV).

Existing property verification technology is poised at an interesting point. The benefits of property verification have been established quite emphatically in the last decade. Researchers have analyzed several historically significant failures and have shown that the use of property verification could have detected the bug in the design. Yet, the adoption of property verification techniques into the pre-silicon validation flow of chip design companies has been retarded by the following factors:

- *FPV Capacity:* Current FPV techniques do not scale beyond small circuit modules.
- *DPV Coverage:* The main criticism of the DPV approach is that only those behaviors that are covered by simulation are examined for property violation.

This thesis has two main motivations:

- We believe that the scalability and efficiency of formal and semi-formal verification can be improved by adopting specification styles that syntactically facilitate abstractions and pruning.
- We believe that the assertion coverage in DPV can be improved by guiding the simulation through formal methods.

The main objective of this thesis is to study the above issues and propose methods for accelerating formal, semi-formal and dynamic property verification. In particular, we have the following contributions:

- *Accelerating Property Coverage in DPV:* We have defined an automated methodology that can analyze formal properties and produce tests that trigger them. This has been integrated within a constrained random test architecture to accelerate coverage of corner case assertions during DPV.
- *Context-sensitive specifications:* A significant fraction of the correctness requirements in standard protocol descriptions are *context-sensitive*, i.e. they apply only when the protocol is in specific contexts. We have formalized a modeling style for expressing such properties and proposed algorithms for formal and semi-formal verification and consistency analysis for such properties.

- *Annotating Temporal Operators for Modular FPV:* To ameliorate the state explosion problem, there has been a paradigm shift from a system level FPV perspective to a modular one. To capture module-level specifications, we have proposed an extension of Linear Temporal Logic by annotating temporal operators with input constraints. In addition, we have proposed a symbolic BDD-based algorithm for verifying such properties.
- *An integrated DPV platform for UML Statechart validation:* We have developed a complete DPV platform for verifying temporal requirements over UML Statecharts. The DPV framework allows the user to specify correctness requirements in a rich assertion specification language and verifies them during simulation.

We believe that the formal methods presented in this thesis will lead to wider adoption of property verification techniques in the design validation flow.