Another Dimension to High Level Synthesis: Verification

Malay K Ganai and Aarti Gupta
NEC Laboratories America, Princeton, USA

Akira Mukaiyama and Kazutoshi Wakabayashi
Central Research Lab, NEC, Tokyo, Japan

Abstract

To cope with the increasing design complexity and size, significant efforts are being made in high level synthesis methodologies, design languages and verification methodologies to leverage the expressive power of high-level models and reduce the design cycle time and cost. High level synthesis is a process of generating a concrete structure to the high level design meeting, three main design constraints: area, timing and power [1-4]. Verification, on the other hand, ensures that the design meets the specification. As of today, verification efforts take about 70% of design cycle. At structural level, the formal verification techniques are quite matured and advanced compared to those at the high level. Part of the reason is due to a general skepticism among designers in regards to the verification efforts made on the abstract model of the design instead on the structured design which is closer to the design on chip.

To leverage off recent advancements made in Boolean SAT-based verification techniques [5] and to reduce the skepticism among the designers, we believe that HLS methodologies should also focus on another dimension, i.e., functional verification. Though this dimension is not part of standard design constraints, but it can help to substantially reduce the verification effort. HLS, in general, does not necessarily synthesize models that are verification "friendly". On at least the following three counts, we find that verification tools perform poorly on the models synthesized by HLS.

- Area optimization: Given limited hardware resources in a typical HLS problem, the structural level design synthesized will have a large number of multiplexers as compared to a design synthesized under the assumption of unlimited resources. This multiplexer count is increased due to sharing of the limited operators. It is quite well-established in SAT community that multiplexers, in general, are not good for SAT engines and hence, SAT-based verification engines are affected adversely.
- Sequential scheduling: Because of the limited resources constraint, the synthesized design is sequentially deeper. Sequential depth can also increase due to multi-cycle operators. Such increase in the sequential depth results in time-consuming deeper searches by verification engines, adversely affecting its performance and also requiring debugging of longer error traces.
- Memory optimization: Traditionally, HLS uses explicit memory modeling for embedded memories. However it has been shown that [6] for embedded memories, Efficient Memory Model (EMM) for verification is far superior to an explicit model.

We have experimented with HLS tool Cyber [1], integrated with a SAT-based model checking tool, DiVer [5]. These tools have state-of-the-art high level synthesis and verification algorithms respectively and are widely adopted by NEC designers. We used Cyber to synthesize a design in Behavioral Description Language (BDL) in two modes: (a) with *maximal* sharing of operators and (b) with *minimal* sharing of operators. Design I synthesized in mode (a) has 2450 multiplexers (muxes) with 31 fsm states. Design II synthesized in mode (b) has 2144 muxes with 25 fsm states. The verification time on a safety property on the synthesized Design I was 12s and on Design II was 6s. Though the area is increased in Design II, the search is improved as the problem partitioning is done at higher level (by removal of muxes). Such partitioning can also be done at bit-level by SAT decision procedure, but is much less effective. Clearly, objectives of optimization for synthesis and verification are not the same.

Verification complexity is growing exponentially with design complexity. As high level synthesis tools are slowly and steadily becoming popular among designers, we believe that it would be almost impossible for HLS providers to ignore "synthesis for verification" paradigm. In order to bridge the widening gap, we propose the use of existing infrastructure of HLS to generate verification friendly models and properties from the given high-level design and specification. The verification can then be carried out on the friendly model to obtain a "golden" reference model. Moreover, as HLS generates both the verification and synthesized structural model, it has the internal signal correspondences between them. This information subsequently can be used by an equivalence checker tool to validate the correctness of structural design (that might have been manually changed to meet design constraints) against the "golden" reference model. We believe that this is also an important step towards removing the skepticism among the designers in regards to verification efforts made on an abstract model and not on the structural model. Currently, we are exploring various high level synthesis heuristics which can be used to generate smarter verification model. In the talk, we would discuss various challenges and issues with such paradigm.

References

[1] K. Wakabayashi, "Cyber: High Level Synthesis System from Software into ASIC," in *High Level VLSI Synthesis*, R. Camposano and W. Wolf, Eds.: Kluwer Academic Publishers, 1991, pp. 127-151.

[2] D. D. Gajksi, N. Dutt, S. Y.-L. Lin, and A. Wu, *High Level Synthesis: Introduction to Chip and System Design*: Kluwer Academic Publishers, 1992.

[3] D. C. Ku and G. D. Michelli, *High Level Synthesis of ASICs Under Timing and Synchronization Constraints*: Kluwer Academic Publishers, 1992.

[4] J. M. Chang and M. Pedram, *Power Optimization and Synthesis at Behavorial and System Levels using Formal Methods*: Kluwer Academic Publishers, 1999.

[5] M. Ganai, A. Gupta, and P. Ashar, "DiVer: SAT-Based Model Checking Platform for Verifying Large Scale Systems," in *Proceeding of TACAS*, 2005.

[6] M. Ganai, A. Gupta, and P. Ashar, "Efficient Modeling of Embedded Memories in Bounded Model Checking," in *Proceedings of CAV*, 2004, pp. 440-452.