On reusability of verification assertions for testing

Maksim Jenihhin, Jaan Raik, Raimund Ubar, Anton Chepurov

Dept. of Computer Engineering, Tallinn University of Technology
{ maksim | jaan | raiub | anchep }@pld.ttu.ee

ABSTRACT: Assertions have proven to be an effective mechanism to improve quality and to speed-up simulation-based design verification. They are created and embedded to the simulatable design description by the designer, the person with the deepest knowledge about the desired functionality and its real implementation. In this paper we propose to reuse this valuable information during the design manufacturing testing phase to increase the test quality and efficiency. The paper considers different types of design properties created for verification such as environmental assumptions and internal signal assertions. The reusable information is proposed to be applied for test pattern generation, embedded test observability improvement and DfT (design for testability) enhancement.

1 Introduction

As the digital hardware complexity grows very fast, its functional verification, meant to ensure that design fulfills the given specifications, is extremely important step in the whole development flow. Simulation-based (dynamic) and formal (static) approaches are two major types of verification. Both of them can employ assertions to improve the quality and speed-up the verification process.

The assertions are created by designer and contain a very valuable knowledge about the design’s functionality and structure. During the verification phase they are simulated together with the design and help to discover design’s incorrect implementation faster and more efficiently (see [1]) or checked formally. Further, the assertions are removed from the design’s synthesizable version.

However the correct implementation of the design’s given specification does not guarantee that the product will get to the customer fault-free. The manufacturing phase should be followed by an appropriate testing as well. The task of the test preparation can be aided by Design for Testability (DfT) techniques performed prior manufacturing. Nevertheless the manufacturing test development for modern circuits remains a very complex task.

In this paper we propose to reuse the verification assertions for manufacturing test development. Please see Fig.1. (Here ASIC stands for Application Specific Integrated Circuit.)

Figure 1. ASIC development flow

There have been proposed methods in our previous works [3],[4] for test generation based on design properties, however the properties themselves were not automatically obtained. The methods considering assertions reusability for testing were also recently proposed by Riazati, Navabi et al in [6],[7] and Boule and Zilic in [8],[9] however they have several limitations and focused only on test pattern generation by the synthesized checkers. The approaches do not consider other aspects of test plan development and lead to large area overhead.

A number of approaches aiming hardware checkers creation from assertions meant for prototype verification/validation by emulation (a development flow phase standing before mass-production), such as [2],[10],[11],[12],[13] can be extended for this purpose. The most widely known commercial tool for usual assertions conversion to their synthesizable form is FoCs from IBM [16],[17].
In this paper we propose a wider range of manufacturing test plan development areas where assertions can be reused.

The rest of the paper is organized as follows. Section 2 discusses test pattern generation based on assumptions. Section 3 proposes an idea for Built-In Self-Test quality improvement with assertions. Section 4 proposes a DfT technique applying assertions for test points insertion. Section 5 presents experimental results for the proof of concept. Section 6 concludes the paper.

2 Assumption-based Test Generation

A part of today manufactured ASICs contains embedded during DfT scan-chains that increase observability and controllability of the circuits under test (CUT) by providing an access to the internal registers. This method allows reaching high fault coverage, however it results in over-testing of the core (see e.g. [14]). In other words, it covers faults that could never influence the functional behavior of the circuit thus reducing the yield. From the other hand, it has been shown that non-scan testing based on pseudorandom test sequences can be highly feasible for several types of designs (such as crypto cores as it was proven in [15]). Finally, not all the circuits may have the embedded scan. The ideas proposed in this paper consider non-scan sequential circuits. However they can be partially applied to scan circuits as well.

The notion of test cubes is applied for test set representation when several bits in the test vectors contain unassigned values X (See Fig. 2). These bits can be assigned to the normal binary values “1” or “0” whether based on some rules or just randomly. The “X”-s add the 3rd dimension to the fully determined 2-dimensional test set.

In case if the test engineer has no prior information about the CUT’s input data dependencies (assumptions) he has to start with a test set filled with all X-s (an empty set). The assumptions provide information how to assign a part of the bits in the test cube which increases the final test quality and eases the test generation process. For example if an assumption provides us information that some particular input (pin) of the CUT is its RESET signal, then the test engineer may decide to set its value to “1” only once in some sequence of cycles. The assumptions may be much more complex and origin from a design specification and its implementation (design) phase.

Both the assumptions (sometimes referred also as environmental constraints) and assertions are design properties. The first ones contain information about the design’s environment and therefore its expected inputs dependencies while the second ones describe the dependencies of the design’s internal signals and outputs [21]. Usually both the assumptions and assertions are described in one of the property description languages such as SVA (System Verilog Assertions, [18]) or PSL (Property Specification Language, IEEE 1850 standard, [19],[20],[21]).

The assumptions can be described explicitly by the designer or implicitly follow from some of the given assertions of the connected designs or design cores. The latter case is shown in Figure 3 and known as assumption-assertion dualism [22]. This method allows obtaining a wider set of assumptions.

![Assumption-assertion dualism](image)

The final set of the assumptions can be divided in two groups. The first group is the assumptions explicitly assigning a set of bits in the test cube. The rest of the undetermined bits should be assigned by Pseudorandom or Genetic TPG. Deterministic TPG may be not reasonably efficient in this case. The second group representing a set of more complex rules not suitable for straightforward bits assignment is meant for an appropriate Deterministic TPG constraints representation. The both groups of the assumptions should be utilized for the maximum efficiency.

The main difference of the proposed approach from the one described in [3],[4] is the formal source and format of the assumptions.

3 Assertion-based BIST

The concern we would like to emphasize in this section is the application of the information provided by assertions for BIST response analyzer observability enhancement. The Built-In Self-Test (BIST) has been
proven to be an efficient approach for manufacturing testing. However in case of non-scan circuits its main bottleneck is observability.

A method for verification assertions synthesis to hardware assertion checkers embedded to the circuits for on-line testing is proposed by Riazati, Navabi et al in [6],[7] and by Boule in [9]. The other approaches meant for prototype validation mentioned in the first section can be extended for this purpose. However they consider TPG to be performed offline.

We propose to apply the verification assertions in the form of hardware assertion checkers for BIST observability enhancement. They could act as separate observers in BIST response analyzer architecture. The second application for them is to aid building complex controller-based BIST observer. One of the task of which would be to inform the analyzer when to check and when to stay in “Silence Mode”.

The main issue here is the observability coverage by the existing assertions. Therefore, the efficiency of this technique should be very much dependent on the particular DUV and the designer’s assertions choice style.

4 Assertion-based DfT

The two main components of CUT’s testability are observability and controllability. The both of them can be significantly improved during DfT phase. These improvements usually include minor rearrangements inside the design or addition of an extra logic (for example scan-chains as it was mentioned in Section 2).

One of the DfT techniques is “test points insertion”, when an additional auxiliary input pin is routed to an internal net of the CUT. The main drawback of this technique is hardware area- and input pins overhead. The second ones are very costly and increase the significance of this drawback. Therefore, only the nets providing good controllability coverage should be chosen. The identification of such nets is a very complex task, usually solved by heuristics.

We propose to use for test points the nets corresponding to the signals (operands) from assertions (Fig. 4). The good candidates for test points’ locations are the nets with hard-to-test faults, due to very low controllability. The other intent for test point insertion would be a fan-out that provides at once a good controllability to a set of middle-level controllability places. Such a test point would shorten the test length.

The both criteria can be usually satisfied by normal assertions, because the assertions reflect the designer’s conception of the most significant cornerstones of the design.

Test points insertion to the sequential circuits may lead at some extend to the same drawback of over-testing as in case of scan-chains insertion mentioned in Section 2.

5 Experimental Results

The implementation of the proposed ideas based on real assertions from industrial design projects is planned as the future work. The experimental data presented in this section originates from our earlier works ([3], [5]) and provided here for the proof of concept.

Table 1 presents the results of a case study for two controller designs without scan chains. The first one is I²C controller from OpenCores library [24] and the second one is an industrial memory controller. The table provides comparison of stuck-at fault coverage achieved by Normal pseudo-random TPG (row 5) and pseudo-random TPG aided by assumptions (row 6). The fault coverage improvement achieved by the proposed approach for the two case-study designs is 1.6 and 3.8 times respectively.

Table 1. Assumption-based TPG case-study

<table>
<thead>
<tr>
<th>Circuit</th>
<th>I²C controller</th>
<th>Memory controller</th>
</tr>
</thead>
<tbody>
<tr>
<td># faults</td>
<td>2440</td>
<td>24242</td>
</tr>
<tr>
<td># flip-flops</td>
<td>114</td>
<td>639</td>
</tr>
<tr>
<td># inputs</td>
<td>20</td>
<td>300</td>
</tr>
<tr>
<td>PR TPG fault coverage, %</td>
<td>Normal</td>
<td>40.87</td>
</tr>
<tr>
<td></td>
<td>Assumption-based</td>
<td>67.95</td>
</tr>
</tbody>
</table>

Table 2 shows the potential of Register-Transfer Level (RTL) test point insertion for sequential non-scan circuits [5]. The second column (initial FC) presents the fault coverage achieved by a sequential ATPG for the unaltered design. The third column (DfT FC) displays fault coverage for the circuit improved by test point insertion. The final column indicates the additional area overhead required by the DfT structures (test points and output response analyzers). In this experimental work test point insertion was not carried out based on assertions but relying on feedback provided by the ATPG. Nevertheless,
they prove that considerable gain in fault coverage can be achieved by the proposed approach.

Table 2. Test point insertion experiments

<table>
<thead>
<tr>
<th>design</th>
<th>Initial FC, %</th>
<th>DfT FC, %</th>
<th>Area overhead, %</th>
</tr>
</thead>
<tbody>
<tr>
<td>gcd</td>
<td>94.79</td>
<td>100</td>
<td>15.6</td>
</tr>
<tr>
<td>mult8x8</td>
<td>66.40</td>
<td>92.39</td>
<td>42.7</td>
</tr>
<tr>
<td>sosq</td>
<td>70.50</td>
<td>90.22</td>
<td>37.1</td>
</tr>
<tr>
<td>risc</td>
<td>98.13</td>
<td>99.95</td>
<td>27.5</td>
</tr>
<tr>
<td>ellipf</td>
<td>88.40</td>
<td>99.05</td>
<td>34.1</td>
</tr>
<tr>
<td>difeq</td>
<td>97.05</td>
<td>98.61</td>
<td>12.9</td>
</tr>
</tbody>
</table>

6 Conclusions and future work

The paper discusses extended reusability of verification assertions for the design manufacturing testing phase to increase the test quality and efficiency.

The paper considers different types of design properties created for verification such as environmental assumptions and internal signal assertions. The reusable information is proposed to be applied for test pattern generation, embedded test observability improvement and DfT (design for testability) enhancement.

The proposed ideas are supported by the proof of concept based on several case-studies and our previous experiments.

As a future work it is planned to perform a comprehensive analysis of the proposed approaches efficiency based on state-of-the-art industrial designs projects.

Acknowledgements

This work was supported in part by EC FP 6 research project VERTIGO FP6-2005-IST-5-033709 [23], Estonian SF grants 7068 and 7483 Estonian Information Technology Foundation (EITSA) and Nations Support Program for the ICT in Higher Education "Tiger University".

References