Facilitating At-speed Test at the Register Transfer Level

Shivakumar Swaminathan
Qualcomm Inc.

Ralph Marlett
Atrenta Inc.

Abstract

We present a case study on preparing multi-frequency ASIC designs conducive for at-speed testing of transition faults. Most of the focus in making design engineers aware of DFT has been geared towards high stuck-at fault coverage by modifying the design for controllability and observability. However, as transistor geometries decrease, speed related defects have become the more predominant failure mode compared to static faults. All modern industry tools perform frequency assignment for at-speed test at the gate level after synthesis. We present a method to design for at-speed-test at the RT level for designs under a generic clock distribution system. The proposed technique aids in creating DFT-friendly design earlier in the design phase, reduces development time and ensures high stuck-at and transition fault coverage.

1. Introduction

There has been an enormous push for DFT at the RT level for the past several years [1, 2]. Leading chip makers realize the importance of looking at the manufacturing test readiness of their SoCs early in the design phase to prevent design iterations motivated by improved test coverage. However, most of this effort has been focused on achieving high stuck-at fault coverage for designs with feature sizes of 90nm and above. As the industry moves towards the deep sub-micron technologies of 65nm and below, speed-related defects are becoming common and gaining more attention. Transition fault coverage and path delay fault coverage have become equally important and are an integral part of any robust DFT strategy today. However, little has been done to make the front end designers aware of at-speed test solutions.

A substantial amount of research has been done in the field of at-speed testing using BIST as well as external testing. With the steady increase in the number and value of the operating frequencies in the design and the inability of low-cost testers to produce high-frequency clocks, on-chip clock controllers are designed to deliver the at speed capture pulses to the sequential elements in the circuit. However, these pulses have been supplied to the storage elements at the leaf stage [3], thereby bypassing the entire clock network in test mode. It has been observed [4] that the number of faults in the clock network has become a significant portion of the total fault population because the clock network has become larger with increasing design complexity and size. This has necessitated the development of a DFT method that reuses the clock network for scan and capture modes to address at-speed coverage requirements [4]. In addition, the difficulties in designing on-chip clock generators to deliver capture clocks at the precise operating frequencies, especially in communication chips, has led to solutions not truly at-speed compliant that results in a significant reduction in test quality of high-speed designs.

In this paper, we present an at-speed test method for multi-frequency ASIC designs. We describe the characteristics of an on-chip clock generator that supplies at-speed pulses at the exact design frequencies at the input of the clock network. We develop a technique wherein each section of the design, running at a particular frequency, is tested at-speed independently by disabling the clock to the rest of the design. This ensures a truly at speed solution that re-uses the system clock network for improved coverage and simpler design. We also show that the solution is generic in the sense that it supports different types of generated clocks used by standard EDA tools for coverage analysis and test vector generation.

The organization of this paper is as follows. At-speed design is briefly described in section 2. We describe the test method in section 3. We propose at-speed rules in section 4 and conclude in section 5.

2. At-speed Design

The at-speed design solution for multi frequency designs presented in this paper requires:

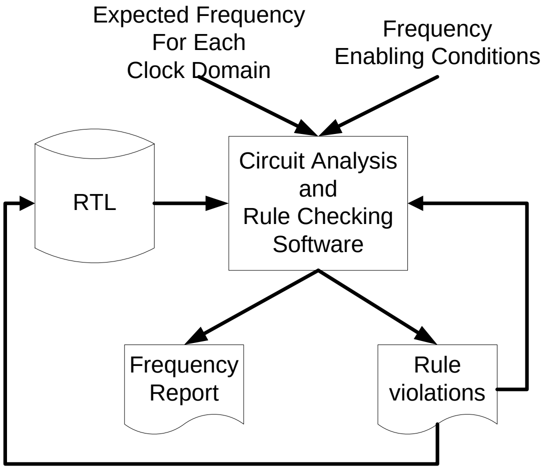
- on-chip clock generation to supply at-speed pulses at the design frequencies to the input of the clock network., and
- a clock distribution technique wherein the clock to each section of the design that runs at a particular frequency is independently tested at-speed by disabling the clock to the rest of the design.

This strategy ensures an at-speed solution that has the added benefit that it re-uses the clock network for improved coverage.

Testing one frequency domain at a time also overcomes the limitations posed by many commercial ATPG tools, which are unable to pulse two or more frequencies for vector generation in large designs with significant amount of inter-domain interaction. Moreover, certain designs may have a large number of internally generated clocks originating from the same root, thereby necessitating at-speed test at a single frequency if the clock network is re-used in test application.

3. At-speed Test Flow

The design principles to fully cover the issues described in the last section, become substantially more complex to implement and verify in circuits with hundreds of clock domains distributed over numerous levels of hierarchy. To ensure that the enabling or disabling of critical branches in the clock tree is performed accurately in a large design, a comprehensive set of rules has been developed that operate as shown below.



In addition to the RTL, the user specifies:
1. frequencies expected for each clock domain
2. frequency enabling conditions

The analysis software checks whether or not the paths from the clock generation blocks are fully enabled to each clock domain and if the expected frequency controls each domain. We describe more details for some rules in the next section.

4. At-speed Rules

The primary constraint for at-speed test can be described as follows: Assume that a design has *t* _ testclocks, namely c_1, c_2, c…, c_t, each of which can produce pulses at *f* frequencies, namely, ci_1, ci_2, …, ci_f using 1-hot enables ei_1, ei_2, …, ei_f where *i* corresponds to the ith testclock. This means $c_i = ci_j$, when $ei_j = 1$ for 1 <= i <= t and 1 <= j <= f.

Rule 1: **There exists a single path from every register to a valid test clock source.**

This rule ensures that there is a single, unique path from test clock source to every register. In order for this rule to pass, any logic gates on the clock path must have non-controlling values on their off-path inputs and mux-selects must be fixed to a known value.

Rule 2: **The clock network should allow only one frequency during at-speed capture.**

This rule ensures that we don't test logic above or below the rated-speed, thus ensuring accurate at-speed coverage numbers from the DFT and ATPG tools used during synthesis. This will also ensure that each frequency domain is correctly isolated.

Rule 3: **Frequencies specified in the test plan for a given node must be realized.**

This rule checks if a user-specified frequency can be reached at any user-specified logic component in the entire design.

5. Conclusions

Design for at-speed test is extremely important to address speed related defects in deep-submicron designs. It has become critical to deal with at-speed compliance as early as possible in the design cycle to avoid design changes that otherwise would be carried out late in the design to accommodate at-speed testing and the associated functional verification overhead that may significantly impact tape out schedules.

We have proposed an at-speed test infrastructure at RT level that enables faster time to market, by eliminating time-consuming testability improvement efforts on gate-level netlists, and improved test coverage by applying the at-speed tests with the system clock distribution logic.

One limitation of the above approach is the inability to test inter-clock-domain transition faults since each clock domain is tested independently. Special clock filter cells have been proposed [4] that allow launch in one domain and capture in another domain at the cost of extra hardware overhead. An alternative solution is to use capture-disable techniques to avoid hold time violations [5]. Another problem is detecting delay faults in multi-cycle paths. Again, special circuitry can be designed [5] that disallow capture at the destination flops in such paths. However, such design structures are a comparatively low percentage of a large SoC and can also be handled by efficient gate-level tools with sdf back-annotation [3]. Further research is being done to incorporate DFT rule checking for these issues.

References

1. Marlett. R, Putting the D back into design for test, *EDA vision article*, 2002.
2. Aktouf C, Why Haven't EDA vendors delivered DFT at RTL, *EDA Alert article*, 2005.
3. *Synopsys DFT-Max User Guide*, Synopsys Inc.
4. Beck, M, Barondeau, O, Kaibel, M, Poehl, F, Xijiang Lin, Press, R, Logic Design for On-chip Test Clock Generation – Implementation Details and Impact on Delay Test Quality, *Proceedings Design, Automation and Test in Europe*, pp. 56-61, vol 1 2005.
5. Pateras S, Achieving At-Speed Structural Test, *IEEE Design and Test of Computers*, vol 20, No. 5, pp. 26-33, Sep-Oct 2003.