3rd International Conference on Electrical & Computer Engineering
ICECE 2004, 28-30 December 2004, Dhaka, Bangladesh

DESIGN CONSIDERATION FOR SUCCESSFUL DELAY FAULT TESTING IN SOC

Sreedharan Baskara Dass(1) *and Aisha Hassan Abdalla*(2)

Trans-Dist Engineering Sdn. Bhd, Putrajaya, Malaysia(1)
Faculty of Engineering, International Islamic University Malaysia, Jalan Gombak, 53100 Kuala Lumpur, Malaysia(2)
(2)Email: aisha@iiu.edu.my

ABSTRACT

Delay Fault Testing using scan patterns has been increasingly popular in the DFT world. There's a debate whether at-speed test with scan patterns can actually replace functional at-speed tests. This paper looks at some of the design considerations for making SoC more delay test friendly and ready. The test chip was designed scan ready but with no delay fault testing constructs.

1. INTRODUCTION

Functional testing verifies that the circuit performs as it is intended to perform. Manufacturing testing verifies that the circuit does not have manufacturing defects by focusing on circuit structure rather than functional behavior [1]. Manufacturing defects include problems such as

- Power or ground shorts
- Open interconnect on the die caused by dust particles
- Short-circuited source or drain on the transistor caused by metal spike-through

Structured DFT techniques, such as internal scan, simplify the test-pattern generation task for complex sequential designs, resulting in higher fault coverage and reduced testing costs. Defects such as slow transition of nodes may go unnoticed since the test is conducted at much lower frequencies. With today's SoCs reaching gigahertz operating range, it's getting more and more urgent for manufacturing test to cover not just stuck-at faults, but delay faults as well. A delay faults is a situation in which a node functions properly but fails to meet its timing requirements [2].

In the following sections we will discuss the ATPG/Scan based delay fault testing design flow and discuss the various issues that must be handled to ensure high fault coverage. An overview of the test chip used to carry out and verify the procedures is outlined below. All the design tasks were carried out using Synopsys DFT-Compiler for scan synthesis and TetraMax for ATPG (scan and delay).

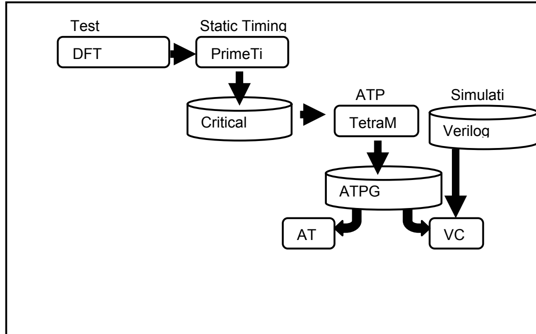
2. PATH DELAY FAULT TESTING METHODOLOGY

Path delay fault tests exercise the critical paths at-speed (the full operating speed of the chip) to detect whether the path is too slow because of manufacturing defects or variations. Path delay faults are tested using the following sequence: The first vector initializes the path before applying the launch event, typically a clock pulse. The launch event generates the second vector, which propagates a logic transition along the entire path. A second clock pulse, occurring one at-speed cycle after the launch clock, captures the resulting transition at the end of the path. Path delay fault ATPG targets individual path delay faults and then simulates each test generated against the remaining undetected faults in the fault list [3]. Figure 1 shows the basic TetraMAX steps and checkpoints to generate an effective set of path delay tests.

A path delay test cycle uses the same order of events as for other fault models

- force primary inputs
- measure primary outputs (optional)
- pulse a clock

Given this order of events, one or two test cycles are required to launch and capture a path delay fault. For most paths, a two-cycle test is generated to apply a launch clock pulse and a capture clock pulse.



]Fig. 1 Basic TetraMax Path Delay Test flow

3. DESIGN CONSIDERATIONS

In this section we consider various design issues that must be taken into consideration when designing SoC which will employ path delay test technique. All this method was tried out on the test chip. Following are methods and technique which helped in increasing fault coverage of circuit using path delay test schemes:

1. Design must be free of any setup/hold timing violation. Design which contains large setup and hold violation should not be employed for path delay tests. Such circuits will flag false violation – these violations are present in the circuit due to design flaws not manufacturing. It's imperative that the design is taken through extensive timing analysis and simulation to weed out all timing bugs.

2. Employ Sequential ATPG for Memory blocks. Memory blocks present themselves as black-boxes and thus are un-testable. While these blocks have their own testing mechanism, they can prevent logic surrounding them from being tested. This is called shadow effect [4]. The overall effect is low fault coverage. The solution is to TetraMax's sequential ATPG is for memory block. This way the path is preserved and at-speed test can be conducted on the actual circuit as is the normal operation mode.

3. Maintain clock edges in functional (normal operating) mode and test mode. For example, a Flip-Flop could be clocked by negative edge clock in functional mode but clocked by positive edge clock in test mode. This can increase the efficiency of stuck-at scan pattern generation but create problems in delay test. This is because a timing critical path in functional mode is masked out in test mode.

4. Combining all the internally generated clocks into one clock source in test mode should be avoided. There are several methods to solve this problem [5]. One method is to have one dedicated pin per internally generated clock in test mode. From ATPG tool's point of view, the design has multiple clocks. All clocks are still clocked at the same time during shift, but the ATPG is now free to handle the clock domains in different ways during the capture cycle. The launch and capture events can now be individually controlled and pulsed for each clock domain.

5. The transition fault test patterns can be generated first and then fault simulated with the stuck-at fault model to obtain the stuck-at fault coverage. If the stuck-at fault coverage is not sufficient at this point, then stuck-at fault test can be run to top up the transition pattern set to increase the stuck-at test coverage.

6. Logic which can cause un-testable paths should be avoided. This includes:
   a. It is a sequentially false path. Such paths cannot be tested in a functional mode, because logic prevents the required state transitions.
   b. Redundant logic (for circuit speed) prevents a single path from being independently tested. Multiplier arrays are a good example of such circuits.

7. Always perform fault simulation with the ATPG pattern set to ensure that the test pattern do capture all violation on a circuit with defects (bad-machine simulation) and does not flag any violation in a defect-free circuit (good machine simulation). With TetraMax, you can write out a Verilog test bench for pattern validation.

4. CONCLUSION

Scan-based delay test has many advantages however; to achieve the high test coverage and quality of result some design effort is required. Careful design planning and consideration to make the chip more delay test ready from the beginning is important. Otherwise the end result can be very discouraging. Strict adherence to DFT rules will ensure the circuit achieves the desired test goals.

Based on our analysis, scan-based path delay tests complements nicely the stuck-at scan test, by combining both its possible to achieve high fault coverage pattern sets to test chip at their normal operating frequencies.

5. REFERENCES

1. *"Test Compiler Tutorial",* Version 1998.08, Synopsys Incorporated,. 1998
2. "Application Note Production Test methodology for Catalina7 Reusable Design Database", Ken Jaramillo, VLSI Technology, Inc., 1999.
3. "Considerations for TetraMAX DelayTest", 2003 Synopsys Solvnet Article
4. "IP Characterization with Pathmill and PrimeTime", Stefan Scharfenberg, SNUG 1999.
5. "Scan Synthesis User Guide", Synopsys, Incorporated, 2002.