

# COMPARATIVE ANALYSIS OF SIMULATION TECHNIQUES: SCAN COMPRESSION AND INTERNAL SCAN

Parul Patel<sup>1</sup>, Arvind Rajawat<sup>1</sup> and Pooja Jain<sup>2</sup>

<sup>1</sup>Department of Electronics and Communication, Maulana Azad National Institute of Technology Bhopal, India

<sup>2</sup>Automotive and Discrete group STMicroelectronics Private Limited  
Greater Noida, India

## ABSTRACT

*With advancement in technology, the feature size of transistors is shrinking and the transistor count in a circuit design is exponentially increasing. As a result, it is hard to control and observe internal nodes leading to complexity in locating and debugging faults specially for sequential circuits. Design for Testability (DFT) provides a way for fault detection of the circuit under test in less simulation duration with little increase in area. Many techniques are proposed under DFT for pattern simulation. In this paper, we have compared two such pattern simulation techniques namely scan compression and internal scan. The experiment is performed on different benchmark circuits, it is observed the simulation time is significantly reduced with increased coverage and a little area overhead.*

## KEYWORDS

*DFT, Scan Compression, Internal Scan*

## 1. INTRODUCTION

CMOS technology has attained very high advancement in scaling i.e., feature size, the designs have become more and more compact and much improved in terms of multi functioning, speed, reliability, and robustness. With these advancements there is high demand of these computing devices now a days in many industries and sectors whether it be medical sector, military sector, telecom department, domestic sectors – homes and shops etc. are all somehow related to this VLSI industry and so the demand of electronic products have boomed to a greater extent. The credit of this expansion can be allotted to the increase in memory density, low computing power and intellectual property modules. With the current trends in VLSI because of the compactness, low power consumption, higher frequency of operation there has been a huge necessity of testing of those components before delivering it to the consumers. The improvement in these areas has led to increased complexity and so the occurrence of manufacturing defects in ICs increases with decrease in transistor sizing.

Design for Testability (DFT)[1] based on scan and ATPG has been adopted as a reliable and broadly acceptable methodology that provides very high test coverage. Though for large circuits, the growing test data volume causes a significant increase in test cost because of much longer test time and elevated tester memory requirements. Scan compression techniques[2] are introduced to reduce test time. The compressed scan architecture is similar to the scan at the chip-level interface, but it contains combinational compression logic and uses many more scan chains of

shorter lengths called compressed scan chains, within the chip core. Figure 1 shows the DFT insertion flow. The first step of the DFT flow is the RTL design. In RTL design the DFT related blocks are inserted into the design. Flip flop are preceded by a multiplexer called a scan cell. Adding Multiplexer to flip flop results in Unrouted netlist. Stitching all such scan cells in chains and making requisite connection forms routed netlist. Routed netlist undergo optimization and design rule check leading to final netlist termed as DFT inserted netlist.

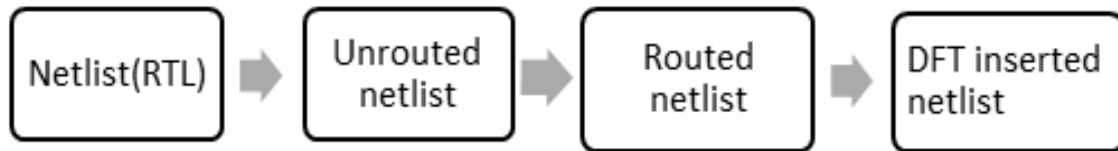


Figure 1. DFT insertion flow

## 2. LITERATURE REVIEW

The first step of the DFT flow[3] is the RTL design. In RTL design the DFT related blocks are inserted into the design and then the design is verified whether the desired response is obtained. The synthesis of the design generating netlist for further verification is carried out. Scan insertion is performed on synthesis netlist. The SDF file and library files obtained after routing and CTS are used for scan insertion. The scan synthesis process reads the RTL design, synthesizes it, tests it again, performs scan insertion and analyses the post DFT- Design. In the next step ATPG is done in which random test patterns are generated. These test patterns are passed as scan chains and simulation is carried out to test the results.

### 2.1. Scan Cell

Figure 2 shows a basic scan cell[4]. Scan cell is the building block of DFT architecture. It has two inputs namely Data in, Scan in and one out Data out. A scan cell can operate in two modes depending on the control signal scan Enable. One mode is functional mode in which the flip flop does its normal function for which it is placed in design with data in as input and another mode is test mode in which it acts as part of the scan chain with Scan in as input. A scan cell always contains at least one memory element.

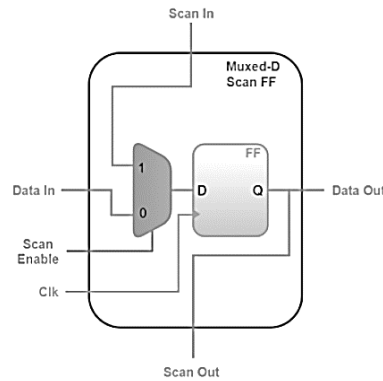


Figure 2. Scan cell

## 2.2. Scan Chain

When all the scan cells are stitched together that is scan out of preceding scan cell is connected to scan in of succeeding scan cell then the resulting chain is termed as Scan chain[4].

Figure 3 shows a design without scan(left), in which flip flop is represented by square and with scan(right) in which scan cell is shown in square where the rectangle inside square represents multiplexer. With full scan design(right), the red line shows connection forming scan chains and all scan cells have the same control signal (scan enable) shown in blue. A practical circuit may contain more than one scan chain.

When scan enable is ‘1’, the scan chain is active i.e., the scan cell is operating in scan mode. When scan enable is ‘0’, the scan chain is inactive i.e., the scan cell is operating in normal mode.

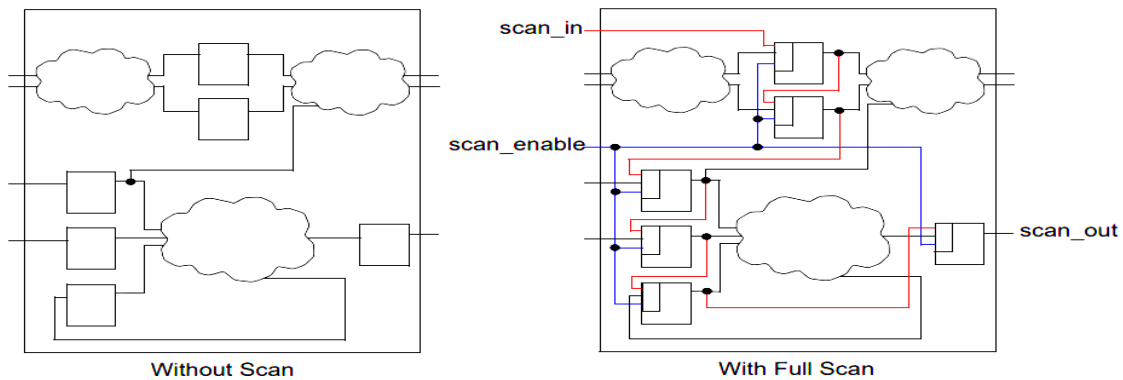


Figure 3. Example of design with and without scan insertion

## 2.3. Scan Operation

Initially, the scan enable is set to ‘1’, activating scan mode[5]. In this mode, Scan data is loaded in scan flops, Shift-In Phase (refer to Figure 4). The clock applied in this period to loading data is termed as shift clock. Once all the scan cells are loaded with test vector then the scan enable is set to ‘0’, activating functional mode. Test vector is forced at primary input and after action of combinational logic, output is available at certain scan cells input. In figure 3, clouds represent combinational logic. One more clock pulse is required to observe the output at the scan out port.

This pulse is termed as capture pulse. Once the output is captured, scan enable is again set to '1', enabling next pattern load.

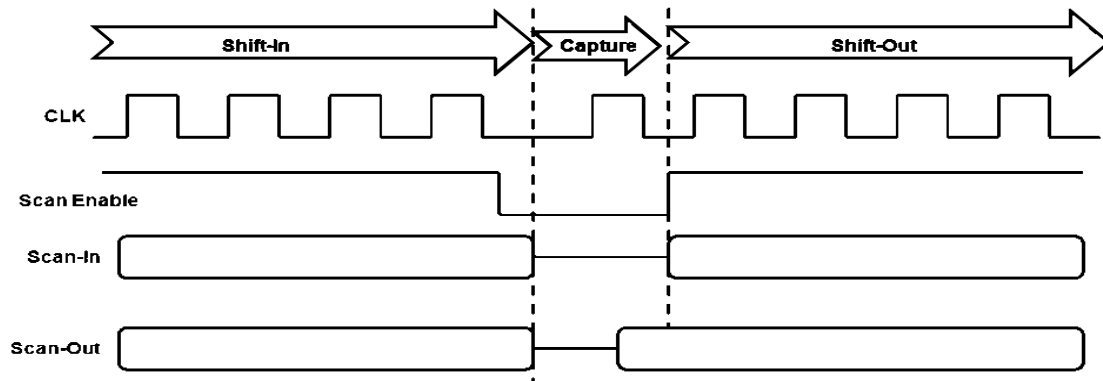


Figure 4. Scan Operation

## 2.4. Fault

Defect arises when the implemented hardware behaviour varies from the desired/required behaviour (design specification). Manifestation of defect at abstracted functional level is termed as fault[6]. Source of fault can be improper masking during fabrication, missing contact window, surface impurities, parasitic formation. Faults are classified as stuck at fault, bridging fault, transition fault, path delay fault, transistor fault. Stuck at fault occurs when a line is stuck at logic '0' or logic '1' implying if a line is permanently connected to ground or permanently connected to power supply. There can be single or multiple stuck at fault. Methodology of representing fault abstractly is called fault model. While designing a fault model to reduce complexity, it is assumed only one fault occurs at a time.

## 2.5. Coverage

All possible defects cannot be detected by single fault model. Effectiveness of any fault model is measured in terms of coverage[5]. Coverage is a quantitative quality formulated as

$$\text{Fault coverage} = \frac{\text{Total number of faults detected}}{\text{Total number of faults}}$$

## 3. INTERNAL SCAN

In this technique[7], a long chain of scan cells is formed such that the design is divided into partially or fully isolated combinational blocks. A design can have more than one scan chain comprising thousands of scan cells. All chains scan enabled are connected, thus all scan chains are in scan mode or normal mode at the same time. This method has simplified the pattern generation problem and enabled serial shift property in sequential elements other than their normal function. This test mode is created along with compressed scan insertion automatically. Refer to figure 5, left side arrangement containing two scan chains refers to internal scan chains.

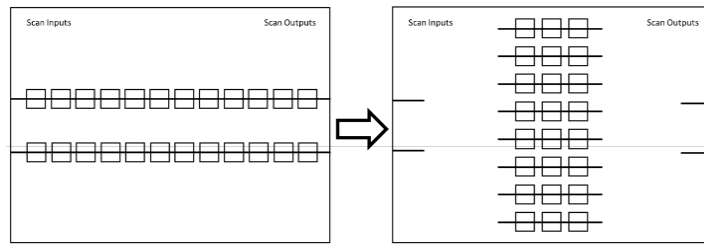


Figure 5. Example of internal scan chain(left) and compressed scan chain(right)

#### 4. SCAN COMPRESSION

In scan compression[7], the long scan chain is divided into smaller scan chains without altering the number of scan in and scan out ports. To do so, a compressor and decompressor are used. Compressor and Decompressor[8] can have different logic to perform compression and decompression action like by applying XOR[9] on smaller chain scan to get final output from compressor. As more elements are added to circuit (codec), there is a little variation in the number of faults of internal scan and scan compression mode. Figure 5 shows that the 2-scan chain arrangement(left) is internally broken into smaller chains(right) but at the chip level the scan-in pins will be same as in internal scan. This methodology reduces the simulation time significantly as the number of scan cells per chain decreases, number of clocks required in shift cycle reduces and all scan chains are acting in parallel. Amount of compression is calculated as

$$\text{Compression ratio} = \frac{\text{Number of internal scan chains}}{\text{Number of scan channels}}$$

Refer to figure 6, compression ratio is 5:2 as the number of internal scan chains (chain within decompressor and compressor) is 5 and 2 scan channels that are feeding the decompressor.

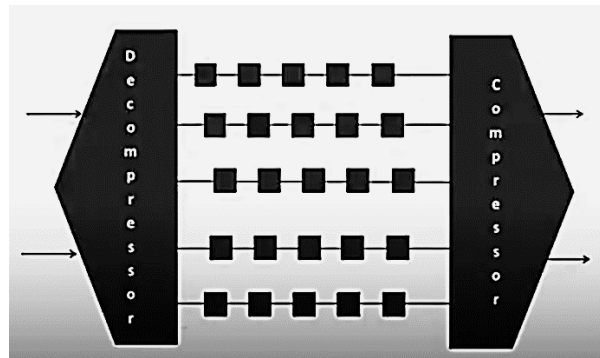


Figure 6. Scan Compression

#### 5. EXPERIMENTAL SETUP AND RESULT

This experiment is performed on 5 benchmark circuits consisting of three ISCAS’89 circuit, one ITC’99 and one FARADAY circuit namely s38584, s38417, s35932, B19, RISC respectively. Refer table 1 for basic information about benchmark circuits used. The Netlist/RTL of mentioned circuits are first converted to DFT inserted netlist using Synopsys –Design Compiler. Synopsys – TetraMAX (Version - p-2019.03-sp5)[10] tool is used to generate test patterns. This tool takes a test protocol file and DFT inserted netlist as input and produces test patterns in stil format. Using

Tetramax utility stil2verilog the test pattern in Verilog format is generated which is used for simulation. Only Static faults are targeted in this experiment. Cadence -Simvision tool is employed (Version Xcelium 19.09.009) for simulation. Table 2 shows the comparison between the methodology in terms of number of scan cells per chain, number of faults detected, simulation time and coverage percentage. As the number of scan cells are less in scan compression, so the number of scan chains are more. Number of flip flops in design (column 2 of Table 1) is greater than the number of scan cells in the internal chain (column 4 of Table 2) as during optimization certain flip are depicted as memory elements and those flops are not part of scan insertion. To test memory flops, usually built-in self-test circuits are embedded in the design.

Table 1. Description of Benchmark circuit

Benchmark Circuit	Number of Flip Flop	Number of inverters	Number of gates
s38584	1452	7805	11448
s38417	1636	13470	8709
s35932	1728	3861	12204
B19	6592	20866	98726
RISC	7599	7370	44872

Table 2. Comparison Table

Benchmark Circuit	Simulation Technique	Scan cells per chain	Number of chains	Fault detected	Coverage (%)	Simulation time(ns)
s38584	Internal Scan	915	1	49529	87.48	2223752
	Scan Compression	14 or 15	79	56810	94.39	84025
s38417	Internal Scan	1425	1	52205	85.21	3210775
	Scan Compression	14 or 15	117	58591	95.64	148025
s35932	Internal Scan	1380	1	53128	80.07	1105625
	Scan Compression	14 or 15	107	60141	90.64	16525
B19	Internal Scan	6142	1	600743	98.91	86476825
	Scan Compression	79 or 80	79	604380	99.51	187025
RISC	Internal Scan	7341	1	328315	97.01	96009750
	Scan Compression	109 or 110	72	330960	97.80	3249025

## 6. CONCLUSIONS

As the technology is shrinking, testability is getting complex. By this comparative analysis in terms of coverage, time, and fault, it is evident that scan compression has the upper hand over internal scan. Scan compression not only reduces test time significantly but also yields better coverage. For s38584, simulation time for internal scan is reduced to 96.2% in scan compression simulation technique. Similarly, reduction in simulation time for s38417, s35932, B19, RISC is 95.4%, 98%, 99.7%, 97% respectively. Though the coverage increase is a little around 9% for

small circuits s38584, s38417, s35932 and around 1% for large circuit B19 and RISC. Above results clearly show the difference in two simulation techniques. When simulation performed on different circuits having different numbers of sequential elements, it is evident that as the number of flip flops increases the simulation time required for internal scan technique is significantly longer than scan compression. Nowadays, even the smallest industrial design contains more than 20,000 flip flops, scan compression gives better coverage in less time and as chains are smaller, debugging is easier.

Figure 7 shows area variation with respect to number of components like AND gate, flip flop, inverter in design. Blue bar is for normal scan insertion and orange bar reflects cells after scan compression architecture is embedded. Using Scan compression is very time efficient. Though area overhead increases nearly by 10-20% after scan compression, 12.2% increase in RISC area and 13.6% increase in B19 area.

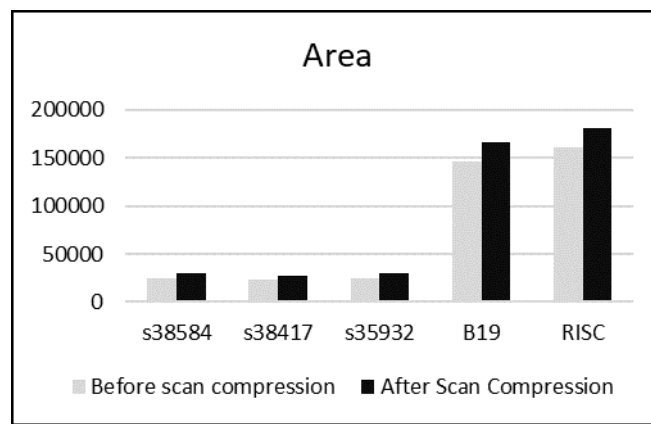


Figure 7. Graph showing area variation.

## 7. FUTURE SCOPE

From the experiment it is clear that scan compression yields better coverage in lesser time, but the area overhead is considerable. In future, work can be done in reducing the area of decoder logic and reducing pattern count keeping into account achieved coverage and time.

## REFERENCES

- [1] J. Chauhan, C. Panchal, and H. Suthar, "Scan Methodology and ATPG DFT Techniques at Lower Technology Node," in 2017 International Conference on Computing Method- ologies and Communication (ICCMC), pp. 508–514, IEEE, 2017
- [2] P. Srinivasan and R. Farrell, "Hierarchical DFT with Combinational Scan Compression, Partition Chain and RPCT," 2010 IEEE Computer Society Annual Symposium on VLSI, 2010, pp. 52-57, doi: 10.1109/ISVLSI.2010.59.
- [3] Laung-Terng Wang, Cheng-Wen Wu, and Xiaoqing Wen., "VLSI Test Principles and Architectures: Design for Testability", Academic Press, 2006
- [4] <https://technobyte.org/internal-scan-chain-structured-dft-techniques/>
- [5] Hur, Yongmin & Choe, Youngcheol. (2014). LOS/LOC Scan Test Techniques for Detection of Delay Faults. The Journal of the Institute of Webcasting, Internet and Telecommunication. 14. 219-225. 10.7236/JIIBC.2014.14.4.219.
- [6] N Burgess, RI Damper, SJ Shaw, and DRJ Wilkins, "Faults and fault effects in nmos circuits-impact on design for testability," in IEE Proceedings GElectronic Circuits and Systems, volume 132, pages 82–89. IET, 1.

- [7] M. L. Bushnell and V. D. Agrawal, "Essentials of Electronic Testing for Digital, Memory and Mixed-Signal VLSI Circuits," *Frontiers in Electronic Testing*, 2002, doi: 10.1007/b117406.
- [8] Synopsys – TestMAX DFT User Guide, Version -P 2019.03-SP4, September 2019
- [9] P. Sakrappanavar, S. Yellampalli and A. Kothari, "Comparative analysis of scan compression techniques," 2014 International Conference on Electronics, Communication and Computational Engineering (ICECCE), 2014, pp. 40-44, doi: 10.1109/ICECCE.2014.7086632.
- [10] Y. Liu et al., "X-Tolerant Compactor maXpress for In-System Test Applications With Observation Scan," in *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, vol. 29, no. 8, pp. 1553-1566, Aug. 2021, doi: 10.1109/TVLSI.2021.3092421.
- [11] Synopsys – TestMAX ATPG and TestMAX Diagnosis User Guide, Version Q-2019.12, December 2019

## AUTHORS

**Parul Patel** is an Intern in STMicroelectronics, India and pursuing M. Tech in VLSI Design and Embedded Systems from MANIT Bhopal, India. She is part of DFT team and deals with scan insertion and boundary scan in design.

