International Journal on Recent and Innovation Trends in Computing and Communication
ISSN: 2321-8169 Volume: 11 Issue: 9
***Article Received:** 25 July 2023 **Revised:** 12 September 2023 **Accepted:** 30 September 2023*

A Total Self Checking Comparator Implementable on FPGAS Using Bist Technology

[1] **Mounika Posham,**

M.Tech (Vlsi) In Dept of Ece in Anurag University, Venkatapur, Ghatkesar Rd, Hyderabad, Telangana 500088.

[2] **Amrita Sajja,**

M.Tech, (Ph.D.), In Dept of Ece in Anurag University, Venkatapur, Ghatkesar Rd, Hyderabad, Telangana 500088.

Abstract— an integrated circuits (IC) "manufacturing tests" may be made easier to administer with the use of design for testability (DFT). Integrated circuits' embedded memory tests make use of the TSC (TSC) approach. We have shown the TSC method and several algorithms used in TSC for the purpose of testing embedded memory in this article. An address generator, controller, comparator, and memory are the four main components of this kind of memory TSC technology. This paper details the three memory TSC controller implementation techniques. The memory TSC controller is modelled in Verilog HDL, and its accuracy is checked using the RTL compiler before synthesis.

Here we provide a way to build TSC comparators for TSC systems that may be implemented on FPGAs—totally self-checking (TSC) systems—that can be used online. By directly measuring the output of each lookup table (LUT), this approach may be utilised to do comprehensive online diagnostics of all LUTs. This entails mapping the basic components of the comparator with a limited number of test patterns. With our technique, we can achieve exhaustive diagnosis with a small number of test patterns on the order of n [O(n)] (where n is the input number to the comparator) while yet covering all bases 100% of the time, even if we are just aware of the LUT's specs and not its exact structure. For systems that need absolute reliability, FPGAs will be a perfect fit. Our experiment also included a single-event upset (SEU) induced by neutron radiation to validate the soft error rate (SER) in a field-programmable gate array (FPGA) based on static random-access memory (SRAM).

INTRODUCTION

More and more complicated integrated circuits (ICs) are being manufactured thanks to the rapid growth of microelectronic technology. Despite the many advantages (more performance, lower cost per function, etc.), this presents significant test and reliability issues [1]-[7]. When high-performance microprocessors are tested at high speeds, the induced IC activity factor (AF) by the test vectors is much greater than when the processors are operated in the field. As a result, the transition times of the circuit under test (CUT) signals may be slowed down due to the generation of excessive power droop (PD). This occurrence is probably going to be mistakenly attributed to mistakes in the delay process.

Consequently, a false negative result will be produced, leading to a rise in the yield loss. Depending on whether the CUT is a sequential one with scan or a combinational circuit, logic TSC (LTSC) may be used to do at-speed testing of logic blocks. LTSCs can be either combinational or scan-based.

An extra phase, a burst phase, is inserted between each shift and capture phase to solve in. An increase in power supply current up to a level comparable to what the CUT absorbs during capture phases is the goal of such a burst phase. For this reason, the inductive part of PD occurs during the burst phase and then disappears before the capture phase. The overall amount of power used during testing and TT is increased by this solution. Omaña et al. put up new methods for the LOS scheme to lessen PD during scan-based LTSC. By enhancing the correlation between neighbouring bits of the scan chains, they make it possible to reduce PD by as much as 87% in and 50% in. Nevertheless, these methods fail to lessen PD in scan-based LTSC that utilises the LOC scheme because they do not enhance the correlation between test vectors that are applied at subsequent capture cycles. To lessen the likelihood of producing false test failures during testing, this study proposes a new, scalable method to decrease PD during the capture stages of scan-based LTSC. By appropriately adjusting the test vectors produced by the Linear Feedback Shift Register (LFSR), our method, which is similar to the methods in, decreases the AF of the CUT in comparison to traditional scan-based LTSC. Although our goal is not to enhance FC (as is often the case with reseeding) but to decrease PD, our method is comparable to reseeding strategies (such as those in) in that it involves appropriately modifying the sequence of test vectors to meet a certain criterion. Our method's foundational principle was first presented in, however in a non-scalable form.

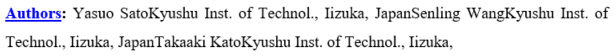
We suggest a scalable method that involves replacing one or more test vectors, called replacement test (ST) vectors, that would normally be applied to the CUT in accordance with traditional scan-based LTSC with new, appropriate ones. In order to decrease the maximum number of transitions between any two subsequent test vectors, the ST vector(s) are produced using the test vectors that will be applied at previous and future capture phases. By comparing this to the first test sequence, we can see that the CUT AF and PD are decreased. In order to lessen the correlation between the test vectors applied to neighbouring scan-chains, a phase shifter (PS) is often used in scan-based LTSC; we take this into account. In most cases, the test vectors that need to be applied to scan-chains at both the beginning and end of the capture process are either provided by the PS itself or may be simply adjusted to do so. Our method takes use of this quality to make its implementation on inexpensive hardware possible. If the scan-based LTSC doesn't have a PS or if the PS doesn't provide the past and future test vectors for all scan-chains, our technique may still be used. It is indeed possible to retrieve the past and future scan-chain test vectors as a linear combination of correct LFSR outputs, as shown in Section IV. With our method, the amount of PD that can be reduced is scalable.

LITERARTURE SURVEY

Low Power BIST for Scan-Shift and Capture Power

Abstract:

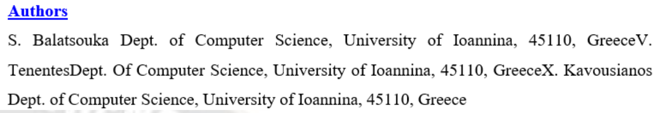
In order to ensure efficient and accurate testing, low-power test technology has been extensively studied. The unpredictable unpredictability of logic BIST is the reason why there aren't as many complex techniques offered for it as there are for scan-test. But logic BIST is become very important for system debugging and field testing right now. In order to decrease shift-power and capture-power, this research suggests a new low-power BIST technique that removes the required high-frequency components of vectors. In addition to lowering test power, the authors demonstrate that the suggested approach maintains test coverage with little loss.


Authors: Yasuo SatoKyushu Inst. of Technol., Iizuka, JapanSenling WangKyushu Inst. of Technol., Iizuka, JapanTakaaki KatoKyushu Inst. of Technol., Iizuka,

Defect Aware X-Filling for Low-Power Scan Testing

Abstract:

For scan testing, there are a number of X-filling approaches that aim to lower the shift and/or capture power. The biggest problem with these approaches is that X-filling for low power results in less defect coverage compared to random-fill. Our proposed scan testing technique is a unified low-power, defect-aware X-filling approach. The power reduction achieved by the suggested technique is equivalent to that of the Fill-Adjacent X-filling method, and it does so while imposing limitations on the peak power during response capture. While reducing the pattern count, this method achieves great defect coverage, which is on par with or even exceeds that of random-fill in many instances. Results from simulations of the biggest ISCAS and IWLS benchmark circuits show the benefits of the suggested approach.

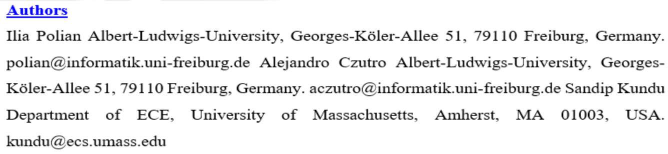

Authors

S. Balatsouka Dept. of Computer Science, University of Ioannina, 45110, GreeceV. TenentesDept. Of Computer Science, University of Ioannina, 45110, GreeceX. Kavousianos Dept. of Computer Science, University of Ioannina, 45110, Greece

Power Droop Testing

Abstract:

The input patterns determine the circuit activity. The voltage supplied by the power source might decrease or increase suddenly in response to variations in circuit activity. Power droop describes this shift, which is a kind of power supply noise. Power droop is a potential source of IC failure, but traditional fault models do not yet account for it, hence it cannot be tested for during testing. We provide a method for detecting these kinds of errors in this study. By adding the high-frequency and low-frequency effects, we may design test sequences that produce the worst-case power decrease. This heuristic technique is proposed here. Even for scan designs, the produced patterns must be consecutive. In order to construct tests, we use a variation of the conventional D-algorithm that is dynamically restricted; that is, the algorithm creates new constraints in real time based on past assignments. Manufacturing testing and early silicon validation may both benefit from the acquired patterns. We create test sequences for ISCAS circuits and develop a prototype ATPG to show that the technique is feasible.


Authors

Ilia Polian Albert-Ludwigs-University, Georges-Köler-Allee 51, 79110 Freiburg, Germany. polian@informatik.uni-freiburg.de Alejandro Czutro Albert-Ludwigs-University, Georges-Köler-Allee 51, 79110 Freiburg, Germany. aczutro@informatik.uni-freiburg.de Sandip Kundu Department of ECE, University of Massachusetts, Amherst, MA 01003, USA. kundu@ecs.umass.edu

A Layout-Aware X-Filling Approach for Dynamic Power Supply Noise Reduction in At-Speed Scan Testing

Abstract:

A new and significant resilience problem in nano-scale a CMOS is power supply noise (PSN). Additional delays may occur if the real supply voltage perceived by individual gates within the circuit is lower compared to the nominal voltage of the supply, which may happen when several gates are switched on at once. Test invalidation owing to excessive PSN might effect yield loss in at-speed scanning testing since the simultaneous switchings are greater than in functional mode. By assigning suitable values to X-bits in partly described test patterns, we provide a Linear Programming-based X-filling method to reduce PSN in at-speed scan tests in this study. This paper addresses the issue of inaccurate dynamic PSN estimation by considering spatial or transition time correlations caused by the circuit layout, power mesh, and netlist. Additionally, it is the first time that the circuit delay is specifically targeted to reduce the impact of PSN during the at-speed scan test.

EXISTING METHOD

TSC COMPARATOR IMPLEMENTABLE ON FPGA:

Tasks Connected to TSC and TSC Comparator When it comes to safety-related applications in particular, TSC is crucial for dependable systems. Next, we will define TSC. A circuit is considered self-testing if, for each error from a specified set, it generates an output that is not core-related for a code input. A circuit is considered fault secure according to definition 2 if it never generates the wrong code output for code inputs, even when faced with a specific set of faults. Thirdly, a circuit is considered TSC if it can detect and repair its own faults.

There are two ways to create TSC LSIs that meet the requirements laid forth in the definitions: (a) utilising ad hoc design for TSC to build the whole circuit, or (b) making copies of the functional blocks (b-1) and checking the output of these copies (b-2) using a TSC data-compare mechanism (b-3). The problem with option (a) is that it requires a completely new design and implementation of all the circuits in an ad hoc manner with severe design constraints. However, if the data-compare mechanism is designed with an ad hoc fail-safe in mind, then technique (b) may be utilised to allow the construction of TSC logic. Put another way, a regular functional block design can be readily applied to a duplicated functional block. Consequently, there will be a significant decrease in development time and expense. Method (b)'s fault-detection coverage is highly dependent on the TSC data-compare mechanism's coverage (Fig.).

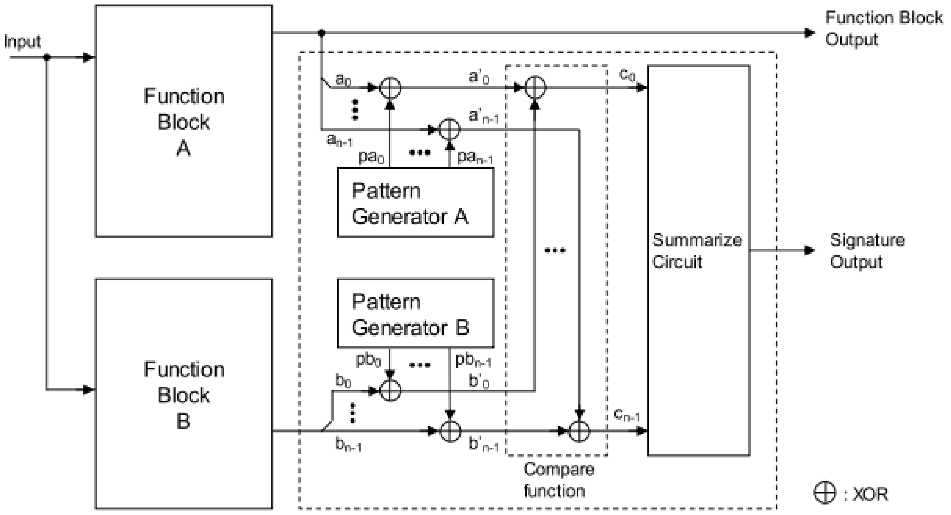


Fig: 1. TSC LSI architecture for approach 2. Two identical function blocks, verification of output from those function blocks, and diagnostics function in comparator (TSC comparator) are included.

Architectural framework for TSC LSI in Approach 2. Included are two similar function blocks, a diagnostics comparator (TSC comparator), and a means to verify the output of those blocks. (b-3) inside the contextualizer. Since certain functions are either not used or irrelevant to the output, it is not always possible to provide 100% coverage for identifying defects in each function block. However, using conventional logic-diagnosis methods, we can detect all faults related to function blocks A and B, which are the most vital to the system, by copying function blocks (b-1) and confirming the results of both blocks in the comparator (b-2). This ensures that we cover all bases when it comes to fault detection. The TSC function for a comparator (b-3) is the last TSC requirement for method (b). To describe a comparator with this capability, we use the acronym TSC. Figure shows one possible layout for a TSC LSI. This TSC comparator compares two n-bit inputs (a0-an−1 and b0-bn−1) from functional blocks A and B of the dual-modular redundancy (DMR) system. With this comparator, you can test for errors in real time. A diagnostic circuit, consisting of a test-pattern generating and injector circuit, usually an exclusive OR (XOR) circuit, is added to provide the TSC function.

DESIGN METHOD OF TSC COMPARATORS:

Works with field-programmable gate arrays while (b-1) and (b-2) are readily accessible in an FPGA using traditional design approaches, (b-3) is inadequate. The following are some of the problems with using comparators in FPGAs, which relate to (b-3). 1) In order to identify errors in a particular failure mode caused by the programmable logic (PL) of an FPGA, when a comparator is connected to LUTs, more test patterns are needed. 2) Because of the SEU in config memory (CRAM), the likelihood of a wire short in FPGAs is greater than in ASICs. Because the approach used in an earlier research is limited to "stack-at faults," it is possible that some errors would go unnoticed. Reports have surfaced about fault models that map circuits to LUTs in FPGAs. A 2k-bit SRAM cell is the basic building block of a k-input LUT. With the 2k test patterns and careful observation of the LUT output, we can diagnose a k-input LUT.

Because of the great degree of implementation flexibility that LUT-based circuits have owing to the programming of LUTs, short circuits of wire between two nets additionally are not inconsequential. Under the black-box implementation of the circuit's function, the total amount of test patterns required for diagnosing an n-input circuit is usually 2n. A deluge of test patterns is produced by the naive application of a TSC comparison on an FPGA. By keeping an eye on the results of each LUT's comparator function, the suggested strategy may cut down on the amount of test designs without sacrificing coverage.

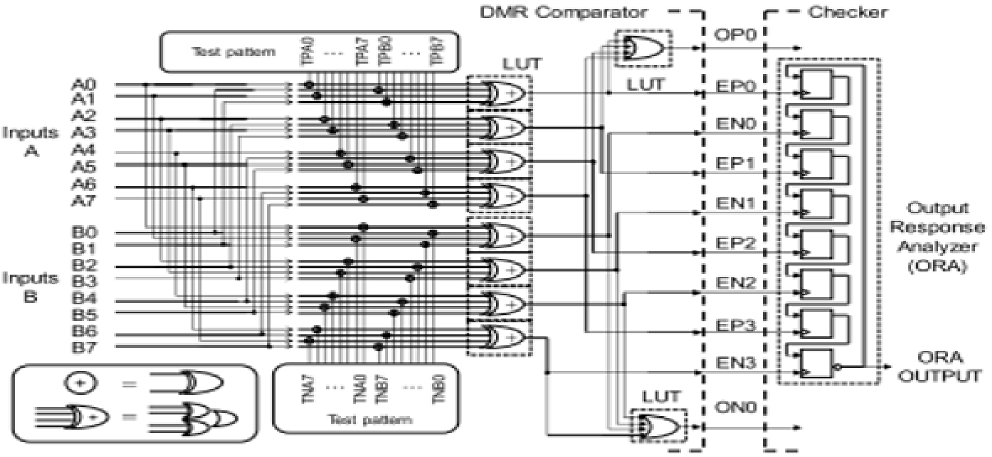


Fig: 2. Example schematic of dual-modular-redundant TSC comparator designed with the proposed method.

Here we see a TSC comparator with 8-bit inputs & a checker that was built using the suggested DMR technique. The XOR circuit test-pattern injector is coupled to two 8-bit inputs, A0-A7 and B0-B7. At the positive pole of another clock edge, one comparator is diagnosed (the upper one in Fig. 2), and at the negative clock edge, the other (the redundant bottom one in the picture) is diagnosed. The top comparator is linked to test patterns TPA0-TPA7 and TPB0-TPB7, while the lower comparator is linked to test patterns TNA0-TNA7 and TNB0-TNB7. In the checker, each flip-flop (FF) has a matching clock terminal, and each LUT has its own set of inputs for the comparators. We refer to this circuit as an output response analyzer (ORA) since it processes the comparator's signature signals. Injection of the test pattern causes a sequential change from "low" to "high" on outputs EP0-EP3 and EN0-EN3.

Consequently, the input of these outputs causes the FFs in the checker to propagate a "high" signal. Figure 3 shows a waveform sample of a comparator operation during a fault injection procedure. A failure signal is injected into each input at the correct moment according to a predefined test pattern. On the positive side of the clock signal, test patterns TPB and TPA are introduced. Figure 3 shows the injection of test patterns TNA and TNB at the negative edge. The checker's FF then uses the inputs of TPB, TNA, so TNB signals to propagate the signal, and the result is ORA

OUTPUT. Eight clock cycles make up the ORA OUTPUT period. Using the CHK_CNT counter, we recorded this duration for a subsequent experiment. The higher comparator's output is denoted as OP0, whereas the redundant comparator's output is ON0.

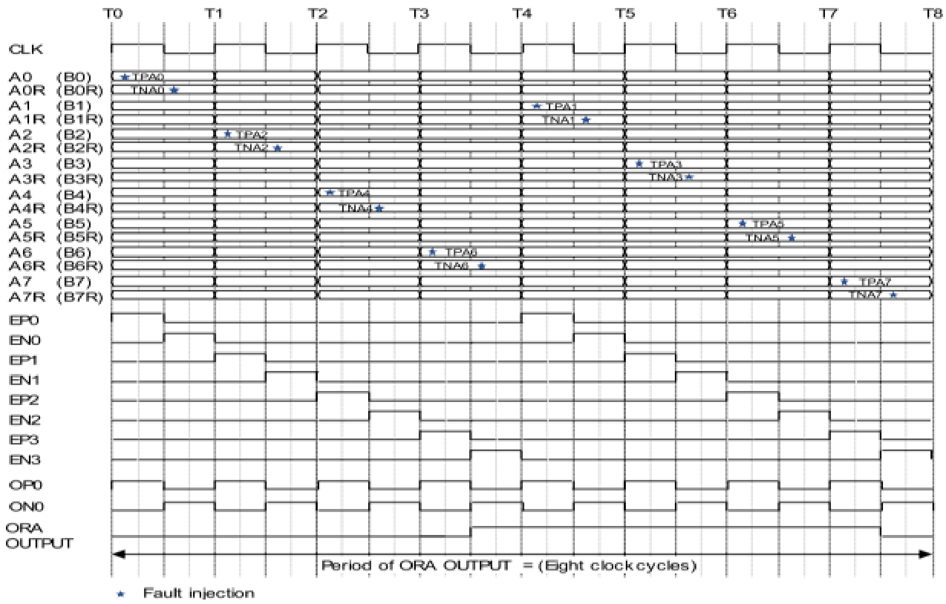


Fig: 3. Example waveforms of TSC comparator. It shows 16 out of 160 patterns. For B0–B7, the corresponding test pattern is replaced with TPB# and TNB#.

PROPOSED METHOD

Random pattern resistant (RPR) defects are the ones that go unnoticed in logic TSC. Typically, these random patterns are stopped when there is no more increase in fault coverage, however the exact moment is unknown. In contrast to traditional testing, which might require an additional millisecond or two to activate and likely wouldn't provide sufficient test coverage without more test vectors, LTSC necessitates more test patterns (wang, 2006). The bulk of users often combine LTSC and ATPG powered tests for industrial testing. When ATPG testing is underway, LTSC is disabled and handled as functional logic. As an alternative to simultaneously running ATPG & TSC, this idea proposes using a collection of ATPG patterns chosen at random for pseudo-random testing. As a result, it achieves LTSC speeds while producing ATPG-level fault coverage. The safety of testing buildings is another driving force behind this effort.

Commonly used for testing and debugging chips, boards, and systems, JTAG is IEEE 1149.1 compliant. To keep JTAG safe when testing and debugging, many techniques are being considered. What follows is an explanation of a few of the earlier security measures. To the most effective of our knowledge, no protective methods have been developed or used to logic TSC as of yet, according to the literature study. The results of the literature review on reducing test data volume indicate the need to investigate potential pattern reduction strategies in order to provide fault coverage comparable to ATPG-based testing while also avoiding power and area overhead. It is required to examine the potential for TSC structural changes, assessment of relevant parameters, and adjustments to testing structures. Research has the potential to provide great results, and there is plenty room for improvement. In order to try to shed light on some of the ambiguities surrounding structurally modified deterministic TSC, this study was motivated by survey data and industry-based white papers. Using ATPG patterns while evaluating structure security, we reduce the test vectors for logic TSCs so that hackers cannot access the chip's hardware information.

There is a little allowable size overhead for logic TSC test structures since they are incorporated into the IC with the design unit (Wang, 1988). A few hundred deterministic tests in ATPG are sufficient for fault coverage, but even for a fairly small design, many thousands of patterns must be created in the LTSC due to the pseudo-random character of the vectors. Consequently, LTSC often requires somewhat longer test sequences and fault coverage is significantly lower than 100%.

Article Received: *25 July 2023* ***Revised:*** *12 September 2023* ***Accepted:*** *30 September 2023*

A reconfigurable LTSC may assist in rearranging various modules, such as the pattern generator, ROM memory, MISR, and others, to meet the requirements of the DUT scan and the test. At runtime, you may change all the settings, including the size and breadth of the registers, as well as their starting and anticipated values and the number of times it takes to execute. Figure depicts a typical layout for an LTSC.

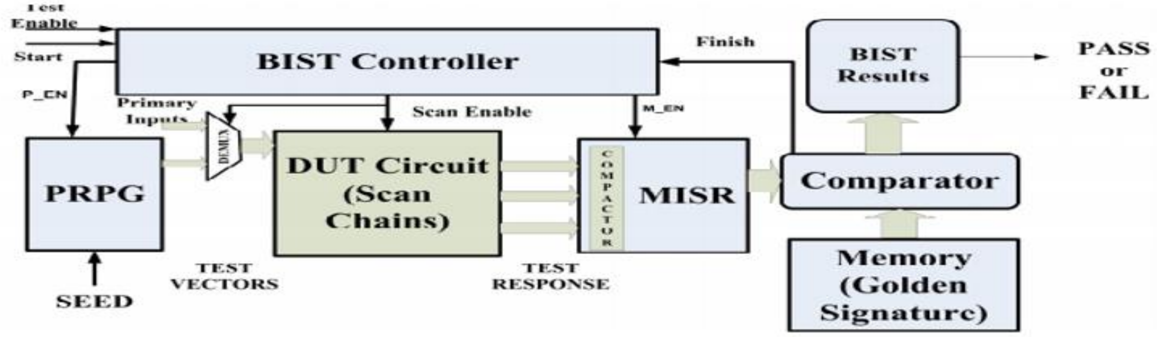


Fig: 4. Logic TSC Architecture.

The exhaustive test sequence created by LTSC is compressed using the seed selection technique. Because to its high compression ratio, fewer test vectors are applied to the DUT, which in turn minimises the power and time needed for test application. A linear feedback shift register (LFSR) generates a pseudorandom sequence of bits, which is used in BitFixing in Pseudorandom Sequences to include deterministic test cubes that identify random-pattern resistant flaws. According to Agarwal (2002), the STUMPS design is extensively utilised in practice and is incorporated in the suggested solution.

The STUMPS design calls for independent LFSRs to provide the circuit's main and scan inputs, allowing for finer-grained control over the shift & capture cycles. The design's scan inputs and main inputs are fed into a parallel LFSR structure, which reduces the TSC area relative to a serial LFSR structure, which is known as the Pseudo Random Pattern Generator (PRPG). All of the suggestions are tested on the ISCAS'89 and ISCAS'99 benchmark designs so that the outcomes may be compared and analysed using the published methodologies. The seed selection approach has reduced the size of the comprehensive test vectors by a factor of 13 (s400) to 2 (s349), even for the modest sized ISCAS benchmark designs, according to the experimental data. By introducing the measure of test cycles per net (TCPN(COV)) for a certain coverage (COV), we may compare the test structure's efficacy. This is the result of dividing the total number pf test cycles by the total quantity of nets (NETS) in the provided net list.

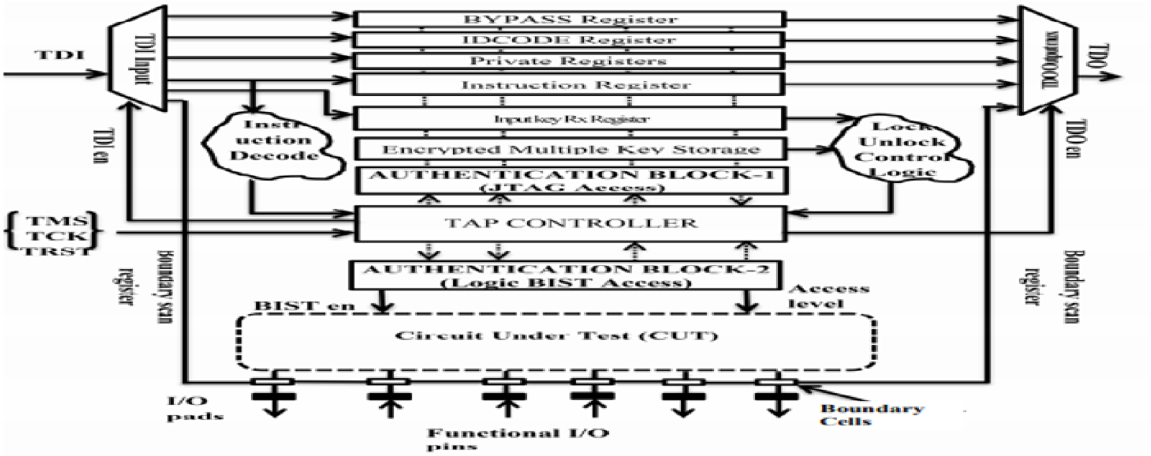


Fig: 5. JTAG structure with dual stage security blocks.

A novel method has been presented for translation predictable motifs to a pseudo-random test sequence, which allows for scalable pattern mapping. Based on the condensed LFSR principle, this method suggests that the deterministic patterns are a subset of the exhaustive test set and have a definite character. Since this technique does not need the addition of

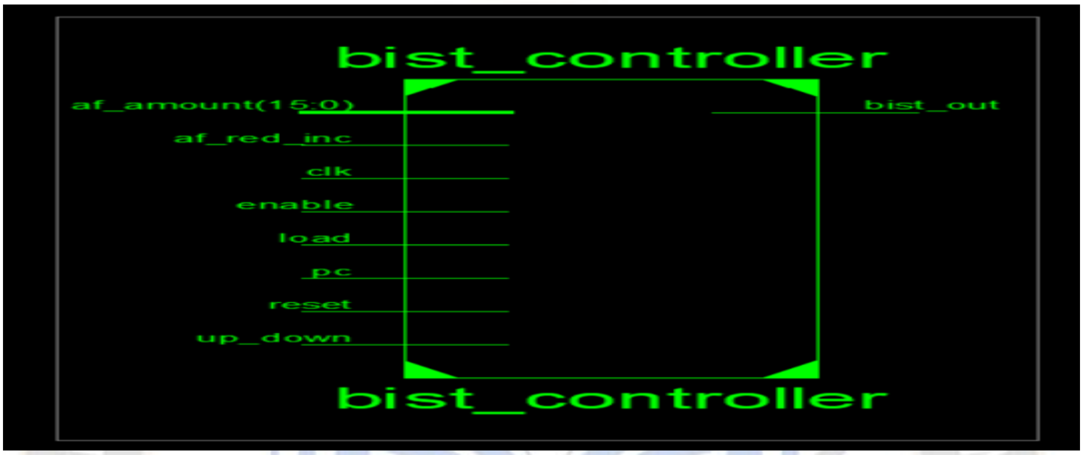
***Article Received:** 25 July 2023 **Revised:** 12 September 2023 **Accepted:** 30 September 2023*

circuitry, it may be used with designs that already have LTSC installed.

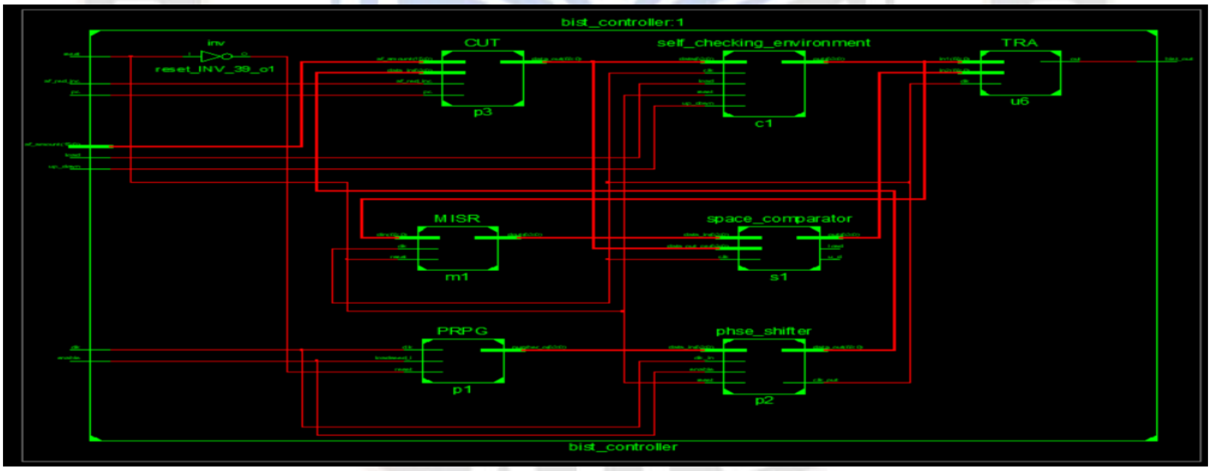
- Method for selecting the shortest test sequence using seeds: A new method for choosing a specific seed (LFSR beginning value) has been developed. In order to compare and analyse, this technique makes use of the whole comprehensive pattern sets of both the TSC and the ATPG. This algorithm produces the ATPG pattern-inclusive shortest random pattern test set as its output. Our evaluation and analysis focus on the fault cover of stopped at fault (modelled flaws) using pseudo-random sequences.

SIMULATION RESULTS

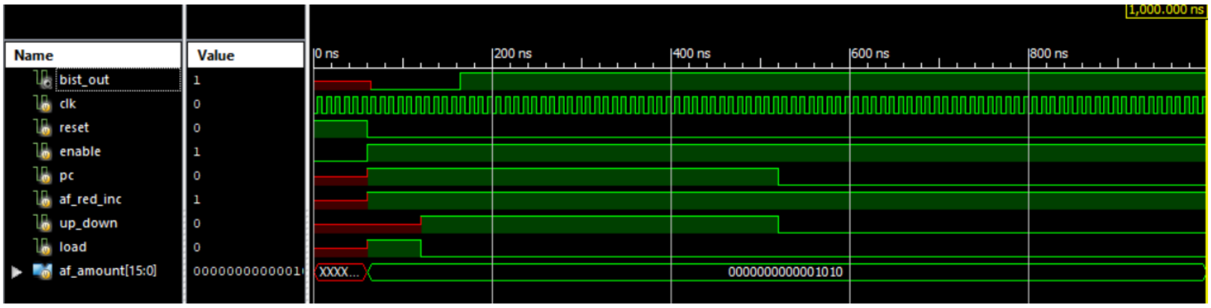
RTL



INTERNAL RTL



SIMULATION RESULTS



Name	Value
bist_out	1
clk	0
reset	0
enable	1
pc	0
af_red_inc	1
up_down	0
load	0
af_amount[15:0]	00000000000010

***Article Received:** 25 July 2023 **Revised:** 12 September 2023 **Accepted:** 30 September 2023*

COMPRESSION TABLE

EXISTING	PROPOSED
AREA: 113 LUTS	10,567 LUTS
DELAY : 9.553ns	0.682ns

CONCLUSION

Through a neutron radiation exposure experiment, we assessed the SER per comparative example and presented a design methodology for TSC comparators that can be implemented on FPGAs using an online fault injection mechanism. The suggested approach was used to do self-diagnostic with 100% coverage in a limited number of test patterns by directly examining the output of each LUT. This significantly decreased the number of test patterns to O(n). We were able to create a TSC system using the suggested TSC comparator and the suggested approach, which could be implemented on an FPGA with a fair amount of test patterns, even though we were unaware of the actual architecture of the circuits in an FPGA. A software-based TSC implementation on FPGAs was made possible by expanding our technique. We tested the efficacy against neutron radiation of a DUT consisting of 1575 DMR comparators to assess the SER of this kind of TSC comparator in an FPGA. A comparator's assessed SER (at NYC sea level) was 0.055 FIT. So, we proved that it is possible to use our method and FPGAs in conjunction with an appropriate design methodology outlined in a functional safety standard, like IEC 61508, to create reliable systems, particularly for safety-critical applications like vehicles, railway systems, chemical plants, or avionics.

FUTURE SCOPE: Using Vertibi Decoder and ASIC Application in High Speed Communication Systems, We Can Take This Project to the Next Level.

REFERENCES

[1] J. Rajski, J. Tyszer, G. Mrugalski, and B. Nadeau-Dostie, “Test generator with preselected toggling for low power TSC,” in Proc. Eur. Test Symp., May 2012, pp. 1–6.

[2] Y. Sato, S. Wang, T. Kato, K. Miyase, and S. Kajihara, “Low power TSC for scan-shift and capture power,” in Proc. IEEE 21st Asian Test Symp., Nov. 2012, pp. 173–178.

[3] E. K. Moghaddam, J. Rajski, M. Kassab, and S. M. Reddy, “At-speed scan test with low switching activity,” in Proc. IEEE VLSI Test Symp., Apr. 2010, pp. 177–182.

[4] S. Balatsouka, V. Tenentes, X. Kavousianos, and K. Chakrabarty, “Defect aware X-filling for low-power scan testing,” in Proc. Design, Autom. Test Eur. Conf. Exhibit., Mar. 2010, pp. 873–878.

[5] I. Polian, A. Czutro, S. Kundu, and B. Becker, “Power droop testing,” IEEE Design Test Comput., vol. 24, no. 3, pp. 276–284, May/Jun. 2007.

[6] X. Wen et al., “On pinpoint capture power management in at-speed scan test generation,” in Proc. IEEE Int. Test Conf., Nov. 2012, pp. 1–10.

[7] S. Kiamehr, F. Firouzi, and M. B. Tahoori, “A layout-aware X-filling approach for dynamic power supply noise reduction in at-speed scan testing,” in Proc. IEEE Eur. Test Symp., May 2013, pp. 1–6.

[8] M. Nourani, M. Tehranipoor, and N. Ahmed, “Low-transition test pattern generation for TSC-based applications,” IEEE Trans. Comput., vol. 57, no. 3, pp. 303–315, Mar. 2008.

[9] N. Z. Basturkmen, S. M. Reddy, and I. Pomeranz, “A low power pseudorandom TSC technique,” in Proc. 8th IEEE Int. On-Line Test. Workshop, Jul. 2002, pp. 140–144.

[10] J. Rajski, N. Tamarapalli, and J. Tyszer, “Automated synthesis of large phase shifters for TSC,” in Proc. Int. Test Conf., Oct. 1998, pp. 1047–1056.