



A LOW-POWER AND AREA-EFFICIENT DESIGN OF A WEIGHTED PSEUDORANDOM TEST-PATTERN GENERATOR FOR A TEST-PER-SCAN BUILT-IN-SELF-TEST ARCHITECTURE

Rubeena Begum¹, Farina Yasmeen²

¹PG Scholar, Department of ECE, Shadan Women's College of Engineering and Technology, Hyderabad, rubeenabegum0707@gmail.com

²Asst Professor, Department of ECE, Shadan Women's College of Engineering and Technology, farinays@gmail.com

ABSTRACT

Test Pattern Generator (TPG) is an essential component in Built-In Self-Test (BIST) architectures, responsible for generating pseudorandom test patterns that can be weighted to improve fault detection efficiency while reducing the dependency on deterministic random number generators. This paper proposes a novel weighted Test Pattern Generator (TPG) along with a deterministic random number generation technique for a scan-based BIST architecture. The primary objective of the proposed design is to generate efficient weighted and deterministic test patterns that enhance fault coverage while minimizing power consumption, hardware overhead, and testing time. The proposed approach maximizes the pseudo-primary seed of the TPG to generate longer weighted and deterministic pseudorandom sequences, thereby increasing the effectiveness of the testing process. Maximum-length weighted patterns are achieved by assigning independent weights to individual scan chains through a weight-enabled clock mechanism. This selective weighting technique improves the quality of test patterns, reduces unnecessary switching activity during scan operations, and significantly lowers dynamic power consumption. Furthermore, the proposed architecture minimizes area overhead by employing an optimized hardware implementation without compromising test performance. In addition, the novel pattern generator is implemented and evaluated on two different test-per-scan BIST architectures to validate its efficiency and reliability. The proposed method demonstrates improved fault coverage, reduced power dissipation, lower hardware complexity, and enhanced overall testing performance compared to conventional scan-based BIST techniques. Experimental simulations and functional verification are carried out using the Xilinx ISE design suite, and the obtained results confirm the effectiveness of the proposed architecture in achieving accurate, reliable, and cost-effective VLSI testing.

INTRODUCTION

Modern technology has focused on developing low-power systems for very-large-scale integration (VLSI) high-speed designs. As a result, several design strategies have been implemented to mitigate trade-offs between performance, power, and area. Instead, several approaches have concentrated on low-power dissipation during BIST normal-mode operations rather than test-mode operations. During the BIST test mode operation, the switching activity in the scan chains and test data compression using the appropriate TPG are crucial. Moreover, this testing should be achieved with high reliability and sensitivity in semiconductor designs. Figure 1 illustrates an example of a conventional

The associate editor coordinating the review of this manuscript and approving it for publication was Wu-Shiung Feng.

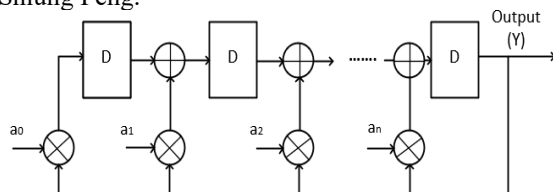


FIGURE 1.1. An example of a conventional pseudorandom TPG;

The TPG consists of the sequence of length n shift registers and input seed bits of $a_0, a_1, a_2, \dots, a_n$. Based on the n -th bit of the shift register, the $(i-1)$ th clock cycle

is updated continuously by the $(n-1)$ -th bit of the shift register and the i -th clock cycle. The TPGs use a high degree of parallelism to achieve high-yield specifications in many practical applications. The TPG linear function is accomplished according to the output feedback signal and the input seed bits. Its linear functionalities are used in many applications such as aircraft systems, cockpit systems, medical systems, audio and video systems, and power generation and distribution systems.

A TPG consists of deterministic, exhaustive, pseudorandom, pseudorandom-weighted, and mixed-mode outputs. The pseudorandom-weighted output is used to achieve higher fault coverage in many BIST structures. The weighted pseudorandom TPG exhibits true randomness and repeatable patterns in all clock cycles. Typically, it requires one seed bit to produce one test pattern for n cycles of the scanning phase in the test-per-scan BIST, where n is the scan chain length. The latest study decreased the switching activity during scan shift cycles. Additionally, the TPG allows the automatic selection of weighted parameters to achieve its low power. The weighted pseudorandom TPG methods, and their implementation can effectively reduce the switching transitions. However, the methods, included additional XOR transitions between the shift registers, it consumed more power and area. The concerned drawbacks are eliminated in the proposed design effectively. The BIST requirements should



mainly focus on the higher fault coverage and the lesser weighted switching activity with lower power and reduced area overhead. To achieve these requirements, two approaches can be utilized. One is to alter the circuit design of the weighted TPG. The other is to include additional hardware in the weighted TPG. Hence, in this paper, a new pseudorandom-weighted TPG is constructed using additional hardware. Additionally, higher fault coverage is achieved in terms of eliminating transition delay faults using test-point insertion. The test-points are inserted for every two NAND gate structures of the over- all design area. The proposed technique involves swapping weighted test patterns to the scan chains using a phase shifter.

The swapping of the weighted patterns considered for selecting the prior scan chains with lesser area is compared with that of the other scan chains. The weighted patterns are hence used with all the scan chains of BIST architecture. This eliminates the faults at a specified output and improves the fault coverages. The TPG also improves its rapid switching activity due to its selected weighted patterns and reduces its average scanning and capturing power consumption during BIST test-per-scan.

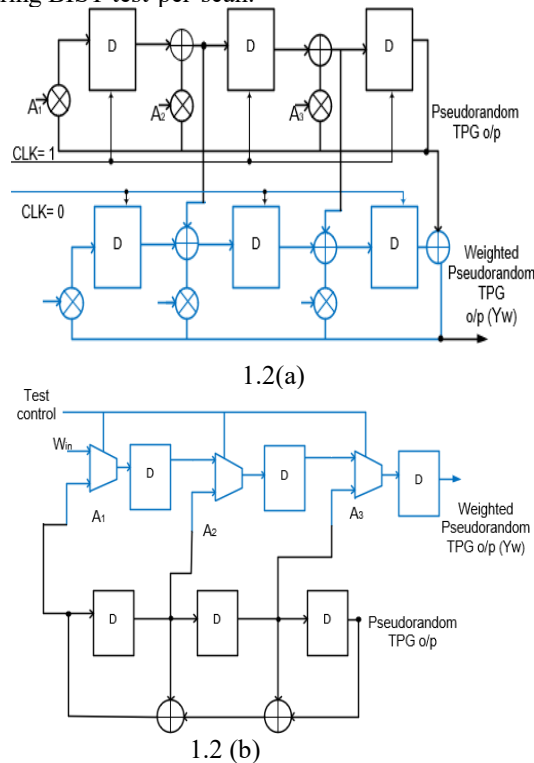


FIGURE 1.2. An example of an existing 3-bit weighted pseudorandom TPG: (a) redundant TPG; and (b) TPG;

The proposed TPG is designed using logic gate techniques and applied in two different test-per-scan BIST architectures. It is implemented and evaluated on a Mentor Graphics IC station using a SilTerra 0.13 μm submicron process. The rest of this paper is organized as follows: The existing weighted pseudorandom TPG is outlined in Section II. In Section III, a new weighted pseudorandom TPG is

proposed. Further- more, the mathematical analysis of the TPG depicts the achievement of maximum-length weighted patterns for the subset of pseudo-primary seeds concurrently.

EXISTING WEIGHTED PSEUDORANDOM TPGs

The primitive polynomial chosen for the TPG is determined by the even or odd tap bits from the register. Generally, primitive polynomials are used for generating pseudorandom patterns. If the tap bit sequences of an n -bit TPG are $n, m, k, l, \dots, 0$, then the co-primes of the tap numbers, such as $n - n, n - m, n - k, n - l, \dots, n - 0$, will also generate the pseudorandom TPG output [16]. Using this concept, the TPG can generate a considerable length of pseudo-primary seeds. In Figures 2 and 3, the black line denotes that the source TPG yields pseudorandom patterns, and the blue line components show the additional hardware used for generating the weighted pseudorandom patterns

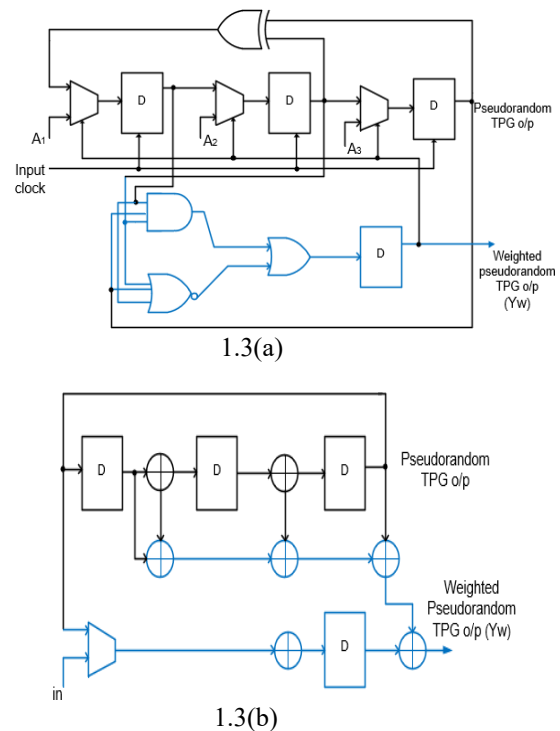


FIGURE 1.3. Examples of existing weighted 3-bit pseudorandom

The existing weighted redundant TPG method in Figure 2 (a) uses either hardware or duplication factors. The hardware redundancy duplicates its function into double modular redundancy, triple modular redundancy, and so on. This can be achieved by duplicating the additional hardware for the source TPG design, thus eliminating the random pattern-resistant faults. However, the hardware redundancy TPG achieves good performance factors; it should be reducible in the hardware overhead. The time redundancy accomplished using the multiple time criteria instead of the hardware using the asynchronous clock values ranges from "0" to "1".



Nonetheless, the same operation is executed using multiple time factors for the weighted patterns. This approach detects numerous resistant faults during different clock cycles.

The Xiang et al. TPG in Figure 2 (b) is mainly used for generating the weighted patterns using the control bits for the multiplexer (Mux). The additional hardware is also used to generate the reseeding bits required for the test patterns to identify their faults. The Mux in the additional hardware identifies the input seeds and the control bits when the constant weight input W_{in} passes the weighted patterns to an output. However, this includes a large area overhead in the scan-forest design of the BIST architecture. This can also be used for the lesser critical path delay, the scanning power, and capture power testing. Additionally, this approach uses a complicated design for the weighted reseeding technique.

To overcome the limitations of existing TPGs, more advanced TPG techniques are introduced later, as shown in Figure 3. These techniques target the weight of the test patterns using techniques such as information redundancy and parallel TPGs. Figure 3 (a) represents the Prasad et al. TPG. This TPG method uses a Mux between the D flip-flops, wherein the control signal is configured to the weighted pattern's initial valid state. Until the valid state of the weighted patterns is generated simultaneously, the additional gates introduce a delay in the circuit. Additionally, the gates introduce high switching transitions in the design. The delay adversely impacts the area and speed of the TPG. Figure 3 (b) shows the Hwasoo et al. TPG. Unlike the existing TPG methods, it uses a small number of hardware logic components rather than a replication. Additionally, the weighted patterns are generated, preferably using only one register function in the blue line. Although the weighted patterns predicted in this existing work have less area overhead, they are insufficient for a large number of pseudo-primary seed bits 'm'. A summary of the existing and proposed weighted TPGs is tabulated in Table 1. The requirements of the existing works are thus improved in the proposed TPG. It included a valid state of generating the weighted patterns for a larger seed bit with less power and area overhead. Significantly, this approach should guarantee low-power operation in all the test-per-scan phases of the BIST architecture.

LITERATURE SURVEY

J. Zhou, Z. Cao, X. Dong, and A. V. Vasilakos, "Security and privacy for cloud-based IoT: Challenges,".

The Internet of Things (IoT) is becoming the next Internet-related revolution. It allows billions of devices to be connected and communicate with each other to share information that improves the quality of our daily lives. On the other hand, Cloud Computing provides on-demand, convenient and scalable

network access which makes it possible to share computing resources, indeed, this, in turn, enables dynamic data integration from various data sources. There are many issues standing in the way of the successful implementation of both Cloud and IoT. The integration of Cloud Computing with the IoT is the most effective way on which to overcome these issues. The vast number of resources available on the Cloud can be extremely beneficial for the IoT, while the Cloud can gain more publicity to improve its limitations with real world objects in a more dynamic and distributed manner. This paper provides an overview of the integration of the Cloud into the IoT by highlighting the integration benefits and implementation challenges. Discussion will also focus on the architecture of the resultant Cloud-based IoT paradigm and its new applications scenarios. Finally, open issues and future research directions are also suggested.

E. Zenner, "Cryptanalysis of LFSR-based pseudorandom generators—A survey,"

Pseudorandom bit generator (PRBG) is an essential component for securing data during transmission and storage in various cryptography applications. Among popular existing PRBG methods such as linear feedback shift register (LFSR), linear congruential generator (LCG), coupled LCG (CLCG), and dual-coupled LCG (dual-CLCG), the latter proves to be more secure. This method relies on the inequality comparisons that lead to generating pseudorandom bit at a non-uniform time interval. Hence, a new architecture of the existing dual CLCG method is developed that generates pseudo-random bit at uniform clock rate. However, this architecture experiences several drawbacks such as excessive memory usage and high-initial clock latency, and fails to achieve the maximum length sequence. Therefore, a new PRBG method called as "modified dual-CLCG" and its very large-scale integration (VLSI) architecture are proposed in this paper to mitigate the aforesaid problems. The novel contribution of the proposed PRBG method is to generate pseudorandom bit at uniform clock rate with one initial clock delay and minimum hardware complexity. Moreover, the proposed PRBG method passes all the 15 benchmark tests of NIST standard and achieves the maximal period of 2^n . The proposed architecture is implemented using Verilog-HDL and prototyped on the commercially available FPGA device.

J. Stern, "Secret linear congruential generators are not cryptographically secure,"

The dual-coupled-linear congruential generator (LCG) (dual-CLCG) is a secure pseudorandom bit generator (PRBG) method among various linear feedback shift register (LFSR), LCG, and chaotic-based PRBG methods for generating a pseudorandom bit sequence. The hardware implementation of this method has a bottleneck due to the involvement of



inequality equations. Initially, a direct architectural mapping of the dual-CLCG method is performed. Since two inequality equations are involved for coupling, it generates pseudorandom bit at unequal interval of time that leads to large variation in output latency. In addition, it consumes a large area and fails to achieve the maximal period. Hence, to overcome the aforesaid drawbacks, a new efficient PRBG method, i.e., “coupled-variable input LCG (CVLCG),” and its architecture are proposed. The novelty of the proposed method is the coupling of two newly formed variable input LCGs that generates pseudorandom bit at every uniform clock rate, attains maximum length sequence, and reduces one comparator area as compared to the dual-CLCG architecture. The proposed architecture is implemented using Verilog-HDL and prototyped on the commercially available field-programmable gate array (FPGA) device. Furthermore, the sequences are captured through the logic analyzer and evaluated for randomness using the National Institute of Standard and Technology (NIST) standard test tool. The experimental result reports that the proposed PRBG method passes all the randomness tests with a high degree of consistency.

W. B. Jone and D. C. Huang and S. C. Wu and K. J. Lee, **An efficient BIST method for small buffers**, Astonishing progress in Silicon devices and circuits and highly reliable mass manufacturing techniques have prompted unprecedented revolutions in electronics for the last 4 decades to the extent that electronics is growingly permeating numerous aspects of our life. The application world, consumer and infrastructure, is now used to exponential performance improvements and high yield. Complexity and competitiveness of modern electronic systems demand for optimization across multiple disciplines. Joint optimization of device, circuit, packaging, and test are increasingly important for high performance systems. Design for Manufacturing and Testing is more critical but increasingly challenging in complex systems. So what does optimization mean for the future of ever-growing and complex solid-state electronics? A daunting challenge worth spending a talk on IEEE Standard 1687-2014 - IEEE Standard for Access and Control of Instrumentation Embedded within a Semiconductor Device, IJTAG - Internal Joint Test Action Group, IEEE Computer Society.

A methodology for accessing instrumentation embedded within a semiconductor device, without defining the instruments or their features themselves, via the IEEE 1149.1(TM) test access port (TAP) and/or other signals, is described in this standard. The elements of the methodology include a hardware architecture for the on-chip network connecting the instruments to the chip pins, a hardware description language to describe this network, and a software language and protocol for communicating with the

instruments via this network. This standard develops a methodology for access to embedded instrumentation, without defining the instruments or their features themselves, via the IEEE 1149.1(TM) test access port (TAP) and additional signals that may be required. The elements of the methodology include a description language for the characteristics of the features and for communication with the features, and requirements for interfacing to the features.

PROPOSED LOW-POWER WEIGHTED PSEUDORANDOM TPG USING THE GALOIS OPERATION WITH A PHASE SHIFTER

Compared with the existing methods, the proposed weighted TPG is designed with some advantages, including fewer switching transitions achieved using the specific weighted patterns and reduced power attained using fewer hardware components in the design. This reduces the hardware over- head and improves the fault coverages in the BIST. The TPG method shown in Figure 1 is the proposed TPG, which includes the Galois operation and additional hardware for weighted pattern generation. The Galois operation in the proposed TPG is shown by the black dashed line and assumes constant pseudo-primary seeds (A, X) for simplification. However, the constant seed bits can be enlarged using the same subset of initial primary seeds. The seed subsets are used to achieve the maximum length in weighted patterns with less switching activity. The additional hardware indicated by the blue line uses a smaller number of components for generating the weighted pseudorandom TPG output.

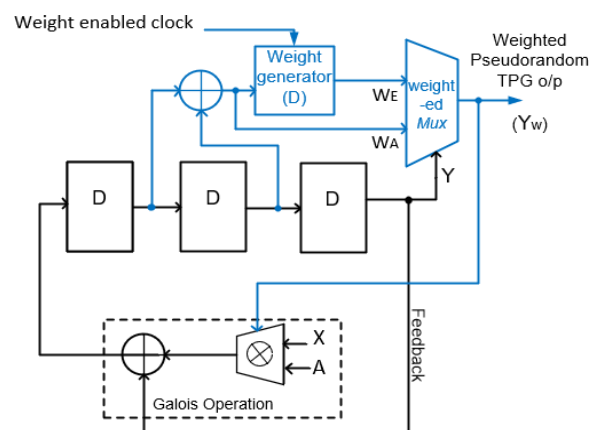


FIGURE 3.1. Proposed 3-bit weighted pseudorandom TPG.

In addition, the additional hardware design uses a weight-Hence, the m-bit TPG is designed using asynchronous clocks in shift registers. The input vector bit (X) is multiplied continuously by the pseudo-primary seed bit (A) and added to the test vectors (Z). In addition, the state of the registers accommodates the multilevel parallelism in the TPGs. Consequently, the next (i 1)-th state after the i-th state is described in terms of the feedback loop structure.



Furthermore, the constant pseudo-primary seeds are enlarged in the Galois operation of the proposed TPG design. This can be extended using the following Galois field Lemma 1 used to identify the subsets of initial pseudo- primary seeds.

Lemma 1: Let A and X be the two input elements in $GF(2^m)$ and Z be their multiplication using the Galois multiplier. If the field is assumed to be without modular reduction, then their weighted patterns are defined as $W_Z = W_A * W_X$.

Lemma 1: Let A and X be the two input elements in $GF(2^m)$ and Z be their multiplication using the Galois multiplier. If the field is assumed to be without modular reduction, then their weighted patterns are defined as $W_Z = W_A * W_X$.

Proof: Let $A = (a_0, a_1, \dots, a_{m-1})$ and $X = (x_0, x_1, \dots, x_{m-1})$ be the two types of elements in $GF(2^m)$, and Z be their output. Then, the output Z can be determined as in [20] $Z = [(a_0 * x_0) + (a_1 * x_1) 2 + \dots + (a_{m-1} * x_{m-1}) 2^{m-1}]$. From the concept of [6], the weighted function satisfies the linear property. It consists of the property of additivity, $W(A_0 + A_1) = W(A_0) + W(A_1)$, and homogeneity $W(c * A) = c * W(A)$, where 'c' is a constant. Hence, the weighted patterns of Z can be achieved by $W(Z) = W[(a_0 * x_0) + (a_1 * x_1) 2 + \dots + (a_{m-1} * x_{m-1}) 2^{m-1}] = W(a_0 * x_0) + W(a_1 * x_1) + \dots + W(a_{m-1} * x_{m-1})$.

Thus,

$$W(Z) = \sum_{i=0}^{m-1} W(a_i * x_i) = \sum_{i=0}^{m-1} W(a_i) * W(x_i) \quad (1)$$

The weighted function is adopted to be between $W(Z) = 0$ for the even weights and $W(Z) = 1$ for the odd weights. This allows for a large number of seed bits A and X as $A = 2^{m-1}$ and $X = 2^{m-1}$, where m is the number of message bits in the field of $GF(2^m)$. Hence, the weighted patterns can also be generated with the maximum length, as shown in equation (1). the conventional TPGs [5] update the next (i+1)-th state as

$$Y_n[i+1] = Y_{n-1}[i] + x_n * Y[i], \text{ for } 0 \leq n \leq m-1 \quad (2)$$

Hence, the additional hardware is designed for seed bit polynomial Z [i] instead of the primitive polynomial Y [i], as in equation (2). The function Z[i] is assumed to be $Z[a_i] = Z[a_0, a_1, \dots, a_{m-1}]$ because the maximum length TPGs are generated to the phase shifter. By applying the weighted function as the polynomial Z[i] in equation (2) on the right side,

$$W_A[i] = W \left[\sum_{i=0}^{m-1} (Z_{n-1}[i] + x_n * Z[i]) \right] \quad (3)$$

As per the linear property, such as homogeneity and the additive property of, equation (3) can be simplified as

$$W_A[i] = W \left[\sum_{i=0}^{m-1} Z_{n-1}[i] \right] + Z[i] W \left[\sum_{i=0}^{m-1} x_n \right] \quad (4)$$

However, for a large number of seed bits with subjective vector bits $g(X) = \sum_{i=0}^{m-1} x_n$, equation (4) is applied as a weighted generator. The conventional

TPGs require '2m-1' binary additions due to the primitive polynomial Y [i]. Concerning the proposed 3-bit TPG, the weight generation is defined as the convolution of the 'm-1' binary additions multiplexed with pseudo primary seeds. Here, the proposed TPG requires only 'm' binary additions and one selection, as shown in equation (4). WE is the estimated weight to be obtained at the (i+1)-th clock cycle, and k denotes the total number of clocks in equation (5). WE assign random weights in the range of '0' to 'k' using the clock delay of the D flipflop. To obtain the estimated we side can be applied with the weighted function as

$$W_E[i+j] = W[Z_n[i+j]] \text{ where } j = \{1, 2, \dots, k\} \quad (5)$$

The additional hardware introduced in the proposed design is indicated by the blue line components. This approach uses an XOR gate, weight generator, and phase shift selection using the weighted Mux. WA is denoted as the tapped convolution value actual weight by the XOR gate, and WE are indicated as an estimated weight by the weight generator. The weight enabled clock activates one of the weighted patterns to the weight generator. The weighted patterns are mathematically calculated using the probability distribution. A flowchart summary of the proposed weighted TPG operation is shown in Figure 5. The weighted Mux acts as a phase shifter to shift the actual and estimated weighted patterns to the scan chains. The weighted Mux also selects the convolution bits WE and WA with self-control, as the weighted patterns WE and WA can be shifted to the output Yw according to the pseudorandom test patterns (Y). The scan chains are identified by the weighted patterns WE and WA using the following essential properties.

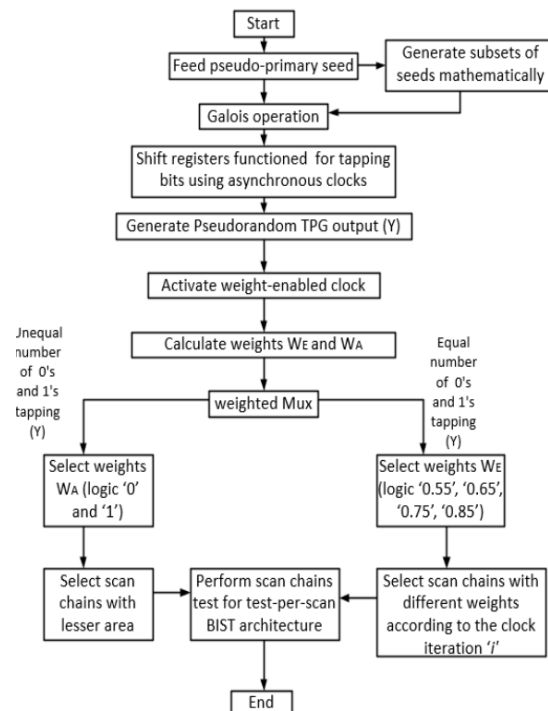


FIGURE 3.2. Flowchart of the proposed weighted pseudorandom TPG operation



First, the weighted patterns WA generated with the probabilities of having '0' or '1' assigned to the certain scan chains occupy a smaller area. The output of the weighted Mux depends on the important feature of the pseudo randomness of the seed inputs. Consider a case in which Y0 will be swapped with Y1, Y2 until Yn, according to the value of the Galois operation (Z) in the proposed 3-bit TPG. Here, Y2 is Y, which is connected to the selection input of the Mux. This determines the weighted pattern. Hence, overall switching transitions in the scan chain primary inputs can be reduced by 25%.

Second, the weighted patterns WE are selected by the weighted Mux if the pseudorandom output (Y) generates an equal number of '0' or '1' values after swapping through the shift registers. If the two scan chain inputs are adjacent to each other at the i-th clock, the probability value is assumed to be 0.55. The two interconnected scan chain cells are assigned a probability value of 0.75.

The random probability value of 0.65 is allocated to the scan chains with similar input vectors whose transition probability has the same value. A probability value of 0.85 is applied to the remaining scan chain cells.

APPLICATION OF PROPOSED WEIGHTED TPG IN TEST-PER-SCAN BIST ARCHITECTURE

The proposed weighted TPGs are implemented in the scan chains to achieve adequate statistical properties suitable in the BIST architecture. BIST architecture is tested using two methods: test-per-clock and test-per-scan. Test per clock is the testing method used to test CUTs individually using the test-point insertion. Test-per-scan is the method used to test the number of scan chains of the BIST in parallel. In general, fault coverage in the test-per-scan BIST can be accurately achieved by using test-point insertion between scan chains. The test-per-scan BIST architecture consists of a TPG, response analyzer, and signature register. The architecture includes the multiple-input signature register (MISR) as a response analyzer used to analyze whether the CUT is fault-free or fault-free. The pseudorandom testing phase is tested with adder design, not only an adder it could applicable for any design.

BIST TECHNIQUE

Built-In Self-Test is a technique of integrating the functionality of an automatic test system onto a chip. It is a Design for Test technique in which testing (test generation and test application) is accomplished through built in hardware features. The general BIST architecture has a BIST test controller which controls the BIST circuit, test generator which generates the test address sequence, response verification as a comparator which compares the memory output response with the expected correct data and a CUT. We have used LFSR and signature analyzer for testing

a three-input combinational logic circuit. The BIST controller can be implemented by either hardwired logic in the form of a Finite State Machine (FSM), microcode controller or processor-based.

The main challenging areas in VLSI are performance, cost, power dissipation is due to switching i.e. the power consumed testing, due to short circuit current flow and charging of load area, reliability and power. The demand for portable computing devices and communications system are increasing rapidly. These applications require low power dissipation VLSI circuits. The power dissipation during test mode is 200% P more than in normal mode. Hence it is important aspect to optimize power during testing. Power optimization is one of the main challenges.

BIST ARCHITECTURE:

A typical BIST architecture consists of

- TPG - Test Pattern Generator
- TRA – Test Response Analyzer
- Control Unit

TEST PATTERN GENERATOR (TPG)

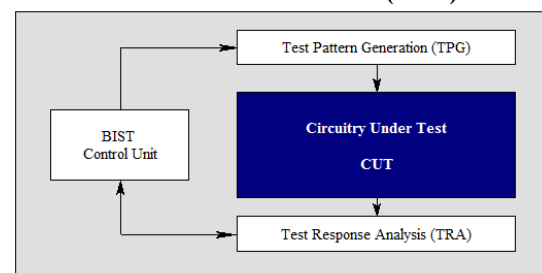


Fig 3.3. BIST architecture

It generates test pattern for CUT. It will be dedicated circuit or a microprocessor. Pattern generated may be pseudo random numbers or deterministic sequence. Here we are using a weighted pseudorandom TPG for generating random number.

TEST RESPONSE ANALYZER (TRA)

TRA will check the output of MISR & verify with the input of weighted pseudorandom TPG & give the result as error or not.

BIST CONTROL UNIT

Control unit is used to control all the operations. Mainly control unit will do configuration of CUT in test mode/Normal mode, feed seed value to weighted pseudorandom TPG, Control MISR & TRA. It will generate interrupt if an error occurs. You can clear interrupt by interrupt_clear_i signal.

CIRCUIT UNDER TEST (CUT)

CUT is the circuit or chip in which we are going to apply BIST for testing stuck at zero or stuck at one error.

Need for using BIST technique

Today's highly integrated multi-layer boards with fine-pitch ICs are virtually impossible to be



accessed physically for testing. Traditional board test methods which include functional test, only accesses the board's primary I/O's, providing limited coverage and poor diagnostics for board-network fault. In circuit testing, another traditional test method works by physically accessing each wire on the board via costly "bed of nails" probes and testers. To identify reliable testing methods which will reduce the cost of test equipment, research to verify each VLSI testing problems has been conducted.

The major problems detected so far are as follows:

- **Test Generation Problems**

The large number of gates in VLSI circuits has pushed computer automatic-test-generation times to weeks or months of computation. The numbers of test patterns are becoming too large to be handled by an external tester and this has resulted in high computation costs and has outstripped reasonable available time for production testing.

- **The Gate to I/O Pin Ratio Problem**

As ICs grow in gate counts, it is no longer true that most gate nodes are directly accessible by one of the pins on the package. This makes testing of internal nodes more difficult as they could neither no longer be easily controlled by signal from an input pin (controllability) nor easily observed at an output pin (observe ability). Pin counts go at a much slower rate than gate counts, which worsens the controllability and observe ability of internal gate nodes.

Integrated Circuits (IC's) are designed and manufactured to meet application specific functional requirements. Some examples of applications are camera-on-a-chip, MP3 player, etc. These functional requirements often need to be balanced with the desired IC performance, maximum allowable power consumption and overall packaged and tested IC cost. Generally total power consumption of the device is a sum of the power consumed by the core and I/O's. Appropriate package attributes such as the material and thermal properties need to be selected to ensure maximum heat dissipation of the die through the package.

IC Architects normally have a specific power consumption budget for the SOC based on the overall system level power budget. Functional operation of the device in a system usually consumes power either at or under the budgeted power for the IC. However, the same device under manufacturing test especially with BIST controller and random pattern generator can consume more power than the budgeted power.

This increase in the power consumption of the IC in the test mode is well known in the industry to cause sudden un-repairable device failures resulting in significant manufacturing fall-out directly impacting the cost of the IC.

Today a combination Q of external Automated Test Equipment (ATE) and internal BIST (Built-In-Self-Test) techniques are used to ensure the highest possible fault coverage of the device at the lowest possible cost IC testing using exclusively external ATE's can

require SOC architects to allocate a fairly large number of pins of the device to invoke the test procedure and run vectors into and through the various blocks of the device such as memory, user defined logic, dedicated functional macros, etc. Combination of external ATE's and internal BIST however can result in, utilizing far fewer external pins on the IC but at the cost of embedding test logic inside device.

PROPOSED DESIGN

Built-In Self Testing (BIST) architectures have gained a lot of consideration in the last fifteen years. Modern complex designs, consisting of multi-cores, demand more reliable and effective embedded testing with no extra complexity and with as much reductions in storage requirements as possible. Replacement of costly Automatic Testing Equipment (ATE) with efficient on-chip embedding architectures is desired to fill the technology gap between ATE and integrated circuits design capabilities. Traditional BIST schemes use pattern generation circuitry, like LFSRs, to test the examined circuit in a pseudo-random manner. However, for more complex designs, it is necessary to apply deterministic test patterns (generated by an appropriate ATPG tool) to cover hard-to-detect faults. Thus, a mixed-mode scheme is usually followed where a pseudorandom sequence of limited length and a sequence of deterministic tests are applied. Popular deterministic BIST schemes involve ATPG and test set compaction, encoding and regeneration of the embedded tests on the chip, and application of the patterns via scan-chain loading. Most of the previous work for deterministic test set embedding for stuck-at faults focuses on proposing new BIST architectures along with efficient test encoding techniques to provide for high fault coverage.

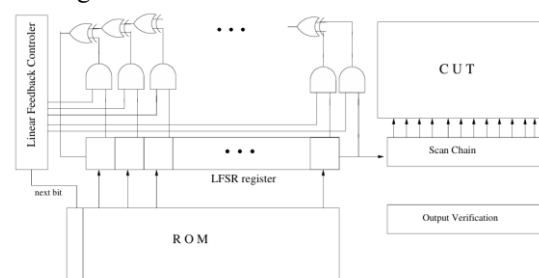


Fig 3.4. BIST using deterministic pseudo random number generator

Most of these techniques employ scan-per-test application to reduce test application time, in contrast to some other methods which use test-per-clock schemes to gain from all intermediate patterns produced by the test regenerating device. Many of the existing encoding techniques try to take advantage of the unspecified bits in the test set to be embedded, in order to reduce the seed storage. For the schemes using the original LFSR reseeding architecture, either with single or multiple LFSR configuration, the hardware overhead is primarily determined by the



storage requirement for the deterministic patterns. Given a test pattern t with $s(t)$ denoting the number of specified bits in t , the test pattern can be encoded, with a very high probability of success, into an $s(t)$ or an $s(t)+20$ bit-wide seed using certain methods. Given a test set $T = \{t_1, t_2, \dots, t_k\}$, the storage amount of the encoded tests is determined by the maximum number of specified bits in $s_{\max} = \max \{s(t) | t \in T\}$ as well as the distribution of the numbers $s(t_1), s(t_2), \dots, s(t_k)$. Even though existing test pattern generators (TPGs) can generate tests with unspecified bits, they do not specifically consider the number and distribution of the specified bits in the final compacted test set. demonstrated that the scheme can benefit greatly, in terms of reducing the storage requirement, if the embedded test patterns are generated by an ATPG tool that (i) maximizes the number of unspecified bits in each generated test and (ii) concatenates tests into sequences that have a number of specified bits equal or slightly greater than $s_{\max}$. In this work, we present a novel ATPG technique for generating test patterns from both weighted random number as well as LFSR based deterministic random number that can be encoded efficiently when using LFSR-based reseeding of test-per-scan BIST. We are motivated by the work of which has shown considerable reduction in the LFSR seed storage when ATPG is guided to find a suitable solution. We use function-based test generation and address the problem of finding test patterns that detect a large number of faults and at the same time constraint the number of specified bits in each generated test. We use Binary Decision Diagrams (BDDs) to represent a fault's test function (set of all possible tests for the fault). We have selected BDDs due to the clear advantage they offer over other function representation methods, for our problem formulations. The canonical structure of BDDs allows us to retrieve the test with the minimum number of specified bits (under some variable ordering) from the test function of a fault, in linear time to the BDD size. This is achieved by finding the largest cube in the BDD using a single traversal. The proposed algorithm pivots on this property to decide whether the test functions of two faults are to be combined (compacted), to get a test function that covers both faults. This is done only if the size of the largest cube in the combined function is close to the sizes of the largest cubes in the individual functions (we precisely define what close means in Section that describes the algorithm). We develop a systematic approach on how to combine test functions by formulating the problem as an iterative modified min-cost max-matching problem on a function compatibility graph. Each vertex in the compatibility graph corresponds to a test function. An edge exists between two vertices only if, when combining the functions of the two vertices, the size of the largest cube of the resulting function is below a user-defined threshold. The rest of the paper is organized as follows. Section 2 overviews basic BIST schemes

with LFSR reseeding. In Section 3 we present the proposed ATPG algorithm which allows for test set generation suitable for on-chip deterministic test set embedding using the considered architectures deterministic test set embedding using LFSR reseeding. We assume a test-per-scan architecture. The size of the scan chain equals the number of primary inputs, let this be denoted by m . The traditional single-polynomial LFSR scheme, used for generating pseudorandom test patterns, has been tuned to regenerate deterministic test patterns, by using different initializing seeds obtained by solving systems of linear equations. Thus, it can be used in a mixed mode approach for both pseudorandom test generation and deterministic pattern decompression and application. This method has shown that using an LFSR of size $s_{\max} + 20$ gives a very high probability of finding a solution to the systems.

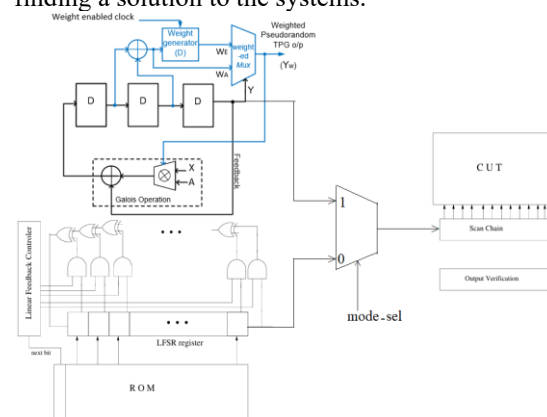


Fig 3.5. Novel BIST architecture using deterministic pseudo random number generator and weighted random number generator.

In the Multiple-Polynomial LFSR (MPLFSR) reseeding scheme of, the LFSR hardware is modified to become reconfigurable, i.e. include logic for reprogramming the LFSR feedback. Essentially, reprogramming refers to the process of changing the characteristic polynomial of the LFSR. This method manages to improve the encoding efficiency of LFSR reseeding using an LFSR size equal to $s_{\max}$ by only considering 16 different polynomials. For the MPLFSR scheme, extra storage is necessary to keep the polynomial selection sequences as well as extra bits for each seed to identify the corresponding polynomials. Figure 1 shows the underlying architecture of the MPLFSR reseeding scheme. Several other LFSR reseeding schemes have been proposed that can further improve the overall seed storage in the expense of some extra BIST hardware to the one shown in Figure 2. proposed a technique for partially reseeding the LFSR instead of fully replacing the LFSR seed, reducing in this way the seed storage. The scan chain is fed by multiple LFSR cells (not only the last one) to provide more useful test patterns. During each phase the same set of seeds is loaded to the LFSR but a different LFSR cell drives the scan chain per phase. In this manner a



smaller set of seeds needs to be stored in the expense of extra time due to the multiple phase application. The proposed structure having both deterministic and weighted random number generators .these two generators are work based selected mode ,if mode value is 1 the patterns are taken from weighted random number otherwise patterns are taken from deterministic random number generator .these patterns are used for BIST application .in future these patterns will be more useful for applying key for advanced encryption standard because of the randomness of the proposed structure is more compared to traditional random number generators ,it will increase the security of the AES algorithm .

RESULTS

New Project Summary:

The image below illustrates the New Project Summary created in the Vivado Design Suite. It summarizes the initial setup parameters for a new RTL project before adding design sources or constraints.

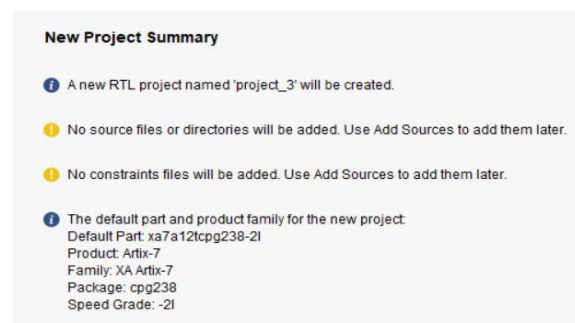


Fig 4.1. New project summary

Figure 4.1 shows the new project summary window displayed after creating a new RTL project in the Xilinx Vivado design suite. this summary provides the final confirmation of the project configuration before the project is created.

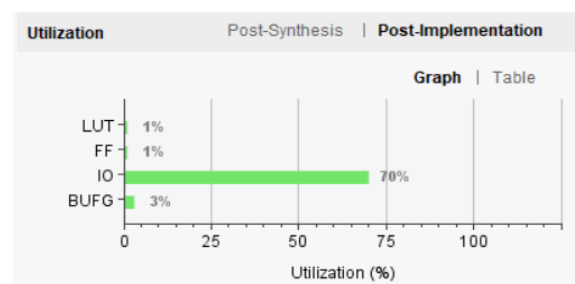


Fig 4.2. Device utilization summary

Figure 4.2 illustrates the FPGA resource utilization after implementation, showing the percentage of hardware resources used in the design. The results indicate minimal usage of LUTs (1%), Flip-Flops (1%), and BUFGs (3%), while I/O resources have the highest utilization at 70%, demonstrating efficient overall resource utilization.

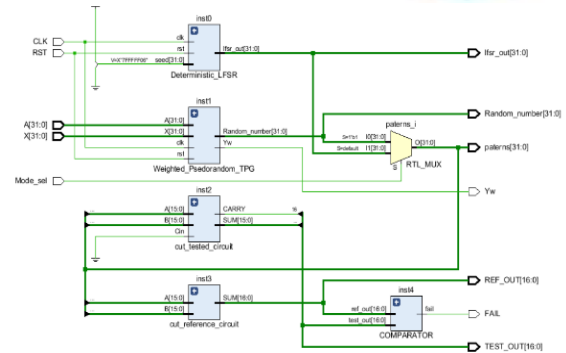


Fig 4.3. Internal structure of RTL Schematic of proposed design

Figure 4.3 shows the RTL schematic illustrates the internal architecture of the proposed design, showing the interconnected logic modules and signal flow between them. It provides a high-level view of the hardware implementation.

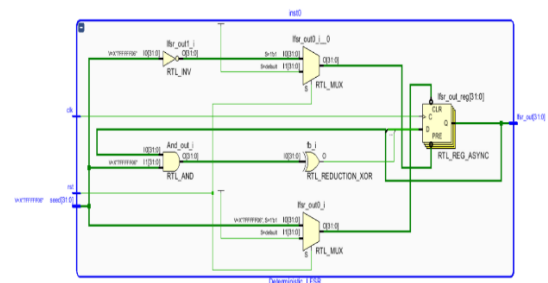


Fig 4.4. RTL Schematic of the Deterministic LFSR Module

Figure 7.4 shows the detailed RTL schematic of the Deterministic Linear Feedback Shift Register (LFSR), highlighting the internal components. It illustrates the data flow and feedback mechanism used to generate deterministic test patterns for the Built-In Self-Test (BIST) architecture.

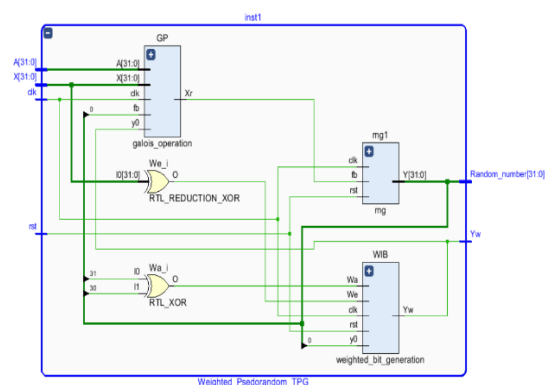


Fig 4.5. RTL Schematic of the Weighted Pseudorandom TPG

Figure 7.5 illustrates the detailed RTL schematic of the Weighted Pseudorandom Test Pattern Generator (TPG), showing its internal modules, logic gates, and register connections. It demonstrates the generation of weighted pseudorandom test patterns using Galois operations and weighted bit generation for effective



Built-In Self-Test (BIST).

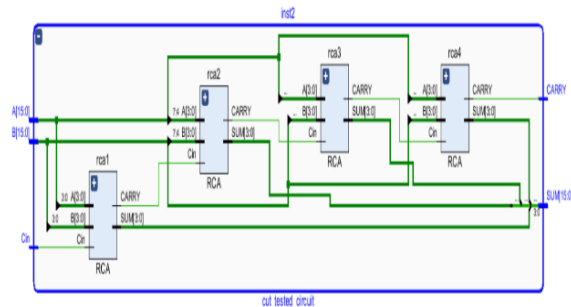


Fig 4.6. RTL Schematic of the Circuit Under Test (CUT)

Figure 7.6 illustrates the detailed RTL schematic of the Circuit Under Test (CUT), implemented using cascaded Ripple Carry Adder (RCA) modules. It shows the internal data flow, carry propagation, and interconnection of the adder blocks used to validate the proposed Built-In Self-Test (BIST) architecture.

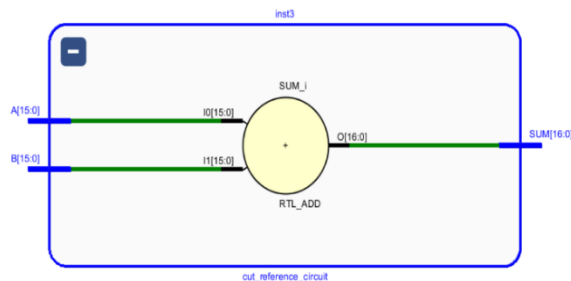


Fig 4.7. RTL Schematic of the CUT Reference Circuit

Figure 7.7 illustrates the RTL schematic of the reference circuit, which performs the ideal addition of the input operands using an RTL adder. The reference output is compared with the Circuit Under Test (CUT) output to verify the correctness of the proposed Built-In Self-Test (BIST) design.

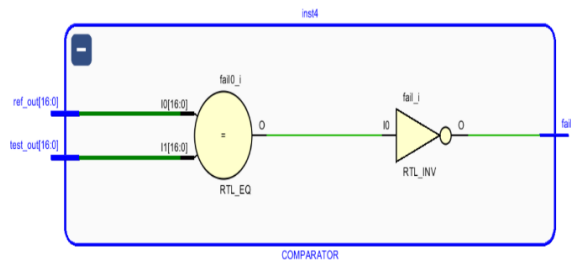


Fig. 4.8. RTL Schematic of the Comparator

Figure 7.8 illustrates the RTL schematic of the comparator, which compares the outputs of the Circuit Under Test (CUT) and the reference circuit. It generates a pass/fail signal by indicating whether the two outputs match, thereby verifying the correctness of the proposed BIST design.

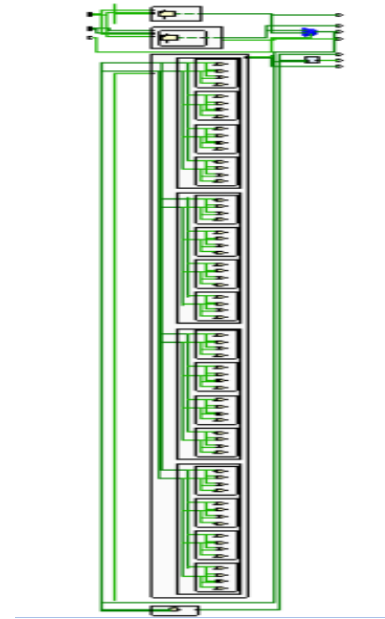


Fig 4.9. Internal logic structure of the RTL schematic of the proposed design

Figure 4.9 shows the detailed internal logic structure of the RTL schematic after zooming into the proposed BIST design. It illustrates the interconnected logic blocks and signal paths that implement the functionality of the proposed architecture.

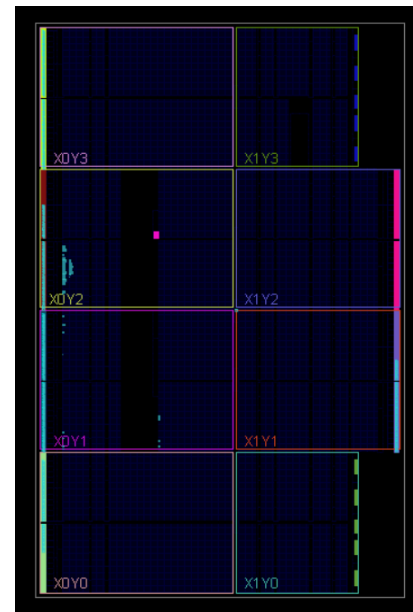


Fig 4.10. FPGA Device Floorplan of the Proposed Design

Figure 4.10 shows the FPGA device floorplan after implementation, illustrating the physical placement of the proposed design on the target Artix-7 FPGA. It provides a visual representation of resource allocation and placement across different regions of the FPGA, confirming successful implementation.



Fig. 4.11: Detailed FPGA Floorplan Showing Logic Resource Allocation

Figure 4.11 presents a detailed view of the implemented design, highlighting the placement of logic resources within the FPGA fabric. It verifies the successful mapping of the proposed design onto the available configurable logic blocks (CLBs) and other hardware resources.

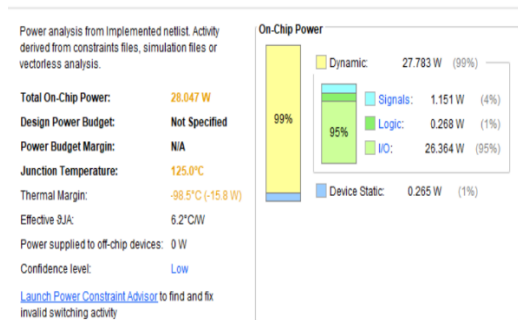


Fig 4.12: Post-Implementation On-Chip Power Analysis of the Proposed Design

Figure 4.12 presents the on-chip power analysis report generated after implementation, showing the total power consumption and its distribution among dynamic and static power components. The results indicate that I/O power (95%) contributes the most to the total power consumption, while logic and static power utilization remain comparatively low.



Fig 4.13: Simulated Waveform of the Proposed BIST Design

Figure 4.13 shows the simulated waveform verifies the functional operation of the proposed Built-In Self-Test (BIST) design by displaying the input, output, clock, and control signal responses over time. The

simulation confirms that the design operates correctly and produces the expected outputs according to the specified test conditions.

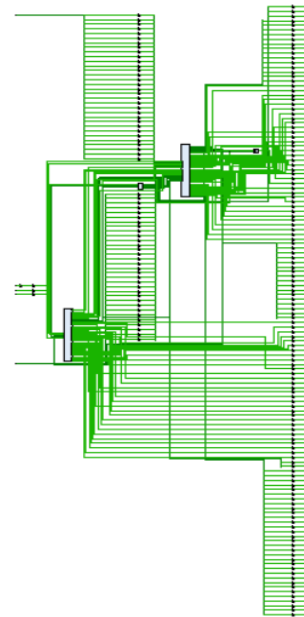


Fig 4.14: Technology Schematic of the Proposed BIST Design

Figure 4.14 shows technology schematic illustrates the gate-level implementation of the proposed Built-In Self-Test (BIST) circuit after synthesis. It shows the optimized logic elements and their interconnections, verifying the hardware realization and efficient mapping of the design onto the target FPGA.

CONCLUSION

A new low-power weighted TPG and deterministic based pattern generators are proposed for generating effective patterns. The subset of the initial weighted pseudo-primary seed bits is calculated mathematically using the Galois operation and deterministic patterns are generated by xor logics, which achieves maximum-length weighted patterns with low transition and more randomness for deterministic generator. The weighted Mux acts as a phase shifter in the design and is used for limiting the switching transitions during the test-per-scan method. The proposed method is enhanced to implement a 32-bit WTPG and deterministic random bits guaranteeing low power consumption on all clock cycles with lower area overhead. The work is implemented with BIST architectures to demonstrate higher fault coverage. Furthermore, this work can be extended to realize sequential faults, redundant faults, scanning and capture phase transition faults, and stuck-open faults. This approach can also be implemented in the multiple scan-forest architectures of the BIST. In future proposed random numbers can be used in cryptographic applications.



REFERENCES

- [1] B. LFSR, S. O. A. Novel, A. S. Abu-Issa, and S. F. Quigley, "Short papers," vol. 28, no. 5, pp. 755–759, 2009.
- [2] I. Pomeranz, "Computing seeds for LFSR-based test generation from non-test cubes," IEEE Trans. Very Large Scale Integr. (VLSI) Syst., vol. 24, no. 6, pp. 2392–2396, Jun. 2016, Doi: 10.1109/TVLSI.2015.2496190.
- [3] A. Sasi, A. Amir Soleimani, A. Ahmadi, and M. Ahmadi, "Hybrid memristor-CMOS based linear feedback shift register design," in Proc. 24th IEEE Int. Conf. Electron., Circuits Syst. (ICECS), Dec. 2017, pp. 62–65, Doi: 10.1109/ICECS.2017.8292094.
- [4] G. N. Balaji and S. C. Pandian, "Design of test pattern generator (TPG) by an optimized low power design for testability (DFT) for scan BIST circuits using transmission gates," Cluster Comput., vol. 22, no. S6, pp. 15231–15244, Nov. 2019, Doi: 10.1007/s10586-018-2552-x.
- [5] J. Zhang, Q. Zhang, and J. Li, "A novel TPG method for reducing BIST test-vector size," in Proc. Int. Symp. High Density Design Package. Microcyst. Integr., no. 149, Jun. 2007, pp. 6–9, Doi: 10.1109/HDP.2007.4283639.
- [6] S. Hellebrand, S. Tarnick, J. Rajski, B. Courtois, and T. I. M. Imag, "Multiple-polynomial linear feedback shift registers," 1992, pp. 120–129.
- [7] X. Lin and J. Rajski, "Adaptive low shift power test pattern generator for logic BIST," in Proc. Asian Test Symp., 2010, pp. 355–360, Doi: 10.1109/ATS.2010.67.
- [8] A. S. Abu-Issa, "Energy-efficient scheme for multiple scan-chains BIST using weight-based segmentation," IEEE Trans. Circuits Syst. II, Exp. Briefs, vol. 65, no. 3, pp. 361–365, Mar. 2018, Doi: 10.1109/TCSII.2016.2617160.
- [9] G. S. Sankari and M. Maheswari, "Energy efficient weighted test pattern generator-based BIST architecture," in Proc. Int. Conf. I-SMAC (IoT Soc. Mobile, Anal. Cloud), I-SMAC, 2019, pp. 448–453, Doi: 10.1109/I-SMAC.2018.8653768.
- [10] R. Kapur, S. Patil, T. J. Snethen, and T. W. Williams, "A weighted random pattern test generation system," IEEE Trans. Comput. - Aided Design Integr. Circuits Syst., vol. 15, no. 8, pp. 1020–1025, Aug. 1996, Doi: 10.1109/43.511581.
- [11] A. Jas, C. V. Krishna, and N. A. Touba, "Weighted pseudorandom hybrid BIST," IEEE Trans. Very Large Scale Integr. (VLSI) Syst., vol. 12, no. 12, pp. 1277–1283, Dec. 2004, Doi: 10.1109/TVLSI.2004.837985.
- [12] D. Xiang, M. Chen, and H. Fujiwara, "Using weighted scan enable signals to improve test effectiveness of scan-based BIST," IEEE Trans. Comput., vol. 56, no. 12, pp. 1619–1628, Dec. 2007.
- [13] H.-C. Tsai, K.-T. Cheng, and S. Bhawmik, "On improving test quality of scan-based BIST," IEEE Trans. Comput.-Aided Design Integr. Circuits Syst., vol. 19, no. 8, pp. 928–938, Aug. 2000, Doi: 10.1109/43.856978.
- [14] N. A. Touba and E. J. McCluskey, "Altering a pseudo-random bit sequence for scan-based BIST," in Proc. IEEE Int. Test Conf., Oct. 1996, pp. 167–175, Doi: 10.1109/test.1996.556959.
- [15] G. Kiefer, H. Vranken, E. J. Marinissen, and H. J. Wunderlich, "Application of deterministic logic BIST on industrial circuits," J. Electron. Test. Theory Appl., vol. 17, nos. 3–4, pp. 351–362, 2001, Doi: 10.1023/A:1012283800306.
- [16] R. Cited and A. Banerjee, U.S. Patent, 2019, vol. 2.
- [17] B. W. Johnson, J. H. Aylor, and H. H. Hana, "Efficient use of time and hardware redundancy for concurrent error detection in a 32-bit VLSI adder," Comput. Arith., vol. 23, no. 1, pp. 171–178, 2015, Doi: 10.1142/9789814641470.
- [18] D. Xiang, X. Wen, and L.-T. Wang, "Low-power scan-based built-in self- test based on weighted pseudorandom test pattern generation and reseeding," IEEE Trans. Very Large Scale Integr. (VLSI) Syst., vol. 25, no. 3, pp. 942–953, Mar. 2017, Doi: 10.1109/TVLSI.2016.2606248.
- [19] H. Shin, S. Choi, J. Park, B. Y. Kong, and H. Yoo, "Area-efficient error detection structure for linear feedback shift register," Electronics, vol. 9, no. 1, pp. 1–10, 2020.
- [20] H. Dau, I. M. Duursma, H. M. Kiah, and O. Milenkovic, "Repairing Reed–Solomon codes with multiple erasures," IEEE Trans. Inf. Theory, vol. 64, no. 10, pp. 6567–6582, Oct. 2018, Doi: 10.1109/TIT.2018.2827942.
- [21] P. Wohl, J. A. Waicukauski, G. A. Maston, and J. E. Colburn, "XLBIST: X-tolerant logic BIST," in Proc. IEEE Int. Test Conf. (ITC), Oct. 2018, pp. 1–9, Doi: 10.1109/TEST.2018.8624738.
- [22] E. Moghaddam, N. Mukherjee, J. Rajski, J. Solecki, J. Tyszer, and
- [23] J. Zawada, "Logic BIST with capture-per-clock hybrid test points," IEEE Trans. Comput.-Aided Design Integr. Circuits Syst., vol. 38, no. 6, pp. 1028–1041, Jun. 2019, Doi: 10.1109/TCAD.2018.2834441.
- [24] H. Tran, "Demonstration of 5T SRAM and 6T dual-port RAM cell arrays," in IEEE Symp. VLSI Circuits, Dig. Tech. Paper, Jun. 1996, pp. 68–69, Doi: 10.1109/vlsic.1996.507719.
- [25] S. Roy, B. Stiene, S. K. Millican, and V. D. Agrawal, "Improved random pattern delay fault coverage using inversion test points," in Proc. IEEE 28th North Atlantic Test Workshop (NATW), May 2019, pp. 1–6, Doi: 10.1109/NATW.2019.8758727.



[26] C. Senthilpari, K. Diwakar, and A. Singh, "Low energy, low latency and high-speed array divider circuit using a Shannon theorem-based adder cell," *Recent Patents Nanotechnol.*, vol. 3, no. 1, pp. 61–72, Jan. 2009, Doi: 10.2174/187221009787003311.

[27] C. Senthilpari, A. K. Singh, and K. Diwakar, "Design of a low-power, high performance, 8-bit multiplier using a Shannon-based adder cell," *Microelectron. J.*, vol. 39, no. 5, pp. 812–821, May 2008, Doi: 10.1016/j.mejo.2007.12.016.