CHAOS BASED BIST DESIGN FOR TESTING OF DIGITAL CIRCUITS

A DISSERTATION

SUBMITTED IN PARTIAL FULFILLMENT OF THE
REQUIREMENTS FOR THE AWARD OF THE DEGREE
OF
MASTER OF TECHNOLOGY (2018-2020)
IN

VLSI DESIGN & EMBEDDED SYSTEM

Submitted by:

VANDANA

2K18/VLS/17

Under the supervision of

Ms. Kriti Suneja

Department of Electronics & Communication Engineering
Delhi Technological University
(Formerly Delhi College of Engineering)
Bawana Road, Delhi -110042

OCTOBER 2020

Department of Electronics & Communication Engineering
Delhi Technological University
(Formerly Delhi College of Engineering)
Bawana Road , Delhi-110042

CANDIDATE'S DECLARATION

I, Vandana, Roll no 2K18/VLS/17, a student of M. tech VLSI DESIGN & EMBEDDED SYSTEM, hereby declare that the project Dissertation titled **"Chaos Based BIST design for Testing of Digital Circuits"** which is submitted by me to the Department of Electronics and Communication Engineering, DTU(Delhi Technological University), Delhi in partial fulfillment of the requirement for the award of the degree of Master of Technology, is original and not copied from any source without proper citation. The work has not previously formed the basis for the award of any Degree, Diploma Associateship, Fellowship or other similar title or recognition.

Place: Delhi **VANDANA**

Date: October, 2020 **(2K18/VLS/17)**

Department of Electronics & Communication Engineering
Delhi Technological University
(Formerly Delhi College of Engineering)
Bawana Road , Delhi-110042

CERTIFICATE

I hereby certify that the Project titled **"Chaos Based BIST design for Testing of Digital Circuits"** which is submitted by VANDANA, Roll No 2K18/VLS/17, Electronics and Communication Engineering, Delhi Technological University, Delhi in partial fulfillment of the requirement for the award of the degree of Master of Technology, is a record of the project work carried out by the student under my supervision. To the best of my knowledge this work has not been submitted in part or full for any other Degree or Diploma to this University or elsewhere.

Place: Delhi **Ms. KRITI SUNEJA**

Date: October,2020 **SUPERVISOR**

ABSTRACT

Due to the increasing design size and complexity of recent VLSI design Circuits (IC) and the decreasing time-to-market, testing is one amongst the key bottlenecks within the IC design and development cycle.

Nowadays, in semiconductor industry technique for testing has been serious concern. A not well-designed testing technique, of any chip would be a harmful for design engineering as well as for verification engineer in many ways like cost, effort and consumer's trust. So now-a-days there is one promising technique for testing of design which is speedily changing with the developments in technology as size of device reduces which is BIST. As a result of small size device, hardware is also becoming complicated, this tendency has shifted to incorporate BISTs in electronic equipments which required high performances for offline testing. In BIST we test a design with the help of pattern generator, output response analyzer and with controller to manage all the actions of testing. Also, a comparison has been made between 'Linear Feedback Shift Register' (LFSR) based and chaotic circuit based BIST using hardware utilization and timing parameters. This report presents the testing of circuit for a BIST functioning in testing mode of operation. For simulations and synthesis, VCS and Verdi synopsis tools for Verilog and MATLAB/SIMULINK has been used. The output waveforms of chua's circuit and MISR/SISR are plotted.

ACKNOWLEDGEMENT

It gives me immense pleasure to express my deepest sense of gratitude and sincere thanks to my highly respected and esteemed guide and supervisor **Ms. Kriti Suneja** (Assistant Professor, ECE), for her valuable guidance, direction, encouragement and help for completing this work. Her useful suggestions for this whole work and co-operative behavior are sincerely acknowledged. And I am also grateful to the Head of Department (Dept. of ECE) for his support and encouragement in successfully completing this work.

At the end I would also like to express my sincere thanks to all my friends who have helped me directly or indirectly during this project work.

VANDANA
VLSI DESIGN & EMBEDDED SYSTEM
IVth Semester
Delhi Technological University
(Formerly Delhi College of Engineering)

TABLE OF CONTENTS

TABLE OF FIGURES

LIST OF TABLES

LIST OF ABBREVIATIONS

BIST	Built-In Self-Test
SoC	System On Chip
IC	Integrated Circuit
CUT	Circuit Under Test
DFT	Design For Testability
FC	Fault Coverage
FSM	Finite State Machine
I/O	Inputs/Outputs
LFSR	Linear Feedback Shift Register
PO	Primary inputs
PI	Primary outputs
MUX	Multiplexer
ORA	Output Response Analyzer
PRPG	Pseudo-Random Pattern Generator
RAM	Random Access Memory
ROM	Read Only Memory
TPG	Test Pattern Generator
RTL	Register Transfer Level
VLSI	Very Large Scale Integration
SISR	Serial input signature register
MISR	Multiple input signature register
XOR	Exclusive OR
ATPG	Automatic Test Pattern Generator

SR	Shift Register
SA	Signature Analyzer
ATE	Automatic Test Equipment
CRCC	Cyclic Redundancy Check code
DUT	Design under Test
PRBS	Pseudo-Random Binary Sequence

CHAPTER-1

INTRODUCTION

The functionality of electronic equipments and gadgets has achieved an outstanding growth over the last more than one decade while their physical sizes and weights have decreased drastically. The key reason is because of the speedy advances in integration technologies, which enables fabrication of many transistors on a single integrated circuit (IC). Every chip within the industry follows Moore's law. In keeping with this law, number of transistors (transistor density) on a chip doubles in every 1.5 years. With the recent advances within the technology, device size reduces up to nanometer scale, but density and complexity of the ICs keep up increasing. This might lead to many manufacturing faults and device failures. To accommodate more number of transistors, the device feature size is reduced. Reduction of the feature sizes leads to increasing the manufacturing faults and fault detection becomes very hard in design. VLSI testing is becoming most important and challenging to know whether a device functions properly or not.

Consequently, an increasing number of chips that have a combination of digital and analog circuits are designed. In various fields of communication, for example satellite, wireless and data exchanging design, use of this combination of both digital and analog IC has highly increased.

Additionally, testing these types of circuits is yet troublesome and exciting task because it must guarantee the full functionality, performance and excellence measure for high and basic functional level operation.

For mixed-signal circuits testing in industry numerous difficulties are exist, so for an alternative, we can use the BIST technique for, testing as a better option. It allows every element independently for testing. This lessens the need for complex functional tests and improves test reuse.[1] In BIST there is pattern generator part which gives input values for testing circuit, if for pattern generation we use random patterns then it will be more secure circuit. So for this pattern generation part we can use chaos based circuit for random patterns. It will be more advantageous for data security also.

Testing is an important part of any design from production to packing of all customer products, in this case VLSI designs. A product before it’s distributed to a customer has got to be tested and must be qualified fault free. However, checking of all IC parts is way more different and complex in comparison to other customer products. Because the decrease in size of the device increases complexity of circuits at a large level, and testing techniques are also obligatory to be much highly complicated. In this background, we suggest through this study chaos based Built in Self-Test abbreviated as BIST, testing methodology.

- **Importance of digital testing:**

It is defined as “testing a digital circuit to verify that it performs the required logic functions in proper time.” [6]. VLSI testing is most crucial step to assure quality of any design and, this testing technique is different from traditional techniques or methods.

IC designing has extended to sub-micro designs which integrated lots of transistors and its speed of processes has crossed up to GHz rate. Clearly, with a greater number of levels of integration the possibility of greater number of faults has increased. So, we have to keep this testing problem dominant.[6]

1.1 MOTIVATION

As very large-scale integrated circuits, become gradually complicated with each generation, as the quantity of test pattern required to realize satisfactory test quality is increasing proportionally. The test data bandwidth between the tester and chip are growing rapidly because of the test pattern storage requirements on an external tester. As volume of test data increases, test power also increases rapidly.

On-chip pseudo random patterns generation based Logic BIST has emerged as a primary solution to test current day large complex designs. The ATE based testing was using deterministic test patterns. Both the deterministic and pseudorandom test patterns which are generally used for testing have their advantages and disadvantages.

Within the on-chip testing scenario, it is comparatively easy to get pseudo-random pattern generator circuits than deterministic pattern generator circuits. It is because, generation of deterministic patterns requires design of highly complex digital circuits adding a tremendous overhead on area, size and complexity. BIST logic coverage typically falls in the range of 75% to 80%. The undetected faults in Logic BIST are usually referred as random pattern resistant (RPR) faults. It is not known precisely when to prevent generating these random patterns and is typically done when no more improvement in fault coverage is seen. Wang (wang, 2006) observed that BIST requires more test patterns than conventional testing, which may take an extra millisecond or two to work, and probably would not provide high enough test coverage without additional test vectors.[8-9]

Second concern for all digital designs is power dissipation. Since most of the switching activities happen in pattern generation, therefore loss of power in BIST design is totally dependent on the TPG.

For battery operated devices low power designs are required and to attain this several techniques has been advanced like reducing the number of registers without disturbing coverage of faults, using single test in one clock type arrangements generation of multiple output test patterns etc. Generators of random binary sequences are widely used in many applications like computer simulation, cryptography, gaming and randomized design. During the last few decades, chaos theory is in use for randomness which provide secure data. The focus of my study is to generate patterns that are truly random in nature by chaotic circuits.

1.2 OBJECTIVE

The main objective of this report is to understand the employment of chaotic circuits for test pattern generator and then implement it in BIST design for testing. They provide more secure and random data for communication. Its non-linear dynamics can be proficiently oppressed to understand low-power protected structures and data can be fault free. Realization of chaotic circuit with MATLAB and Verilog for pattern generation is explained in the report.

1.3 OVERVIEW

In direction to explain the contents of this research, an outline of this study is ordered in the following form:

This report has been organized into 4 chapters. Chapter 1 deals with the introduction. It also includes the motivation and objective of the report. Chapter 2 has review of some research papers related to my thesis. In chapter 3, Architecture of Built-in self-test with its parts are introduced. Chapter 4 deals with Design and Methodology and in next chapter 5 results and analysis of circuits and design has been shown.

CHAPTER-2

LITERATURE REVIEW

This chapter includes a thorough study of previously presented works on BISTs implementation using Chua's Circuit. Reviewing these literatures helps one to identify the technological gap in research and ideates to implement the said technique to infer and conclude for the same. The progress is then made by designing what is that gap and how one bridges it.

[11] Lahcene Merah, Adda Ali-Pacha, Naima Hadj Said and Mustafa Mamat

"A Pseudo Random Number Generator Based on the Chaotic System of Chua's Circuit, and its Real Time FPGA Implementation"

In this research paper authors presented appropriate technique to produce a strong PRNS which satisfy all the needs of security from a chaos based system of Chua's circuit. Evaluation of randomness of these numbers is done by NIST-800-22 statistical tests, and used to encrypt an image. FPGA is used here for a hardware implementation of this system. All the pattern sequence clear all the NIST statistical tests which approve its efficiency for cryptographic issues. In this study FPGA employment results demonstrate that the design has a low-cost implementation and inexpensive FPGA circuit (Spartan6 LX45familly in our case) is appropriate for the design. Moreover, the hardware employment offers an output of 1.9 Gbps which is appropriate for the furthermost actual cryptographic necessities.

[12] Recai Kilic, Fatma Yildirim Dalkiran

“Utilizing SIMULINK in Modeling and Simulation of Generalized Chaotic Systems with Multiple Nonlinear Functions”

Authors presented in this paper about the execution of two general chaotic structures, which have the most design part and variable dissimilar nonlinear function blocks by using SIMULINK software. With the anticipated sole and compress SIMULINK models for general chaotic structure which can be realized with different nonlinear functions, it will be easily investigated the roles and effects of various nonlinear functions in their own chaotic system structure. The likely other considerations for these function block is assessed in these compact models which are based on mathematics. The models presented here are often utilized in courses which covers non-linear designs/circuits, device and chaos. And engineering students can easily redesign these circuits and they can employ these designing techniques supported SIMULINK in investigating and designing of other chaotic systems.

[13] Marta Blaszczyk and Dr. Richard A. Guinee

“A novel modelled true random binary number generator for key stream generation in cryptographic applications”

Authors used a unique technique for producing true random binary sequence in this research. Even though the proposed chaos based TRBS generator topology has been used elsewhere, for binary stream generation, the proposed PRBSG employment for the TRBSG response data scrambling leads to an effective key stream generation in accord with NIST Test Suite. In this study full circuit execution has been used, and PSpice and MATLAB simulation is used for validation. To attain a constant time chaotic oscillator 3 current feedback operational amplifiers AD 844/AD are used with the voltage comparator LM311 this comparator is used for implementation of chaotic behavior. This circuit could simply be constructed in integrated module form along with the decorrelating PRBNG for key stream generation in encrypted component applications. Using the NIST and Diehard Test Suites they checked all bit pattern we generated for its non-periodic behavior. The mathematical state-space model has been found and implemented in MATLAB/Simulink. Then data result of NIST test will present for data found from simulation for both PSpice and MATLAB model implementation.

[14] Preethy K John and Rony Antony P

"Optimized BIST Architecture for Memory Cores and Logic Circuits using CLFSR"

Authors presented in this paper that BIST which is a technique to test embedded device. It offers simple automated testing methods to distinguish errors or faults at core memory and logic level of a system, which is called MBIST and LBIST respectively. To test a logic of circuit, test patterns are required. But in MBIST we require address sequences as well as test patterns. Author's work is focused in this paper to design an improved version of BIST architecture to test memory and logic circuits by CLFSR. Here CLFSR is designed effectively that produces all the possible combinations of zero's and one's for n-bits. In the BIST technique CLFSR as ATPG to generate patterns for memory or logic make design less complex. Correspondingly, with CLFSR, area overhead of traditional technique is also reduced because of use one common circuit here in place of two. By the execution outcomes, it has been verified that CLFSR produces all patterns of zero's and one's including all-zero pattern. Therefore, the design of CLFSR structure can be efficiently used for the generation of Memory address. sequences and test pattern sequences for BIST applications.

[15] Manjunath Mosalgi and Ganapathi Hegde

“Power Optimized TPG For BIST Architecture”

Authors in this paper proposed a new method of test pattern generator based on LFSR using clock gating method which lessens dissipation of power in BIST. When the logic component is not executing any function in digital device then that can be remove for the time being which reduces power dissipation in testing. Without disturbing to the fault coverage this proposed TPG averagely decreases power by 15% and area overhead is 11% according to ISCAS C17, C432, C499, C880, C1355, C1908, C2670, C3540, C5315, C6288 and C7552 benchmark circuits. The power consumption of proposed method can still be reduced in upcoming research. Because of extra circuit of clock gating in LFSR based TPG, this combinational circuit still consumes some power and that can be decreased with the help of pass transistor logic for the execution of AND and XOR gates.

[16] Abhilash Bagalkoti, Suhas B Shirol, Ramakrishna S, Rajashekar B.S. and Priyatamkumar

“Design and Implementation of 8-bit LFSR, Bit- Swapping LFSR and Weighted Random Test Pattern Generator: A Performance Improvement”

Authors presented a qualitative study of different type pattern generation techniques on the basis of delay and power. Individuality of this research implementation in the area of BIST for digital circuits to facilitate testing of design turn out to be easier with less power dissipation. These designs can be used in upcoming research for reversible logic so that loss in power can be minimized. Power dissipation in the design testing is more than that of the regular working of the IC. Because of the contraction in size of chip the power distribution of the circuit is also diminished. Adding a slight overhead to the design for the purpose of testing which can eliminate the use of complex exterior testing module is the method called as Built In Self Test (BIST). In BIST the LFSR is used to generate the random sequence for testing of a VLSI design.

[17]- Aixue Qi, Chunyan Han and Guangyi Wang

"Design and FPGA Realization of a Pseudo Random Sequence Generator Based on a Switched Chaos".

In this research Authors have presented a new switched chaotic system for generating good pseudo-random pattern for chaos-based wide range communications. In this paper a chaotic pseudo-random pattern generator which uses the proposed chaotic system is considered and realized using FPGA. In digital systems, the fact that chaotic systems are realized with finite precision contributes to the dynamic deterioration of novel systems. The presence of short cycle duration is the most notable issue of degradation. This paper introduces a new switched chaotic method to produce pseudo-random (PR) sequences for spread spectrum communications and develop its statistical properties, consisting of four separate subsystems, it can alter its configuration from one to another on a random and timely basis using two switching functions. A PR sequence generator is designed and implemented using the switched method, based on FPGA (field programmable gate array). This approach is based on the principle of combining the DSP Builder tool with applications from Matlab / Simulink and Quartus II, and ensures production of repeatable, compatible sequences based only on the initial values. In addition, the proposed PR sequence generator is subject to four simple statistical tests using the well-known FIPS-140-1 test suite. This PR sequence generator efficiently passes all essential statistical analysis which can be used in the design of different FPGA-based spectrum distribution communications.

[18] Ihsan Pehlivan, Ersin Kurt, Qiang Lai, Aziz Basaran, and Mustafa Cagri Kutlu

“A Multiscroll Chaotic Attractor And Its Electronic Circuit Implementation”

As per the author Scientists, engineers and researchers in the fields of physics, mathematics, electricals and electronics may need to model these equations and build electronic circuits to create new chaotic systems with complex and different dynamic behaviours. The physical realization of electronic circuits of some chaotic systems can be relatively difficult and complex. For example, in electronic circuit simulations of chaotic systems output amplitude values are not a problem, but in real circuit applications, the values that the state variables are limited by the supply voltage values of the operational amplifier or similar operating element (e.g. +15 V, -15 V). In this study, a multiscroll chaotic Lu-Chen system was scaled to be an example of systems that require scale surgery with differential equations. The comparative simulations and oscilloscope outputs of the chaotic multiscroll-scaled Lu-Chen system show that a successful scale operation and an electronic circuit design can be performed. Chaotic systems are hypersensitive to initial conditions, non-periodic, noise-like broadband systems. Therefore, they help to understand many systems that are considered random. There are many nonlinear structures in the universe.

[19]- M. S. Azzaz and M. A. Krimil, F. Labiod and A. Kadir, D. Teguig

“FPGA Hardware Design of a Unified Chaotic System for CTRNG”

In this paper authors propose a digital operation of unified chaotic system via hardware tactic. This method is based on basic parallel and pipelined hardware description in HDL language, VHDL. Experimental outcomes on Xilinx FPGA Zybo board validates that we can get different chaotic circuits namely Lorenz, Chen and Lu by monitoring the parameter α physically by an external switch or automatically by a MUX with a chaotic selection. Then these produced random patterns are used to construct CTRNG which can also be used in various embedded devices where cryptography is needed. The random chaos-based nature of the so-called unified chaotic system was examined and studied. In this paper Hardware Description Language (VHDL) is used to propose hardware RTL architecture which define the Euler mathematical determination technique. All Simulation and synthesis of design are completed on Xsim simulator and Vivado tools. Execution outcome are targeted on the Xilinx FPGA ZYBO board which embed Zynq device Z-7010. Then these random patterns were discovered for designing Chaos-based TRNG. Synthesis implementation results and statistical DIHARD tests validate that the proposed hardware strategy permits a finest cooperation between high security characteristics and embedded system necessities in terms of price, output and resources.

[20] Parvathy Chandra and Vishnu V. S.

“Circuit under Test Verification with MSIC Test Pattern Generator”

Authors presented on low-power test pattern generation method. Simulation outcomes presented that an MSIC pattern had the satisfactory features of a smaller number of variations and uniform distribution of sequences. A flexible test per scan is developed with the combination of Reconfigurable Johnson counter and seed generator. The MSIC sequence produced by the MSIC test pattern generator may contain repeated test patterns but switching activity which results in error is reduced. Thus the system improves the test efficiency. This method also lessens the power consumption during testing mode with smaller number of switching activities between test patterns. After generating the MSIC patterns, the validation process is conducted on the combinational logic circuit and the response is verified via two techniques, the reversible technique and the LUT technique. From the comparison between these two techniques, a conclusion can be made that, in application where area is not a major concern, LUT method can be used. In case of compact devices, where area is considered LUT method of testing is not used.

[21] Manish J Patel, Nehal Parmar, Vishwas Chaudhari

“Design and Implementation OF Logic-BIST Architecture for I2C Slave VLSI ASIC Design Using Verilog”

Authors presented the method of testing that includes driving control indicators from an inbuilt controller, producing pseudorandom sequences on the chip with the help of LFSR, passing the sequences via CUT, compress the output patterns from the device under test into signatures, then pass it through the output response analyzer (signature analyzer) to evaluate these signatures to compare with the fault free signature which is requirement and lastly give output of the pass or fail signal. Because the size of particular IC is fixed so with only the clock signal and the test mode signals the specific IC can be checked. The result of this is also simplified with only one signal indicative of the pass or fail status after the comparison of golden signature to output of signature analyzer.

[22] Xi Chen Fei Xiang Lili Zhang

“Optimal Design of Double-precision Chaotic Signal Generators Based on IEEE-754 Standard”

In this paper a finest method for the field programmable gate array (FPGA) double-precision chaotic signal generator based on IEEE-754 standard is proposed. The aim of this research is to execute a double-precision chaotic signal generator on FPGA. With the use of these timing controls to reuse designs and implementing parallel architectures, sense of balance is attained on each side of chip area and working speed. This technique has a certain range of generalization. It can be used to produce all the types of continuous chaotic signals; change a continuous design into digital design by Euler algorithm. Initially, split the complete design into different segments as black boxes according to different roles. Then connect all sequential controls using these modules according to the actual request.

[23]- Fernando Corinto, Oleh V. Krulikovskyi and Serhii D. Haliuk

“Memristor-based chaotic circuit for pseudo-random sequence generators”

In this paper, authors have explored chaotic circuits which are based on memristor for PR sequence generators. Specifically, the simplest memristor-based chaotic circuit is used. This circuit comprises of only three series devices: inductor, capacitor and memristor. The statistical properties of the produced PR patterns have been tested by means of the NIST test. The outcomes of these tests illustrate that this advanced generator we can use in real protected applications which use cryptography for security.

CHAPTER-3

BIST(BUILT-IN SELF-TEST)

BIST is a technique used in DFT that manually puts the testing logic design with the circuit under test (CUT). The basic BIST is the combination of 3 major hardware devices a TPG, an ORA, and a controller. The test vectors are generated by *test pattern generator* for the CUT which is under test. There are many techniques for generation of test vectors like deterministic pattern generator which uses ROM to store patterns, using counter, and using a shift register with XOR gate. And ORA is mainly comparator type circuits. ORA compresses and compare the test vectors to find accuracy of the CUT using mainly XOR gate. Controller block is also compulsory for activation of TPG and ORA to check and evaluate the output responses. [7]

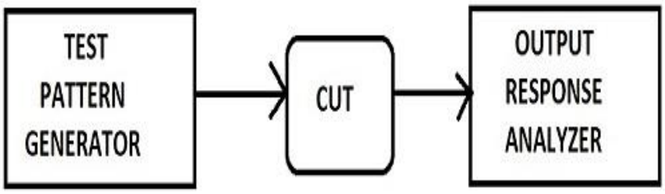


Fig 3.1: BIST basic structure

The connection lines between PI to MUX and between circuit response to can’t be checked by this BIST method. In ordinary operation mode, the CUT takes its inputs from one unit and accomplishes the task which is defined in that unit block. But in testing mode, which is controlled by controller a TPG gives test vectors to CUT, and it’s responses are checked by ORA.

There are some terminologies, which are used in testing defined as below:

Fault Coverage: The part of faults which can be found by the test vectors out of total existing faults.

Test set size: It is the test patterns generated by the TPG in total, and it is related to coverage of faults. Usually, if test set size is large that means high fault coverage.

Hardware overhead: This is extra requirement of hardware in BIST. High value of it is not acceptable in mostly real type microcontroller type systems.

Drawbacks of BIST:

- Because of test controller, pattern generator, response analyzer and testing of BIST hardware additional active area required.
- Because of input MUX at least one extra pin is required to activate operation of BIST.
- Because of BIST extra path delays are added.
- Increase in yield loss leading to increased chip area
- Time and effort of design rises.
- Complex hardware.

Advantages of using BIST:

- External test equipment which are costly can be easy to remove,
- Parasitic effects can be avoided which occur due to equipment connecting cables,
- Speed of testing is good,
- Vertical testability,
- High diagnostics resolution
- It can also examine from remote areas because of self-testing feature.

There are high number of testing problems that must be conveyed during the process of designing and development of a that product, but the final aim is to give superiority checking in a useful way. This aim has become highly problematic to attain as VLSI design and PCB component concentrations rise to the level that many industries report that costs of testing are often high as 55% of the total cost of product [9-10]. So, the best way to ensure that a product is testable is to involve Design for Testability (DFT) from the very launch, that is during the design phase of the product.

3.1 ARCHITECTURE OF BIST:

The basic requirement for BIST is a test pattern generator, a response analyzer, a circuit for test and a test controller. CUT is design which is under testing it takes input from the test pattern generator which generates the test vectors using some designs like Linear Feedback Shift Register LFSR, LFSR/SR, and then output of CUT goes to the output analyzer which can be MISR. The controller present in the circuit regulates both TPG and ORA by making necessary choices such as what should be the size of sequence which goes to circuit which is under the testing. The basic architecture of BIST as described above is shown in Fig 3.2

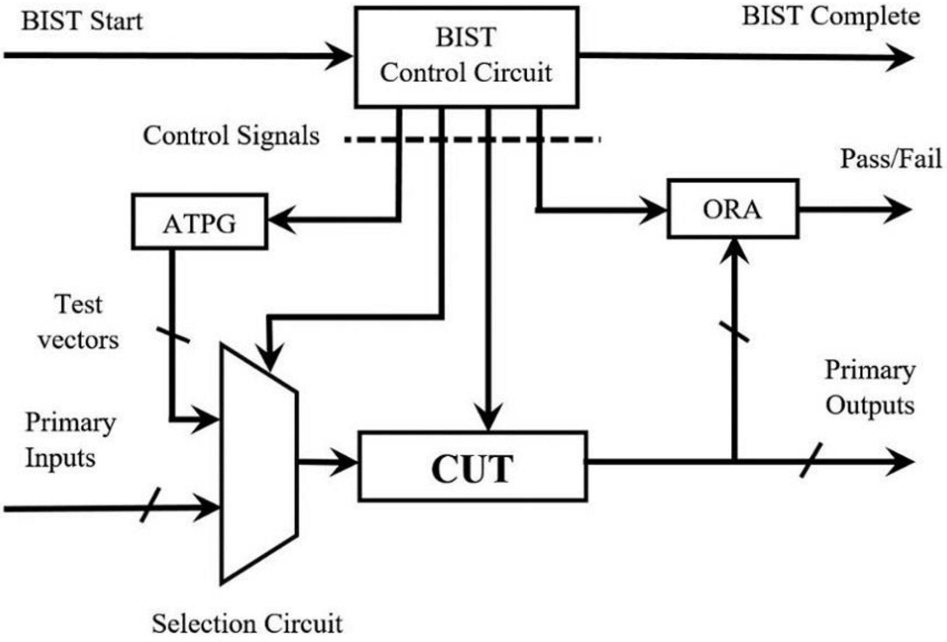


Fig 3.2: ARCHITECTURE of BIST [37]

This can be applied to non-concurrent and testing of the system with its logic and memory parts, which is also called LBIST and MBIST respectively. It may also be considered for low fault latency periodic testing. This needs including a process of testing into the design under test that assure the recognition of all errors in a fixed time.

The test generator and response analyzer are usually designed to guarantee coverage of all the specific faults, minimum extra hardware required, and test set size should also be reasonable.

TPG and ORA are generally executed by simple circuits like counters and shift registers, especially linear-feedback shift registers (LFSRs). These are shift registers designed with standard flip-flops, and feedback with the output of some flip-flop with modulo-2 to the input of shift register. LFSR is also used for generation of pseudo-*random* tests. An LFSR is also used to form ORA to compare output of CUT.

The response pattern sequences of LFSR are interrelated to each other because these are shifted with time and repeat themself. So, it decreases the efficiency of detection of fault by LFSR. In order to de-correlate these patterns, EX-OR gate network is used. The output pattern of the circuit of testing is usually compressed by ORA technique into a signature of small length, that will be compared to pre stored golden signature we have, to know circuit nature.

3.2 TEST PATTERN GENERATOR

Test pattern generation techniques of BIST can be classified into four categories:

I. Exhaustive Patterns

II. Pseudo-exhaustive Patterns

III. Pseudo-random Patterns

IV. Deterministic or stored Pattern

3.2.1 Exhaustive Pattern Generator:

This technique of BIST removes the process of test pattern generation and it's coverage of fault is very high. For the testing of an n number of inputs of combinational logic, it needs to apply all combinations of n which is equal to 2^n test vectors to the CUT. For high clock speeds, this technique is impractical to use for large number of n. It is practically useful for only a smaller number of inputs.

Exhaustive Patterns generator:

1. Binary Counter
2. Complete LFSR

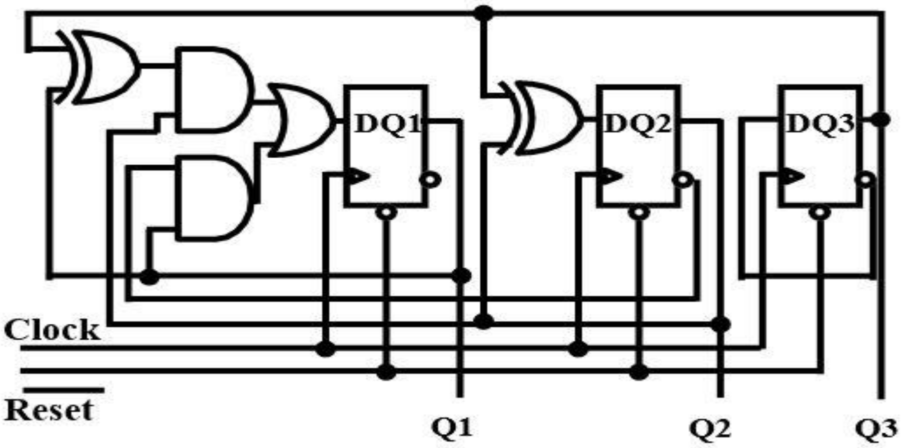


Fig 3.3: Exhaustive Patterns Generator [31]

3.2.2 Pseudo-exhaustive Pattern Generator:

It is the modified form of exhaustive testing. In this pattern generation technique, the main circuit is divided into some smaller sub-circuits. After that every probable test vector is applied to each sub-circuit exhaustively. The foremost motive of this technique is to maintain fault coverage equal to exhaustive testing and, as well as reducing the testing time with this. Since almost complete coverage of fault is certain, so there is no requirement of simulation of fault for both exhaustive testing and pseudo-exhaustive testing.

Extra effort is required in designing of the main circuits into small testable pseudo-exhaustive circuits, which increases hardware requirements also and will degrade the overall performance. Additionally, there is a major issue regarding the distribution of test patterns and test responses. *Cone Segmentation* is one technique of circuit partitioning for this pseudo-exhaustive pattern testing as shown below in Figure 3.4. In this technique, a cone type structure is defined as the Fan-In of an output of circuit.

Let’s assume size of the main cone is N as shown below, this circuit is divided into two parts or sub-circuits based on the Cone segmentation. For cone1 the primary output is *h* while for cone2 the primary output is *f*. So the total number of test vectors required for exhaustive pattern method of cone1 and cone2 is equal to (2^5 +2^5) = 64 and for exhaustive test total number of test pattern required with 8 primary inputs are 28 = 256.

There are some other techniques also present for this circuit partition pattern generators like:

1) Syndrome driver counter
2) Constant weight counter
3) LFSR/EX-OR

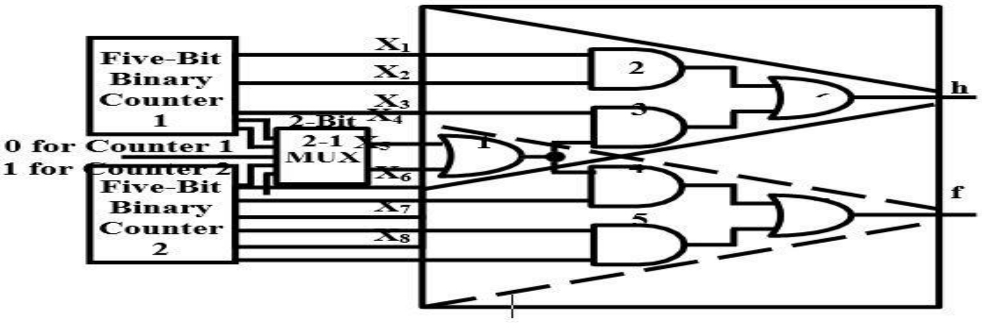


Fig 3.4: Pseudo-exhaustive Pattern Generator[31]

3.2.3 Pseudo-Random Pattern Generator:

Any sequence of zeroes and ones is known as a PRBS when the bits of that sequence seem to be non-periodic in the normal behavior, but actually they are repeating in other way. Pseudo random numbers are sufficiently random in nature for replacing true random numbers. For generating these PR patterns LFSR generator is widely used, because it is easy to design and implement. Usually, this technique needs high number of test vectors than stored TPG, but less than that of exhaustive testing. In other hand this method may require a testing time higher than other methods and require estimation of fault coverage by simulation of fault in contrast with other methods. This pattern testing though, has the probability of less requirement of hardware as well as less performance overhead and less struggle to design than the previous strategies. Drawback of this pattern generation technique as far as random pattern resistant fault detection is concerned arises, because each bit has probability of 0.5 of being either 0 or 1.

In order to measure effectiveness of this testing technique there are thee issues which needs to be considered:

1) The number of test patterns determination

2) Fault coverage evaluation

3) And random pattern resistant faults detection.

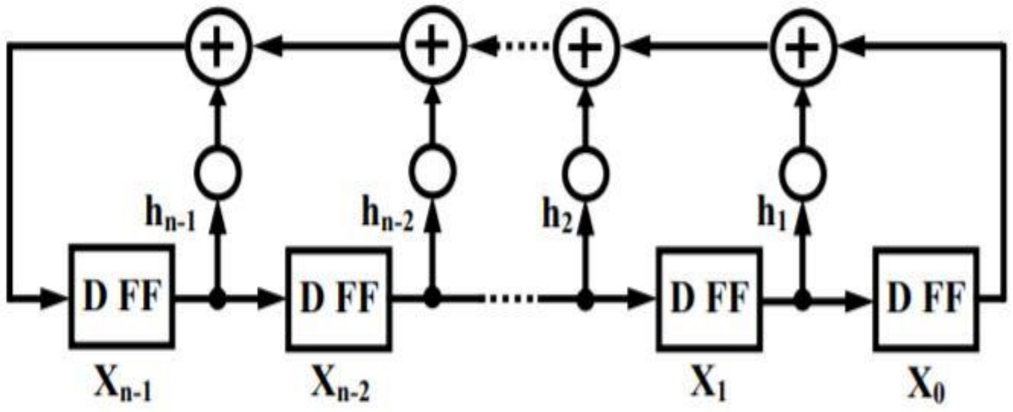


Fig 3.5: Pseudorandom Pattern Generator [31]

3.2.4 Stored pattern Generator:

This is the traditional technique which is used for generation of test patterns and can be applied to CUT in normal mode of BIST. Test vectors and the equivalent patterns of output set are placed in a read only memory on the IC. When activation of testing is started, then stored input vectors for testing are applied to the input of CUT and comparison will occur between their output vectors and their equivalent stored output patterns. If it is not a match then fault is present in the circuit. Even though fault coverage of this strategy is excellent, but it is inadequate for some applications because of extra hardware requirement.

The categories of test vectors defined before are not totally mutually exclusive. Like, pseudo- random test vectors can behave like pseudo-exhaustive and visa-versa. Many BIST applications may use several types of test vectors; PRN test vectors can be used in combination with stored type test vectors.

Comparison among Test Generation Methods:

Applying a Built-in self-test approach, the key problems are fault coverage, extra hardware requirement, extra testing time, and more effort for design. All problems of BIST we discussed are in complex relationship. Table 3.1 gives the brief about the features of the test generation methods we discussed before on the basis of the some of the above discussed issues.

Table 3.1 Comparison of different test pattern generation methods:

Test Generation Methods	**Fault Coverage**	**Hardware overhead**	**Design Effort**
Exhaustive type	High	Low	Small
Pseudo-Exhaustive type	High	High	Large
Pseudo-Random type	Low	Low	Small

Stored Pattern type	High	High	Large

3.3 OUTPUT RESPONSE ANALYSIS:

Output Response Analyzing techniques which compared compressed type data response with a defined golden or fault-free signature response are classified as shown below:

I. Transition Count

II. Signature Analysis

III. Syndrome Testing

3.3.1 Transition Count:

This technique is defined as total number of counts of transitions of either 1 to 0 or 0 to 1 in output response corresponding to the given input test vector. This is relatively same to another technique called ones count. Figure 3.6 shows an architecture of transition count technique. It has simple hardware consisting of D-flip flop with E-XOR gate to detect a transition of 1 to 0 or 0 to 1 and has one counter also to count number of transitions. Aliasing probability of it is less than ones count technique. Test sequences cannot be permuted in this because permutation of input patterns will modify the number of transitions. And, on rearrange the order of test sequence we can reduce or increase the changes, hence, to reduce the masking possibility. It only records transition count instead of output sequence.

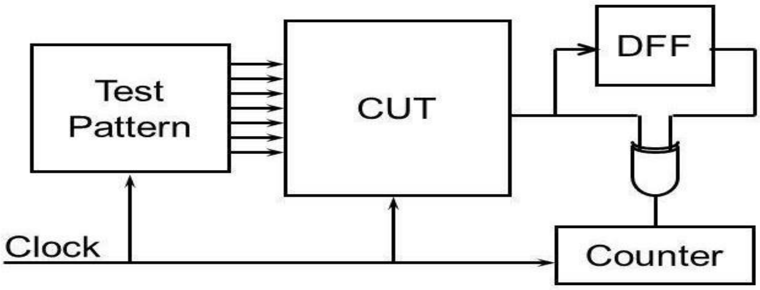


Fig 3.6: Transition count structure [31]

Output Pattern Compression/Compaction Techniques of BIST:

In BIST, large number of bit sequences in DUT responses are fed to output response analyzer. A circuit having multiple outputs may produce the need for a large number of random patterns, which is turn is practically infeasible to manage.

Therefore, it is essential to compress this huge number of circuit output responses to a controllable size so it can be easily store on the integrated circuit. This very long test response compression is done by response analyzers in BIST. The compressed pattern is called a signature. This signature is then verified from the already stored golden signature. If our output value matches to that fault free golden signature, then CUT is observed good circuit otherwise it will fail.[29]

Compression: It is used to suppress the multiple bit stream response in to smaller size and this is reversible process. It is hard to implement in hard ware.

Compaction: It is also used to suppress the multiple bit stream response in to smaller size and this is an irreversible (lossy) process.

Some techniques in brief:

a) For computing parity in bit stream is parity compression technique.

b) To counts the number of 1's in the sequence is syndrome testing.

c) Counting of the total number of toggling between 1 to 0 or 0 to 1 is transition count.

d) To count CRC check word on the sequence, Cyclic Redundancy Check (CRC) is used.

Aliasing: Multiple bit sequence changing into a lesser number of bit sequence is called compression as we discussed earlier. It is like mapping of many-to-one. Therefore, in input bit stream errors can occur. So, a faulty output can also have the value which may be same as the golden signature. If this happens, then the circuit will be reported same as the fault-free circuit. This kind of situation is called aliasing/masking. This is the probability that a faulty output will be treated as fault-free response. For example,

Let's assume that there are 2^m number of input bits, 2^k response signatures and 2^{m-k} number of PI sequence which map to that response signature. Then calculation for possibility of masking is,

$$P(A) = \frac{\text{Number of erroneos input patterns map into golden value}}{\text{Number of faulty input patterns}}$$

$$= \frac{2^{m-k}-1}{2^m-1} \qquad (1)$$

$$\approx \frac{2^{m-k}}{2^m} \quad \text{for high value of m}$$

$$= \frac{1}{2^k} \qquad (2)$$

3.3.2 Signature analysis:

This technique was founded by HP Ltd. It notices errors in bit patterns caused by hardware faults and its compact long data sequence design response into golden signature. Then the actual response which is found during testing is compared to the golden signature. Lastly, after this comparison we can find whether the circuit is faulty or fault free.

If we have n stages of signature generator then 2^n signatures will be generated. And if the length of the input sequence is m then 2^m input sequences map into 2^n signatures. Only one out of all input sequences will be error free and generate correct signature.

Probability P that an input pattern deteriorated into another having same signature can be calculated on the basis of assumption that any of the possible input patterns may be fault-free or faulty:

$$P = \frac{2^{m-n}-1}{2^m-1} \qquad (3)$$

For m>>n above expression will be:

$$P = \frac{1}{2^n} \qquad (4)$$

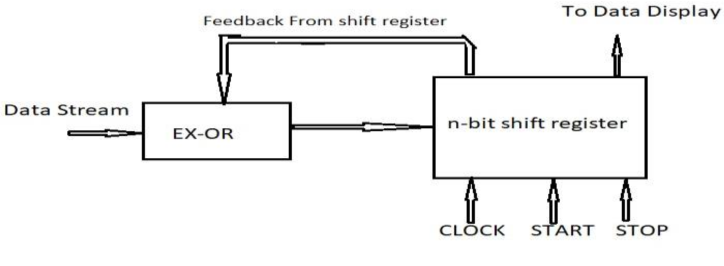


Fig 3.7: Signature Analyzer

Response compaction using LFSR in Signature Analyzer:

- For response compactness this technique uses CRCC generator using LFSR.
- Data patterns from circuit’s primary outputs to be compressed to reduce the number of polynomial coefficients.
- Primary output of polynomial will be divided by its characteristic polynomial and using CRCC and its remainder will be left in LFSR. And initial value of LFSR should also be stored to generate numbers before testing.
- Once the signature is generated, it is compared to the already stored golden signature for testing to be completed.

MISR:

An MISR will be used as ORA in BIST. It performs same function as the signature analyzer. This testing technique is an important technique because it generates different response values for faulty and fault free designs and then compare these values to each other to know if the circuit is good or not.

For TPG if Pseudo-Random Pattern Generator is used then it will generate the required sequence which is provided as primary input pattern for the DUT. The output value of this DUT is given to ORA which is MISR here. This will suppress all output values into one LFSR because if we want less hardware requirement then this compression of bit patterns is necessary. There are many ways to form MISR by connection of the inputs of LFSRs. Since the EX-OR function is linear and associative in nature, $(A\oplus B) \oplus C = A\oplus (B\oplus C)$, so this is preferably used. [8]

- The Traditional LFSR response compacter has a drawback that there is an excess requirement of hardware.

- LFSR follows *superposition principle* and it is linear also so by using LFSR.
- All output patterns are placed over to single LFSR. EX-OR addition of remainders of division of polynomials of every PO by the characteristic polynomial is the final value of the remainder.

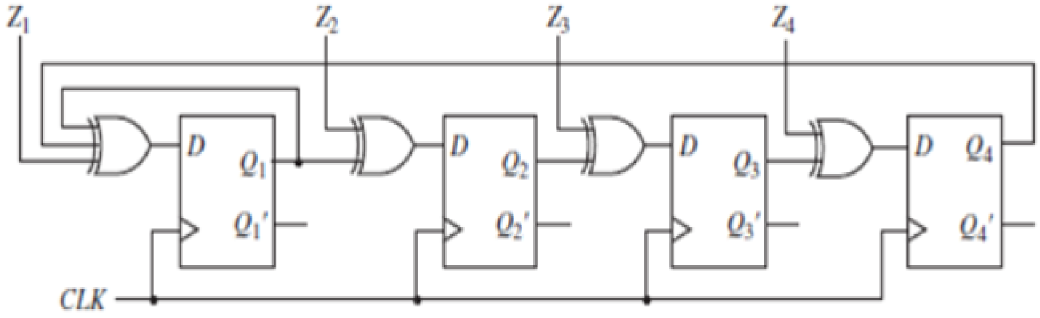


Fig 3.8: MISR

3.3.3 Syndrome Checking:

This testing technique is defined by way of the possibility of 1’s in the output response of design under test. The syndrome of three-input AND gate is *1/8* and *7/8* of a three-input OR gate because syndrome is functional property. Figure 3.9 shows an architecture of the syndrome count. The highly notable characteristic of this technique is that the final response value is not dependent on the execution. It is completely determined by its function of the circuit.

The syndrome of any Boolean function can be found by $S = K / 2^n$, where n is the total number of input lines and K is the total number of min-terms realized by function.

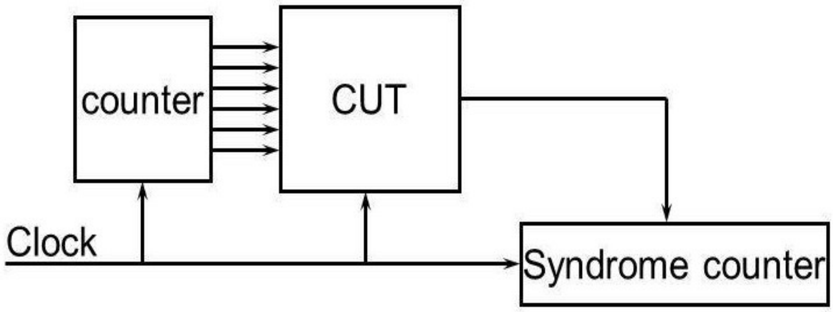


Fig 3.9: Syndrome Testing

3.4 CUT:

This CUT is the main design which is used in BIST for fault detection purpose. It can be die, wafer, packaged chip and any simple circuit which is being tested. It is connected to LFSR at input side so input pattern will be fed in to it by TPG and at output side ORA is connected which gives final output of this CUT. Whenever requirement occurs for testing then this circuit will connect to main integrated circuit and start the testing.

3.5 BIST CONTROLLER

According to the specifications of the CUT of testing controller controls the TPG and the ORA. The required pattern length will be present in the TPG and ORA areas. So essential patterns value will be called by this regulator and then required number of primary inputs and response patterns of design under test will be provided.

3.6 BIST APPLICATIONS:

Manufactures are increasingly employing BIST in testing of real products. Examples of such applications are given to illustrate the use of BIST in semiconductor industry, safe and secure data communication, and computer industries etc.

It is also used in some other application for example;

- In automotive electronics devices; like sensors of antilock brake system in vehicles.
- In aviation; to minimize the time on ground needed for repair and increase the safety level.
- In medical field; it is used in safety-critical devices for periodic testing to ensure that device is safe to use.
- In military; this is used in missile (in U.S.), to control testing which reduces the number of cable and weight of missile.
- In electronic devices; in IC and in embedded system testing.

CHAPTER-4

DESIGN AND METHODOLOGY

In this project chaotic circuit is used as TPG for random pattern generator instead of LFSR in general BIST. When implementing BIST, it is essential to review the test vector generator. Because the test vector generator is one of the most significant apparatuses in the BIST. The fault coverage of numerous fault models is directly dependent on the input vectors which are used generated by ATPG and gave to the circuit which is under the testing. Finally, we implement the ATPG which improve security of data, that is chaotic circuit.

4.1 RANDOM PATTERNS GENERATOR:

4.1.1 Chaotic circuit:

Chua's circuit is a normal circuit that shows typical random nature. In other words, this circuit is like "non-periodic oscillator"; Because oscillating waveform generated by this, which is different from a regular electronic oscillator, due to there is no “repetition’’. Construct of this circuit is simple that’s why this circuit became a universal real-world example for a chaotic structure, leading some to declare it "a paradigm for chaos"[20]

The fast growth in technical knowledge has directed to a enormous development in power of computation in latest expertise. Use of deterministic chaos theory in security of data is a promising area of research. Chaos is a long-term changeable nature of deterministic dynamical device which is sensitive to initial conditions

[19]. With the help of these properties, new algorithms can be perceived for data security.

This chaotic structure has many properties like ergodicity, insufficient certainty, non-periodic, wide range communication, sensitivity for initial values, complicated structure & deterministic dynamics etc. A wide range communication with secure signals are main ability of this chaos theory based systems. Furthermore, this encrypted system for communication can easily realize on the part of IC like very simple design. [21].

The contributions of chaos theory give new solutions by analyzing the appropriate data and through the use of current advances in different areas, especially in chaotic simulation methods and techniques.

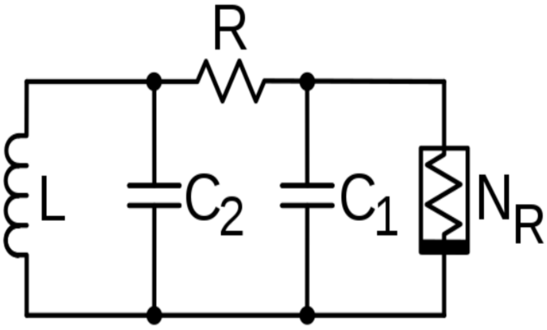


Fig 4.1: - Chua's circuit [28]

Characteristic equations of this circuits Fig 4.1 are:

$$\dot{x} = \alpha[y - x - f(x)] \qquad (5)$$

$$\dot{y} = [x \dot{-} y + z] \qquad (6)$$

$$\dot{z} = -\beta y \qquad (7)$$

$$f(x) = m_1 * x + (m_0 - m_1)/2(|x + 1| - |x - 1|) \qquad (8)$$

For Verilog code of this circuit, I have used Forward-Euler numerical model. From this model we evaluate discretized equations, as follow [33]:

$$X[k+1] \ = \ X[k] + h\alpha(Y[k] - X[k] - f(X[k])) \qquad (9)$$

$$Y[k+1] = Y[k] + h(X[k] - Y[k] + Z[k]) \qquad (10)$$

$$Z[k+1] = Z[k] - h\beta Y[k] \qquad (11)$$

From these equations, flow diagram can be derived which will help in HDL coding. So there are some flow diagrams for derivatives of X, Y and Z as follows:

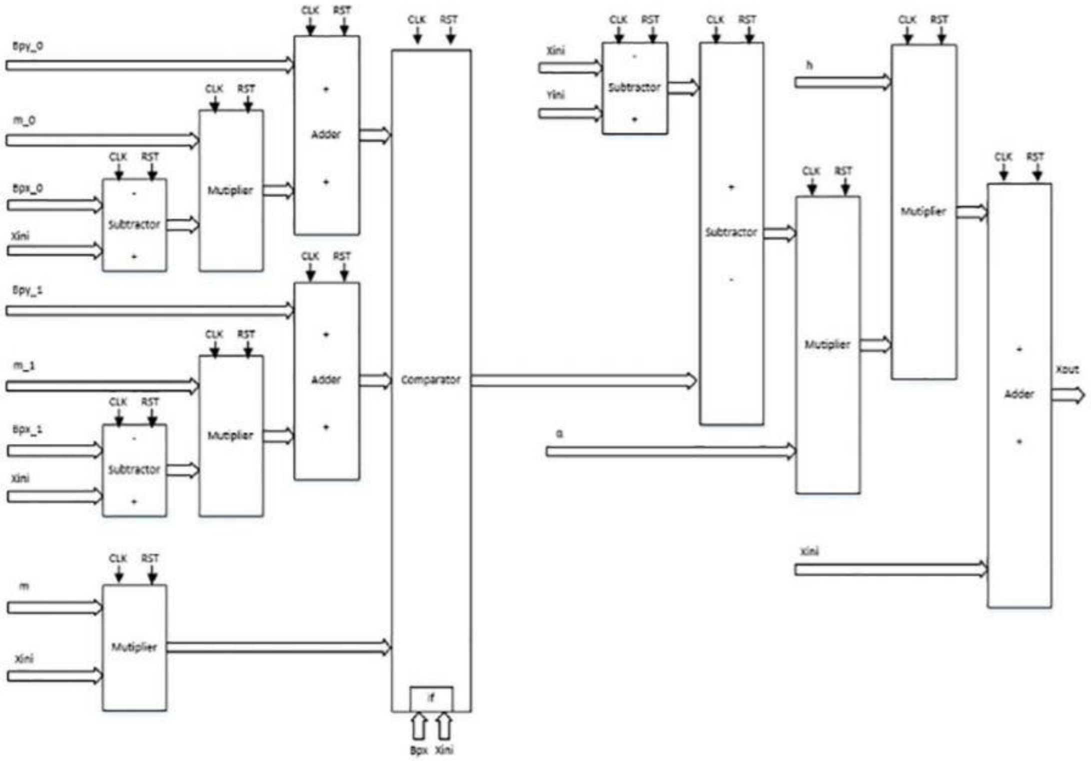


Fig 4.2: Flow Diagram for X[K+1] value [33]

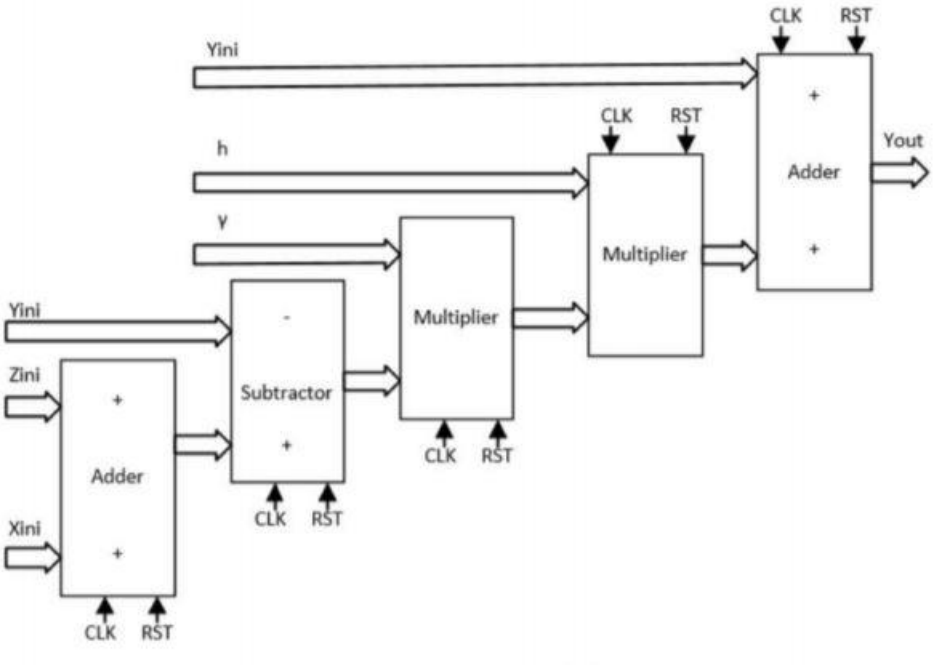


Fig 4.3: Flow Diagram for Y[K+1] value [33]

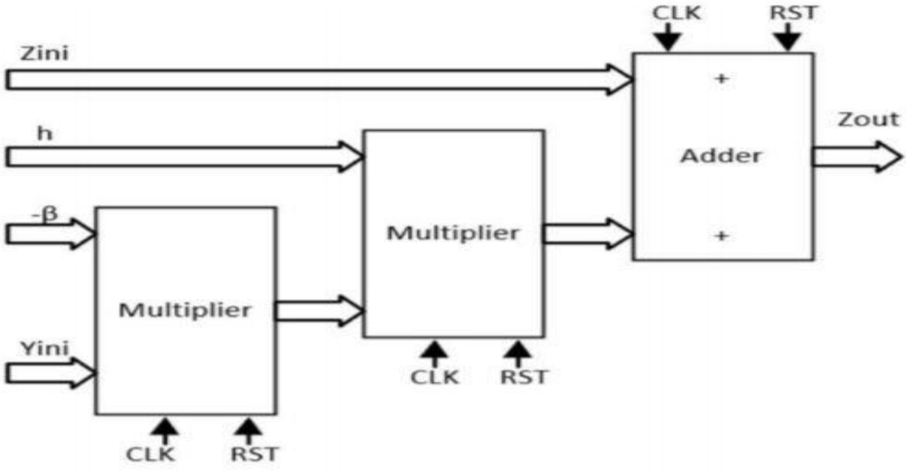


Fig 4.4: Flow Diagram for Z[K+1] value [33]

4.2 MISR AS ORA:

As we discussed above, response analyzer reduces the higher response rate sequence from the design under testing in lower response signature and after this it goes for comparing and then generate the result in one bit - it is either pass or fail like indicator.

This compaction is completed primarily for reducing timing to compare every bit from design which is under test with fault free signature we have. The final output response which indicates whether circuit is faulty or fault-free is necessary. Here I am using MISR for ORA for implementation of BIST.

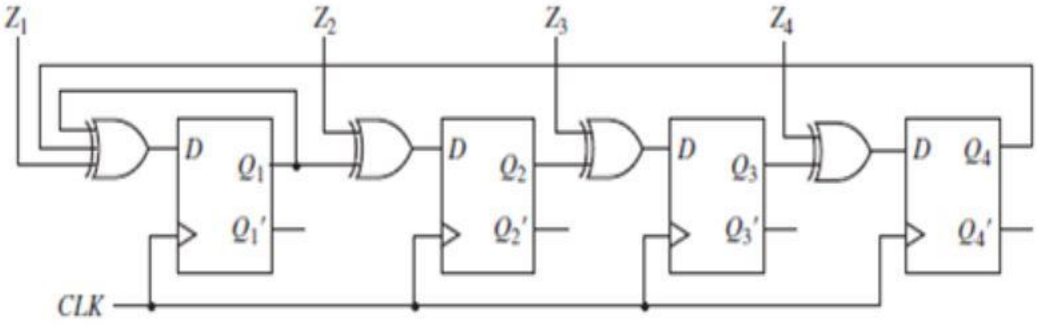


Fig 4.5: MISR as ORA

CHAPTER-5

SIMULATION & RESULTS

RTLs of any circuit helps to know the basic structure of the design, this also helps in knowing the number of basic blocks used for that specific execution.

Alternative characteristic of design is waveform related to timing analysis check, where function of circuit is checked and concluded to the correctness of the applied design. We study in VLSI design that the switching or the dynamic power consumption has the major percentage of the total power dissipation of a system. So, we can also find about power dissipation by this if required.

5.1 RTL ANALYSIS:

a) RTL of derivative of Y is shown follow:

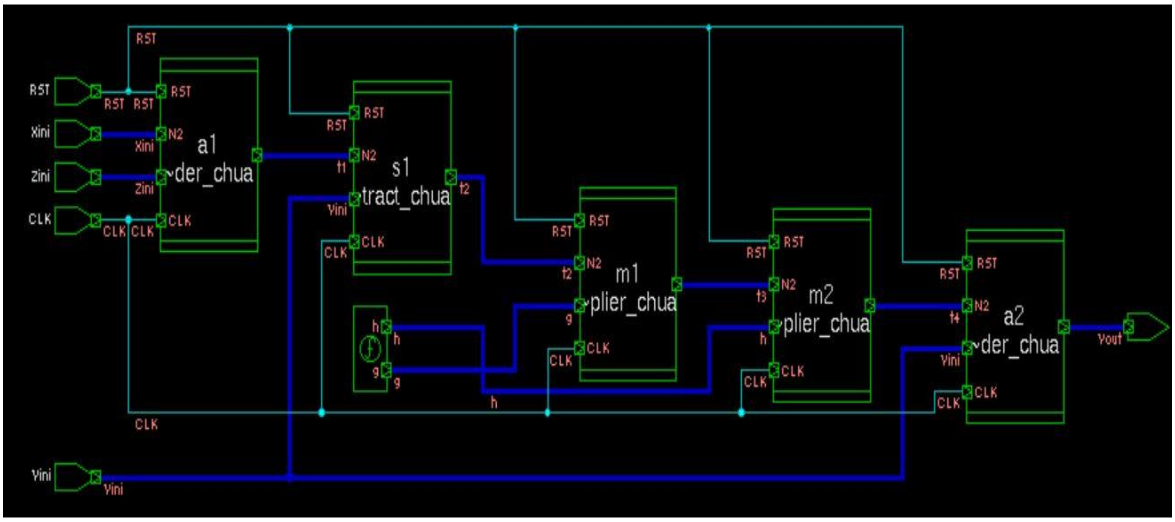


Fig 5.1: Schematic of dy/dt

b) RTL of derivative of X is shown follow:

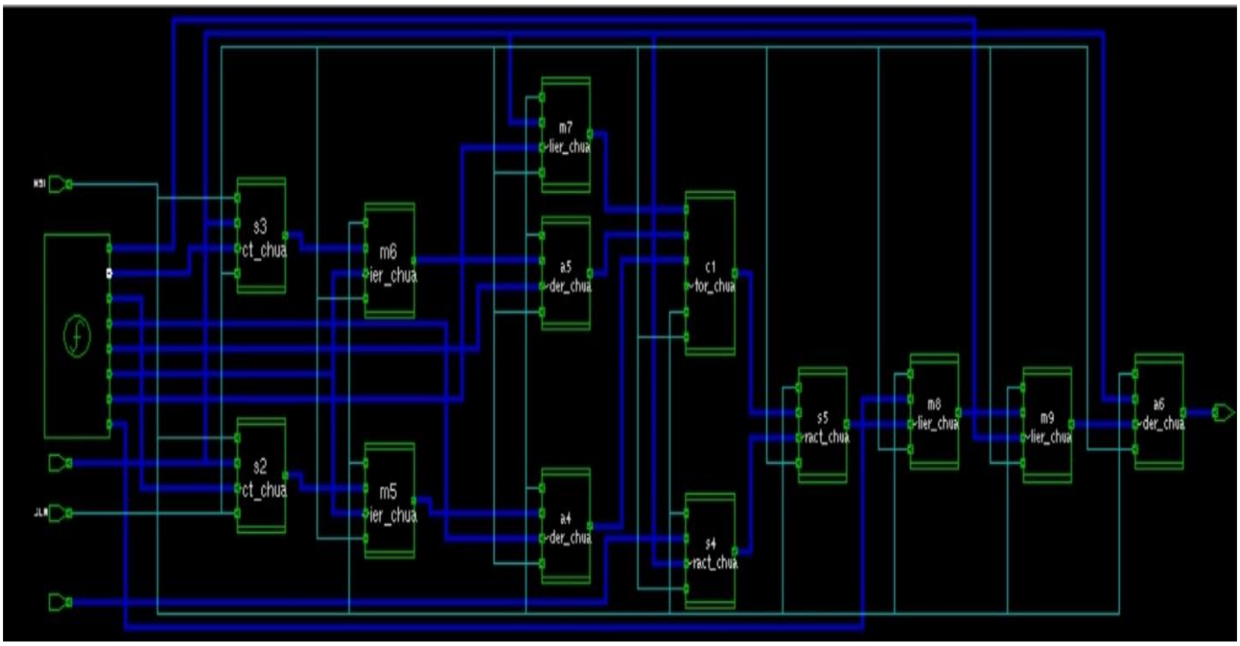


Fig 5.2: schematic of dx/dt

c) RTL of derivative of Z is shown follow:

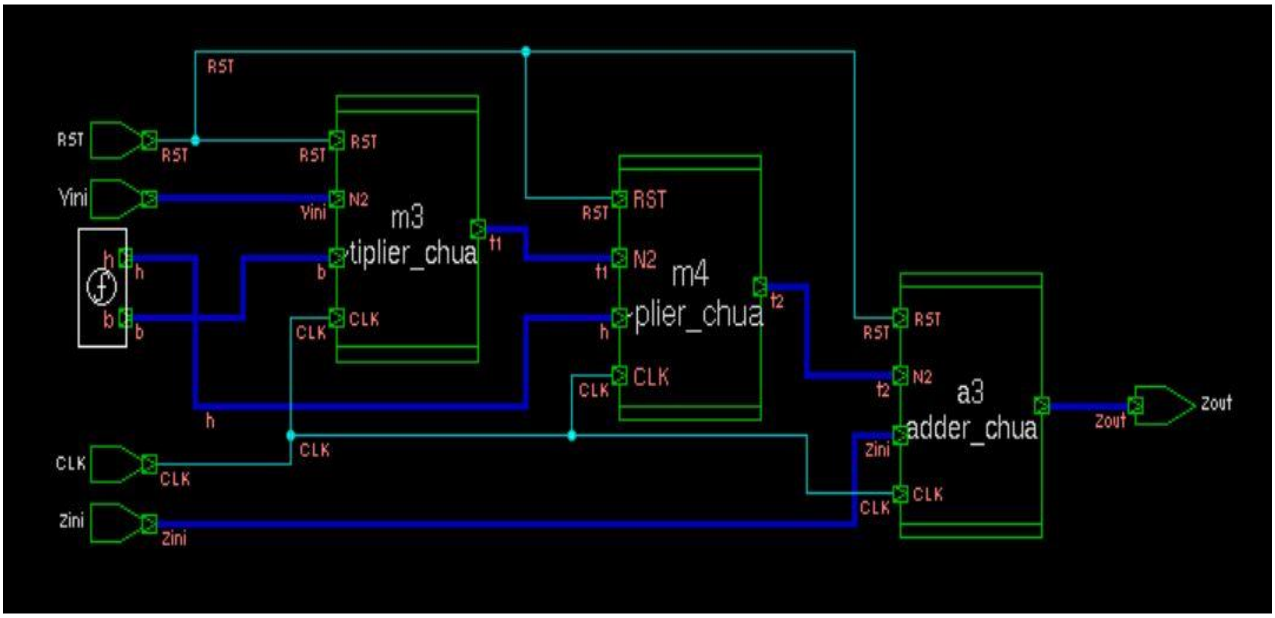


Fig 5.3: Schematic of dz/dt

For a test CUT, a fundamental circuit 10-bit RC Adder were analyzed. RTL for the proposed architecture with both ripple carry adder is shown follow:

d) RTL of circuit under test :

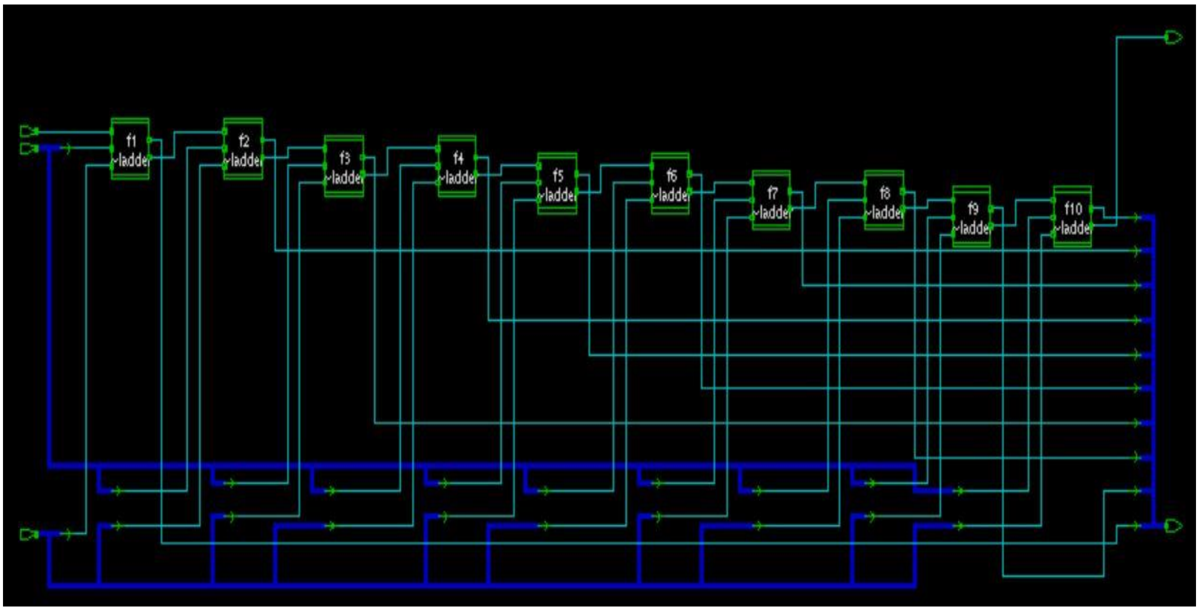


Fig 5.4: Schematic of ripple carry adder as CUT

e) RTL of MISR as ORA is shown follow:

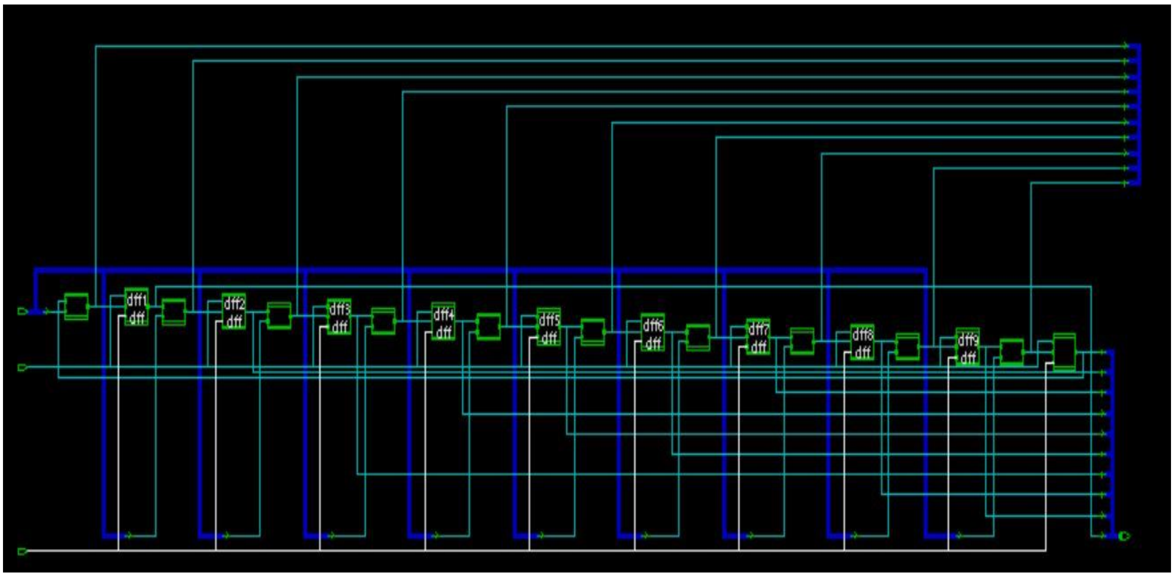


Fig 5.5: Schematic of MISR as ORA

f) RTL of LFSR as pseudo-random TPG is shown follow:

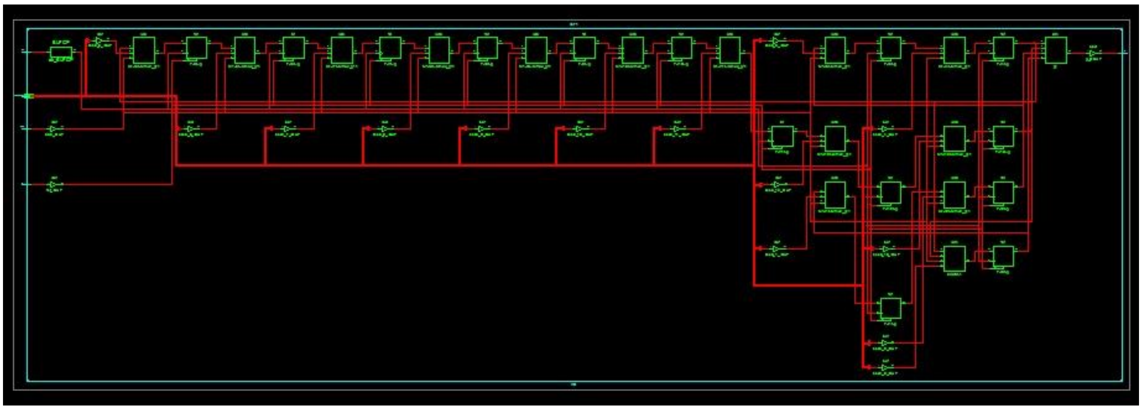
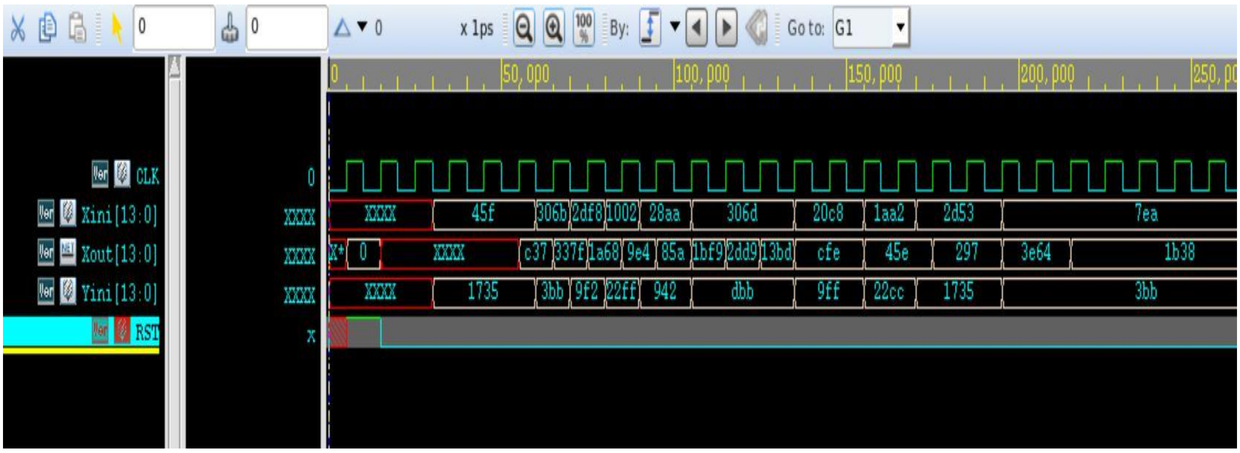


Fig 5.6: Schematic of LFSR as TPG

5.2 TIMING RESPONSE:

a) Timing response of Xout which is derivative of dx in normal and in analog waveform:



0
0
x 1ps
By:
Go to: G1
0
50,000
100,000
150,000
200,000
250,000
CLK
Xini[13:0]
Xout[13:0]
Yini[13:0]
RST
0
XXXX
XXXX
XXXX
x
XXXX 45f 306b 2df8 1002 28aa 306d 20c8 1aa2 2d53 7ea
X* 0 XXXX c37 337f 1a68 9e4 85a 1bf9 2dd9 13bd cfe 45e 297 3e64 1b38
XXXX 1735 3bb 9f2 22ff 942 dbb 9ff 22cc 1735 3bb

Fig 5.7(a): Timing waveform of dx/dt

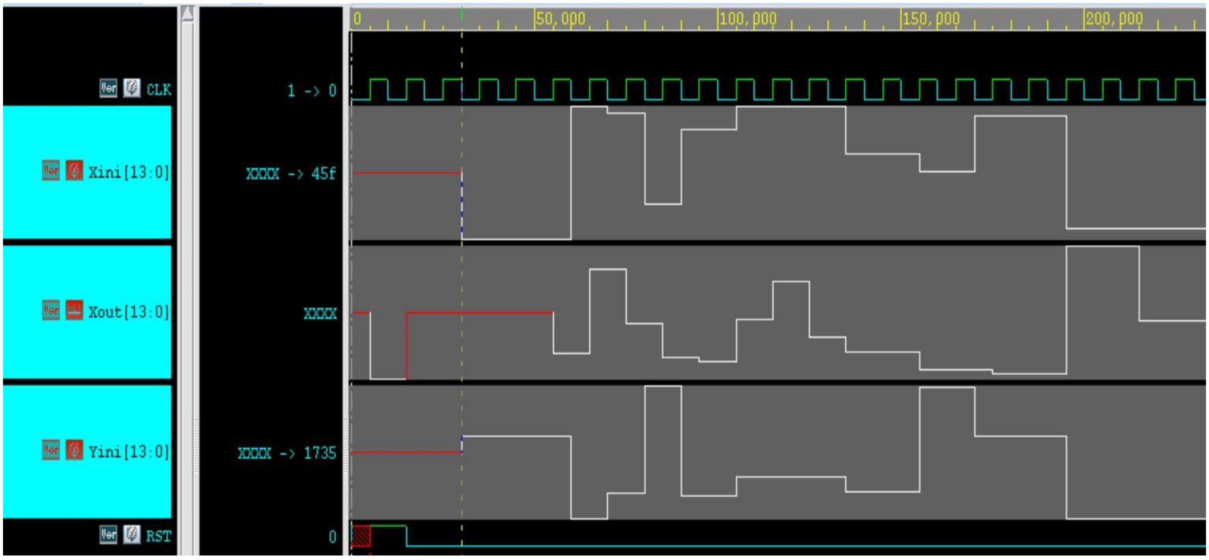


Fig 5.7(b): Timing waveform of dx/dt in analog form

b) Timing response of Yout which is derivative of dy in normal and in analog waveform:

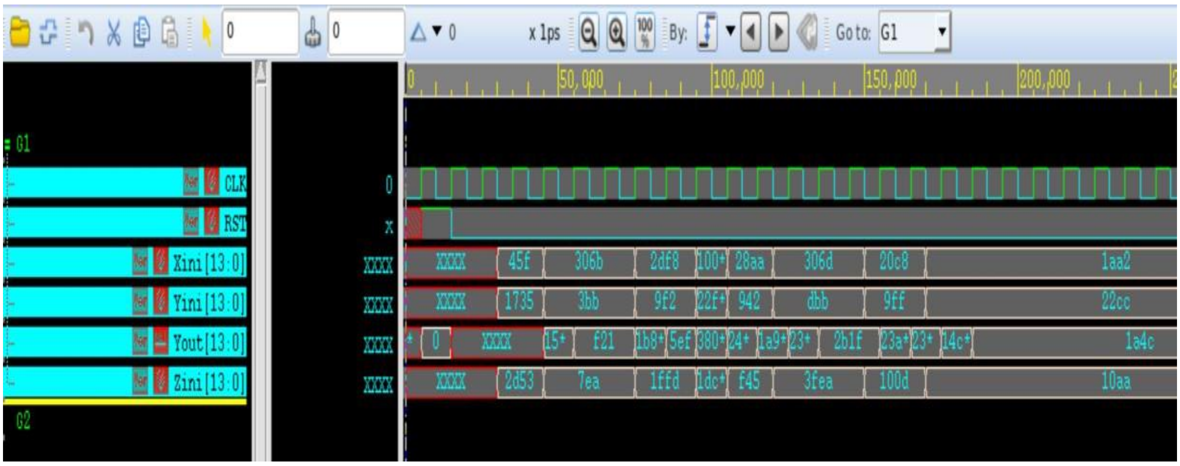


Fig 5.8(a): Timing waveform of dy/dt

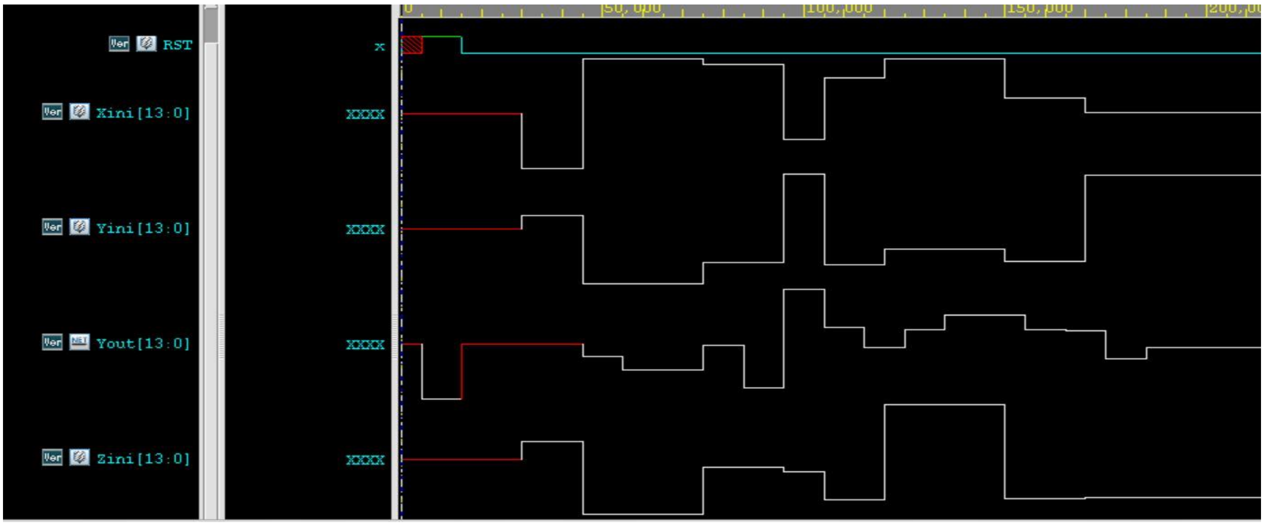


Fig 5.8(b): Timing waveform of dy/dt in analog form

c) Timing response of Xout which is derivative of dx in normal and in analog waveform:

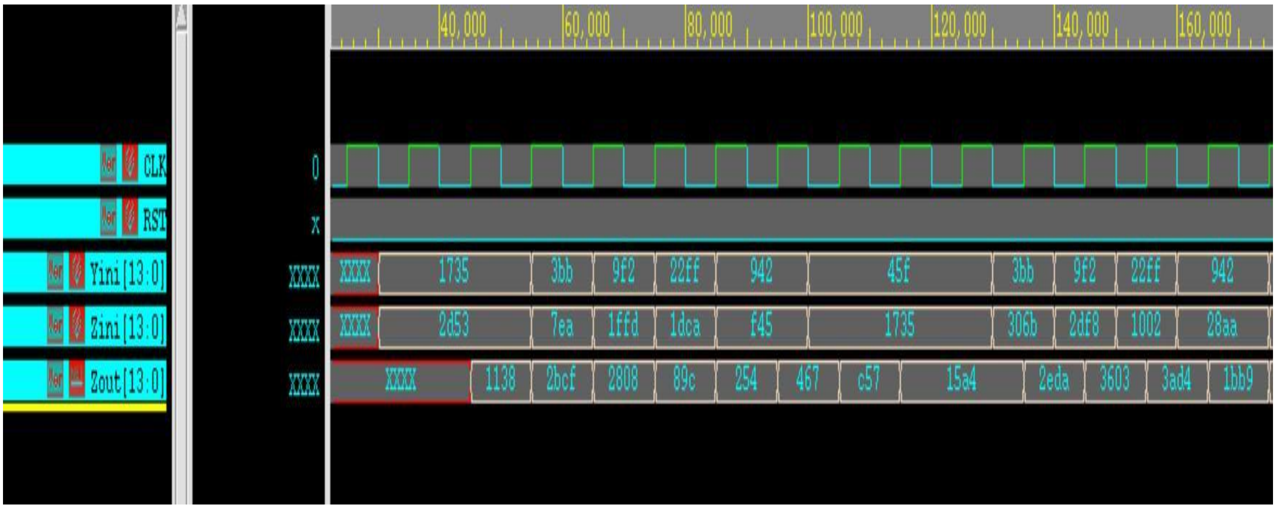

CLK
RST
Yini[13:0]
Zini[13:0]
Zout[13:0]
0
x
XXXX
XXXX
XXXX
40,000
60,000
80,000
100,000
120,000
140,000
160,000
XXXX 1735 3bb 9f2 22ff 942 45f 3bb 9f2 22ff 942
XXXX 2d53 7ea 1ffd 1dca f45 1735 306b 2df8 1002 28aa
XXXX 1138 2bcf 2808 89c 254 467 c57 15a4 2eda 3603 3ad4 1bb9

Fig 5.9(a): Timing waveform of dz/dt

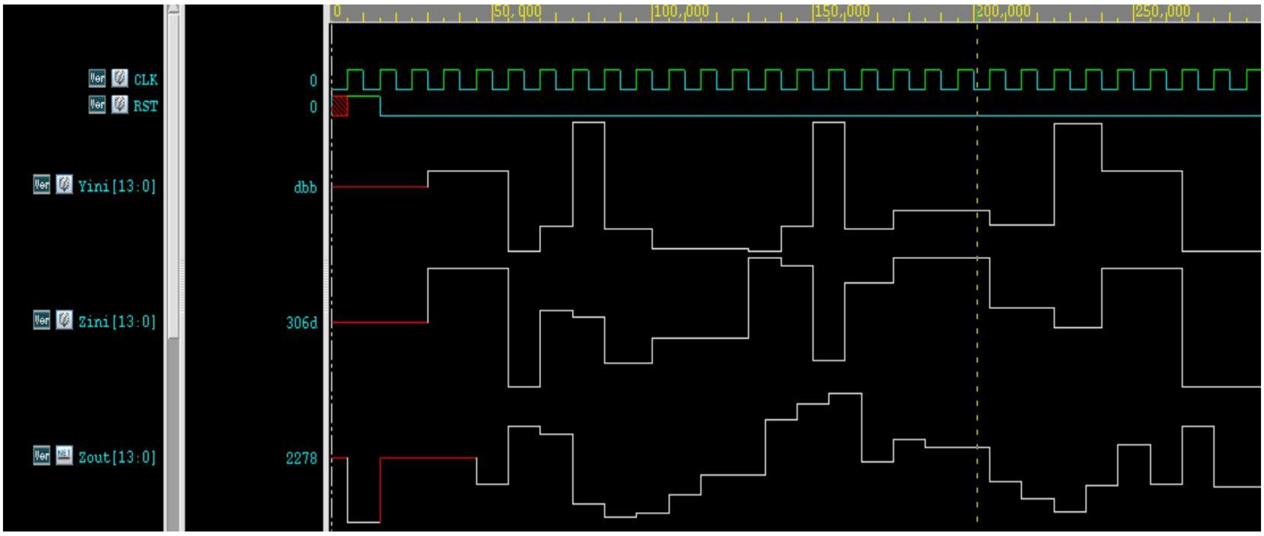


Fig 5.9(b): Timing waveform of dz/dt in analog form

d) Timing Response of circuit under test :

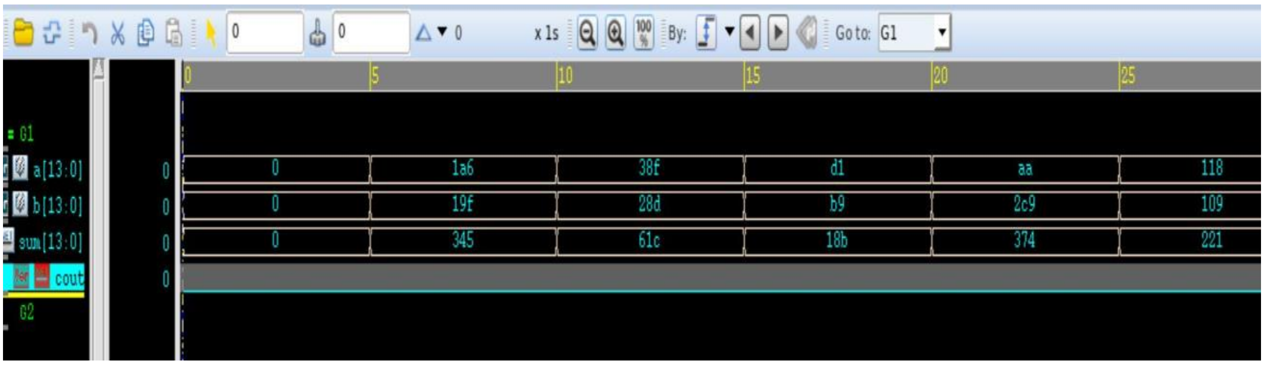


Fig 5.10: Timing waveform CUT

e) Timing response of MISR :

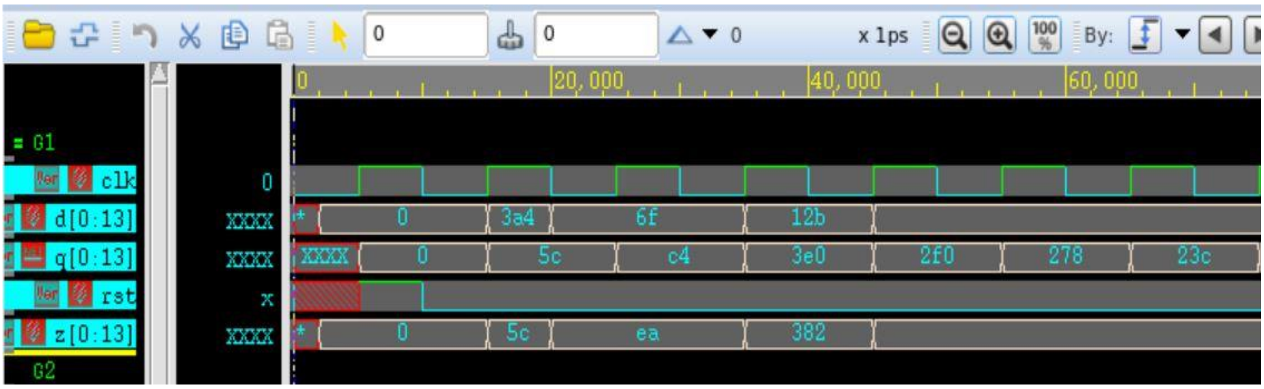


Fig 5.11(a): Timing waveform MISR

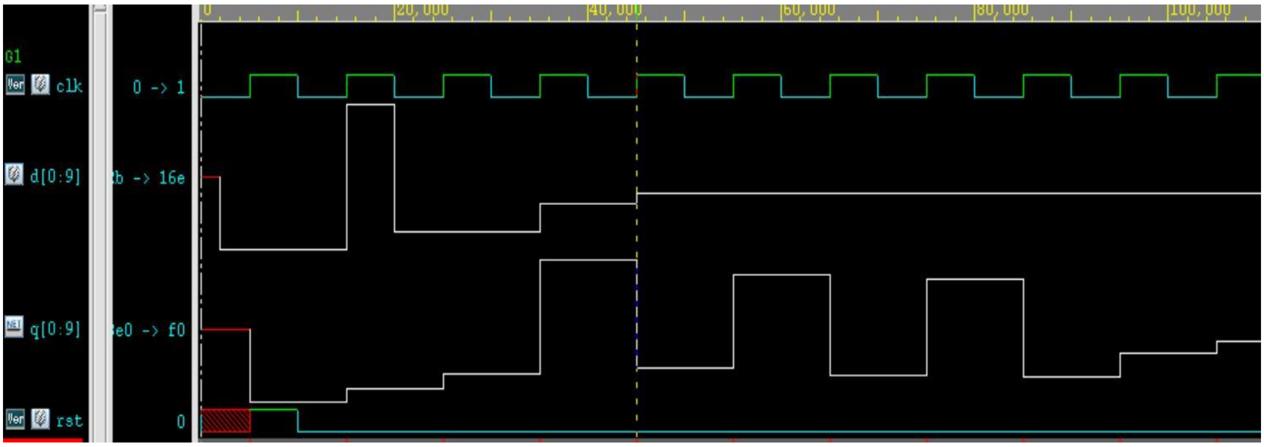


Fig 5.11(b): Timing waveform MISR in analog form

Timing Response of Chua's circuit in MATLAB:

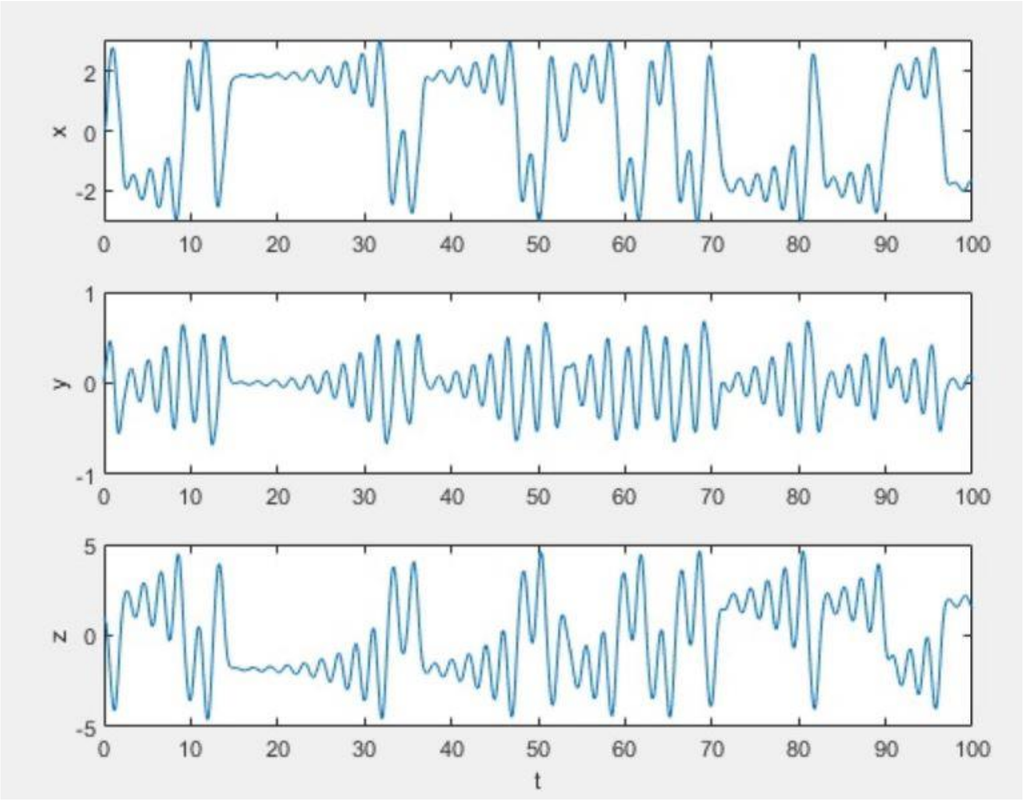


Fig 5.12: Timing response of x, y and z

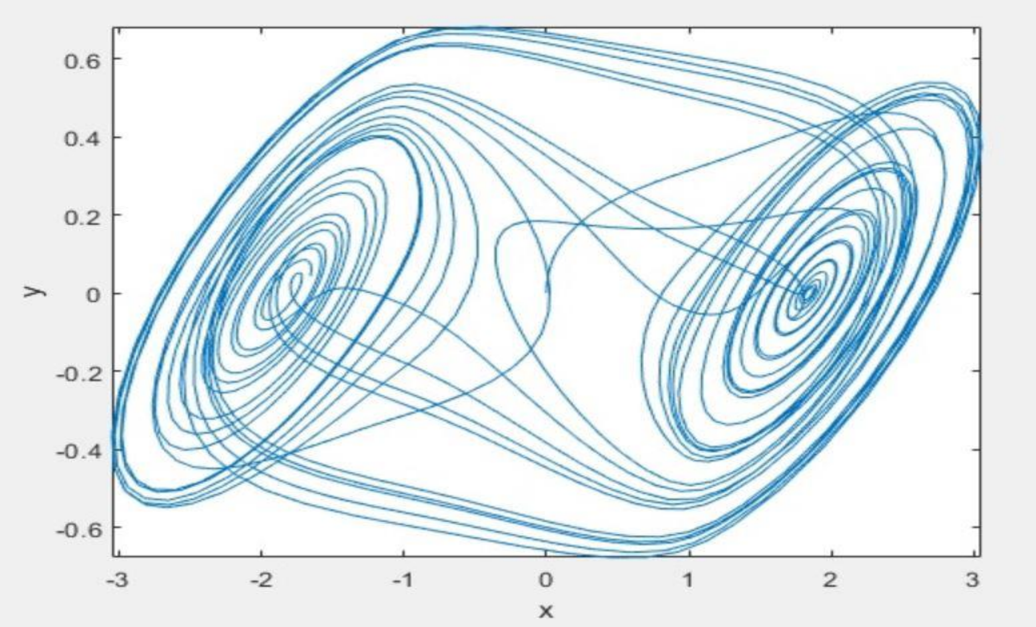


Fig 5.13: x-y response of chua's circuit

Timing Response of ORA using MATLAB/Simulink:

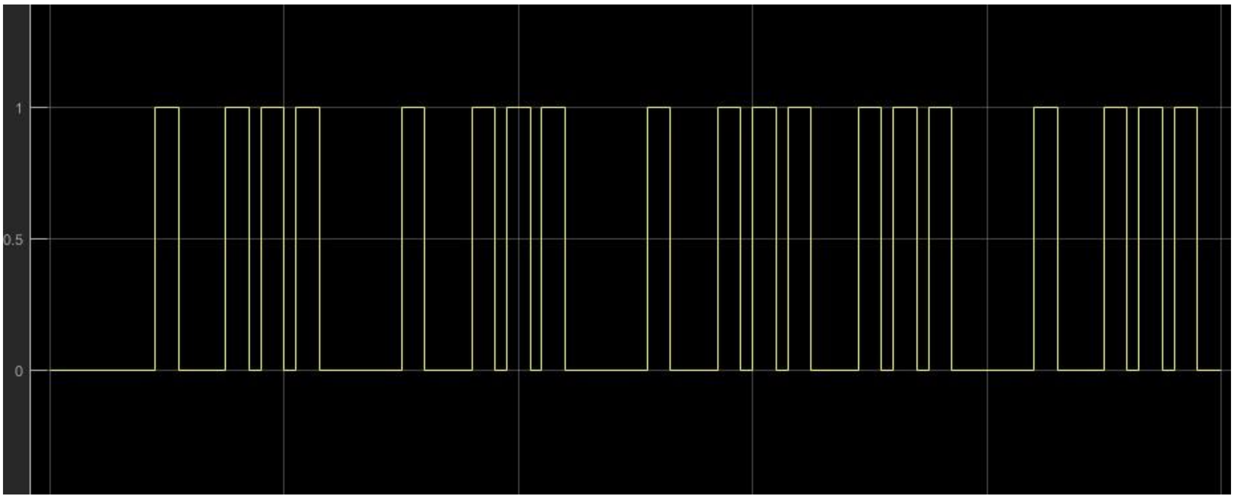


Fig 5.14: Timing response of SISR by Simulink

RTL schematic of SISR using MATLAB/Simulink:

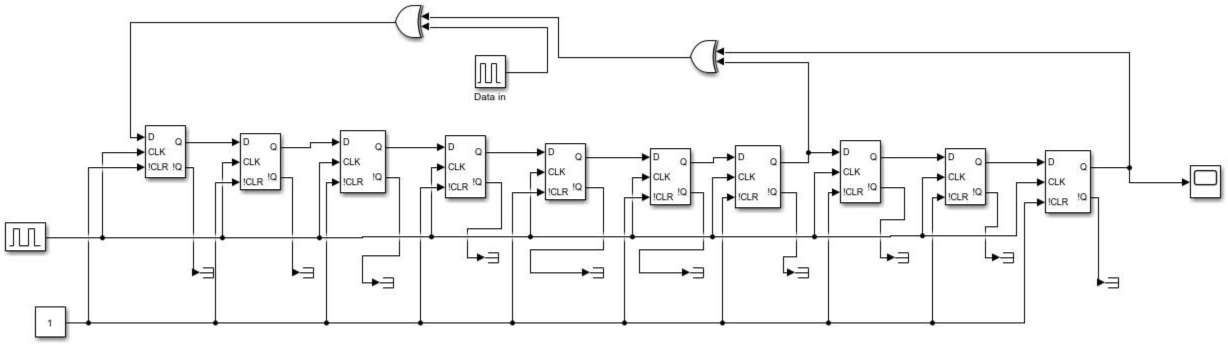


Fig 5.15: Schematic of SISR by Simulink

Comparison between traditional LFSR TPG and proposed TPG:

Table 5.1: Comparison based on delay, area and power:

Type of Pattern generator	LUTs	FFs	Maximum Delay	Maximum Power
LFSR	15	14	1.298ns	1.191W
dx Patterns	52	88	3.751ns	14.694W
dy Patterns	43	70	3.379ns	14.398W
dz Patterns	15	42	3.370ns	13.416W

CHAPTER-6

CONCLUSION

The implemented work, BIST using Chua’s circuit for random pattern generation for RC Adder was designed. It was found that BIST implemented with Chaotic circuit as TPG gave more secure to data transfer and for testing it is better for high fault detection as compared to LFSR as TPG. Timing waveforms were shown with normal and analog form for both TPG and ORA circuits with HDL(Verilog) and MATLAB/Simulink. VCS Synopsis RTL design compiler was used here to implement BIST architecture.

For further work includes implementing the chaotic circuit for ORA also so data can be decrypted. Alternative possible method may be employing architecture on FPGAs and accumulate the results to analyze the faults occurred by hardware on them and test the functionality.

On comparison of traditional LFSR circuit as pseudo-random patterns generator with this chaotic circuit for random pattern generator, has complex structure and use more number of LUTs and flip flops than that of LFSR but it has high fault coverage than normal TPG and it is highly secure design for any model testing due to its random nature.

All the design has been implemented in VCS tool, written in Verilog HDL language.

REFERENCES

[1] V. D. Agrawal, C. R. Kime, and K. K. Saluja, “A tutorial on built-in selftest. I. Principles,” IEEE Des. Test Comput., vol. 10, no. 1, pp. 73–82, Mar. 1993.

[2] E. Tlelo-Cuautle, M. A. Duarte-Villasenor and J. M. Garcia-Ortega, "Modeling and simulation of a chaotic oscillator by MATLAB," in IEEE Latin America Transactions, vol. 5, no. 2, pp. 94-97, May 2007, doi: 10.1109/TLA.2007.4381349.

[3] Wannaboon, N. Jiteurtragool and T. Masayoshi, "Chaotic oscillation-based BIST for CMOS operational amplifier," 2014 International SoC Design Conference (ISOCC), Jeju, 2014, pp. 130-131, doi: 10.1109/ISOCC.2014.7087607.

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

[5] K.M. Krishna, M. Sailaja, “Low Power Memory Built in Self Test Address Generator Using Clock Controlled Linear Feedback Shift Registers”, J Electron Test 30, 77–85 (2014). https://doi.org/10.1007/s10836-014-5432-1.

[6] Design Verification and Test of Digital VLSI Circuits NPTEL.

[7] Version 2 EE IIT, Kharagpur, Module 8 Testing of embedded systems, Lesson 40 Built in Self-Test BIST for Embedded Systems, pages 3-16.

[8] S. Jamuna, Asst Professor, Department of ECE, DSCE, “Implementation of BIST Structure using VHDL for VLSI circuits”,International Journal of Engineering Science and Technology (IJEST) Bangalore, Vol. 3 No. 6 june 2011.

[9] L. Michael Bushnell, D. Vishwani Agrawal, “Essentials of Electronic Testing For Digital, Memory And Mixed Signal VLSI Circuits”, Boston: Kluwer Academic Publishers,2002,volume 17 ISBN:978-0-7923-7991-1.

[10] L. T. Wang, C. W. Wu, and X. Wen, Eds., “VLSI Test Principles and Architectures: Design for Testability”, Morgan Kaufmann Publishers Inc. (2006), ISBN:978-0-08-047479-3.

[11] M. Lahcene, A. P. Adda, N. Said & M. Mustafa. (2013), “A Pseudo Random

Number Generator Based on the Chaotic System of Chua's Circuit, and its Real Time FPGA Implementation", Applied Mathematical Sciences. 7. 2719-2734. 10.12988/ams.2013.13242.

[12] K. Recai & D. Fatma, "Utilizing SIMULINK in Modeling and Simulation of Generalized Chaotic Systems With Multiple Nonlinear Functions", Computer Applications in Engineering Education. 2010,18. 684 - 693. 10.1002/cae.20273.

[13] B. Marta & G. Richard. (2008), "A novel modelled true random binary number generator for key stream generation in cryptographic applications", 1 - 7. 10.1109/MILCOM.2008.4753211.

[14] K. Preethy John and P. Rony Antony, "Optimized BIST Architecture for Memory Cores and Logic Circuits using CLFSR", 2017 International Conference on Intelligent Computing, Instrumentation and Control Technologies (ICICICT) (2017): 1266-1270.

[15] M. Manjunath & H. Ganapathi. (2017), "Power Optimized TPG for BIST Architecture", 1-4. 10.1109/ICCIC.2017.8524577.

[16] B. Abhilash, S. Shirol, J. Ramakrishna, B. S. Rajashekar and K. Priyatam, "Design and Implementation of 8-bit LFSR, Bit- Swapping LFSR and Weighted Random Test Pattern Generator: A Performance Improvement", 82-86. 10.1109/ISS1.2019.8908063.

[17] Q. Aixue, H. Chunyan and W. Guangyi, "Design and FPGA realization of a pseudo random sequence generator based on a switched chaos," 2010 International Conference on Communications, Circuits and Systems (ICCCAS), Chengdu, 2010, pp. 417-420, doi: 10.1109/ICCCAS.2010.5581965.

[18] P. Ihsan, K. Ersin, L. Qiang, B. Aziz, and C. Mustafa Kutlu., "A Multiscroll Chaotic Attractor And Its Electronic Circuit Implementation", 1. 29-37 2019

[19] M. S. Azzaz, M. A. Krimil, F. Labiod, A. Kadir and D. Teguig, "FPGA Hardware Design of a Unified Chaotic System for CTRNG," 2018 International Conference on Signal, Image, Vision and their Applications (SIVA), Guelma, Algeria, 2018, pp. 1-4, doi: 10.1109/SIVA.2018.8661042.

[20] C. Parvathy and V. S. Vishnu, "Circuit under Test Verification with MSIC Test Pattern Generator", International Journal of Science and Research (IJSR), Volume 4 Issue 7, July 2015, 2374 - 2378

[21] J. Manish Patel, P. Nehal, V. Chaudhari., “Design and Implementation OF Logic-BIST Architecture for I2C Slave VLSI ASIC Design Using Verilog”, ASEE journal ISSN: 0975 – 6779| NOV 12 TO OCT 13 | VOLUME – 02, ISSUE - 02

[22] X. Chen, F. Xiang and L. Zhang, "Optimal design of double-precision chaotic signal generators based on IEEE-754 standard," 2012 IEEE International Conference on Automation and Logistics, Zhengzhou, 2012, pp. 466-469, doi: 10.1109/ICAL.2012.6308251.

[23] F. Corinto, O. V. Krulikovskyi and S. D. Haliuk, "Memristor-based chaotic circuit for pseudo-random sequence generators," 2016 18th Mediterranean Electrotechnical Conference (MELECON), Lemesos, 2016, pp. 1-3.

[24] T. Stojanovski and L. Kocarev, (2001)“Chaos-Based Random Number Generators—Part I: Analysis”, Circuits and Systems I: Fundamental Theory and Applications, IEEE Transactions on. 48. 281 - 288. 10.1109/81.915385.

[25] S. Saini and J. S. Saini, "Secure communication using memristor based chaotic circuit," 2014 International Conference on Parallel, Distributed and Grid Computing, Solan, 2014, pp. 159-163. doi: 10.1109/PDGC.2014.

[26] Charles E. Stroud, “A Designer's Guide to Built-in Self-Test”, Springer 2002.

[27] B. John, Christy M. Philip, Agi J. George, “Design and Implementation of Built in Self Test (BIST) forVLSI Circuits using Verilog”, International Journal of Engineering Research & Technology (IJERT) NCETET – 2015 (Volume 3 – Issue 05).

[28] J. Rajski, J. Tyszer, and N. Zacharia, “Test data decompression for multiple scan designs with boundary scan”, IEEE Transactions on Computers, 47, pp. 1188-1200, 1998.

[29] N. A. Tauba and E. J. MaCluskey, “Altering a pseudo-random bit sequence for scan based”, in Proceedings of International Test Conference, 1996, pp. 167-175.

[30] S. Hussain, K. Padma Priya, “Test Vector Generator (TPG) for Low Power Logic Built In Self-Test (BIST)”, International Journal of Advanced Research in Electrical, Electronics and Instrumentation Engineering, Vol. 2, Issue 4, April 2013.

[31] P. Kasunde, Dr. K. B. ShivaKumar, Dr. M. Z. Kurian, “Improved Design of Low Power TPG Using LPLFSR”, International Journal of Computer & Organization

Trends – Volume3 Issue4 – May 2013.

[32] https://en.wikipedia.org/wiki/Chua%27s_circuit

[33] E. Tlelo-Cuautle, AJ Quintas-Valles, LG de la Fraga, JJ Rangel-Magdaleno, "VHDL Descriptions for the FPGA Implementation of PWL-Function-Based Multi-Scroll Chaotic Oscillators", PLoS One. 2016 Dec 20;11(12):e0168300. doi: 10.1371/journal.pone.0168300. PMID: 27997930; PMCID: PMC5173382.

[34] V. Kirthi, Dr. G. Mamatha Samson, "Design of BIST with Low Power Test Vector Generator", IOSR Journal of VLSI and Signal Processing (IOSR-JVSP) Volume 4, Issue 5, Ver. II, Sep-Oct., 2014.

[35] Annie Chandra, K. Thatchayani, "Power Reduction in TPG Based Built- in Self-Test(BIST) using LP/BS-LFSR on FPGA", International Journal of Engineering Research & Technology (IJERT), ISSN: 2278-0181, Vol. 3 Issue 4, April – 2014.

[36] R. Trivedi, S. Dhariwal & K. Abhishek. (2018), "Comparison of various ATPG Techniques to Determine Optimal BIST", 93-98. 10.1109/ICICS.2018.00031.

[37] F. Corinto, O. V. Krulikovskyi and S. D. Haliuk, "Memristor-based chaotic circuit for pseudo-random sequence generators," 2016 18th Mediterranean Electrotechnical Conference (MELECON), Lemesos, 2016, pp. 1-3.

[38] A. S. Abu-Issa and S. F. Quigley, "Bit-swapping LFSR for low-power BIST", ELECTRONICS LETTERS volume 44, number 6, (2008).

[39] M. Bushnell and V. Agrawal, "Essentials of Electronic Testing for Digital, Memory and Mixed-Signal VLSI Circuits", Boston: Kluwer Academic Publishers, 2000.

[40] P. Bardell, W. McAnney and J. Savir, "Built-In Self-Test for VLSI: Pseudorandom Sequences", Somerset, New Jersey: John Wiley & Sons, 1987.

[41] S. Swapna, S. Sunilkumar Manvi, "Review of LP-TPG Using LP-LFSR for Switching Activities", International Journal of Advanced Research in Computer Science and Software Engineering, Volume 5, Issue 2, (2015).

[42] Miss. T. Patil and Mr. A. Dhankar, "A Review on Power Optimized TPG Using LP- LFSR for Low Power BIST," IEEE Sponsored World Conference on Futuristic Trends in Research and Innovation for Social Welfare (WCFTR'16).