Simulation of BIST based Multiplier using FPGA

G.Sathwik[1], Mohammed Ali
[1]*M.Tech Student, ECE Department, J. B. Institute of Engineering and Technology, Hyderabad*
[2]*Guide, Associate Professor, ECE Department, J. B. Institute of Engineering and Technology, Hyderabad*

***Abstract*- The ever increasing applications of integrated circuits in the day-to-day useful electronic gadgets is the driving force for the development of low power designs of configurable hardware designs. High speed and low power are the main parameters that are targeted by modern circuit designers. Multipliers are the very important logic operational unit of any processing unit in digital signal processing applications. The speed and performance of multiplier is among the efficiency improvement parameters of any digital hardware design. Another important feature of hardware designs is self-testing ability. The built-in-self test (BIST) feature helps in quick diagnosis of the hardware functional authenticity. A low power Test Pattern Generator (TPG) is involved in the design for self-test design realization. The low power performance is analyzed with different operating clock frequency and Different FPGA devices with an improvement of around 10% of power with respect to previous design. The designs works at high speed of 3.288Gbps Throughput and 1.5 clock cycle Latency.**

I. INTRODUCTION

The main challenging areas in VLSI are performance, cost, testing, area, reliability and power. The demand for portable computing devices and communication system are increasing rapidly. These applications require low power dissipation. The main aim of these devices is to reduce the power dissipation with high fault coverage. Generally power dissipation of a system in test mode is more than in normal mode. The project is an approach towards design of BIST based multiplier using VERILOG on FPGA. This section describes the domain details about the multiplier in general, and about BIST multiplier along with project objective and motivation and expected outcome.

Multipliers are extensively used in digital signal processors (DSPs). Many current Field Programmable Gate Arrays (FPGAs) incorporate embedded cores such as memory and DSPs in addition to logic blocks. For example, Xilinx Virtex-2 and Spartan-3 series FPGAs incorporate 18×18K-bit random access memories (RAMs) with 18×18-bit multipliers associated with each RAM in addition to the configurable logic blocks (CLBs). Xilinx Virtex-4 and Virtex-5 series FPGAs incorporate DSP cores. Since the multipliers are often part of larger complex circuits like DSPs they are less controllable and less observable and hence an effective Built-In Self-Test (BIST) approach is required. While developing BIST approaches for the multipliers in DSP modules in Virtex-4 FPGAs, we investigated various previously proposed test approaches in an attempt to find an architecture independent test approach that effectively tests multipliers regardless of their architecture.

In general an N×M-bit multiplication is performed by multiplying each of the M-bits of the multiplier with all N bits of the multiplicand resulting in M N-bit wide partial products. These partial products are then added together to generate the final product. Faster multiplication can be achieved by reducing the number of partial products. The multiplier inside the DSPs in Virtex-4 FPGAs is an 18×18-bit 2's complement multiplier that produces two 36-bit partial products. The DSPs in Virtex-5 devices have larger 25×18-bit 2's complement multipliers that produce two 43-bit partial products. These partial products are sign extended to 48 bits and added using a separate 48-bit carry look ahead (CLA) adder.

Knowing the architecture of the multiplier is important to design an effective test but the Xilinx data sheets for the multipliers and DSPs present inside the FPGAs provide little if any information about the architecture of these multipliers. Since there is no mention of clock cycle latencies in data sheets we can eliminate sequential logic architectures. Therefore, we performed fault simulations on various non sequential multipliers to find an effective test approach that tests multipliers

regardless of their architecture. The various choices for multipliers in our fault simulation analysis include array multipliers, signed array or Baugh Woolley multipliers, modified Booth multipliers, modified Booth Wallace tree multipliers and a custom multiplier obtained by making a modification to the modified Booth multiplier

In complex integrated circuits, low controllability and observability of embedded blocks impose severe testability problems. In order for the final chip to become a viable product such embedded blocks must be well tested. Built-in self-test (BIST) structures are well suited for testing such embedded blocks, since they cut down the cost of testing by eliminating the need of external testing as well as they can apply the test vectors at speed. High fault coverage (FC), small area overhead and small application time have been the traditional objectives of BIST designers. Even though these objectives still remain important, a new BIST design objective, namely low power dissipation (PD) during testing.

Multipliers are commonly used as embedded blocks in both general purpose data path structures and specialized digital signal processors. Effective low power BIST schemes for both carry save array multipliers and modified Booth multipliers with carry save addition of the partial products have recently been proposed. Wallace tree summation along with Booth encoding are the most common techniques for designing fast multiplier blocks. Booth encoding aims to reduce the number of partial products whereas Wallace tree summation and carry look ahead (CLA) addition in the final stage of the multiplier aim at the faster addition of the partial products. A BIST scheme for such multipliers has recently been proposed in. This BIST scheme does not take into account the low PD objective. We will use this BIST scheme as the basis for our comparisons. The organization of the paper is follows: Section-2 contains Literature survey of this paper. Section-3 describes the Stride BV Algorithm and its general architecture. Section-4 contains implementation of the proposed system architecture and its details. Section-5 contains results and analysis. Section-6 describes the dissertation conclusion and future work.

II.RELATED WORK

Many architectural modifications are proposed by many scholars and researchers in their work regarding low power design of BIST based logic circuit for hardware design applications. In [1] a low power test pattern generator design is proposed using a low-power Linear Feedback Shift Register for BIST structures. This design follows the approach of reducing the switching activity based on single input change pattern generated by a counter and a gray-code converter. Reference [2] presents FPGA Implementation of an LFSR based Pseudorandom Pattern Generator for MEMS Testing. This design has the characteristics of high speed, low power consumption and it is especially suited in the processors where uniform distribution random numbers are required. A Low Power linear feedback shift register based low power test pattern generator design is proposed in [3]. This design mainly focuses on how test vectors are generated in the BIST and how to reduce the power consumption. In this paper the transition is reduced by generating the gray-code with 1-bit distance. Reference [4] shows FPGA implementation of 16-bit BBS and LFSR PN Sequence Generator. This paper features the change in the logic of PN sequence generator by changing the seed in LFSR or by changing the key used in BBS.

A paper with FPGA based N-bit LFSR to generate random sequence number design is proposed in [5]. This design presents study the performance and analysis of the behavior of randomness in LFSR. A review of LP-TPG using LP-LFSR for Switching Activities is presented in [6]. This paper presents structures of multiplier, LFSR, LP-TPG and BIST. In [7] the author presents a simulation study of TPG using Shift Register based on 16th Degree Primitive Polynomials. The study in this paper focuses on a comparative study of different types of implementations for a LFSR for 16th degree irreducible or primitive polynomials. Generation of Pseudo-Random number by using WELL and Re-seeding method is presented in [8]. In this paper a random number is generated by using WELL method first and its performance was analyzed. For avoiding the repeating pattern the Reseeding method is used. A number of researches are also performed on logic operational units for high speed applications using FPGA devices.

A review on Vedic Mathematics for digital signal processing operations is present in [9]. This paper deals with exhaustive review of literature based on Vedic Mathematics. An improved efficiency of Vedic multiplier is proved over conventional multiplier in this paper. An FPGA based implementation of high speed 16-bit Vedic multiplier using LFSR is presented in [10]. This paper describes the implementation of 16-bit Vedic multiplier enhanced with propagation delay and automatic insertion of all possible combinations of inputs. The TPG is the major component of BIST hardware design. Many BIST application circuits are proposed and simulated by researchers to propose power and speed optimized designs based on FPGA implementation. Reference [11] presents FPGA implementation of BIST enabled UART for Real Time Interface Applications. This paper shows functional verification of various block of UART. A concurrent BIST architecture for online input vector monitoring design is proposed in [12]. This paper is based on the idea of monitoring a set of vectors reaching the circuit inputs at the time of normal operation and the use of a SRAM like architecture that store the relative locations of the vectors that reach the circuit inputs. A BIST enabled I2C protocol design implementation on FPGA is presented in [13]. This design enables self-test of a common hardware interface protocol for high speed communication device.

The requirement of the today's hardware designs is low power circuit implementation of BIST based logic circuits on FPGA to achieve high speed operational circuits. Reference [14] shows an advanced BIST architecture with Low Power LBIST and BDS oriented March Algorithm for Intra Word Coupling Faults. This paper addresses read faults with classic faults with an improvement in the efficiency of the architecture and test time in detecting the faults. In the present paper a critical consideration is given to low power BIST implementation. A multiplier with two 4-bit inputs is taken as a test design for low power implementation on FPGA with self-test capability. The self-test feature is provided using a low-power test pattern generator design. The test pattern is designed using a modified architecture by reducing the number of sequential component as compared to the conventional design components.

III.SYSTEM DESIGN DESCRIPTION

BIST block diagram

Fig 1 shows the architecture of BIST. In built- in self-test (BIST) design, parts of the circuit are used to test the circuit itself. On-line BIST is used to perform the test under normal operation, whereas off-line BIST is used to perform the test off-line. In this paper, BIST methodology will be used for verification of these multipliers. Built -in self test (BIST) techniques constitute a class of schemes that provide the capability of performing at-speed testing with high fault coverage, whereas simultaneously they relax the reliance on expensive external testing equipment. BIST techniques are typically classified into offline and online. Offline architectures operate in either normal mode (during which the BIST circuitry is idle) or test mode. During test mode, the inputs generated by a test generator module are applied to the inputs of the circuit under test (CUT) and the responses are captured into a response verifier (RV). Therefore, to perform the test, the normal operation of the CUT is stalled and, consequently, the performance of the system in which the circuit is included,

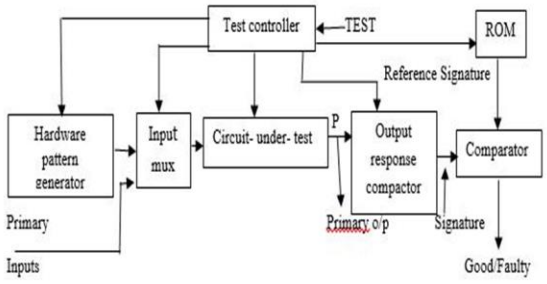

Figure 1: Block Diagram of BIST Architecture

Figure shows the architecture of IVM with BIST the block contained in this architecture were two MUX that produces the output depends on the mode if the mode is normal it selects the normal vectors, if the architecture performs in test mode MUX selects TA, TB as its inputs. This architecture depends on two mode i.e normal mode and test mode, a 4 bit multiplier block that generates the 4 bit product output, a test pattern generator that comes in work when test mode is activate, another block is IVM BIST control unit role of this unit is to produce a decoded value to be checked and final block is response verifier that compares the output of 4 bit multiplier and the IVM BIST control unit output. If

the values don't match then it will show that there are errors that have to be corrected. RV is enabled when the output of the decoder and the 4-bit multiplier does not match. The architecture performance depends on two modes named as normal or test mode. The functionality of the system and named as *T/N*.

If the value of T/N is 0 systems operates in normal mode and the inputs of multiplier driven by normal input vector i.e., A and B are the inputs. The outputs of multipliers are provided to the 4-bit multiplier and it generates an output by multiplying the inputs. At the same time the output of the MUX is given to the IVM BIST control unit that also decodes the actual output. And check is performed by response verifier. This unit only activated when the output of the IVM BIST control unit and the output of the 4 bit multiplier doesn't match. If there is no error, both the outputs are same then product output is obtained.

If the value of T/N is 1 system operates in test mode and the input selected by the multiplier driven by the test pattern generator, i.e., TA and TB. And this test pattern generator is controlled by IVM BIST control unit. The procedure is same MUX selects TA, TB and given to the 4 bit multiplier that generates an output and this output is then compares with the IVM BIST control unit value at response verifiers and follow the same explained above.

The generation of this concurrent BIST test pattern includes the following steps:

- The number of bits generated per pattern is restricted to some limit. Generation to some extent particular test set with minimum no of bits as possible.
- Target efficiency is achieved by selecting the patterns from this test.
- Comparison is done in appropriate manner by selecting output values.
- Continuously monitor the input output value Generate by the BIST control unit and the 4-bit multiplier.

IV. PROPOSED WORK

The block diagram consists of different functional blocks namely:

a. TPG: For the BIST implementation, a test pattern generator with random output value is required. For TPG realization, a low-power modified design of linear-feedback-shift-register (LFSR) is used in this design implementation.

b. Signature memory: Indicates whether the output is correct or not by analyzing the signature generated by the PRBS sequence generator.

c. Counter: It is used to keep track of the number of test vectors fed to the combinational block.

d. Programmable Clock generator circuit: It generates different clock frequencies.

e. Programmable Skew generator circuit: It generates the skew required for the output register.

f. Combinational Block : It is used to design of Multiplier Block acts as a Design Under Test (DUT)

V. RESULT ANALYSIS

5.1 Device utilization summary:

Selected Device	-LFSR_LP	
Number of occupied slices-	4	
Number of slice Flip Flops	-	6
Number of 4 input LUT's	-	3
Number of IOs	-	3
Number of bounded IOBs	-	11

5.2 Simulation Results of Design.

LFSR (Linear Feedback shift Register) :

Which generates a set of outputs by ex-oring from the initial set of inputs. Depending up on the position of ex-or gates connected in the path or out of path those lfsr are named as INTERNAL type lfsr and EXTERNAL type lfsr.

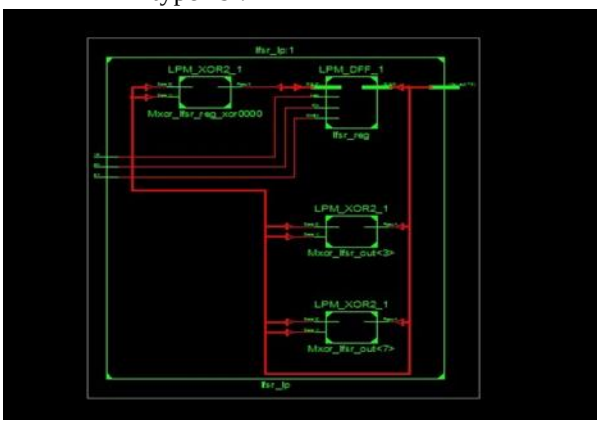

Fig 5.2a LFSR Schematic view

MULTIPLIER:

The multiplier does its action in Digital Signal Processors, which multiplies two set of input binary

data .The below figure shows the RTL schematic view of Multiplier module.

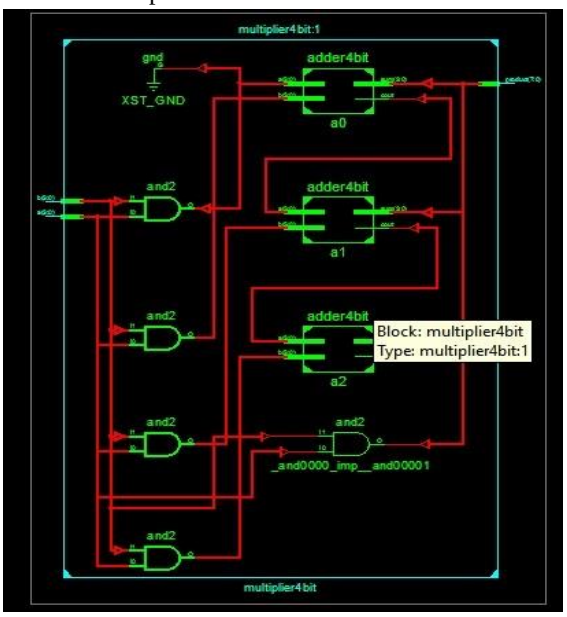

Fig: 5.2 b Multiplier Design RTL view.

MULTIPLE INPUT SIGNATURE REGISTER (MISR).

MISR is used to Reduce test logic by using multiple bit streams to create a signature. The MISR is formed by adding the ex-or gates at the each flip-flop. Depending on the input provided to the MISR we get an output and depending on the no of registers we capture the output and compare it with the signature register contents.

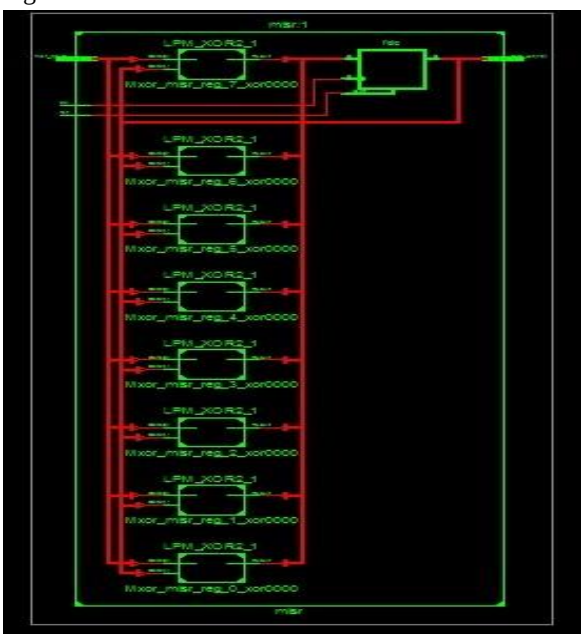

Fig : 5.2.c MISR RTL schematic view.

TEST RESPONSE ANALYZER:

In short Test Response Analyzer is TRA which compares the response values(golden values) and the generated values , so if the test is successful then logic design is error free. The Test Response Analyzer is as shown in the below figure(5.2d) .

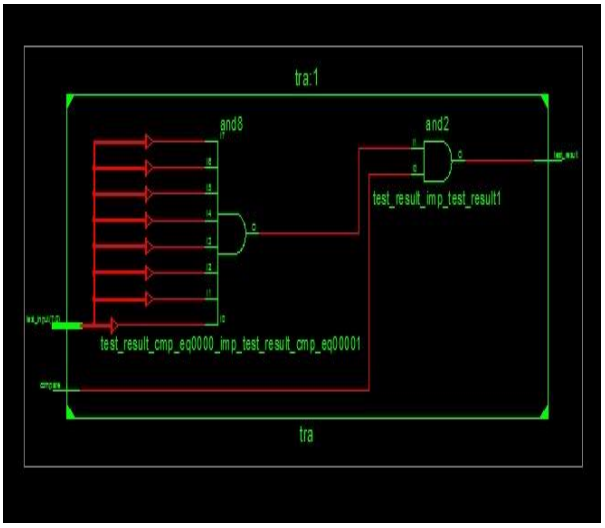

Fig 5.2d: Test Response Analyser RTL schematic View.

The below figure describes the functionality of the TRA whether the test is successful or failed .so ,at a certain positive edge of clock the test response is noticeable along with other parameters .

Fig 5.2e :simulation result of bist pass or fail.

BUILT IN SELF TEST:

BIST controller is the main module in the project which controls the bist operation. It is a state machine used for state transition from one state to other state. The BIST controller also decides the input to the circuit under test based on whether the module is in normal mode or test mode on seeing the Test Mode (Test) input.

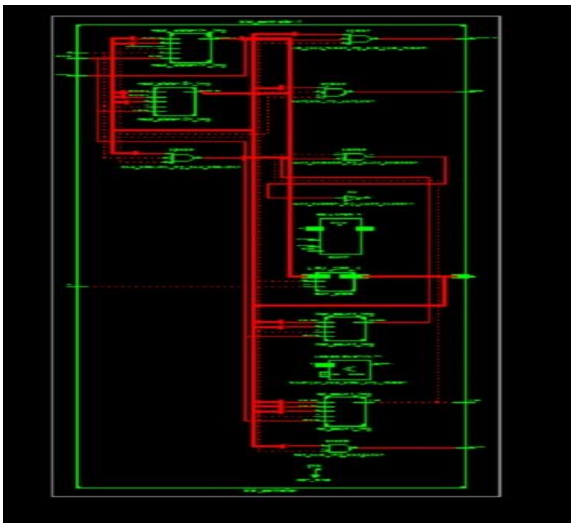

Fig 5.2f BIST top module Schematic view

Total REAL time to Xst completion: 37.00 secs
Total CPU time to Xst completion: 37.71 secs
Total memory usage is 224616 kilobytes
Number of errors : 0 (0 filtered)
Number of warnings: 0 (0 filtered)
Number of infos : 0 (0 filtered)

5.3 POWER REPORT

The power report consists of types of power utilized is like Dynamic power, Total power, Quiescent power and Thermal properties.

Here the Dynamic power is 0.174Watts, Quiescent power 0.028Watts and total power is 0.202Watts .

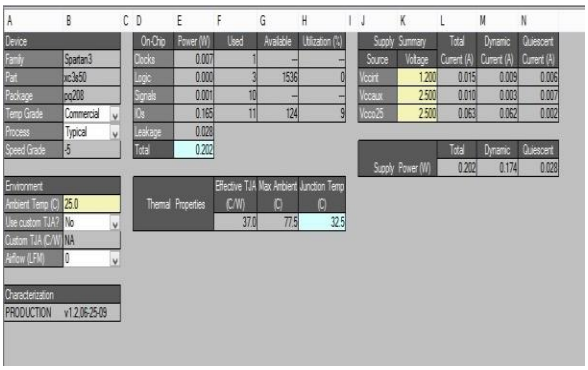

Fig: 5.3 Power Report indicating various powers.

VI. CONCLUSION

6.1 Advantages and Disadvantages

Advantages

1. The main advantage is the testing cycle decreases drastically i.e., the ATE (Automatic Test Equipment) verifies with a slower rate than self testing type.
2. In LPLFSR power dissipation is very low compared to LFSR.
3. More fault coverage area.
4. Reduce the test time.

Disadvantages:

1. The embedding of self test logic increases the on chip size, but the area over head on the chip is negligible.
2. The at most care has to be taken that self test logic itself free from errors, else the entire scenario changes one should test.

6.2 Applications

1. Testing the fault tolerant design.
2. The self testing technique is applied for routing the circuit interfaces to fault free designs when there are faults crept in the testing design.
3. By low power design the low power devices can be developed and able to design portable designs.

VII. CONCLUSION

In the present work the logic design that is used in the built-in-self-test application is a 4 -bit multiplier and the test pattern generator is also designed for generating a random 8-bit number. The test pattern generator is a modified design that has a low register-to-bit ratio, i.e., the number of out bits in the generated sequence is more than the number of registers in the generator circuit. Thus, with respect to the previously proposed designs of TPG it involves less number of registers and hence a low- power design realization is obtained using this TPG.

This work simulation and synthesis clearly indicates that low power implementation hardware can be used for applications with a configurable IC that has low internal voltage (Vint) and low auxiliary voltage (Vaux).The power utilization with Linear Feedback Shift Register multiplier Design is more compared with Low Power Linear Feedback Shift Register logic Design multiplier design.

The present design uses a modified test pattern generator design for BIST implementation. This design can be modified by multiple series combination of similar architecture of logic circuit for a long bit sequence generation of random number.

6.4 Future Scope

In future the multiplier and the test pattern generator can be configured to match the application specific requirement of the design for a BIST based hardware design implementation. The present work also has the scope of combining other existing hardware designs with this design for a complex logic implementation.

REFERENCES

[1] A. Kavitha, G. Seetharaman, T. N. Prabhakar and Shrinithi, "Design of Low Power TPG using LP-LFSR",

[2] International Conference on Intelligent Systems Modelling and Simulation, IEEE Computer Society, pp.334-338, 2012.

[3] Md. Fokhrul Islam, M. A. Mohd. Ali and Burhanuddin Yeop Majlis, "FPGA Implementation of an LFSR based Pseudorandom Pattern Generator for MEMS Testing", International Journal of Computer Applications, Volume-75 No-11, pp.30-34, August 2013.

[4] Syed. Mujeeb Raheman and M. Basha, "Low Power Linear Feedback Shift Register based Low Power Test Pattern Generator", International Journal and Magazine of Engineering, Technology, Management and Research, Volume-2 Issue-6, pp.579-584, June 2015.

[5] Mohammad Moinuddin and Sadgunn Kumari V, "FPGA implementation of 16-bit BBS and LFSR PN Sequence Generator", International Journal of Scientific Engineering and Technology Research, Volume-4 Issue-31, pp.6082-6084, August 2015.

[6] Babitha P. K., Thushara T. and Dechakka M.P., "FPGA based N-bit LFSR to generate random sequence number", International Journal of Engineering Research and General Science, Volume-3 Issue-3, pp.6-10, May-June 2015.

[7] Swapna S. and Sunilkumar S. Manvi, "Review of LP-TPG using LPLFSR for Switching Activities", International Journal of Advanced Research in Computer Science and Software Engineering, Volume-5 Issue-2, pp.157-160, February 2015.

[8] Mirella Amelia Mioc, "A simulation study of TPG using Shift Register based on 16th Degree Primitive Polynomials", New Developments in Pure and Applied Mathematics, pp.363-369, ISBN: 978-1-61804-287-3.

[9] V. Divya Bharathi and Arivasanth M., "Generation of Pseudo-Random number by using Well and Re-seeding method", International Journal of Advanced Research in Electronics and Communication Engineering, Volume-4 Issue-3, pp.604-609, March 2015.

[10] Kaustubh M. Gaikwad and Mahesh S. Chavan, "Vedic Mathematics for Digital Signal Processing Operations: A Review", International Journal of Computer Applications, Volume-113 N0-18, pp.10-14, March 2015.

[11] Neha Shaktawat and Ghanshyam Jangid, "FPGA Implementation of High Speed 16-Bits Vedic Multiplier using LFSR", International Journal of Computer Science and Mobile Computing, Volume-4 Issue-8, pp.102- 108, August 2015.

[12] L. Supriya, J. Lingaiah and G. Kalyan, "FPGA Implementation of BIST (Built-in-Self-Test) Enabled UART for Real Time Interface Applications", International Journal of Science, Engineering and Technology Research.

[13] S. Abirami, Nikitha S. Paulin, S. Prabhu Venkateshwaran, "A concurrent BIST architecture for online input vector monitoring", International conference on Science, Technology and Management, pp.1411-1488, Feb 2015.

[14] Jayashri H. M. and Shilpa K., "Design and Implementation of I2C with BIST Technique on FPGA",International Journal of Research in Information Technology, Volume-3 Issue-5, pp.428-438, May 2015.