International Journal of Innovative Technology and Exploring Engineering (IJITEE)
ISSN: 2278-3075 (Online), Volume-9 Issue-12, October 2020

A BIST Methodology to test CLB Resources on an SRAM-Based FPGA using Complementary Gates Configuration

Rajesh A, Jameer Basha SK, Francis Xavier, Hari Babu S

Abstract ***This paper primarily focuses on designing a new Built in self test (BIST) methodology to test the configurable logic blocks (CLBs) which is the heart of field programmable gate array (FPGA). The proposed methodology targets stuck-at-0/1 faults on a RAM cell in an LUT which constitutes about 90% of the total faults in the CLBs. No extra area overhead is needed to accommodate the test pattern generators (TPGs) and output responses analyzers (ORAs) as they are realized by the already existing configurable resources on the FPGA.A group of CLBs chosen as block under test (BUT) are configured as complementary gates (AND/NAND, OR/NOR, XOR/XNOR) to successfully test the aforementioned faults. The proposed BIST structure when implemented on Xilinx Virtex-4 FPGA proved 100% fault coverage and minimized test configurations.***

Keywords- FPGA, BIST, CLB testing, complentary gates, look up table, fault coverage.

I. INTRODUCTION

A Field-programmable Gate Array (FPGA) is a configurable integrated circuit that can implement an arbitrary logic design. A typical FPGA architecture consists of an array of programmable logic blocks (PLBs), Input/output (I/O) cells, and a programmable interconnect network, as shown in Figure 1. The PLBs in FPGAs typically consist of look-up tables (LUTs), multiplexers, and flip-flops. The PLBs are configured individually to implement low level digital logic. Then, the PLB inputs and outputs are connected together via the programmable interconnect network, also known as the programmable routing network, to implement larger circuits [1]. The routing network also provides signal paths from the I/O cells to the PLBs. The I/O cells provide a means for the FPGA to exchange information with external devices also implemented in a system or directly with system inputs and outputs. Broadly two types of FPGA's exist based on their programmable nature; a onetime programmable Antifuse FPGA and the unlimited programmable SRAM-based FPGA [2]. The logic functions and the interconnections among the logic resources are determined by the configuration bit stream that must be downloaded into the FPGA.

Revised Manuscript Received on October 30, 2020.
* Correspondence Author
Mr. Rajesh A, Assistant Professor, Department of EECE, GITAM Hyderabad.
Mr. Jameer Basha SK, Assistant Professor, Department of EECE, GITAM Hyderabad.
Mr. Francis Xavier, Assistant Professor, Department of EECE, GITAM Hyderabad.
Mr. Hari Babu S, Assistant Professor, Department of EECE, GITAM Hyderabad.

This programming data is stored in the configuration memory and each configuration bit controls the state of logic or routing element such as a transmission gate or multiplexer [5]. As the technology advanced further, the ability to integrate more number of components on the FPGA also augmented exponentially [3]. As to the date of this work, the gate count in the FPGA has crossed one billion. To test an FPGA under this condition will be significantly complicated. Moreover, to test the complete resources on FPGA, it has to be configured in incredibly large number of ways which is quiet impractical. So, a set of test configurations (TC's) has to be developed, regardless of any particular user application.Of all the resources present on the FPGA, the CLBs are the main logic resource for implementing sequential as well as combinatorial circuits. The LUT present in the CLB implements any combination circuit by storing their truth table values in it [4]. It can thus be considered as a memory cell storing a single bit. Due to the manufacturing or operational defects, there is every possibility that a particular memory cell in the LUT can be stuck-at-0/1 that could affect the logic of the user application. This paper targets such faults through a new BIST methodology where a part of CLBs are configured as complementary gates. Several previous works were developed to address the CLB fault testing [6]-[10]. The main advantage of the proposed methodology is that the number of test configurations is reduced as compared to the above said works. The conventional BIST introduces area and performance overhead by adding an extra hardware to account for TPGs and ORAs. But the proposed methodology utilises the already existing CLB resources to realise both the TPGs and ORAs thus resulting in zero area overhead virtually. Firstly, a part of CLBs are configured as TPGs, BUTs and ORAs and the test vectors are applied. Secondly, the roles of complementary gates are exchanged and are subjected to the same test vectors. The outputs thus obtained are sent to ORA for checking the faults. The ORA chosen in this work is a parity checker which is an XOR gate. Then, the roles of the CLBs are swapped until every CLB is tested.

The rest of the paper is organised as follows. In section II the background information about architecture of an SRAM based FPGA's CLB and its testing considerations are presented with reference to Xilinx Virtex-4 FPGA. Section III presents the proposed BIST structure and the test strategy followed. Section IV covers the test results and discussions. Section V concludes the paper.

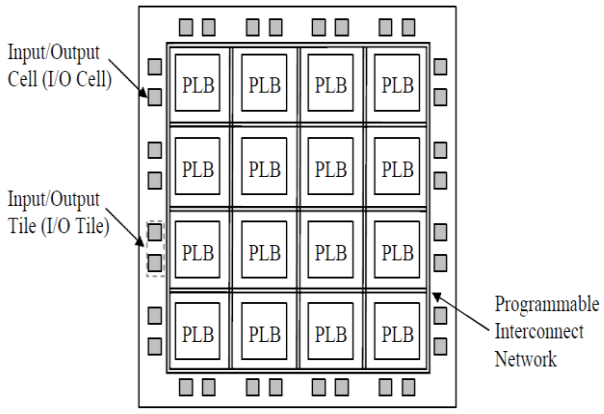

Figure 1: Typical FPGA Architecture

II. BACKGROUND

A. *Xilinx Virtex-4 FPGA*

The Virtex-4 PLB shown in Figure 2 consists of four SLICES;in those two are SLICELs and remaining two are SLICEMs [4]. A SLICEL is consists of two 4-input LUTs, two flip-flops/latches, and some secondary logic gates and multiplexers (Figure 2.3).

The LUTs consist the truth table of the combinational logic functions configured into a SLICE. The flip-flops / latches are used for any sequential logical functions that need to be performed by the SLICE. The secondary logic gates and multiplexers control the internal signal routing within the SLICE as well as specialized logic functions such as fast carry logic of adders. A SLICEM is a more complex SLICEL. It includes all of the internal characteristics of a SLICEL with added functionality to be used as shift registers or small RAM memories. The slices in Virtex-4 contain two 4-input LUTs, labeled F and G, two logic-carrying storage elements, multiplexers, and some arithmetic gateways. The LUTs can be used as a 4-input LUT, up to a 16-bit shift register (SliceM only) or a 16-bit LUT RAM (SliceM only). The storage elements can be configured as positive or negative edge-triggered flip-flops or active high or active low level-sensitive latches with clock-enable control capability.

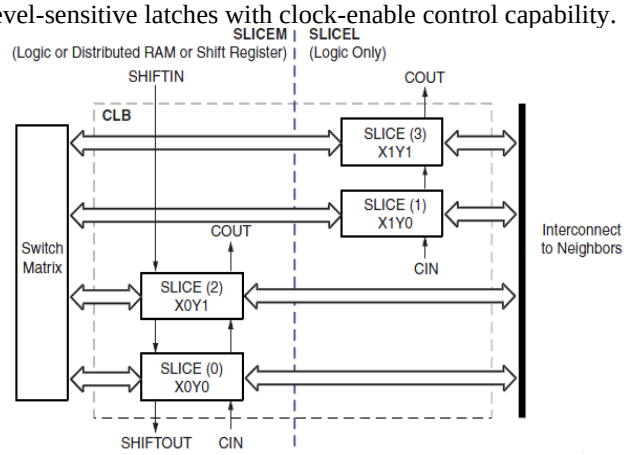

Figure 2: Arrangement of Slices within the CLB

B. *Testing considerations*

The two type of faults that are dealt in this paper, namely stuck-at-0 and stuck-at-1 faults on RAM cell of LUT are the most commonly observed faults. When a particular memory cell in an LUT is always stuck at logic '0', then it is called a stuck-at-0 fault and vice-versa [5]. It is to be noted that only single bit value can be stored in an RAM cell of LUT. Figure 3 illustrates both the faults.

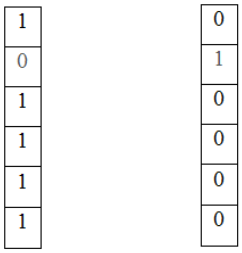

Figure 3: stuck-at-0 fault stuck-at-1fault

To test the above faults, a BIST circuitry is required. The BIST circuitry must have a TPG from where the test patterns are generated and an ORA, where the outputs of BUT are checked to determine if there are any faults. Initially the entire FPGA is configured in to the TPGs, BUTs and ORAs. The BIST structure implemented is as shown in Figure 4.

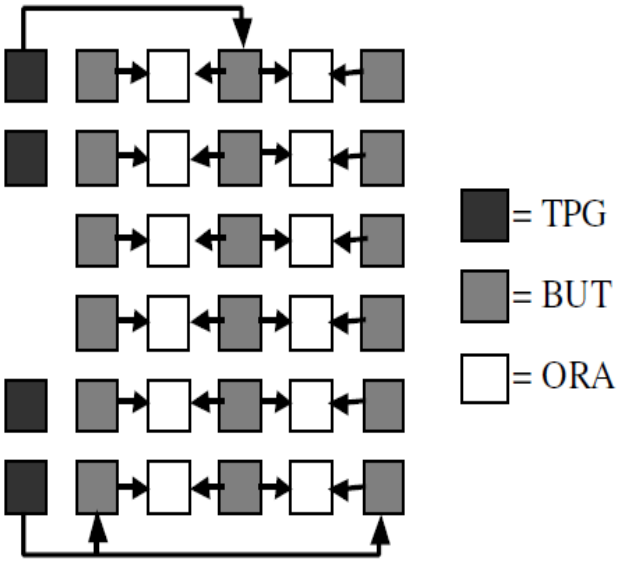

Figure 4: BIST structure

Traditionally the test pattern generator can be realised by implementing a counter or a linear feedback shift register [5]. In this paper we configure the existing CLB's to store the test patterns. This can be possible as LUTs in CLBs act as a RAM. The test patterns thus generated are directly applied to BUT. The ORA which is used to compare the outputs of BUT can be realised by configuring a part of CLBs as parity checker. An example of a simple parity checker is XOR gate. The above process is repeated swapping the roles of the CLBs in different sessions thus covering all the CLBs on FPGA as illustrated in Figure 5.

Retrieval Number: 100.1/ijitee.L79851091220
DOI: 10.35940/ijitee.L7985.1091220
Journal Website: www.ijitee.org

Published By:
Blue Eyes Intelligence Engineering
and Sciences Publication

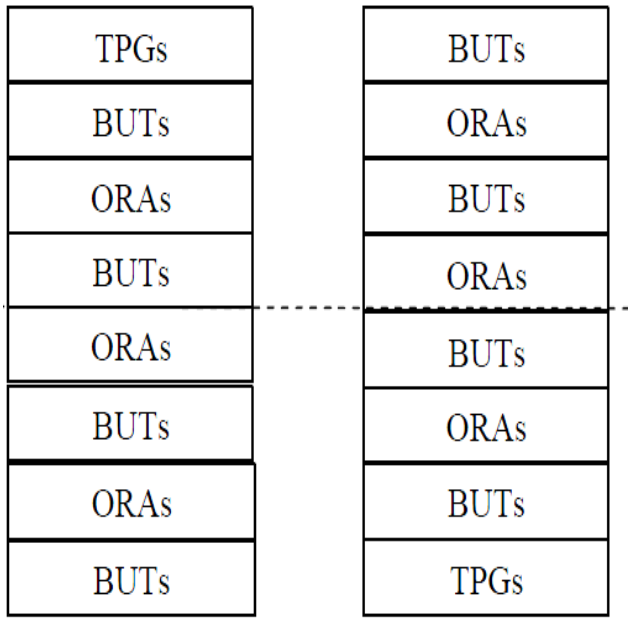

Figure 5:Test configuration (a) session 1 (b) session 2

III. PROPOSED METHODOLOGY

The proposed methodology utilises four CLBs to realize the TPG, two CLBs to realise ORA and four CLBs as BUT. Figure 6 shows the circuitry of a TPG.

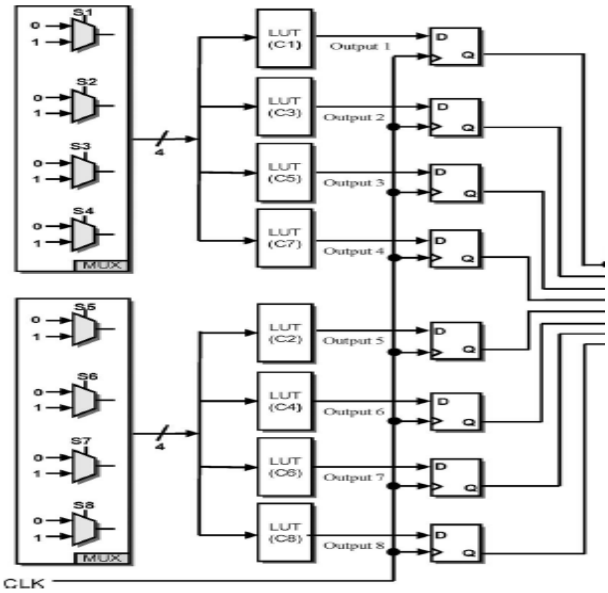

Figure 6: TPG of the proposed methodology

The selection lines s1-s8 is used to control the values at the outputs. Initially all the c1-c8 values are configured according to table I. The BUT which contains four CLBs are configured as complementary gates i.e. CLB1, CLB2, CLB3, CLB4 are configured as AND, NAND, OR, NOR respectively as shown in figure 7. The test patterns to the BUT must be generated such that same logic values are given as inputs to the complementary CLBs i.e. (CLB1 and CLB2), (CLB3 and CLB4). The reason for such arrangement can be understood by observing the truth tables of all the four gates as shown in table II. It can be observed that for the same input to the complementary gates the outputs are also complement to each other. Thus, when the outputs of the complementary gates are compared with a parity checker, any miss match in the output could be detected by the pass/fail (P/F) signal. It is also to be remembered that the eight inputs of the BUT are coming from the TPG.

TABLE I: Test patterns in LUT

Address	C1	C2	C3	C4	C5	C6	C7	C8
0000	0	1	0	1	0	1	0	1
0001	0	1	0	1	0	1	1	0
0010	0	1	0	1	1	0	0	1
0011	0	1	0	1	1	0	1	0
0100	0	1	1	0	0	1	0	1
0101	0	1	1	0	0	1	1	0
0110	0	1	1	0	1	0	0	1
0111	0	1	1	0	1	0	1	0
1000	1	0	0	1	0	1	0	1
1001	1	0	0	1	0	1	1	0
1010	1	0	0	1	1	0	0	1
1011	1	0	0	1	1	0	1	0
1100	1	0	1	0	0	1	0	1
1101	1	0	1	0	0	1	1	0
1110	1	0	1	0	1	0	0	1
1111	1	0	1	0	1	0	1	0

TABLE II. Truth Table comparision of BUT

I/P	AND	NAND	OR	NOR	PARITY
0000	0	1	0	1	2
0001	0	1	1	0	2
0010	0	1	1	0	2
0011	0	1	1	0	2
0100	0	1	1	0	2
0101	0	1	1	0	2
0110	0	1	1	0	2
0111	0	1	1	0	2
1000	0	1	1	0	2
1001	0	1	1	0	2
1010	0	1	1	0	2
1011	0	1	1	0	2
1100	0	1	1	0	2
1101	0	1	1	0	2
1110	0	1	1	0	2
1111	1	0	1	0	2

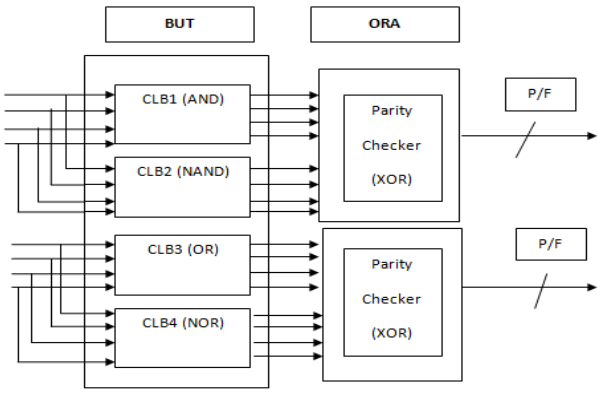

Figure 7: BUT and ORA STRUCTURE

When each CLB in the BUT is configured as complementary gates as described above, the logic values shown in table II will be placed on the LUT's of the respective CLBs. CLB1 can then detect a stuck-at-1 in the top 15 cells and a stuck-at-0 in the bottom cell.

The same analysis follows to other CLBs. To facilitate the complete detection of faults we swap CLB1 and CLB2.Same principle applies to CLB3 and CLB4.Thus all the 32 possible stuck-at-0/1 faults in a CLB can be detected.

IV. RESULTS AND DISCUSSIONS

To have a realistic platform the proposed methodology is implemented on a Xilinx virtex-4 FPGA. The BIST structure was described in Verilog HDL and synthesised by Xilinx XST synthesizer. The design is placed on the FPGA at the desired positions using place and route tool provided by the vendor .A verilog program is written to change the TCs. The faults are injected using FIAT tool and the results are observed at the ORA. Logic simulation is performed by Modelsim. For instance, figure 8 shows a stuk-at-0 fault considered at16th cell. The results prove 100% fault coverage and the total number of test configurations required are reduced to two. The location of the faults (diagnosis) can also be known by passing all the failed output results to a fault dictionary. When a fault is encountered the faulty response is compared with the dictionary contents and the reason for the faults can be traced. Also the propose methodology checks can detect the stuck open /close faults on local interconnects inherently.

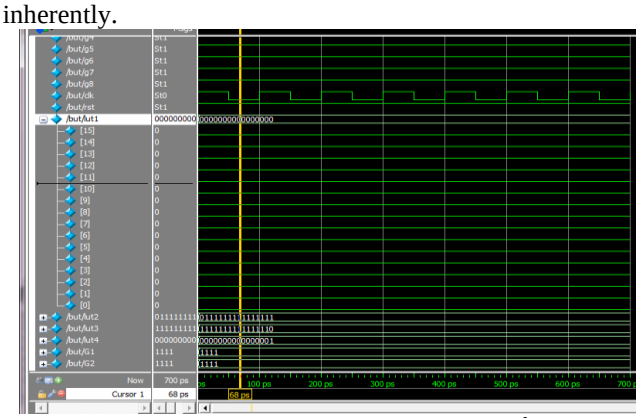

Figure 8: stuck-at-0 fault on 16th cell

V. CONCLUSION

This paper focused on detecting the stuck-at-0/1 faults on a RAM cell of an LUT by configuring the CLBs as complementary gates. Using the proposed methodology 100% fault coverage can be achieved. Besides, the number of test configurations is reduced to two.

REFERENCES

1. V. Betz, J. Rose, and A. Marquardt, *Architecture and CAD for Deep-Submicron FPGAs*, Boston, MA: Kluwer Academic Publishers, 1999.
2. O. Heron, T. Arnaout, and H. J.Wunderlich, "On the reliability evaluation of SRAM-based FPGA designs," in *Proc. Int. Conf. Field Programmable Logic Appl.*, Aug. 2005, pp. 403–408.
3. V. Betz, J. Rose, A. Marquardt, *Architecture and CAD for Deep-Submicron FPGAs.* Boston, MA: Kluwer Academic Publishers, 1999.
4. Xilinx Inc., "Virtex-4 FPGA User Guide," UG070 v2.5, 2008
5. L.Wang,C.Stroud and N.Touba, System on chip Test Architectures, Amsterdam: Elsevier,2007
6. M. Abramovici and E. Charles, "BIST-based test and diagnosis of FPGA logic blocks," *IEEE Trans. Very Large Scale Integr. (VLSI) Syst.*, vol. 9, no. 1, pp. 159–172, Feb. 2001
7. S. Xiaoling, J. Xu, and P. Trouborst, "Testing Xilinx XC4000 configurable logic blocks with carry logic-modules," in *Proc. IEEE Int. Symp. Defect Fault Tolerance VLSI Syst.*, Oct. 2001, pp. 221–229.
8. E. Atoofian and Z. Navabi, "A BIST architecture for FPGA look-up
9. table testing reduces reconfigurations," in *Proc. IEEE Asiau Test Symp.*,Nov. 2003, pp. 84–89.
10. S. K. Lu, J. S. Shih, and C. W. Wu, "Built-in self-test and fault diagnosis for lookup table FPGA," in *Proc. IEEE ISCAS*, May 2000, pp. 80–83.
11. P. Girard, O. Heron, S. Pravossoudovitch, and M. Renovell, "An efficient BIST architecture for delay faults in the logic cells of symmetrical
12. SRAM-based FPGAs," *J. Electron. Test., Theory Appl.*, vol. 22, no. 2,pp. 161–172, Apr. 2006.

AUTHORS PROFILE

Rajesh Adluri has completed Bachelor of Technology(B.Tech) Degree in Electronics and Communication Engineering(ECE) from JNT University, Hyderabad, India and Master of Technology(M.Tech) Degree in VLSI Design & Automation Techniques from NIT Hamirpur, India. He is currently working as Assistant Professor in the Department of Electronics and Communication Engineering, School of Technology, GITAM (Deemed to be University), Hyderabad, India. His research interests are in VLSI Designing and Testing

.

Jameer Basha SK has completed Bachelor of Technology(B.Tech) Degree in Electronics and Communication Engineering(ECE) from JNT University, Hyderabad, India and Master of Engineering(M.E) Degree in Embedded Systems from BITS,Pilani-Pilani Campus, India. He is currently working as Assistant Professor in the Department of Electronics and Communication Engineering, School of Technology, GITAM (Deemed to be University), Hyderabad, India. His research interests are in GPU Computing, Intersection of Embedded Systems & Deep Learning and Computer Architectures for High-Performance Computing.

Francis Xavier S is currently working as an Assistant Professor in the department of Department of Electronics and Communication Engineering, School of Technology, GITAM (Deemed to be University), Hyderabad, India. He has 15 years' experience in the field of communications and signal processing. His research interests are in signal processing and communications.

Haribabu.sanamu has completed Bachelor of Technology(B.Tech) Degree in Electronics and Communication Engineering(ECE) from Nagarjuna University, Guntur, India and Master of Technology(M.Tech) Degree in Electronic design and Technology from NIT Calicut-Calicut , India. He is currently working as Assistant Professor in the Department of Electronics and Communication Engineering, School of Technology, GITAM(Deemed to be University), Hyderabad, India. His research interests are in GPU Computing, Intersection of Embedded Systems and Computer Architectures for High-Performance Computing.

Retrieval Number: 100.1/ijitee.L79851091220
DOI: 10.35940/ijitee.L7985.1091220
Journal Website: www.ijitee.org

Published By:
Blue Eyes Intelligence Engineering and Sciences Publication