Australian Journal of Basic and Applied Sciences, 9(27) August 2015, Pages: 251-257

ISSN:1991-8178

Australian Journal of Basic and Applied Sciences

Journal home page: www.ajbasweb.com

FPGA based Self Healing Solution for Fault Tolerant Digital Circuits

[1]S. Meyyappan and [2]V. Alamelumangai

[1]*Assistant Professor, Dept. of E&I Engg., Annamalai University*
[2]*Professor, Dept. of E&I Engg., Annamalai University*

ARTICLE INFO

Article history:
Received 23 June 2015
Accepted 25 August 2015
Available online 2 September 2015

Keywords:
BIST, Fault injection, FPGA, LFSR, Self healing, TMR, VHDL.

ABSTRACT

The paper endeavours to unleash a design strategy to retain the true nature of the output in the event of occurrence of faults at the interconnect level of cascaded digital circuits. The operational pattern of the combinational circuit facilitates the creation of a self healing attribute and ensures the reliability of the digital architecture. The proposed scheme inserts faults randomly into the system at the interconnect levels and fosters to predict its behaviour in response to a fault. The scheme enjoys the benefits of employing software implemented fault injection in the form of an LFSR to aid in the process of injecting both stuck at 0 and 1faults. The design encompasses ways to intrigue the state of the intermediate signals and carries it with steps to rig out the true values at the primary output lines. The Modelsim based simulation results obtained for a decoder designated as the circuit under test emphasize the suitability of the scheme and avail the facilities of a FPGA processor to exhibit its practical viability.

© 2015 AENSI Publisher All rights reserved.

To Cite This Article: S. Meyyappan and V. Alamelumangai., FPGA based Self Healing Solution for Fault Tolerant Digital Circuits. ***Aust. J. Basic & Appl. Sci.,*** *9(27): 251-257, 2015*

INTRODUCTION

The performance of a digital system owes its competence to the selection of highly reliable components and the use of proven methods for their interconnection. The system requires an extensive verification of the logic design, programs and final hardware using simulation, diagnostic and functional tests in the expected environmental conditions. In spite of these reliability assurance techniques, the system may still fail during normal operation due to uncontrollable or undetectable faults. These are caused by undetected design errors, random failures of components or connections, and externally induced malfunctions during the operation of the system. An immediate way to cope up with faults in digital systems leads to incorporate fault tolerance thereby restoring the normal operation of the system even in the presence of faults (Lach, J. *et al*., 1998) and (Lala, P.K., 2001).

The complexity of present-day VLSI devices continues to rise and the chips are therefore becoming untestable by external Automated Test Equipments (ATE) (Lala, P.K. and B.K. Kumar, 2003) and (Lala, P.K., *et al*., 2006). The test lengths appear to rapidly increase in line with the testing times and the ATE memory requirements, in light of which the Built-In Self-Test (BIST) establishes itself as a necessary part of VLSI circuits. The circuit equips with it to test itself by BIST without using any ATE equipment or when used together with an external tester. The BIST is an on-chip testing system that generates test vectors to detect faults and elongates to verify the fault prone nature of the hard-ware.

A fault injection system provides the capability to induce a fault at any desired location and the philosophy allows faults to be injected at varying hierarchial levels of the testable system. The process of fault injection appears to be crucial in any digital measurement system. It enables the designer to introduce faults in the connecting lines to arbitrate on the need for a self healing strategy in order to make it fault tolerant. The faults in a testable system are assumed to be mainly single bit faults where a single bit is flipped from logic 1 to a 0 or vice-versa.

The theory of fault injection relates to the intentional insertion of faults into a system for the purpose of studying its response. The techniques for fault injection fall into three categories: (i) simulation based fault injection in which fault is injected into the simulation models of systems, (ii) software implemented fault injection that contains fault injection module in the VHDL description itself and (iii) physical fault injection capable of injecting faults into physical systems. The advantage of simulation-based fault injection extends to the fact that it can be used early in the design cycle.

However, the main drawback of simulation-based fault injection lies in the context that it is time-

Corresponding Author: S. Meyyappan, Assistant Professor, Dept. of E&I Engg., Annamalai University
E-mail: meys.narayan@gmail.com

consuming and the associated overhead limits the extent of usage, while injecting faults in the VHDL description offers many advantages that include platform independence and portability among design packages. A realistic fault injection system necessitates the capability to access most signals within the VHDL format including the inputs and outputs of the description which is crucial for both on- and off-line testing.

A detailed survey on fault injection techniques has been carried out (Haissam Ziade, *et al.*, 2004) with a purpose to bring out the relative merits and demerits of the various methodologies developed to inject faults into a system prototype or model. A probabilistic model has been developed (Trailokya Nath Sasamal and Anand Mohan, 2011) to derive the optimum faulty period in a digital system. The faults with transient nature have been injected using VHDL program and the results used to validate the distribution functions in the model.

A transient fault injection has been outlined at the VHDL level (Lala, P.K., 2012) to establish its ability to evaluate the testability for digital circuits. An FPGA based fault injection tool that supports synthesizable fault models of digital circuits have been implemented using VHDL (Swathi Rudrakshi, *et al*., 2012). The design has been validated using state machine based examples for different types of faults. A circuit instrumentation approach has been adopted (Civera, P., *et al*., 2001) to inject faults in the system emulated to FPGA. The effectiveness of fault injection environment has been tested to provide speed up factors over other fault injection techniques.

A switch architecture for concurrent testing and diagnosis for faults in multistage interconnection networks has been proposed (Minsu Choi, *et al.*, 2003). The compound effect of fault tolerant operation has been evaluated and the results show graceful performance enhancement due to fault tolerance. A partially self checking scheme for combinational circuits with concurrent error detection facility has been presented (Goran Lj. e*t al*., 2004). The results have been related to significantly reduce the area overhead in two level circuits. An FPGA architecture that is composed of functional cells has been discussed (Lala P.K., 2008) to bring out its error correction capabilities. An architecture that enables tolerance of single bit errors in a functional cell of the FPGA has been presented.

The occurrence of faults relate to unpredictable changes in the components of a logic circuit and permanently alter the logic function in the sense it may lead to deviations from the specified values of logic variables. Thus, an astute fault generator along with a facility to arrive at the correct state of the variable is of crucial significance to ensure the dedicated purpose of the digital system in use.

Proposed Methodology:

The main focus corners to formulate a sequence with a view to inject faults in the lines that interconnect the combinational circuits in a digital system. It examines the signal status to articulate a healing sense and there from arrive at the error free value of the measured variable. The scheme derives the benefits of the Modelsim platform to implement the methodology on a Spartan board to eschew its applicability in the practical world.

A key factor in the design of fault-tolerant digital systems owes its ability to address issues of dependability requirements. The fault injection defines itself to be an effective method to study error behaviour, measure the dependability parameters and evaluate fault-tolerant digital systems.

Generally, highly critical applications rely on hardware redundancy to increase their reliability. The best known Triple Modular Redundancy (TMR) achieves fault tolerance without actually detecting any fault. In this method, each module, which may be a complete system, such as a computer, or a less complex unit, like a microprocessor or even an adder or a gate replicates itself three times. The voting element collects the outputs from the three sources and delivers the majority vote at its output.

The choice of a decoder as Circuit Under Test (CUT) owes to its exclusive dependence in any digital measuring environment and the fact that the philosophy can easily relate to other circuits. The expressions given below establish the relationship between the inputs and the desired outputs of the 2:4 decoder in the fault free state.

$$Dout(0) = (\overline{Din(1)}\ \boldsymbol{and}\ \overline{Din(0)})$$
$$Dout(1) = (\overline{Din(1)}\ \boldsymbol{and}\ Din(0))$$
$$Dout(2) = (Din(1)\ \boldsymbol{and}\ \overline{Din(0)})$$
$$Dout(3) = (Din(1)\ \boldsymbol{and}\ Din(0))$$

When the CUT operates in fault free state, it produces the output as expected for a given input combination in accordance with its input – output relationship. On the other hand, the appearance of stuck at faults at the intermediate lines causes the circuit to generate faulty output. Fig. 1 depicts one such turbulent state of the CUT in which the interconnect levels $\overline{Din(0)}$ is stuck at 1 and Table 1 displays the faulty operating condition of the circuit in consequence of the presence of stuck at fault occurred at the interconnect levels.

Australian Journal of Basic and Applied Sciences, 9(27) August 2015, Pages: 251-257

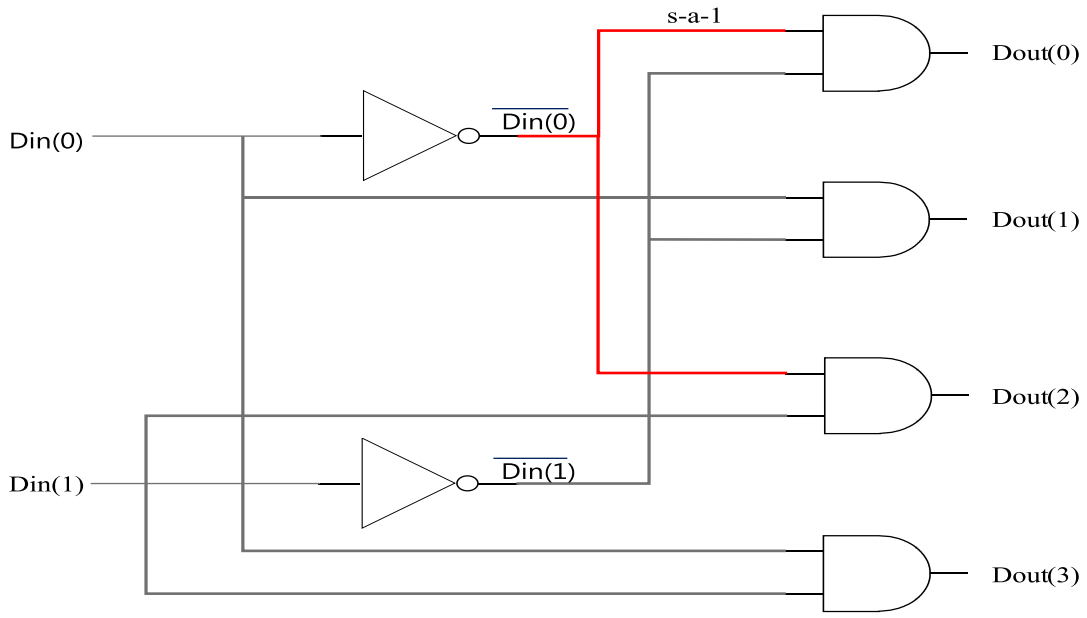

Fig. 1: The CUT in the presence of stuck at faults

Table 1 Faulty operating condition of the CUT

Din(1)	Din(0)	$\overline{Din(1)}$	$\overline{Din(0)}$	Dout(3)	Dout(2)	Dout(1)	Dout(0)
0	0	1	1	0	0	0	1
0	1	1	(1)	0	0	1	(1)
1	0	0	1	0	1	0	0
1	1	X	X	1	0	0	0

(1) - Stuck at 1

The expressions presented below represent the operational behaviour of the proposed self healing scheme which houses three identical copies of the CUT along with the constituent components of the voting system. The design exhibits excellent resilience and possesses the ability to provide the correct values on the primary output lines even in the presence of faults as displayed in Table 2 in which two forms of stuck at faults are considered for instance with a view to articulate the fault tolerant capability of the proposed approach.

$$Dout(0) = \left(D1(0)\ \boldsymbol{and}\ D2(0)\right)\ \boldsymbol{or}\ \left(D1(0)\boldsymbol{and}\ D3(0)\right)\ \boldsymbol{or}\ \left(D2(0)\ \boldsymbol{and}\ D3(0)\right)$$
$$Dout(1) = \left(D1(1)\ \boldsymbol{and}\ D2(1)\right)\ \boldsymbol{or}\ \left(D1(1)\boldsymbol{and}\ D3(1)\right)\ \boldsymbol{or}\ \left(D2(1)\ \boldsymbol{and}\ D3(1)\right)$$
$$Dout(2) = \left(D1(2)\ \boldsymbol{and}\ D2(2)\right)\ \boldsymbol{or}\ \left(D1(2)\boldsymbol{and}\ D3(2)\right)\ \boldsymbol{or}\ \left(D2(2)\ \boldsymbol{and}\ D3(2)\right)$$
$$Dout(3) = \left(D1(3)\ \boldsymbol{and}\ D2(3)\right)\ \boldsymbol{or}\ \left(D1(3)\boldsymbol{and}\ D3(3)\right)\ \boldsymbol{or}\ \left(D2(3)\ \boldsymbol{and}\ D3(3)\right)$$

Table 2: Signals flow of the fault Tolerant decoder

Inputs			Interconnect Levels			outputs
D1(0)	D2(0)	D3(0)	A D1(0) AND D2(0)	B D1(0) AND D3(0)	C D2(0) AND D3(0)	A OR B OR C Dout(0)
0	0	0	0	0	0	0
(1)	0	0	0	0	0	0
1	1	1	1	1	1	1
1	1	(0)	1	0	0	1

(0) - Stuck at 0
(1) - Stuck at 1

The theory of Linear Feedback Shift Registers (LFSR) treads to generate pseudo-random numbers that can be used as test patterns in logic circuit testing. The outputs of a selected number of stages in

a shift register connected to its input through an EX-OR network form the LFSR. The output of any stage results as a function of the initial state of the bits in the register and of the outputs of the stages in the feedback path. The selection of feedback paths thus becomes crucial in the construction of an LFSR to allow it to perform in tune with the design.

The fig. 2 displays the logic structure of a 4-bit LFSR capable of generating maximal length sequence for a given seed value while the polynomial of *degree 4* shown below holds the responsibility for the creation of maximal length sequence in the sense it makes the LFSR generate $2^n - 1$, i.e. $2^4 - 1 = 15$ different pseudo random binary patterns.

$x^4 + x + 1$

which can be rewritten as

$a(x) = 1 * x^4 + 0 * x^3 + 0 * x^2 + 1 * x + 1$

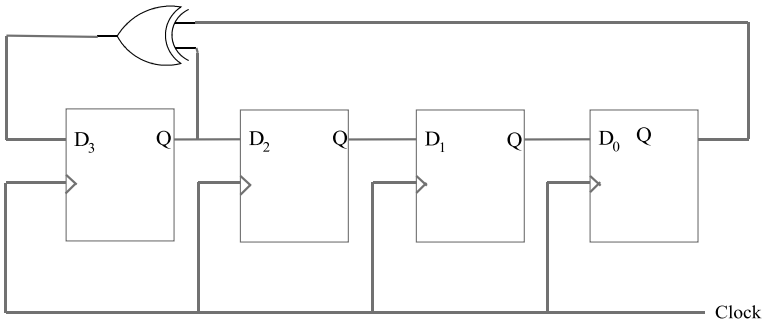

Fig. 2 Logic structure of 4-bit LFSR

The scheme relieves itself from the tedious and tiresome process of manually injecting faults during the time of simulation by employing a four bit LFSR as the fault injection module. The presence of a LFSR leads to introduce faults into the CUT and bring out the random nature of the occurrence of faults in a real time environment. The creation of test patterns adds a new dimension to the approach in the sense it forges the unintended nature of the behaviour of the digital systems and carries the mechanism to perform correctly even under extraneous circumstances.

The methodology explained using the flow diagram in Fig. 3 pronounces a procedure to inherently introduce faults in multiple interconnect levels of the CUT and interleave out a self healing stream to arrive at a fault tolerant circuit.

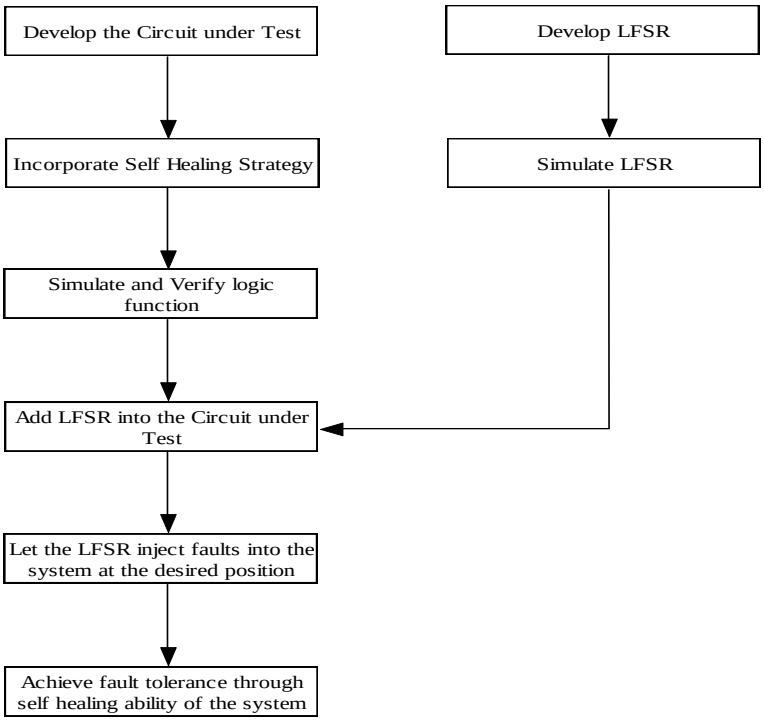

Fig. 3: Flow diagram of the fault tolerant design

Simulation Results:

The modern digital systems experience the merits in the use of high level language such as VHDL. The actual implementation of the system follows the specification and reverberate the need for the compliance of testability, power consumption and a sense of reliability. The ability to simulate the occurrence of a fault in the VHDL description of a circuit assumes extreme importance to heave the circuit with a built-in on-line fault detection capability. The internal signal requires to be accessed at the VHDL level for the purposes of injecting faults to ensure greater coordinative characteristics for the system (Luis Entrena, *et al.*, 2001).

The strategy avails the role of Modelsim platform to illustrate the veracities through VHDL description. The signals seen in Fig. 4 relates to the events take place when the LFSR is connected in place of the first decoder to elucidate the operational behaviour the proposed self healing system against the injection of faults into the CUT. The Figs. 5 and 6 echo the similar events when the LFSR is connected in place of the second and third decoders respectively. The deployment of the LFSR as the fault injector projects the ability of the design to randomly bring in faults into the CUT and still ensure the desired primary output of the system. The Figs. 4 to 6 elucidate the innate ability of the scheme to counter the presence of faults introduced naturally by LFSR and proclaim the fact that the proposed scheme exhibits total insensitivity to faults and comes up with true values on its primary output lines even in erroneous operating conditions.

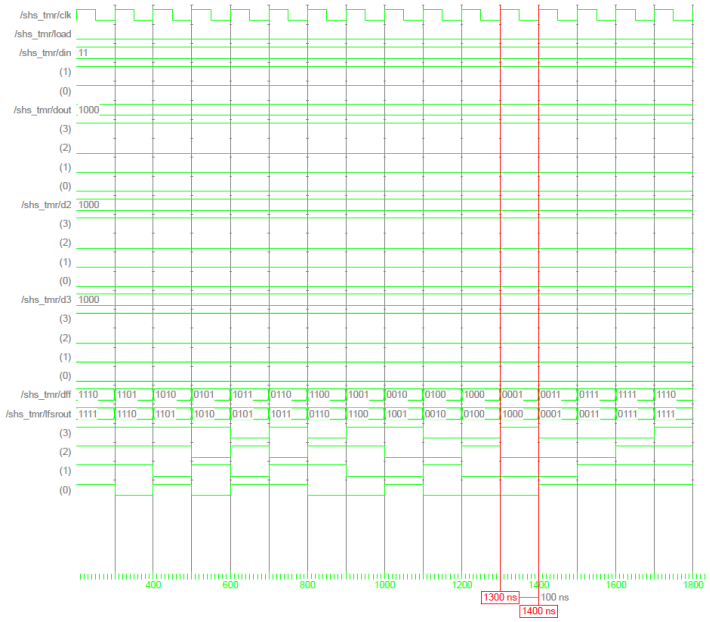

Fig. 4: Output of the self healing system with faults in the output lines of the first decoder

Fig. 5: Output of the self healing system with faults in the output lines of the second decoder

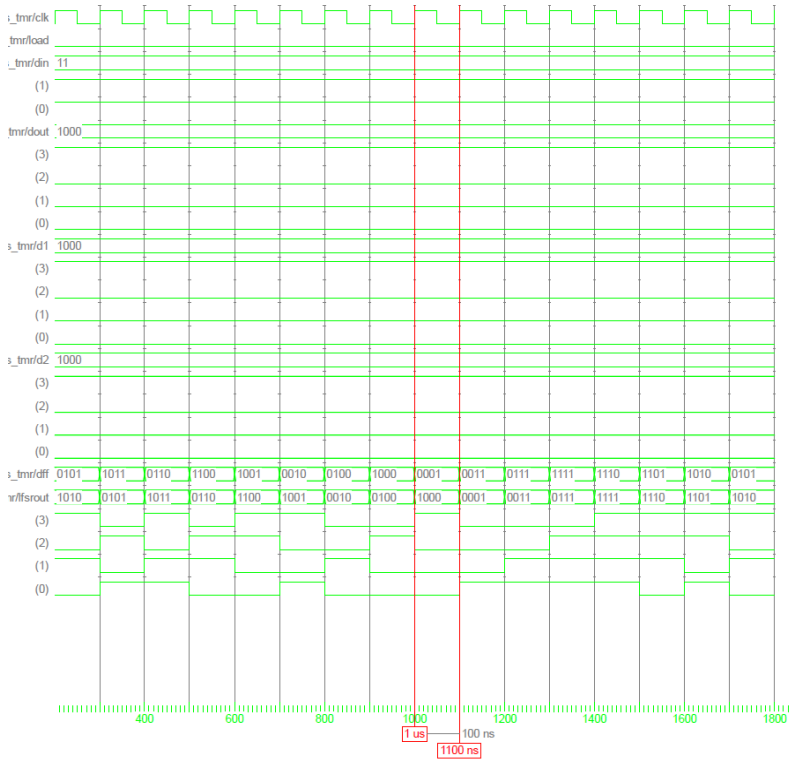

Fig. 6: Output of the self healing system with faults in the output lines of the third decoder

Hardware Implementation:

The hardware redundancy appears to be a preferred choice to ensure the reliability of dependable systems in highly critical applications. The FPGAs in light of their inherent runtime reconfigurable ability and function independent stature acclaim their suitability for fault tolerant implementations (KShirsagar, R.V., R.M. Patrikar, 2008).

The RTL schematic shown in Fig. 7 displays the components that constitute the self healing architecture. The XC3S500E FPGA synthesizes the VHDL code generated to realize the fault tolerant design on the Xilinx Foundation 9.2.i platform. The very fact that the synthesizable code serves to validate the cohesive nature of the strategy and authenticates the use of the proposed approach for tracing the real output in the event of fault interruptions in a digital system.

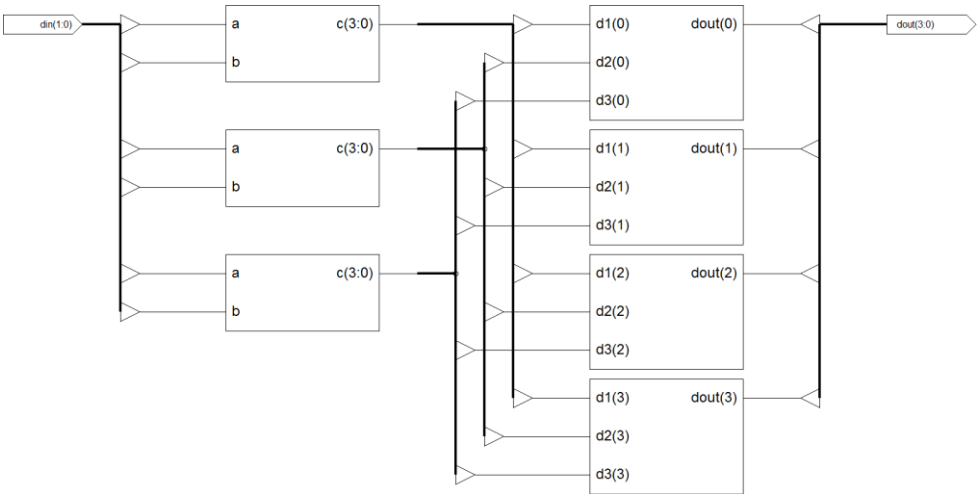

Fig. 7: RTL schematic of the fault tolerant decoder

Conclusion:

The artefacts of the basic operational theory behind the TMR have been eschewed to articulate a self healing solution for combinational digital circuits. The formulation has been enlivened through the natural introduction of faults with the use of an LFSR on one or more lines over which the signal flows. The benefits of simulated fault injection approach have been reaped in the process of fault generation. The Modelsim based simulation results have been obtained from the VHDL code tailor made to trace the fault free output for the multiple combinations of testable patterns in the chosen decoder. The capabilities of XC3S500E FPGA have been sought to encrypt a synthesizable RTL circuitry and encompass an environment suitable to validate the proposed scheme. The inert facility to address stuck at faults in the chosen digital architecture has been focussed to reiterate the fault tolerant aspect and acclaim for itself a wider spectrum of use in a band of digital circuit related paraphernalia.

REFERENCES

Civera, P., Macchiarulo, L., Rebaudengo, M., Sonza Reorda, M and Violante, M., 2001. “Exploiting FPGA-based Techniques for Fault Injection Campaigns on VLSI Circuits”, Torino, Italy.

Goran Lj. Djordjevec, Mile K. Stojcev and Tatjana R. Stankovic, 2004. “Approach to Partially Self-Checking Combinational Circuits Design”, Microelectronics journal, 35(12): 945-952.

Haissam Ziade, Rafic Ayoubi, and Raoul Velazco, 2004. “A Survey on Fault Injection Techniques”, The International Arab Journal of Information Technology, 1(2).

Lach, J. and W.H. Mangione-Smith and M. Potkonjak, 1998. “Low Overhead Fault-Tolerant FPGA Systems”, IEEE Trans. on VLSI Systems, 6(2): 212-221.

Lala, P.K., 2012. “Transient and Permanent Fault Injection in VHDL Description of Digital Systems”, Circuits and Systems, 3: 192-199.

Lala, P.K., 2001. “Self-Checking and Fault Tolerant Digital Design”, Morgan Kaufmann publishers.

Lala, P.K. and B.K. Kumar, 2003. "An Architecture for Self-Healing Digital Systems," Journal of Electronic Testing:Theory and Applications, 19: 523-535.

Lala, P.K., B. Kiran Kumar, J.P. Parkerson, 2006. “On Self-Healing Digital System Design”, Microelectronics Journal, 37(4): 353-362.

Lala, P., 2008. “On FPGA Design with Self Checking and Fault Tolerance Capability”, International Conference on “Engineering of Reconfigurable Systems and Algorithms.

Luis Entrena, Celia López and Emilio Olías, 2001. “Automatic Insertion of Fault-Tolerant Structures at the RTL Level”, International On-Line Testing Symposium – IOLTS, pp: 48-50.

Minsu Choi, Nohpill Park and Fabrizio Lombardi, 2003. “Modeling and Analysis of Fault Tolerant Multistage Interconnection Networks”, IEEE Transactions on Instrumentation and Measurement, 52(5).

KShirsagar, R.V. and Patrikar, R.M., 2008. “A New Design Approach For Tolerating Inter connect Faults in Digital Circuits”, Proceedings of SPIT-IEEE colloquium and international conference, Mumbai, India, 2: 117-120.

KShirsagar, R.V. and Patrikar, R.M., 2008. “A Novel Fault Tolerant Design and an Algorithm for tolerating Faults in Digital Circuits”, 3rd International Design and Test Workshop, Monastir, Tunisia.

Swathi Rudrakshi, Vasujadevi Midasala and Naga Kishore Bhavanam, S. 2012. “Implementation of FPGA based Fault Injection Tool (FITO) for Testing Fault Tolerant Designs”, IACSIT International Journal of Engineering and Technology, 4(5).

Trailokya Nath Sasamal and Anand Mohan, 2011.“A Specially Designed Transient Faults Injection Technique at the VHDL Level and Modelling”, IJRRAS 9(2).