ANALYSIS OF FAULTS IN AN N-BIT SELF CHECKING REGISTER

T. Shunbaga Pradeepa[1] and S. Uma Maheswari[2]

[1]Assistant Professor, Department of Electronics and Communication Engineering, Coimbatore Institute of Technology, Coimbatore
[2]Professor, Department of Electronics and Communication Engineering, Coimbatore Institute of Technology, Coimbatore

ABSTRACT

Soft errors which are random errors induced by radiations may be produced due to transient faults and upsets in electronic systems. From the survey, it has been observed that the existing error correcting techniques and models have some limitations. The conventionally used error detection method named Triple Modular Redundancy (TMR) method has large overhead which makes it uneconomical. In this paper, the existing techniques like Time Redundancy based error Detection (TRDED) has been implemented and verified for different intervals of errors. It has been observed that only particular errors can be detected and no corrections are done. The modified circuits abbreviated as SETTOFF can be used for Soft Error and Timing Error Tolerant Flip Flop. These circuits which have both error correction and detection has been implemented and verified for different intervals of time. Since the chances of induced errors are increasing, there is a great necessity for developing a technique to provide more reliability and performance. Targeting towards the above features, self-checking register architecture for multi-bit error detection has been proposed and analyzed using Xilinx ISE Simulator for transient fault occurrence and has been analyzed.

KEYWORDS

Transient Fault, Self-checking register, Single Event Upset (SEU), Multi bit error detection, Single Event Transient (SET)

1. INTRODUCTION

Errors that occur randomly are Soft errors. They are induced by radiations that may be produced due to transient faults and upsets in electronic systems. Transient faults occur in 2 different ways. One is Single Event Upset that will change the state value in the storage cell. The other is Single Event Transients which generate transient voltage pulses in combinational gates. Memory elements are easily affected by SETs and sometimes they turn into soft errors. SEUs are a major concern in both dense memory arrays and sequential logic. The SEUs can be protected efficiently by conventional Error Correcting Code (ECC) techniques. But, since ECC is distributed across the entire system, they are not applicable in sequential logic. Therefore, there exists a challenge in achieving efficient error mitigation in general logic. The most widely used safety technique is Triple Modular Redundancy (TMR). This method eliminates errors in general logic. Although TMR is highly reliable, it requires large area which makes it uneconomical for most non-safety-critical electronics. Demands in Technology and customer are pushing performance and energy efficiency. However, the soft errors are becoming a major concern at the same time. To balance these conflicts, it is better to provide a convenient safety method in supporting non-safety-critical electronics. The First contribution of the work is the design and implementation of an error

N. Ramasubramanian et al.(Eds) : SRAIC-2019
pp. 01-17, 2019. © CS & IT-CSCP 2019 DOI: 10.5121/csit.2019.91501

tolerant D flip flop. In this method, to analyze cost-efficient error-tolerance in general logic a new design was done. The design is named Soft Error and Timing Error Tolerant Flip Flop. It is abbreviated as SETTOFF. This design can correct error upsets and detects transient errors. The errors that occur naturally are Timing Errors (TE).

The objective of this work evolves from the great interest in developing a technique to provide a better performing safety method that supports non-safety-critical electronics and in achieving more reliability and performance in detecting and correcting errors due to upsets and transients. Most sequential circuits do not have error correcting capability. So, they are easily induced to soft error especially in case of redundant circuits. If the combinational logic blocks with redundancies are unprotected, it will produce SET pulses which in turn may lead to occurrence of errors. If the redundancy is stable the particle striking can produce SEU as like in latch. So there exists a great need to achieve efficient error mitigation logic. The existing techniques use replication to improve the error tolerance level in any electronic system. But there exists some drawbacks. The techniques covers the errors occurred in the actual circuit but if they occur in the safety replication module those errors are not protected. Those circuits are not checking themselves. It has become essential for Integrated Circuits to have some kind of circuits that detects soft errors as well as timing faults. Such protection is needed for all designs. One such tolerant design is TMR, but this seems to be costly.

Improvement in soft and timing error detection using time redundancy method have been implemented in Lorena Anghel and Michael Nicolaidis [4] based on time redundancy. Lin, M. Zwolinski, and B. Halak [15] have discussed a new architecture for Flip-Flop called SETTOFF which improves circuit performance to radiation hits against the existing ones. But, the cost area and performance are high. Yang Lin et al. has proposed [14] a technique to check circuits by itself for soft error based on SETTOFF. Sheng Lin et al. has proposed [11] circuits based on Schmitt trigger. The circuit uses conventional latch which increases the area consumed. Hsuan-Ming Chou et al. has presented [3] a design to protect from soft error targeting different applications with trade-off in performance, power, and reliability.

The flow of paper is like: Section 1 describing the overview and its related literature survey. Section 2 describes about the triple modular redundancy fault tolerant techniques. Basic idea about transient fault detection is given in Section 3. The study SETTOFF is briefed in Section 4. The architecture of proposed self-checking register is described in Section 5. Section 6 includes the implementation results of this work followed by conclusion in Section 7.

2. ARCHITECTURE OF TRIPLE MODULAR REDUNDANCY

Triple modular redundancy (TMR) is a method adopted in early days to obtain better safety system. The system has a majority voter. It reads data from three duplicate circuits. It then compares for majority of the outputs.

2.1. TMR Systems

TMR is a kind of fault tolerant system that is implemented in most computing system in the form of N modular redundancy. As the value of N is assumed as 3 meaning Tri, it is called Triple Modular Redundancy. This method has three systems performing the same process and the result of the process is taken by considering the majority value of the output. In this if one system is faulty, the other two systems will mask the fault in the system and it will correct the error automatically. The TMR is applicable to many redundant forms and also it is applicable in Error Correction Codes. TMR makes use of three identical and redundant form of the original system to

compute its output. If there are no errors or faults in the systems all the redundant modules will produce the same output. If there are any errors then the outputs will be different.

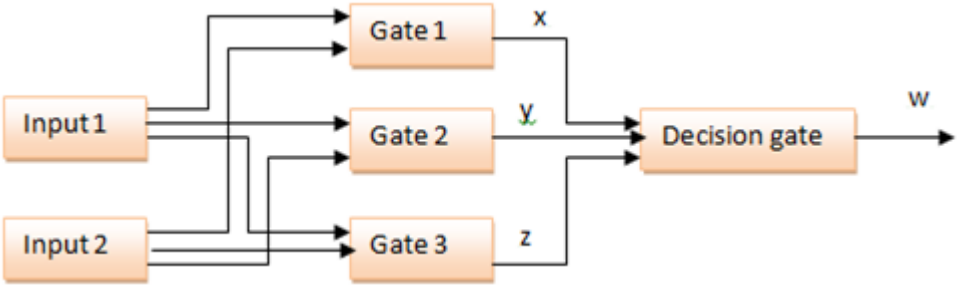

Figure 1. Triple Modular Redundant architecture

The circuit in Figure 1 represents the Triple Modular Redundant architecture where majority logic is used for finding the exact output in the circuit. If the circuits work properly without error, the outputs are same. The outputs will be different if the circuit has any failure. Majority gate is used in the circuit. It will help in finding the actual output. The logic output will be high value if more than two outputs are logic high. The logic output will be a low value if more than two inputs are logic low. The logic equation of the majority gate can be given as in Equation (1) where x, y and z are the inputs and w is the output of the majority gate.

$$w = (xy) \text{ or } (yz) \text{ or } (xz) \quad (1)$$

This has used AND logic as well as OR logic. Essentially the majority logic gate is a voting mechanism.

Consider the following scenarios that might occur in the majority gate. Let's say that logic 1 represents that there is no failure and logic 0 means the system has some failure.

Case (i) if there are no errors in the system, then all three modules will produce an output of 1, and the majority gate also produces a value of 1.

Case (ii) if any failure occurs in one of the modules then it produces an output of 0, but the other two modules are error free and produces an output of 1, the majority gate produces an output of 1. It is seen that even if one of the modules fail the error is masked by the other two modules.

Case (iii) if all the modules are producing an output of 0 then it will be reflected in the majority gate output. Conventionally, a fault-tolerant machine uses replicated elements which are operating parallel.

The TMR may be a robust form of error correction scheme but it doesn't indicate in which module the error has occurred. Also replication of the modules three times increases the hardware size required for fault free implementation. Although TMR is highly reliable, the large area consumption makes it uneconomical for most non-safety-critical electronics applications. There are other techniques to achieve cheaper solutions, but they are normally less reliable than TMR.

3. TIME REDUNDANCY BASED ERROR DETECTION

The Time Redundancy Based Detection abbreviated as TRD technique detects SET. It is found at the input of a flip flop by comparing the sampled data at two time instants delayed by delta. The error tolerance overhead of TRD is small as there is no duplication done. An SET pulse whose width is not greater than delta can be detected since it doesn't overlap the two time instances. The

TRD technique will detect timing errors that are caused due to previous logic modules, and the SEU occurring in the flip flop before the second time instance. Since the triplication is done the correct value is easily identified. The hardware redundancy achieves good tolerance in transient fault detection. The system is costly as because it is not suitable in commodity products. So, an alternate method of injecting transient faults is used.

3.1. Operating Principle

SET occurring in logic blocks can be corrected by themselves within a short period of time. They also recover automatically. There is no hardware duplication in the technique shown in Figure 2. TRD can detect SET for an the input of the flip flop whose pulse width can be maximum of Dtr <= delta - Dsetup - Dcomp,

where Dsetup is the time for setup of the error flip flop and Dcomp is comparator delay. The same SET if captured by the main flip flop at a time interval of t0, can be recovered at

t0 + δ - Dsetup - Dcomp , whereas the comparator will produce an error output when the inputs are not consistent.

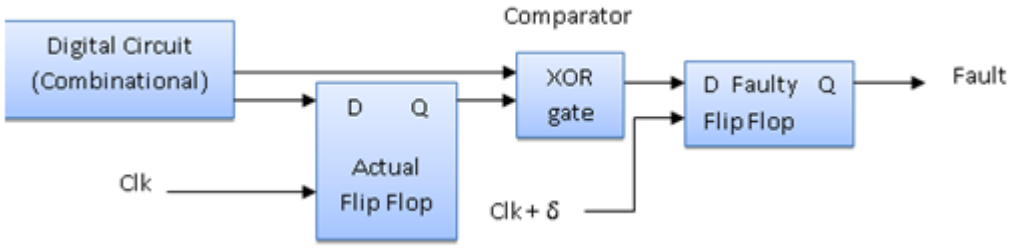

Figure 2. TRD Flip Flop

When the delay of TE is not greater than Dtr the fault can be detected and corrected for the input D. This architecture can also detect SEU in the main flip flop from t0 to t0 + δ − Dsetup − Dcomp, which is called the TRD interval. TRD can detect but cannot correct errors. SEU in the main flip flop but outside the TRD interval cannot be detected by this module.

4. ERROR TOLERANT FLIP FLOP FOR SOFT AND TIMING ERROR

SETTOFF [15] overcomes the drawbacks of previous techniques and achieves a higher error-tolerance with lower cost. The errors occurring during a write cycle of SETTOFF are detected and are easily corrected. Other errors which corrupt the data stored in SETTOFF are detected and corrected internally. If these errors are found to occur during a hold cycle, it is difficult to find it. The SETTOFF architecture is shown in Figure 3. The main flip flop is a normal flip flop. The last stage element has a pair of inverters. These inverters drive the output of the flip flop. They are now replaced by a correction XOR-gate. Therefore, in normal operation, the output variable Q is inverse of actual node N. The TRD interval of the clock phases high and low are as shown in Figure 4.

Module I is an adapted TRD architecture. It contains a XOR gate for detection and delayed clock is used to drive faulty Flip-Flop. The delay element δ is the sum of Dhclk (period of the high clock phase), Ddxor (delay of XOR-gate detector) and Dsetup (faulty Flip-Flop set up time). The TRD interval is equal to δ - Ddxor - Dsetup= Dhclk. During the write cycle of the main flip flop the error flip flop is enabled. Module I detect the type of error based on interval of TRD. It will be detected as SET if the L1 stage has a pulse width which is not greater than Dhclk. It detects as

timing error when a delay of module is not more than Dhclk and it detects as SEU if there is a change of state in node N during Dhclk. The error flip flop generates a signal when error is detected. Module I detects errors occurring during the write operation. Module II detects and corrects SEU occurring during the other half of the clock phase. Transition detector is present which monitors the internal node N. There is an XOR-gate which is used for correction purpose. Those SEU that corrupt the last stage of actual flip flop are considered, whereas others are masked. When TD is disabled the output (Error_SEU_bar) will remain high. This indicates errors are not present. The correction XOR-gate inverts N to Q.

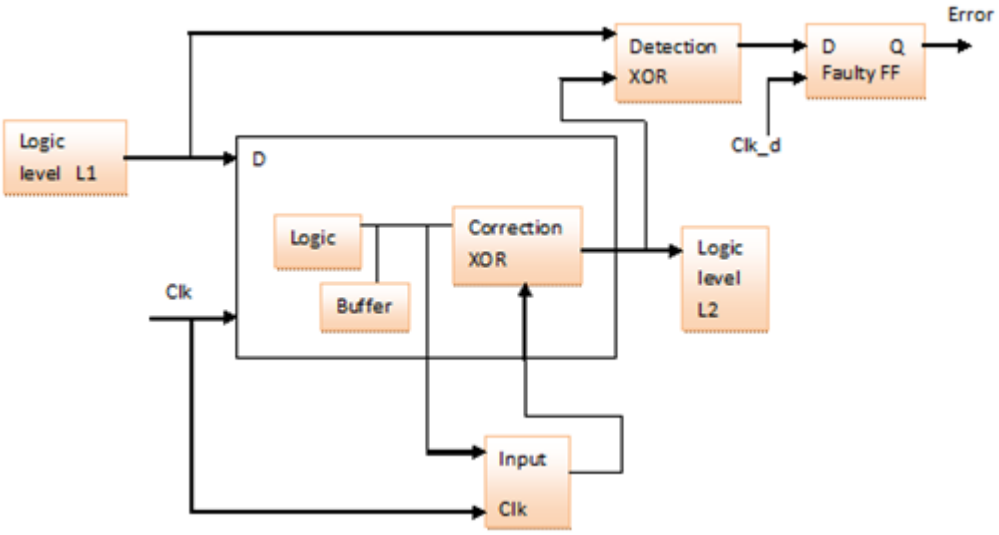

Figure 3. SETTOF Architecture

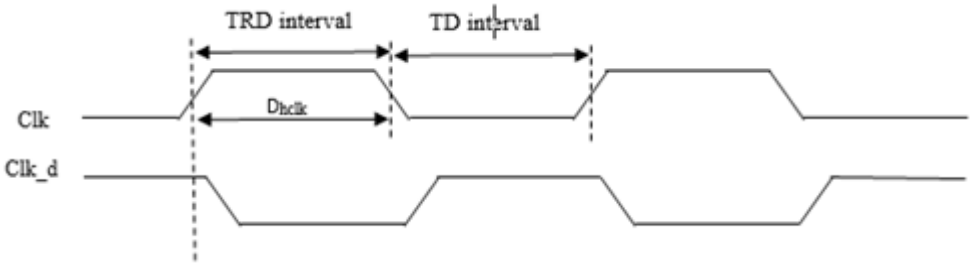

Figure 4. SETTOFF's TRD and TD interval [13]

To further illustrate this, consider the three conditions shown in Figure.5.

Condition (i) Consider that SEU is correcting while writing in the flip flop, it will capture the input and checks if there have been any change in the bit value in the rising edge of the clock. Also, Error_SEU_bar is asserted and make the correction XOR-gate invert N to Q.

Condition (ii) the next case is when SEU is correcting when it is holding the data. Flip flop typically holds any of the two architectures either multiplexer based architecture or the clock gating based architecture. In a multiplexer based architecture, the input is in a hold cycle if the Flip-Flop captures an error, but the output Q is selected by a multiplexer to feed back into the input D. The Flip-Flop then captures the corrected Q to overwrite the SEU stored in the last inverter pair during the hold cycle. Error_SEU_bar is set to 1 at the same time.

Condition (iii) in a clock-gating-based architecture, the driving clock in the flip flop is gated where there is no input capturing in hold cycle. The bit-Flip error remains at node N. This ensures that the flip of the bit remains corrected at Q. The process generates a correction glitch in the output of SETTOFF due to the delay in the propagation of the TD. The width of the fault is very small. This is due to the fact that the TD is relatively fast and the correction process is incorporated in the flip flop architecture. Even if the correction fault propagates, it is not fatal. SETTOFF has the ability of both error detection and correction. Some cases of errors have been detected and corrected using the modified D Flip Flop. SETTOFF can efficiently tolerate error upsets and timing errors.

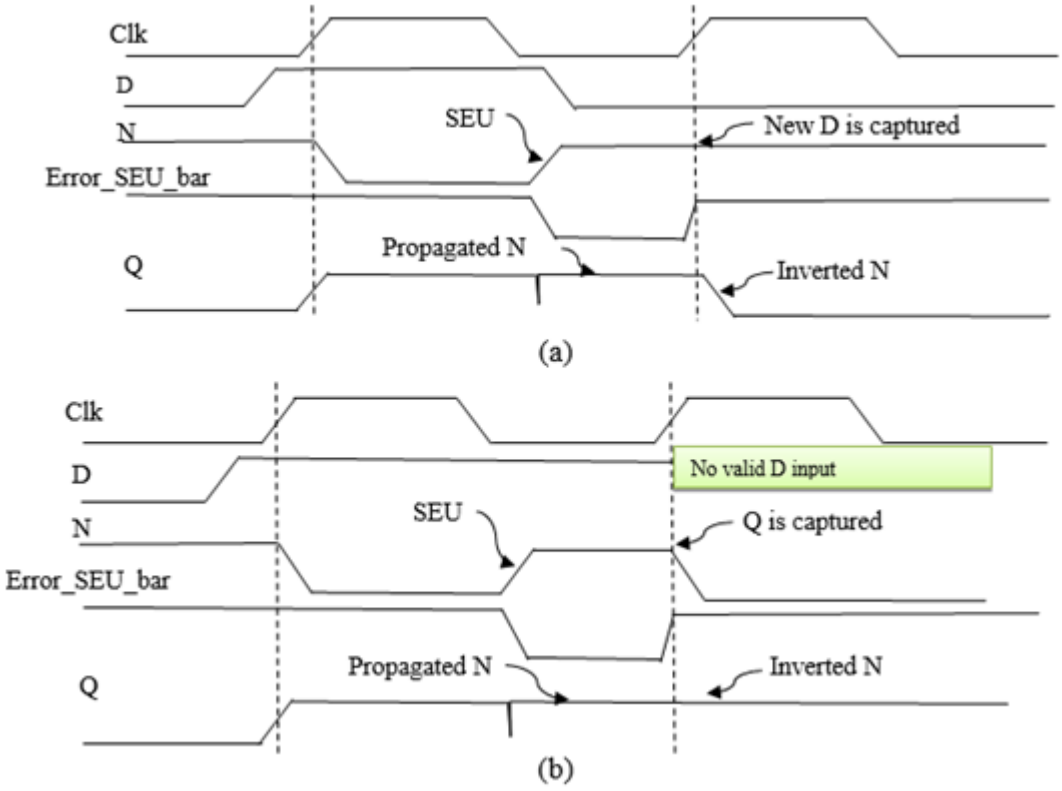

Figure 5. SETTOFF Operating principle [13]

5. SELF- CHECKING ERROR TOLERANT REGISTER

Self-checking capability is not available in almost all pipeline protection techniques and hence they are easily affected by soft errors. The probability that the circuit is affected is determined by the area and size of the redundancies. But, the charge is used to determine the circuit's vulnerability. If the redundant modules are unprotected then ECC is needed at appropriate stages. The proposed technique takes has a self checking method implemented in register architecture.

Then bit self checking register [14] is shown in Figure 6. It has n SETTOFF blocks which are shared with a self checker block. It also has a glitch filter (GF), and also includes a TD checker. The errors are combined together and are stored in the flip flop. As Module II is affected the error affects the output of a TD. The presence of self checker makes the process of monitoring the outputs. It then detects errors in Module II when there are any faulty transitions at these outputs.

The outputs of each bit are connected to the parity checker through n input XOR tree. The output is logic high when a fault is detected. This fault detection is made to pass through the Glitch

Filter. These transients may in turn induce glitches in the parity checker [16] output. The glitches are filtered and prevent them from passing. The errors that are detected can also be corrected.

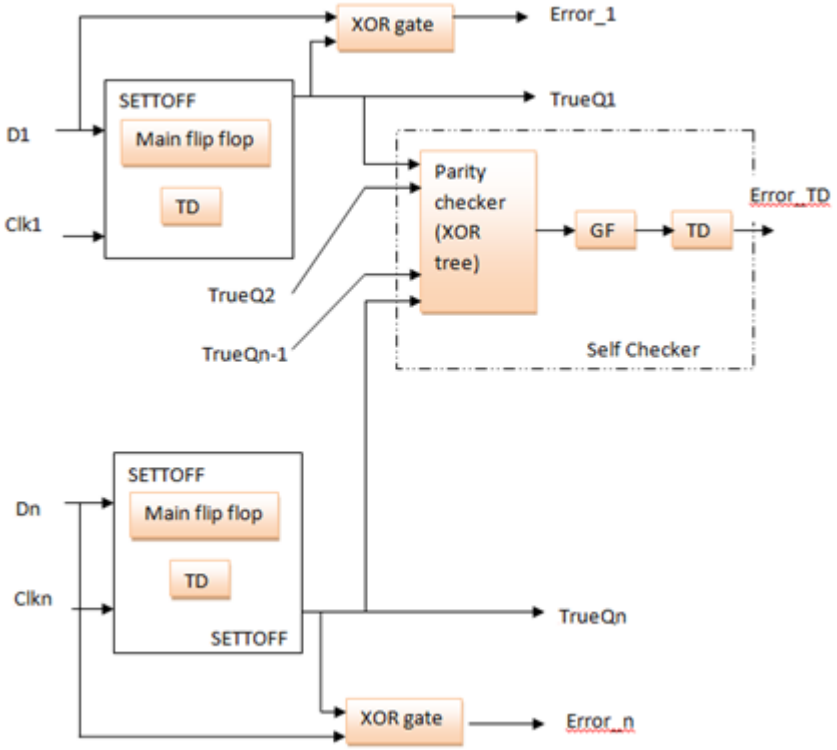

Figure 6. n-bit self-checking register

5.1. Proposed Self-Checking Register for Multi-Bit Error Detection

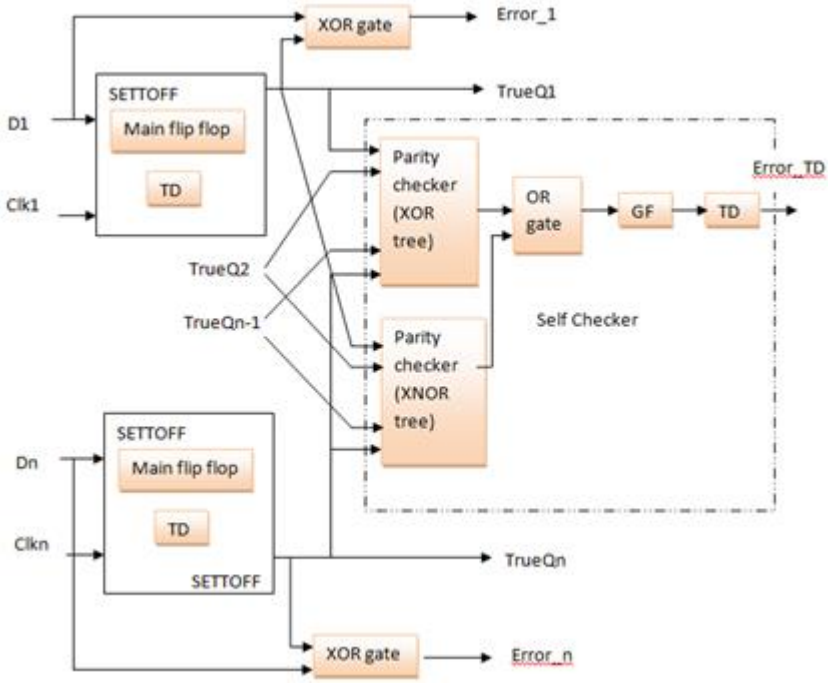

Figure 7. Proposed n-bit self-checking register with multi-bit error detection

The Figure 7 gives the n bit self-checking register architecture block. It has n SETTOFF and also has a self checking mechanism inbuilt as already in the Figure 6. This modified self checker can detect both odd and even number of soft errors. A self checker which is inbuilt monitors the outputs of all SETTOFF. The presence of parity checker helps in detecting odd number of errors using n input XOR tree. To detect even number of errors an n input XNOR tree is used. If errors are detected, the parity checker circuit generates a change in state. Thus, Self checking register gives more protection in pipeline architectures. The self checker has the capability of monitoring the outputs of each Flip Flops.

6. RESULTS AND DISCUSSIONS

The simulation results obtained in implementing various error tolerant techniques including time redundancy based error detection SETTOFF and the proposed self checking register for pipeline architectures are presented. Error tolerance analysis has been done using Xilinx ISE and Microwind Dsch tool was used for circuit analysis of SETTOFF.

6.1. Error Tolerance Analysis

The discussed error resilient techniques were analyzed for different time intervals of forced errors.

6.1.1. Time Redundancy based Error Detection (TRD)

Figure 8 shows the RTL schematics of the TRD Flip Flop. It consists of an XOR gate with D Flip Flop. The XOR gate acts as the comparator. Comparator generates high value when an error is found.

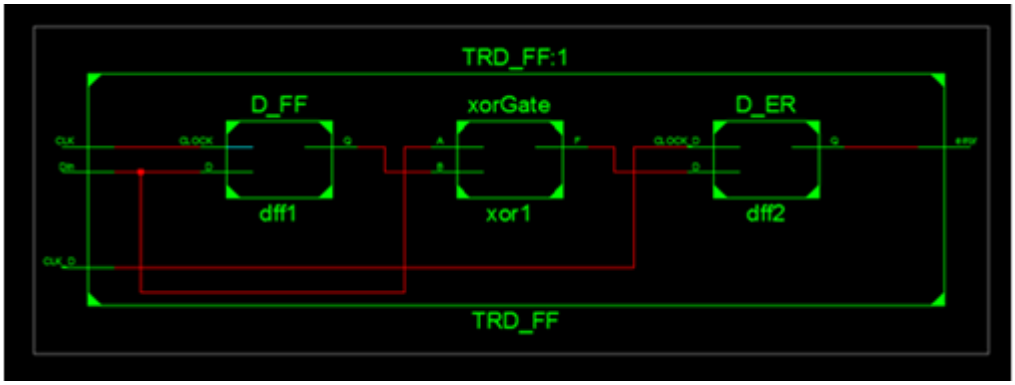

Figure 8. RTL schematics of TRD

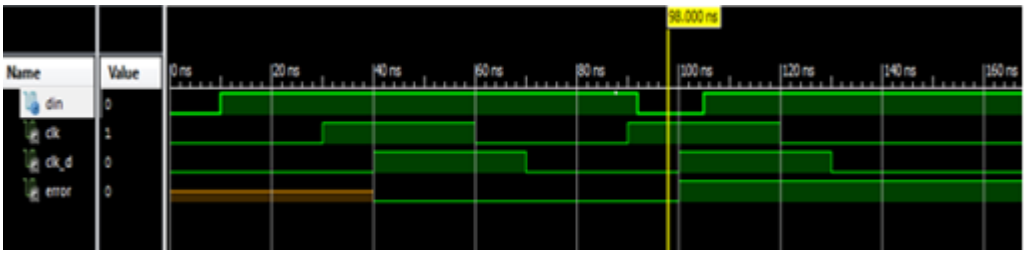

Figure 9. Error occurring from 92 ns to 105 ns

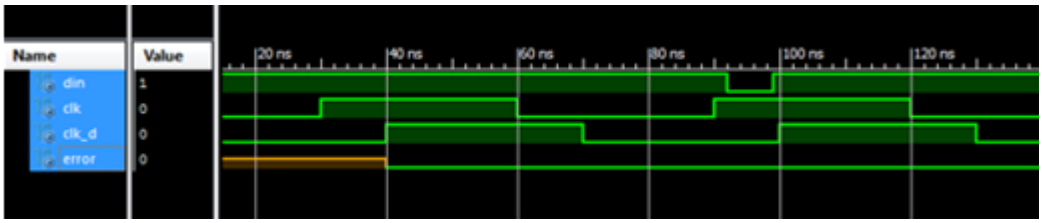

Figure 10. Error occurring from 92 ns to 99 ns

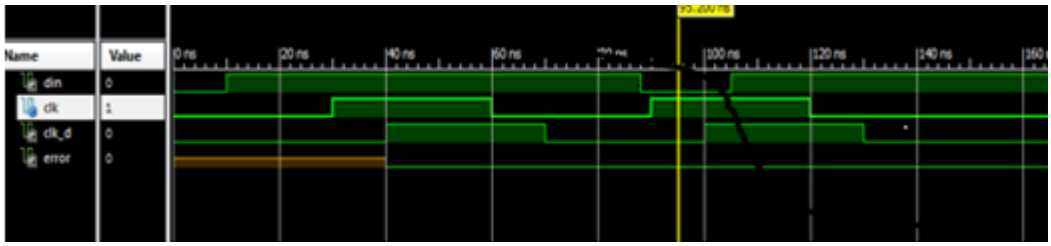

Figure 11. Error occurring from 88ns to 105 ns

Figure 9 to Figure 12 shows the waveforms obtained for different intervals of errors that have been forced on the input Din. Clock period is taken as 60 ns and delay is given to the error flip flop as 10 ns. Since delay of clock transition occurs at 100 ns, the errors occurring as transitions before 90 ns and before 100 ns as well as after 90 ns and after 100 ns have been detected. But upsets which have transitions after 90 ns and before 100 ns are not getting detected. As well as, transitions occurring before 90 ns and after 100 ns are not getting detected.

Table 1. Error tolerance analysis in TRD

S.No	Error type	Inference
1	Captured transients	Output degraded, abided
2	Error upsets (FF)and (TRD)	Output degraded but not abided

Error tolerance analysis of TRD architecture is briefed and given in Table 1 and it shows that captured transients occurring external to the main Flip Flop will be corrupting the output and can be tolerated using the TRD architecture. But error upsets occurring in main Flip Flop and TRD architecture are not getting corrected even though they corrupt the output. So it can be inferred that TRD architecture doesn't have the capability to correct the error upsets occurring in the main Flip Flop.

6.1.2. SETTOFF

RTL schematic of SETTOFF is given in Figure 13. It consists of detection and correction modules for the conventional Flip Flop.

TRD part consists of the detection XOR gate and the error D Flip Flop. Correction part has correction XOR gate and the transition detector. The corrected output is taken from the correction XOR gate and error indication is taken from the TRD part.

The error detection waveforms given in Figure 14 to Figure 16 are same as in TRD architecture. Here clock period is taken as 60 ns and delay is 20 ns.

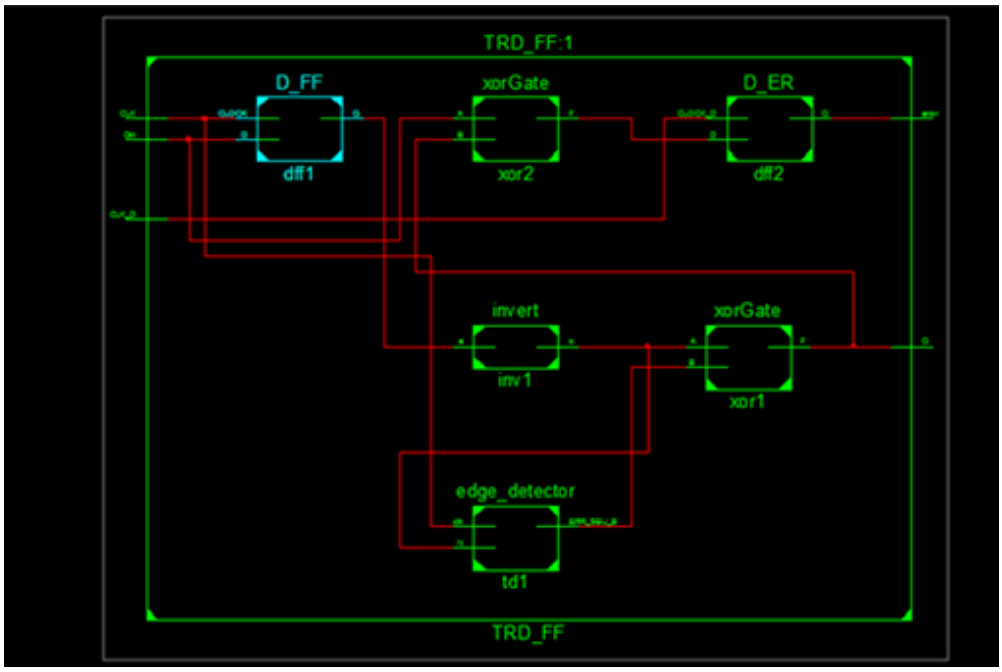

Figure 12. RTL schematic of SETTOFF

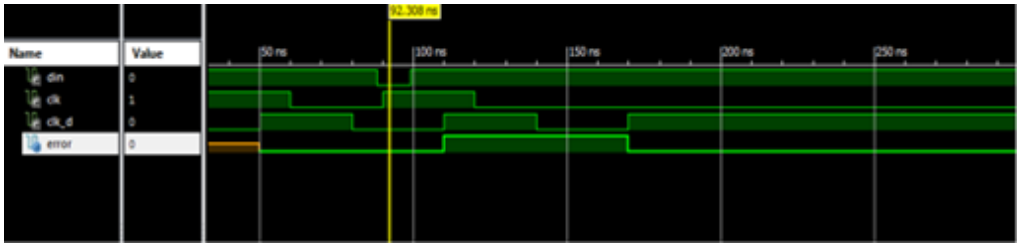

Figure 13. Error occurring from 88 ns to 99 ns

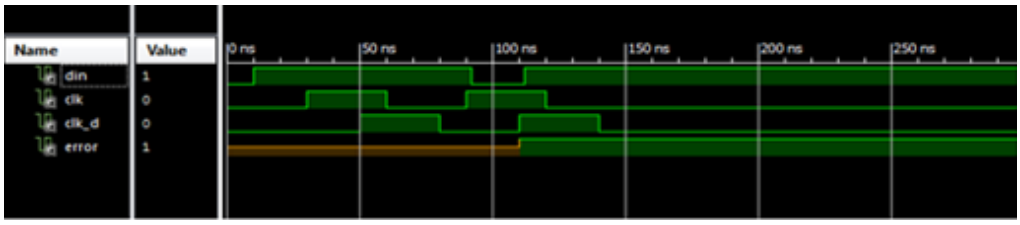

Figure 14. Error occurring from 92 ns to 112 ns

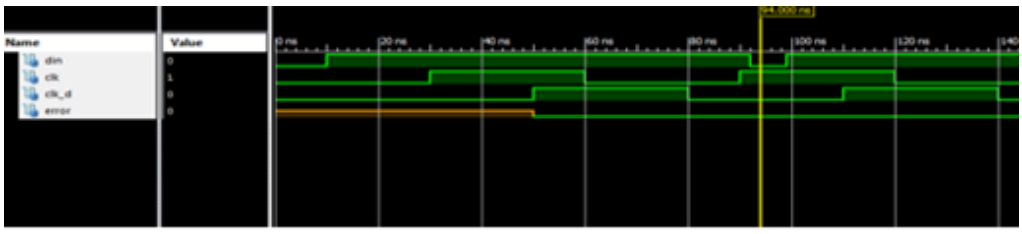

Figure 15. Error occurring from 92 ns to 99 ns

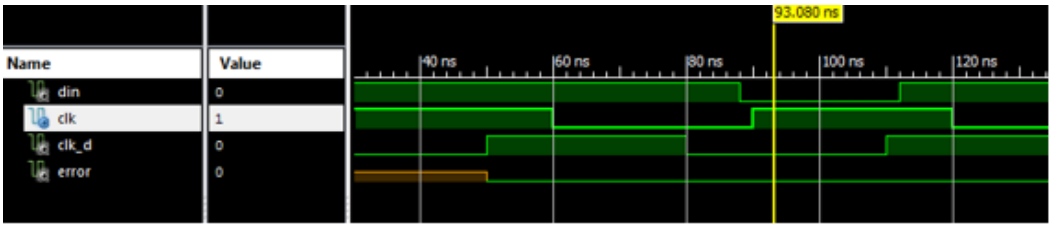

Figure 16. Error occurring from 88 ns to 112 ns

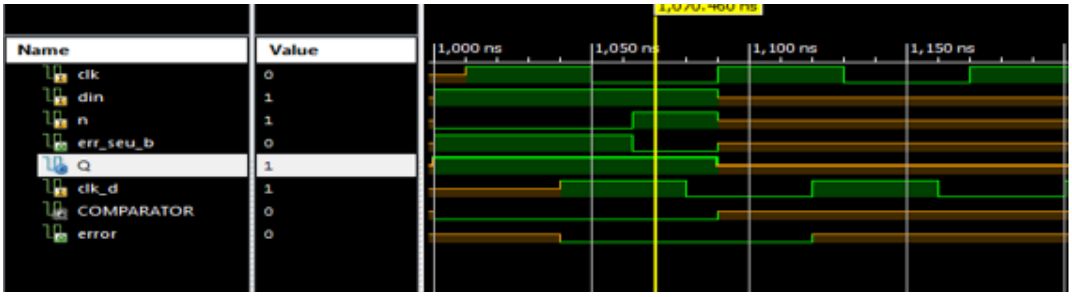

Figure 17. Captured Q value

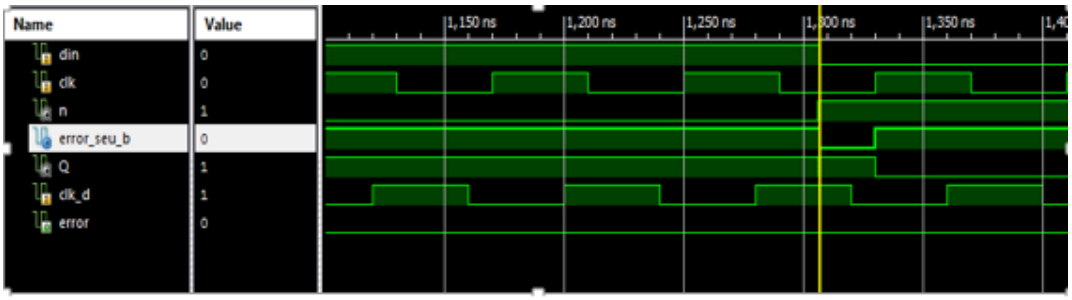

Figure 18. Propagation of n

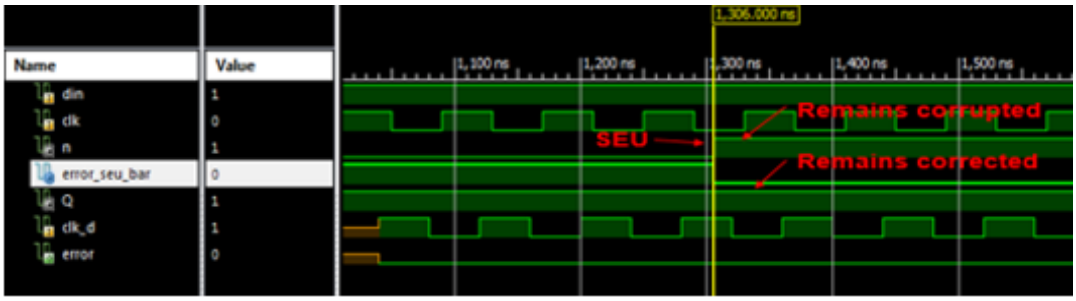

Figure 19. Corrected Q

The delayed clock transition occurs at 110 ns. The errors having transitions before 90 ns and before 110 ns have been detected. But upsets which have transitions after 90 ns and before 110 ns are not getting detected. As well as, transitions occurring before 90 ns and after 110 ns are not getting detected.

For analyzing the correction process, errors have been injected at node ‘n’. In all the above three cases which are given in Figure 18 to Figure 19, value of Q won’t be interrupted. In Figure 17 Q value will be captured when upset is occurring at node ‘n’. Figure 18 shows that the value at node

'n' will be propagated to output Q when error occurs. When the value at node 'n' remains as corrupted, the value at output Q remains as corrected and this is given in Figure 19.

Table 2. Error tolerance in SETTOFF

S.No	Error type	Inference
1	Captured transients	Output degraded, abided TRD
2	Error upsets (FF)	Output degraded, abided TD based architecture
3	Errors (TRD)	Output not degraded and abided
4	Errors in (TD)	Output degraded and not abided

Table 2 gives the error tolerance analysis of SETTOFF. Particular intervals of errors can be detected and correction occurs for SEUs induced at the node 'n', which is the input of correction XOR gate. When compared to TRD architecture, SETTOFF has the capability of correcting the error upsets in the main Flip Flop.

6.1.3. Self checking register

The self checker module has a parity checker for checking errors, a glitch filter for filtering and a transition detector as given in Figure 20.The error signals from each Flip Flop is given to an OR gate.

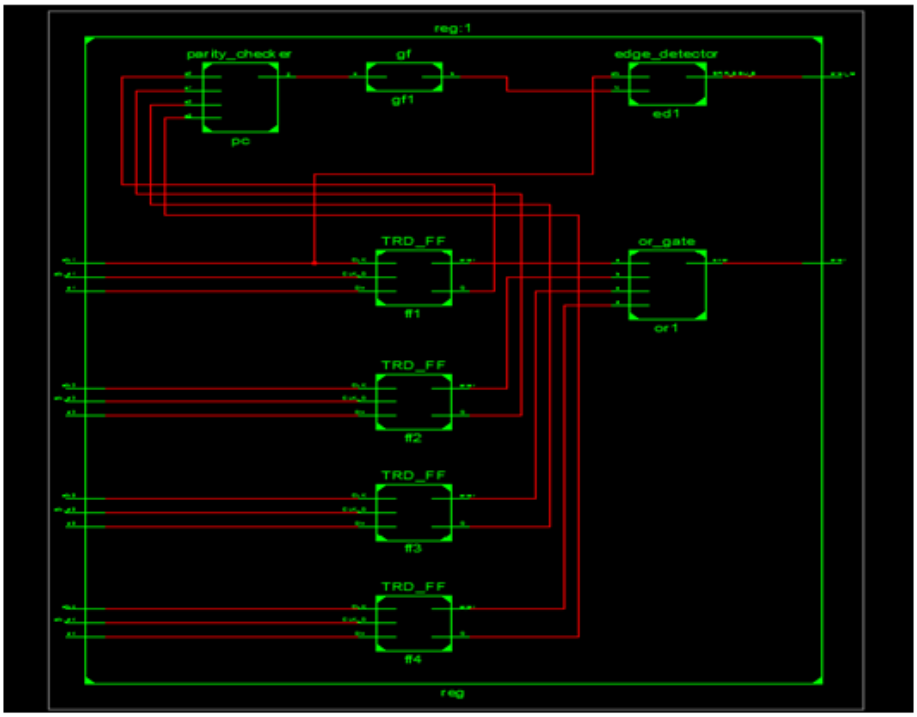

Figure 20. RTL schematic of self-checking register

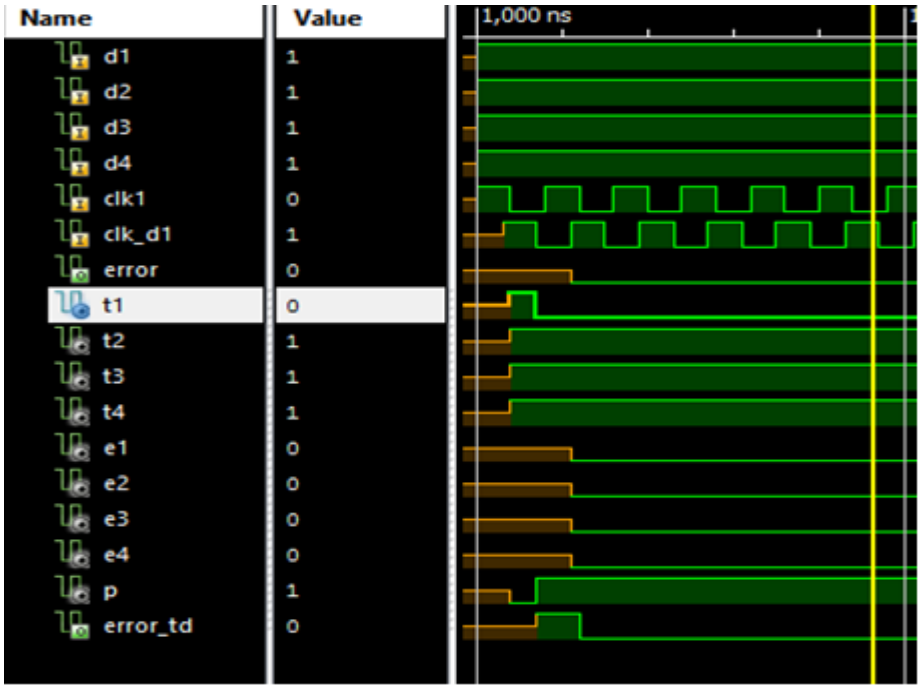

Figure 21. Output waveforms of self-checking register for single error detection

A single event transition is applied at the output of the first flip flop and the transition occurring at parity checker output is detected by the transition detector. As an indication of the error occurred at the first Flip Flop output, the output of the transition detector, error_td signal, gets asserted.

6.1.4. Self-checking register for multi-bit error detection

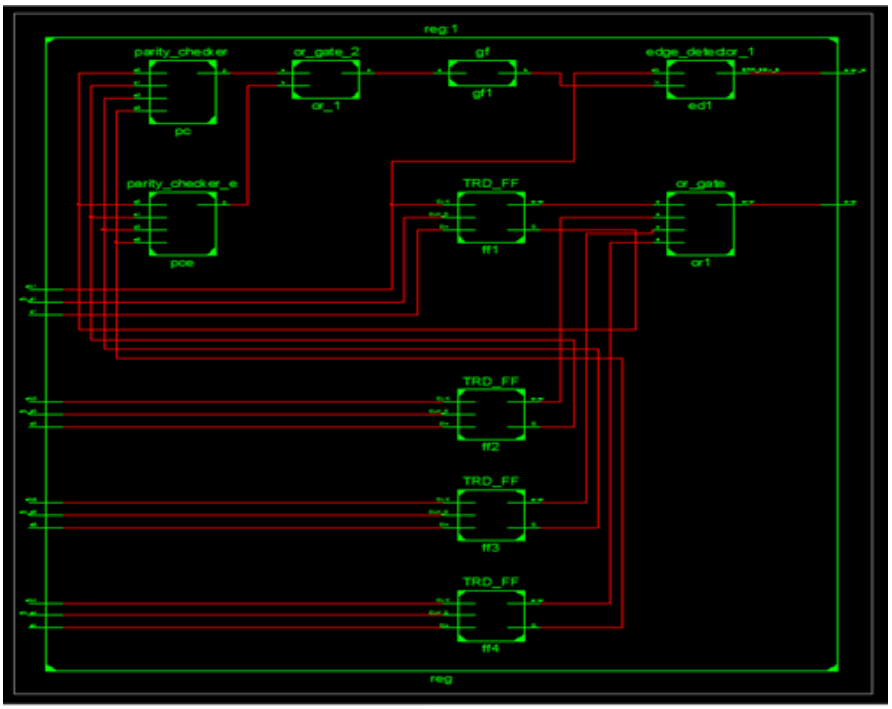

Figure 22. RTL schematic of self checking register for multi-bit error detection

The architecture in Figure 22 show self checking register for multi-bit error detection. An additional parity checker is given along with the self-checker to detect the multi-bit errors. The outputs from the parity checkers are given as the inputs to an OR gate and the output from this gate is then given to the glitch filter.

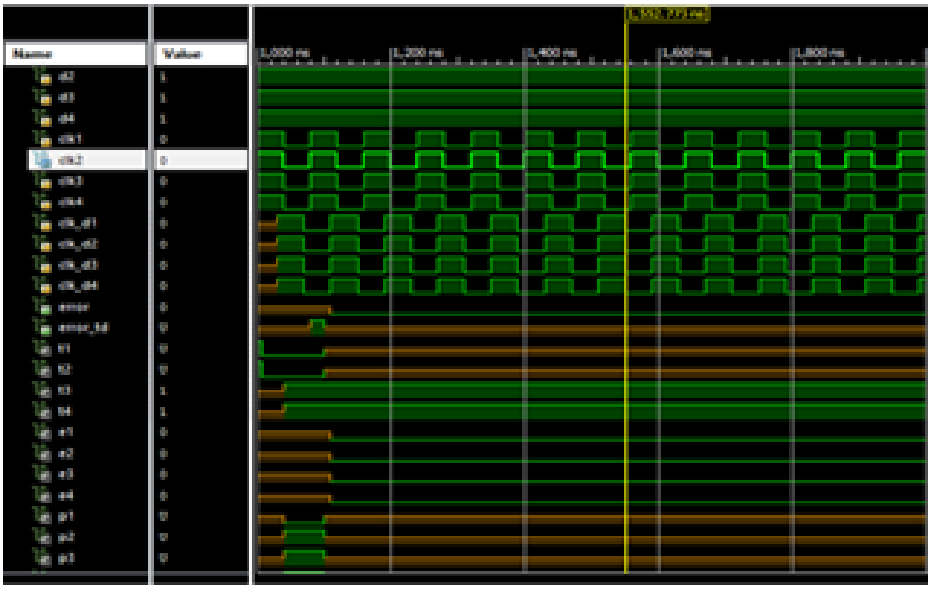

Figure 23. Output waveforms of self-checking register for single error detection

Table 3 Analysis of error tolerance in Self-checking register

S.No	Error type	Inference
1	Captured transients	Output degraded, abided TRD
2	Error upsets (FF)	Output degraded, abided TD
3	Errors (TRD)	Output not degraded and abided
4	Errors (TD)	Output degraded, abided Self-checker
5	Errors (self-checker)	Output not degraded and abided

Table.3. summarizes the analysis of error tolerance in self checking register. Self-checking register has more error tolerance capability. It detects the errors occurring in the TD-based architecture also compared to SETTOFF architecture.

6.2. Circuit Analysis

Figure 24 shows the circuit diagram of SETTOFF which has been implemented in Microwind Dsch tool. Fault analysis results are given in Figure 25 and Figure 26. Figure 25 shows the truth table of SETTOFF which is similar to that of conventional D Flip Flop. Test vector analysis of SETTOFF with stuck at faults at clock, input and internal nodes are given in Figure 26.

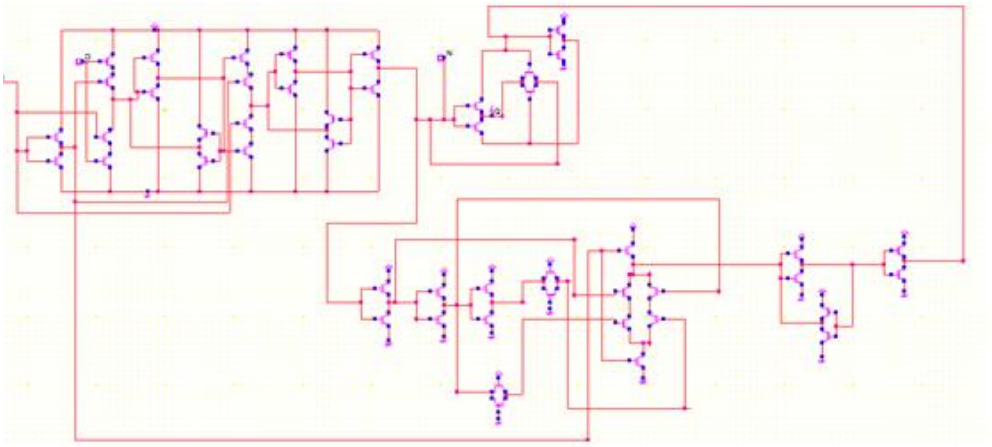

Figure 24. Circuit diagram of SETTOFF

Truth-Table | Test Vectors

clk	D	N	Q
0	0	0	65535
0	0	1	65535
0	1	0	65535
0	1	1	65535
1	0	0	0
1	0	1	0
1	1	0	1
1	1	1	1

Figure 25. Truth table of SETTOFF

Truth-Table | Test Vectors

clk,D,N	000	001	010	011	100	101	110	111
Q(fault-free)	65535	65535	65535	65535	0	0	1	1
clk@0	x	x	x	x	x	x	x	x
clk@1	0	0	1	1	0	0	1	1
D@0	x	x	x	x	0	0	<0>	<0>
D@1	x	x	x	x	<1>	<1>	1	1
N@0	x	x	x	x	<1>	<1>	1	1
N@1	x	x	x	x	0	0	<0>	<0>
Detect score					2/6	2/6	2/6	2/6

Figure 26. Test vector analysis of SETTOFF

Error tolerance analysis gives a comparison between the different techniques. The proposed one has more error tolerance capability.

7. CONCLUSION

A register with self checking capability has been proposed. The proposed system has been analyzed by introducing faults of different time intervals. It is found that the proposed system is error tolerant towards single event upset and timing errors. The system detects the error as well as corrects by itself with the help of in built self checking capability. The multiple errors are also detected and corrected. The fault analysis can be extended to other modules of digital sub blocks so that failures can be prevented. The system has a drawback of consuming more area. Techniques to reduce the area can be done in future.

REFERENCES

[1] C. L. Chen, M. Y. Hsiao, (1984) "Error-correcting codes for semiconductor memory applications: A state-of-the-art review", IBM Journal of Research and Development. Vol.28, Issue.2, pp. 124 – 134.

[2] C.H. Chen, D. Blaauw, D. Sylvester, Z. Zhang, (2014) "Design and evaluation of confidence-driven error-resilient systems", IEEE Transactions on Very Large Scale Integration (VLSI) systems,Vol.22 , Issue.8, pp. 1727–1737.

[3] Hsuan-Ming Chou, Ming-Yi Hsiao, Yi-Chiao Chen, Keng-Hao Yang, Jean Tsao, Chiao-Ling Lung, Shih-Chieh Chang, Wen-Ben Jone, and Tien-Fu Chen, (2015) "Soft-error-tolerant design methodology for balancing performance, power, and reliability", IEEE Transactions on Very Large Scale Integration (VLSI) Systems. Vol.23 , Issue.9, pp.1628–1639.

[4] L. Anghel and M. Nicolaidis, (2000) "Cost reduction and evaluation of a temporary faults detecting technique," Proceedings Design, Automation and Test in Europe Conference and Exhibition 2000 (Cat. No. PR00537), Paris, France, pp.591-598.

[5] M. Ebrahimi, A. Evans, M. B. Tahoori, R. Seyyedi, E. Costenaro and D. Alexandrescu, (2014) "Comprehensive analysis of alpha and neutron particle-induced soft errors in an embedded processor at nanoscales", Design, Automation and Test in Europe Conference and Exhibition (DATE), Dresden, pp.1-6.

[6] M. Favalli and C. Metra, (2004) "TMR voting in the presence of crosstalk faults at the voter inputs",IEEE Transactions on Reliability, Vol. 53, No. 3, pp. 342-348.

[7] M. Nicolaidis, (1999) "Time redundancy based soft-error tolerance to rescue nanometer technologies", Proceedings 17th IEEE VLSI Test Symposium (Cat. No.PR00146), Dana Point, CA, USA, pp. 86-94.

[8] M. Zwolinski, (2001) "A technique for transparent fault injection and simulation in VHDL", Microelectronics Reliability, Elsevier Ltd., 41 (6), pp.797- 804.

[9] N.D.P. Avirneni and A. Somani, (2012) "Low Overhead Soft Error Mitigation Techniques for High-Performance and Aggressive Designs", IEEE Transactions on Computers, vol. 61, no. 4, pp.488-501,April 2012

[10] N. Gaitanis, (1988) "The design of totally self-checking TMR fault-tolerant systems", IEEE Transactions on Computers, Vol. 37, No. 11, pp. 1450-1454.

[11] S. Lin, Y. Kim and F. Lombardi, (2011) "Design and Performance Evaluation of Radiation Hardened Latches for Nanoscale CMOS", in IEEE Transactions on Very Large Scale Integration (VLSI) Systems,Vol.19, No. 7, pp. 1315-1319.

[12] S. Mitra, N. Seifert, M. Zhang, Q. Shi and K. S. Kim, (2005) "Robust system design with built-in soft-error resilience", Computer, Vol.38, No.2, pp. 43-52.

[13] T. Calin, M. Nicolaidis and R. Velazco, (1996) "Upset hardened memory design for submicron CMOS technology", IEEE Transactions on Nuclear Science, Vol. 43, No. 6, pp. 2874-2878.

[14] Y. Lin and M. Zwolinski, (2014) "A cost efficient self checking register architecture for radiation hardened designs", IEEE International Symposium on Circuits and Systems (ISCAS), Melbourne VIC, pp. 149-152.

[15] Y. Lin, M. Zwolinski and B. Halak, (2014) "A low-cost radiation hardened Flip-Flop", 2014 Design, Automation and Test in Europe Conference and Exhibition (DATE), Dresden, pp. 1-6.

[16] Y. Lin, M. Zwolinski and B. Halak, (2016) "A Low-Cost, Radiation-Hardened Method for Pipeline Protection in Microprocessors", IEEE Transactions on Very Large Scale Integration (VLSI) Systems, Vol. 24, No. 5, pp. 1688-1701.

AUTHORS

T. Shunbaga Pradeepa is obtained her B.E. Degree in Electronics and Communication Engineering and M.E. Degree in VLSI Design. She has 4 years of industry (R&D) experience, currently she is working as Assistant Professor in the Department of Electronics and Communication Engineering at Coimbatore Institute of Technology, Coimbatore. She has around 10 years of teaching experience. She is currently doing research in the automotive safety architecture design. Her research interests are in the design and testing of analog and digital systems, Embedded Systems, VLSI design.

Dr.Uma Maheswari obtained her B.E. Degree in Electronics and Communication Engineering from Government College of Technology, Coimbatore, M.E. Degree in Applied Electronics from Coimbatore Institute of Technology and Doctoral degree in Electrical and Electronics Engineering with specialization in Biometrics, from Bharathiar University, Coimbatore. She is presently working as a Professor in the Department of Electronics and Communication Engineering, Coimbatore Institute of Technology, Coimbatore. She holds around 32 years of teaching experience. Her research interests are VLSI Design, Digital Image Processing and Digital Signal Processing. She is also guiding research scholars in different disciplines.