IIOAB JOURNAL
ISSN: 0976-3104

ARTICLE

CONCURRENT ERROR DETECTION WITH SELF-CHECKING MAJORITY VOTING CIRCUITS

V. Elamaran[1*], VR. Priya[2], M. Chandrasekar[1], Har Narayan Upadhyay[1]

[1]*Department of ECE, School of EEE, SASTRA University, Thanjavur, INDIA*

[2]*Department of CSE, SRC Campus, SASTRA University, Kumbakonam, INDIA*

ABSTRACT

The reliability of a system can be improved by designing it with relevant fault-tolerant approaches. The hardware redundancy is often used technique to mitigate errors in a system by placing duplicate copies of original modules. The faults are masked using this kind of passive redundancy approach instead of detecting errors with the help of majority voting circuit. So the reliability of this voter is more important. This study exemplifies five different self-checking voter circuits by means of computing majority and minority and with relevant additional logic in each. Self-checking voter circuits are implemented using mirror concept, mux-and-or approach, mux-xor approach, conventional approach and Manchster carry style. The critical path delay, power dissipation, layout area, number of transistors and figure-of-merit (FOM) are the performance metrics for a comparative study. Simulation results are obtained using Microwind layout editor tool with 120 nm, 90 nm and 70 nm CMOS process technologies. The approach using Manchester carry obtain better FOM (78.85), least layout area (227.0 μm²), lower power dissipation (8.464 μW) as compared with other circuits. The Mux-and-or circuit obtains lower critical path delay (0.420 ns) as than other approaches.

KEY WORDS
CMOS, Fault-tolerance, Majority function, Self-checking, VLSI design, Voter circuits.

Published: 2 December 2016

INTRODUCTION

Business-critical applications demand fault-tolerant system designs so that the transactions don't fail especially in stock exchange, banking and other time-shared systems. For example, the unavailability of ATM service should be 10 hours per year; unavailability of airline reservation should be around 1 minute per day [1]. These applications require very high reliability so that the probability of failure rate is close to zero. The identification of faults/errors and repairing the faulty modules are not possible in applications like spacecrafts and satellites [2].

The reliability improvement is achieved through fault-tolerant approaches. It does not mean that fault-tolerant systems have always high reliability; this depends upon the probability of error occurring in a system [3]. We can design a system to tolerate any single error, but the poor reliability may occur if the probability of such error is too high. But fault-tolerance helps to improve the system reliability by obtaining correct outputs under the environment with software or hardware errors. A system can be designed with high quality components to reduce the failure rate; the system cannot function if the hardware components fail [1].

If there is a possibility of quick repairing faulty components, the availability of a system is improved. Reliability would depend on an interval of time whereas the availability would depend on an instant of time [4]. By reducing the mean-time-to-repair, the reliability of a system is improved. Applications like Supercomputer servers should have high availability for example the unavailability is 10 days per year. Telecommunication service is an example where the unavailability should be around 5 minutes per year [1].

The safety of a system is more important i.e., it is safe if it operate correctly or it is in the safe state during the occurrence of failures [4] [5]. Applications like banking, railway signaling, nuclear energy demand high safety during the operation time. The safety operations are like money should not be distributed if any doubt (banking); Reactor should be stopped if there is any problem; all signal indicators should glow red (railway signaling) [1].

The ability of a system to serve its intended level of service to the customers is known as "dependability". The dependability of a system can be improved by fault removal, fault tolerance, fault prevention and fault forecasting. The faults, errors and failures are the dependability impairments. The sole objective of a system is to increase the dependability of a system [6-8].

This paper is organized as follows. The section 2 contains materials and methods of five self-checking majority voter circuits with critical path delay findings. The section 3 discuss simulation results of power dissipation and layout area used in all designs. The conclusions are made in the section 4.

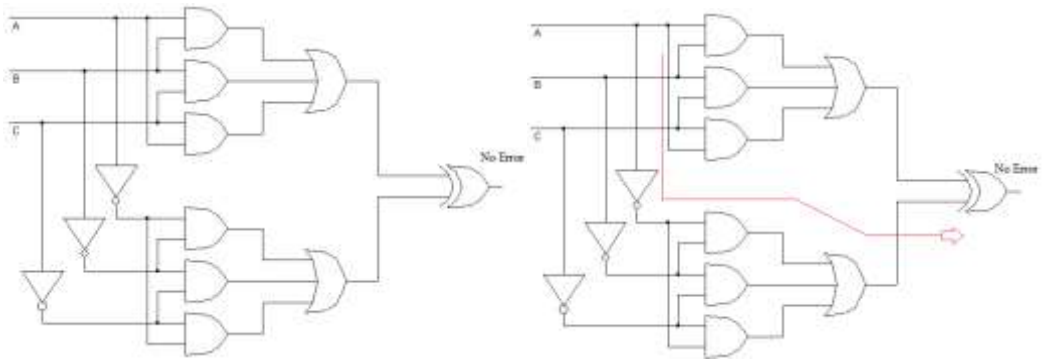
MATERIALS AND METHODS

Self-checking voter circuits are designed and simulated using Microwind and DSCH software tools. DSCH is a digital schematic editor tool whereas the Microwind is a layout editor tool [9-12]. Schematics which are designed using DSCH are converted in to Verilog hardware description language (HDL) script. The Verilog scripts are then compiled using Microwind tool to generate a corresponding layout.

***Corresponding Author**
Email:
elamaran@ece.sastra.edu

Self-checking voter (SVC) using mirror concept

In this study, self-checking voter circuits are implemented by comparing majority computation and minority computation logic outputs; if they are different there is no error in the circuit [13]. The majority value is computed using (AB+BC+CA) expression, whereas the minority value is computed using the same expression but with complement inputs as in [Fig. 1].



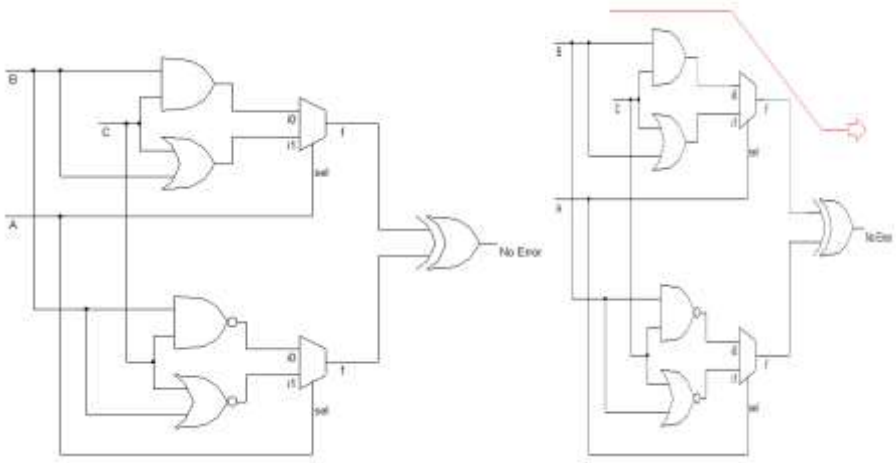
(a) (b)

Fig. 1: SVC using mirror concept.

The critical path delay is evaluated as in [Fig. 1b]. This circuit requires one inverter (0.17 ns), one AND gate (0.16 ns), one OR gate (0.13 ns) and one XOR gate (0.16 ns) for the critical path; hence the total delay is 0.62 ns.

Self-checking voter (SVC) using mux-and-or

If A=0, then the majority becomes "BC" else "B+C". Similarly, if A=0 then the minority becomes "$\overline{BC}$", else "$\overline{B+C}$" as in [Fig. 2]. The critical path of this circuit requires one AND gate (0.16 ns), 1 mux (0.1 ns) and one XOR gate (0.16 ns); hence the total delay is 0.420 ns as portrayed in [Fig. 2b].

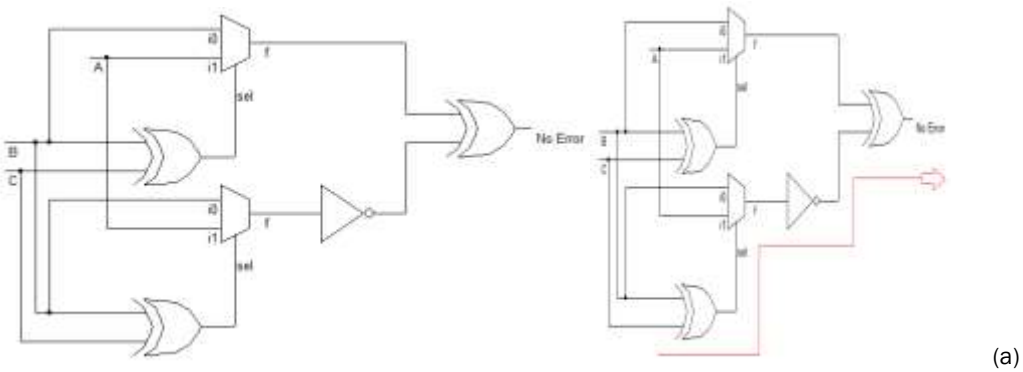


(a) (b)

Fig. 2: SVC using mux-and-or.

Self-checking voter (SVC) using mux-xor

If B=C, then the majority becomes "B" else "A". The minority value is computed using similar concept as in [Fig. 3]. The critical path of this circuit requires one XOR gate (0.16 ns), one mux (0.1 ns), one inverter (0.1 ns) and one more XOR gate (0.16 ns); hence the total delay is 0.520 ns as depicted in [Fig. 3b].



(a)

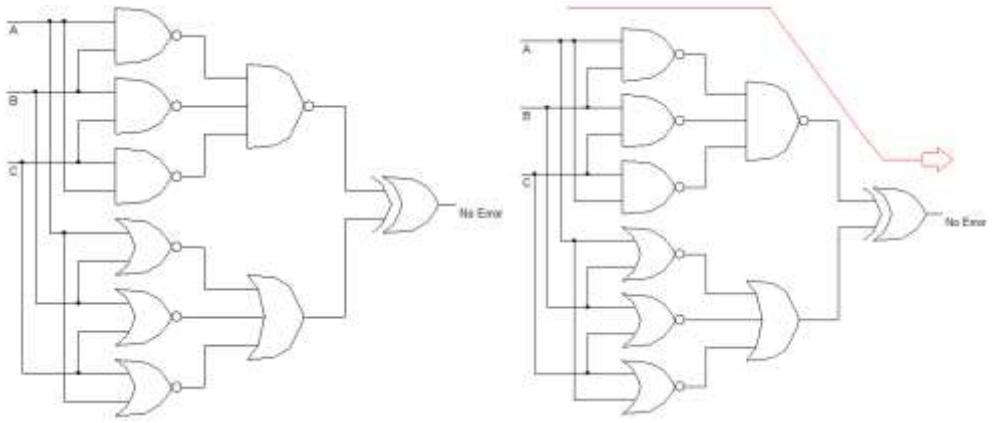
(b)

Fig. 3: SVC using mux-xor.

.....

Self-checking voter (SVC) using conventional method

The majority and minority values are computed using conventional method "AB+BC+CA" by NAND/NOR gates as in [Fig. 4]. The critical path of this circuit requires one NAND2 gate (0.16 ns), one NAND3 gate (0.16 ns) and one XOR gate (0.16 ns); hence the total delay is 0.480 ns as shown in [Fig. 4b].



(a) (b)

Fig. 4: SVC using conventional method.

.....

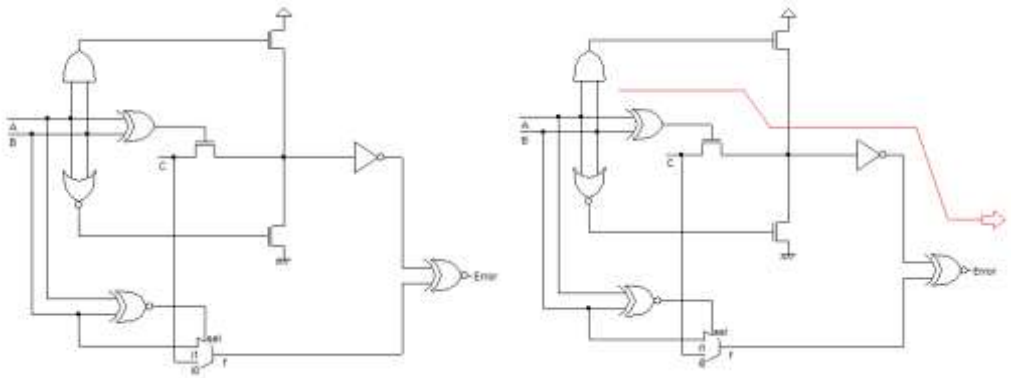
Self-checking voter (SVC) using Manchester carry chain

The Minority value is computed using Manchester carry chain and the minority value is computed using mux-xnor principle as in [Fig. 5] [14]. The Manchester carry chain algorithm is described as follows:

i) If A and B are different, then the minority value becomes $\overline{C}$.

ii) If A=B=1, then the minority becomes 0.

iii) If A=B=0, then the minority becomes 1.



(a) (b)

Fig. 5: SVC using Macnchester carry chain.

RESULTS AND DISCUSSION

The power dissipation and layout area results are discussed in this section. An another performance metric is a figure-of-merit (FOM) is expressed as in Equation (1),

$$FOM = \frac{1}{Power \times Delay \times Area} \quad (1)$$

Since applications demand low power, less area and high performance, an optimized VLSI design should have a high FOM [4].

Power dissipation results

The circuit functionalities are verified in both using DSCH tool and Microwind tool with schematic and layout respectively [15]. The power dissipation simulation results of self-checking voter circuits are obtained in [Fig. 6] in 120 nm, 90 nm and 70 nm process technologies. In 120 nm foundry, SVC using Manchester carry chain consumes power of 8.464 µW which is less as compared with remaining circuits.

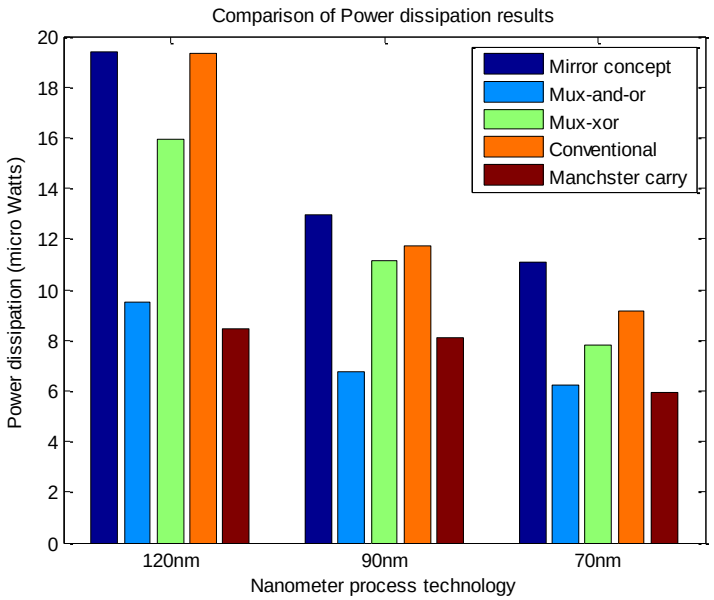


Fig. 6: Power dissipation results.

Critical path delay results

The critical path delay results of self-checking voter circuits are obtained in [Fig. 7] in 120 nm process technology. SVC using mux-and-or had lower delay as 0.420 ns than others.

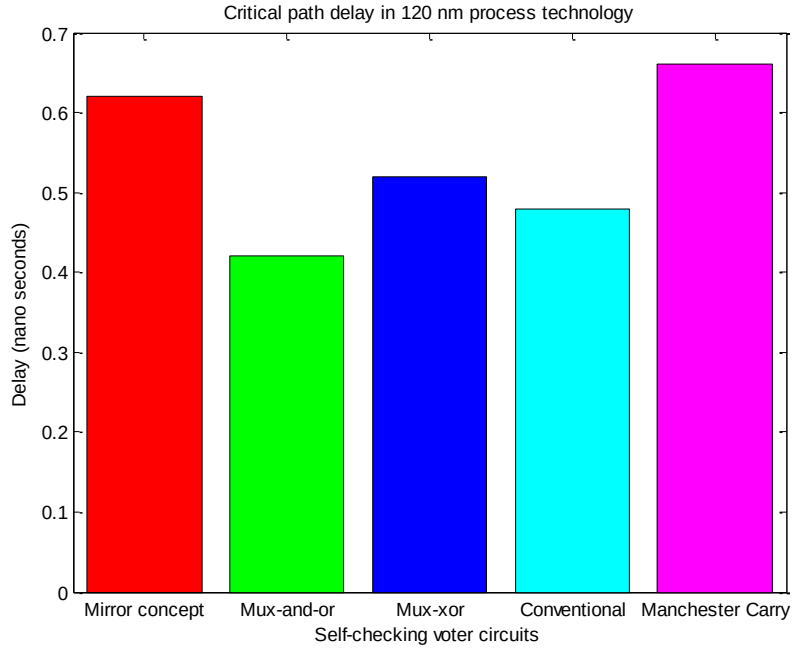


Fig. 7: Critical path delay results.

Layout area results

Self-checking voter circuits using conventional (AB+BC+CA) method obtain less layout area in all 120 nm, 90 nm and 70 nm process technologies as in Figure 8; whereas SVC using Manchester carry chain stands next with low area. In 120 nm foundry technology, SVC using conventional had 170.4 μm^2 and SVC using Manchester carry chain had 227.0 μm^2.

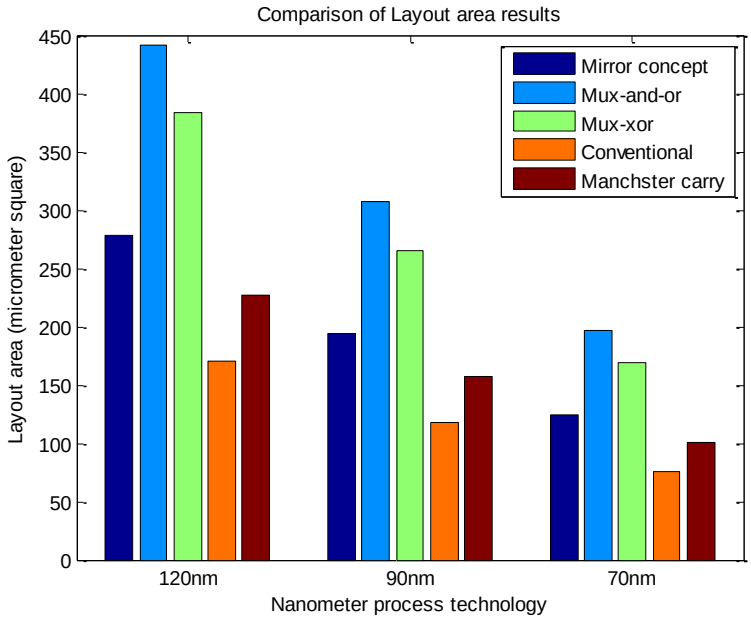


Fig. 8: Layout area results.

Figure-of-merit results

FOM results are obtained using the Equation (1) and are plotted in [Fig. 9] in 120 nm process technology. The SVC using Manchester carry chain had a better FOM as 78.85×10^5 than other designs.

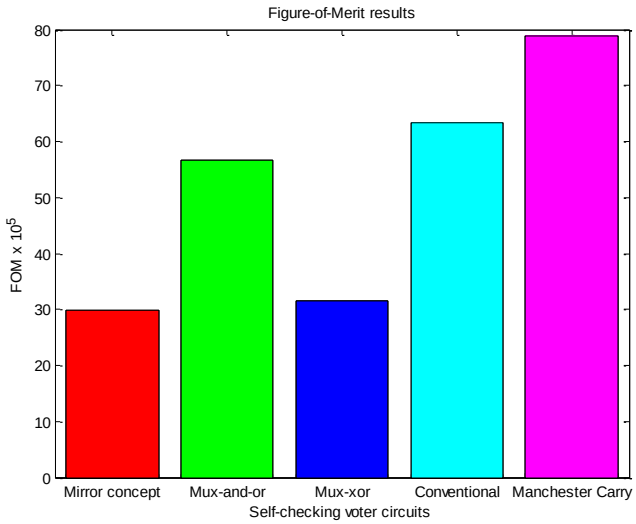


Fig. 9: Figure-of-merit results.

.....

Total number of transistors and failure rate

Failure rate (λ) of a circuit is determined using $\sqrt{g}$, where g is the total number of logic gates or transistors in a design. It is obvious that the design is good if it contains few number of logic gates in order to have a lower failure rate. The reliability of a system is improved by reducing failure of rate.

The total number transistors and the corresponding failure rate results are obtained in [Fig. 10 and 11] respectively. Results show that the SVC using mux-xor design requires only 32 transistors (16 nMOS and 16 pMOS) with the lower failure rate as 5.6568 than other circuits. The SVC using mux-and-or design stands second best method with 38 transistors and 6.1644 failure rate. The SVC using mirror concept had a higher failure rate due to more number of transistors (64).

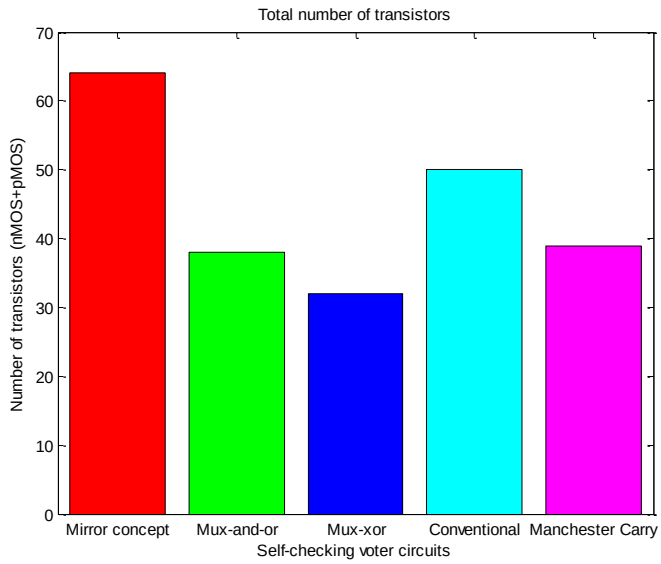


Fig.10: Results of total number of transistors in SVC.

.....

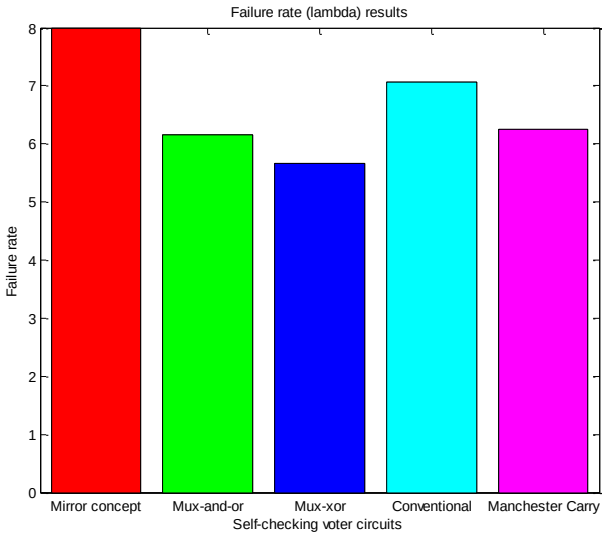


Fig. 10: Failure rate results.

CONCLUSION

In the current semiconductor technology evolution, reliability of microelectronic circuits play a vital role in any system design. The hardware redundancy is often used to remove or mask faults/errors in a circuit by incorporating redundant modules. The fault masking can be done using the majority voter circuits in a triple modular redundancy (TMR) system for better reliability. The TMR system fails if the voter circuit does not function correctly. This study exemplifies five difference self-checking voter circuits along with a comparative analysis in area, power and delay results. In real time, some applications may demand low power; some may demand high performance; few require less area. So, the trade-off is important for many applications. Designers may able to achieve low power circuits but at the cost of area or performance and vice versa. This work can be further extended self-checking five input majority voter circuits; 7-input majority voter circuits, etc.

CONFLICT OF INTEREST
There is no conflict of interest.

ACKNOWLEDGEMENTS
None

FINANCIAL DISCLOSURE
None

REFERENCES

[1] Dubrova E.[2013] Fault-Tolerant Design. New York: Springer Science+Business Media.
[2] Lala PK. [2001] Self-Checking and Fault-Tolerant Digital Design. Morgan Kaufmann publishers.
[3] Koren I, Mani Krishna C. [2013] Fault-Tolerant Systems. Morgan Kaufmann publishers.Design of a novel fault-tolerant voter circuits for TMR implementation to improve reliability in digital circuits.
[4] Kshirsagar KV, Patrikar RM. [2009] Microelectronics Reliability, 49: 1573-1577.
[5] Majority function computation using different voter circuits – a comparative study. V Elamaran, et al. 2015. International Journal of Pharmacy and Technology, 7: 9764-9773.
[6] Balasubramanian P. [2016] ASIC-based design of NMR system health monitor for mission/safety-critical applications. SpringerPlus, 5: 1-16.
[7] Elamaran V, Har Narayan Upadhyay. [2015] CMOS VLSI Design of Low Power SRAM Cell Architectures with new TMR: A Layout Approach. Asian Journal of Scientific Research, 8: 466-477.
[8] Elamaran V, Har Narayan Upadhyay. [2015] Low Power Digital Barrel Shifter Datapath Circuits using Microwind Layout Editor with High Reliability. Asian Journal of Scientific Research, 8: 478-489.
[9] Hari Hara Subramani S. et al. [2014] Low-energy, low power adder logic cells: A CMOS VLSI Implementation. Asian Journal of Scientific Research, 7: 248-255.
[10] Elamaran V. et al. [2015] A handy approach for teaching and learning digital VLSI design using EDA tools. International Journal of Pharmacy and Technology, 7: 9935-9944.
[11] Rajesh KSSK. et al. [2014] CMOS VLSI design of low power comparator logic circuits. Asian Journal of Scientific Research, 7: 238-247.
[12] Pradhisha R. et al. [2015] FPGA Implementation of Self-testing Logic gates, Adders, Multipliers.. Indian Journal of Science and Technology, 8: 1-6.

[13] Balasubramanian P, Maskell .L.. [2015].A distributed minority and majority voting based redundancy scheme. Microelectronics Reliability, 55: 1373-1378.
[14] Uyemura, J.P. Introduction to VLSI Circuits and Systems. India: Wiley India publishers, 2006.
[15] Sayapaneni P, Elamaran V. [2014]s.l.: IEEE Press, A comparative study on low power adders using Microwind EDA tool. International Conference on Computational Intelligence and Computing Research, pp. 1-5.