Redundancy Based Design and Analysis of ALU Circuit Using CMOS 180nm Process Technology for Fault Tolerant Computing...

Article · January 2015

DOI: 10.12785/ijcds/040106

3 authors, including:

Tejinder Singh
University of Waterloo
36 PUBLICATIONS **76** CITATIONS

International Journal of Computing and Digital Systems
ISSN (2210-142X)
Int. J. Com. Dig. Sys. 4, No.1 (Jan-2015)
http://dx.doi.org/10.12785/ijcds/040106

Redundancy Based Design and Analysis of ALU Circuit Using CMOS 180nm Process Technology for Fault Tolerant Computing Architectures

Tejinder Singh[1], Farzaneh Pashaie[2] and Rajat Kumar[3]

[1]*Discipline of Electronics and Electrical Engineering, Lovely Professional University, Phagwara 144 402, PB, India*
[2]*Department of Mechatronics, Islamic Azad University, South Tehran Branch, Iran*
[3]*Hardware Development Division, Gingerbox InfoTech Pvt. Ltd., Tamil Nadu 600 116, Chennai, India*

Received 2 Apr. 2014, Revised 15 Oct. 2014, Accepted 1 Nov. 2014, Published 1 Jan. 2015

Abstract: As the technology entering into Nano dimensions, the manufacturing processes are becoming less reliable, that is drastically impacting the yield. Therefore, fault tolerant systems are becoming more important, particularly in safety-critical applications. In this paper, we present the design and analysis of 4-bit Arithmetic and Logical Unit (ALU) circuit designed using CMOS 180 nm process technology for fault tolerant computing architectures. As, ALU is a functional block of the Central Processing Unit (CPU) of a computer system. It is highly recommended that the ALU block must be fault free or fault tolerant one. In order to have high reliability and high up time of the system, we have used the classical Triple Modular Redundancy (TMR) technique in which three redundant subsystems are used in order to attain high reliability. We have achieved lower power dissipation with higher reliability of ALU circuit. The Voter Logic and Fault detection circuits are also designed and reported in this paper.

Keywords: Fault Tolerant ALU Design, Triple Modular Redundancy, 180nm Process Technology, Schematic Design

1. INTRODUCTION

In the past few years, the design of computing architectures and circuits has become remarkably complex and dense, while the role and importance of the systems have increased explosively [1,2]. As the scale of integration increased from small/medium to large and to today's very large scale, the reliability per fundamental computing function require dramatic improvements. Due to the fact that as CMOS technology is also following Moore's law, the implementation of more number is modules per unit chip are is mandatory. But, as we plan to shrink the technology scale further, there we can see a lot of complications like reliability concerns, accuracy parameters, precise results of circuits and various other circuit level performance parameters. As the demand of enhanced functionality is increasing, the complexity of systems has increased and that has raised the probability of complete failure of system. Moreover, our dependence on computing systems has grown to an extent that it has became impossible to return to less sophisticated mechanisms [1]. High reliability and uninterrupted operations in computing systems are much more vital in certain applications such as spacecraft navigation, aircraft flight control and landing systems, nuclear power plants, and chemical industries. Failures or malfunctioning of any equipment or its components in any such application leads to disastrous effects [5]. So our main concern is to have high reliable system in critical applications.

In order to achieve high reliability of computing systems, we approach a fault tolerance mechanism for the Arithmetic and Logic Unit (ALU), which will avoid the unexpected breakdown of the system. We have reported design and analysis of a complete 4-bit ALU circuit. Complementary Metal Oxide Semiconductor (CMOS) 180nm process technology is used to design the system. Triple Modular Redundancy (TMR) technique is adopted and applied to the system to have fault free system. In this fault tolerant mechanism, majority voter logic module is designed to select the correct output even in the case of failure of system and disagreement detector is used to

E-mail address: tejinder.singh@ieee.org, st_f_pashaie@azad.ac.ir , kumar.rajat@outlook.com

detect the faulty ALU and provide output to the used that in case of any fault occurs.

The structure of the paper is organized as - the theory of fault tolerant system is given in section 2 followed by design mechanism of fault tolerant system in section 3. The various redundancy techniques are described in section 4. The design of fault tolerant ALU system is given in detail in section 5 followed by results and discussion in section 6 with simulated results of fault tolerant behaviour of system and at last the conclusion or outcome is given in section 7.

2. FAULT TOLERANT SYSTEM

Fault tolerance or in simple words the graceful degradation is the property is a system that enables that system to continue its operation properly even in the event of failure of few of system's internal components. if the quality of operation decreases anyhow, the decrease is proportional to the failure's severity as compared to a naïvely implemented system in which even a minor failure leads to total breakdown of system. Fault tolerance mechanism or graceful degradation is particularly sought after in life-critical and highly available systems [15].

Fault tolerance is the property that enables a system to continue operating in the event of the failure of some of its components as described [15]. Several applications areas need systems to maintain correct (predictable) functionality in the presence of faults: Banking systems, Control systems, manufacturing system.

There are mainly three types of faults [8] viz- Permanent fault: the faults are perpetual and can be caused by physical damage or design errors, Intermittent fault: The faults occur periodically and typically result from unstable device operation and Transient fault: It is often caused by external disturbances, exist for a finite length of time and are non-recurring in nature.

3. DESIGN MECHANISM

In fault tolerant designs, redundancy is used to provide the information needed to negate the effects of a failure. Basically four types of redundancy are considered: hardware, software, information and temporal, time redundancy [3]. Hardware redundancy is perhaps the most commonly used redundancy and can be employed in several forms [1]. Hardware redundancy consists in employing several identical circuits to perform the same computation at the same time [9].

The faults can be detected by the duplication or masked by the triplication by comparing the redundant outputs through a comparator/voter [4]. Information redundancy involves the addition of redundant information to the original data, i.e. it is the number of bits used to transmit a message minus the numbers of bits of actual information in the message. Informally, it is the amount of wasted space used to transmit certain data [1]. Temporal redundancy consists in forcing the system (or a sub- system) to repeat a given operation and then compare the results with those of the previous operation. Such a redundancy is able to tolerate transient or intermittent errors but not permanent errors thus making this solution not suitable for our study. In software redundancy, error detection and recovery are based on replicating application processes on a single or multiple computers [3]. Time Redundancy consists in re-executing the same operation at different time and comparing results to detect faults [4].

The use of redundancy is proposed not as a replacement, but rather as a supplement to the two cardinal principles of reliable design [6]. One is to use the most reliable components and and other is to use the least possible complexity consistent with required system performance

A. Redundancy

The two functions of redundancy are – Passive redundancy that uses excess capacity to reduce the impact of component failures. One common form of passive redundancy is the use of high stress handling bond wires in IC, whereas Active redundancy eliminates the performance decline by monitoring performance of individual device and this monitoring is used in voting logic. The voting logic is linked to switching circuit that automatically reconfigures the components.

4. REDUNDANCY TECHNIQUES

Redundancy is the provision of functional capabilities that would be unnecessary in a fault-free environment. This can consist of backup components which automatically "kick in" should one component fail. For example, large cargo trucks can lose a tire without any major consequences. They have many tires, and no one tire is critical (with the exception of the front tires, which are used to steer). The idea of incorporating redundancy in order to improve the reliability of a system was pioneered by John von Neumann in the 1950s.

Two kinds of redundancy are possible: space redundancy and time redundancy. Space redundancy provides additional components, functions, or data items that are unnecessary for fault-free operation. Space redundancy is further classified into hardware, software and information redundancy, depending on the type of redundant resources added to the system. In time redundancy the computation or data transmission is repeated and the result is compared to a stored copy of the previous result.

In order to tolerate the defects due to manufacturing processes, we have considered the hardware based redundancy technique [3]. Triple Modular Redundancy (TMR) is commonly used for designing dependable systems to ensure high reliability, availability and data integrity. Triple Modular Redundancy has been extensively used as a building block of fault-tolerant computing architectures. The idea of this technique is very fundamental: A TMR unit consists of three computing

modules and a voter logic module. Three modules perform the same computation in parallel manner and there results are applied to the voter as shown in Fig. 1. If any one of the three modules fails, the other two modules can correct and mask the fault [7]. With the failure of voter system, the complete system can collapse. However, in a good TMR system, the voter is much more reliable than other TMR components. The voting logic compares the outputs of all the modules pass the majority output i.e. if all three outputs are same then it becomes the final output and if two out of three outputs are same then the two same outputs become the final output. Also, if the two same outputs are erred output then it will become the final output [9].

A. Arithmetic and Logical Unit

The processors found inside modern CPUs and graphics processing units accommodate very powerful and very complex ALUs, a single component may contain

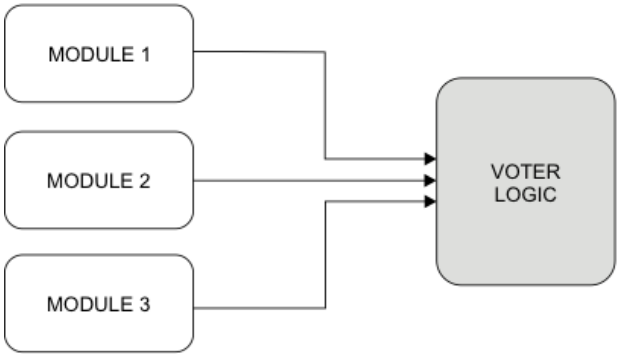

Figure 1. Block diagram of triple modular redundancy technique. Three similar modules and the fault is detected by Voter logic circuit.

a number of ALUs. An ALU loads data from input registers, executes the operation and stores the result into output registers [13]. The design and function of an ALU may vary between different processors. For example, some arithmetic and logical units only perform integer calculations, while others are designed to handle floating-point operations. Regardless off the way an ALU is designed, its primary job is to handle integer operations. Therefore, a computer's integer performance is tied directly to processing speed of ALU [15]. ALUs can perform the following operations:

- Integer arithmetic operations (addition, subtraction and multiplication)
- Bitwise logic operations (AND, NOT, OR, XOR)
- Bit-shifting operations (shifting a word by a specified number of bits to the left or right).

5. FAULT TOLERANT ALU DESIGN

The fundamental function of a processor is to execute sequences of instructions that are stored in main memory, which is external to the processor or central processing unit of a computer system. The processor also monitors and inspects the other system components, usually via dedicated control signals. For example, the processor directly or indirectly controls input/output or I/O operations viz. data transfers between primary memory and I/O devices.

The processor contains various registers, which are used for the temporary storage of various instructions and operands, and an arithmetic and logical unit (ALU), which executes instructions related to data processing. It is proposed that redundancy to the critical components of a CPU may provide a feasible alternative to the multiprocessor architecture. ALU is only responsible for all the computational tasks, hence the ALU should be fault tolerant in any case for reliable systems.

A design of fault-tolerant system enables a system to continue its described and expected operation, possibly at a degraded or reduced level, rather than showing a sign of complete failure, at a time when some part of the system fails to operate, The fault-tolerant term is most

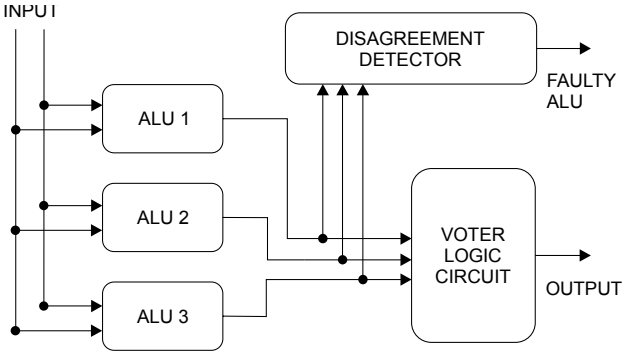

Figure 2. Block diagram of fault tolerant system. Three similar ALU circuits provide output to voter logic to select the correct output in case of any fault and disagreement detector detects the faulty ALU module.

commonly used for computing architectures or computing systems that are designed and customized to continue more or less complete operational with, might be, an increase in response time or reduction in throughput in the event of partial failure of the system. That is, the whole system is not stopped due to hardware or the software problems. An example in another field is a motor vehicle designed so it will continue to be drivable if one of the tires is punctured. A structure is able to retain its integrity in the presence of damage due to causes such as corrosion and fatigue manufacturing flaws, or impact.

Triple modular redundancy is basically the replication of a component into three identical systems where all systems or sub modules contain the same information and perform the operation at the same time. The output of all the three sub modules is then voted

upon to select the majority output by a voter module. Therefore, if two of the three sub modules are functionally correct, the voter will provide the correct output because of majority. TMR is arguably the most reliable technique in fault tolerance, but it tends to be bulky when used with large system components.

Fault tolerance is not just a property of individual machines; it may also characterize the rules by which they interact with each other. Recovery from errors in fault-tolerant systems can be characterized as either roll-forward or rollback. When the system detects that it has made an error, roll-forward recovery takes the system state at that time and corrects it, to be able to move forward. Roll-back recovery reverts the system state back to some earlier, correct version, for example using checkpoints, and moves forward from there. Rollback recovery requires that the operations between the checkpoint and the detected erroneous state can be made idempotent. The designed fault tolerant system is shown as a top-level block diagram in Fig. 2.

Within the scope of an individual system, fault tolerance can be achieved by anticipating exceptional conditions and building the system to cope with them, and, in general, aiming for self-stabilization so that the system converges towards an error-free state. However, if the consequences of a system failure are catastrophic, or the cost of making it sufficiently reliable is very high, a better solution may be to use some form of duplication.

Arithmetic and logical unit (ALU) is a digital circuit that is divided into two distinct parts, the AU (arithmetic unit) and the LU (logical unit). The AU performs the arithmetic operations such as Add, Subtract, Multiply etc. and the LU performs the logical operations such as AND, OR, NOR etc. ALUs are designed to perform integer calculations. The ALU is a fundamental building block of the central processing unit of a computer [13].

A. Majority Voting Logic

It will allow the voting of correct output value in the presence of faults. To design majority voter circuit, we used the following method given below in Fig. 3. Here we use three AND gate and one OR gate [9]. Suppose we have three inputs A0, A1 and A2 and we have to calculate the majority output out of three. Than we can find out it by the Boolean expression:

$$OUTPUT= [(A0\&A1) + (A0\&A2) + (A1\&A2)]$$

B. Disagreement Detector

In triple modular redundancy, it is also important to find the error part. Disagreement detector is used to find the error part of the system. We can design it by various methods. We have used XNOR operation method to implement the system. Disagreement detector has three inputs and three outputs. Different outputs give different type of information that it describes in Table 1. It is designed by XNOR gate. We know that for a XNOR gate if inputs are different than it gives a low output and if inputs are same than it give high output. The disagreement circuit is shown in Fig. 4.

Table 1. Disagreement detector outputs

Inputs			Outputs			Analysis
A	B	C	X1	X2	X3	
1/0	1/0	0/1	1	0	0	C is faulty
1/0	0/1	1/0	0	0	1	B is faulty
0/1	1/0	1/0	0	1	0	A is faulty
1/0	1/0	1/0	0	0	0	All faulty
1/0	1/0	1/0	1	1	1	No fault

Table 2. Voter logic circuit selection criteria

Selection Criteria	Output
ALU1 = ALU2 and ALU1 ≠ ALU3	ALU1
ALU2 = ALU3 and ALU2 ≠ ALU1	ALU2
ALU3 = ALU1 and ALU3 ≠ ALU2	ALU3

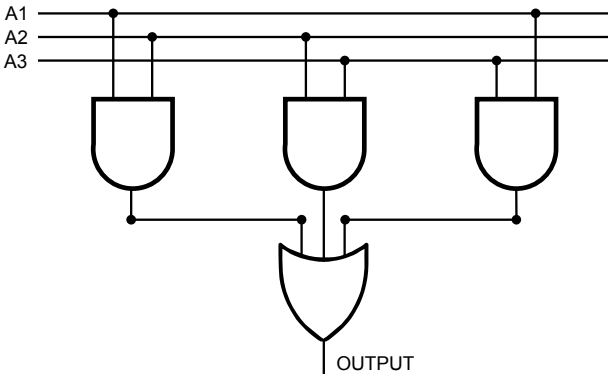

Figure 3. Majority voter logic circuit designed using AND gates and an 3 input OR gate.

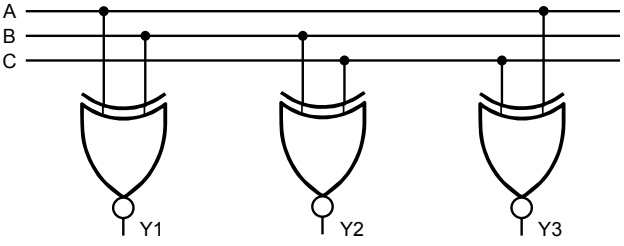

Figure 4. Schematic of digagreement detector circuit used to detect the faulty ALU module from the redundant modules.

□□□□□□3□ Functions implemented in 4-bit ALU

No.	Command	Function
01	0 0 0 0	NOT
02	0 0 0 1	AND
03	0 0 1 0	NAND
04	0 0 1 1	OR
05	0 1 0 0	NOR
06	0 1 0 1	XOR
07	0 1 1 0	XNOR
08	0 1 1 1	BUFFER
09	1 0 0 0	ADDITION
10	1 0 0 1	SUBTRACTION
11	1 0 1 0	MULTIPLICATION
12	1 0 1 1	INCREMENT
13	1 1 0 0	DECREMENT
14	1 1 0 1	SHIFT RIGHT
15	1 1 1 0	DIVISION
16	1 1 1 1	SHIFT LEFT

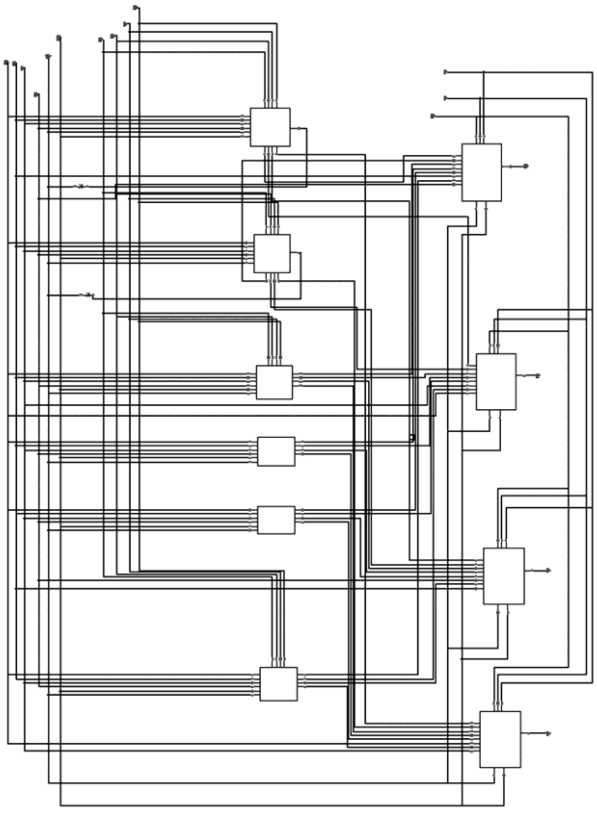

Figure 5. Schematic design of Arithmetic Unit designed in Cadence Virtuoso Schematic Editor.

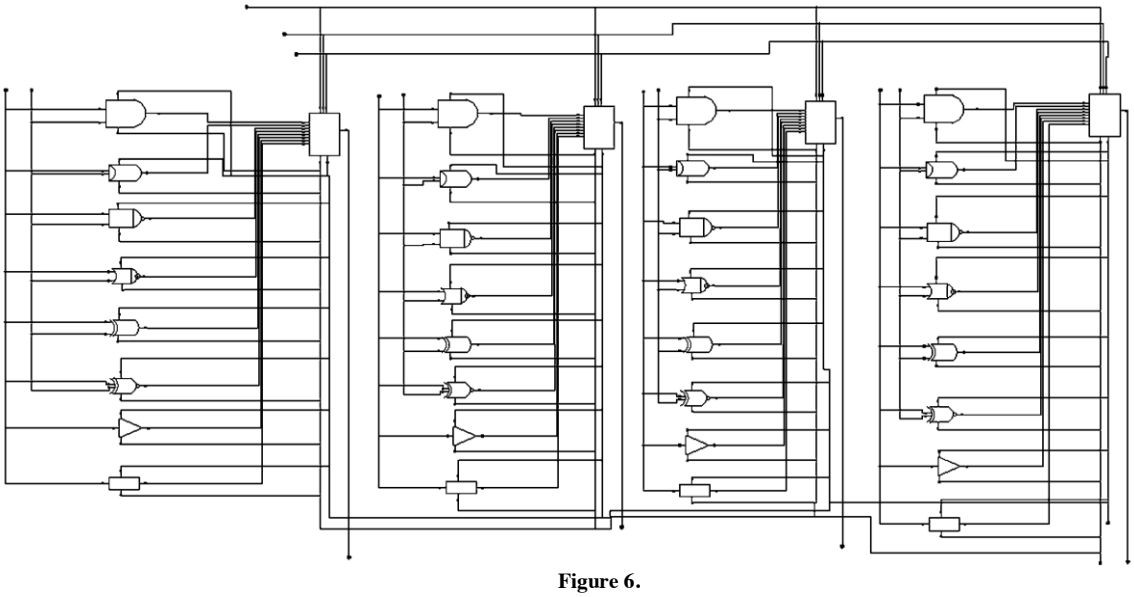

Figure 6.

Figure 6 Schematic of 4-bit Logical Unit

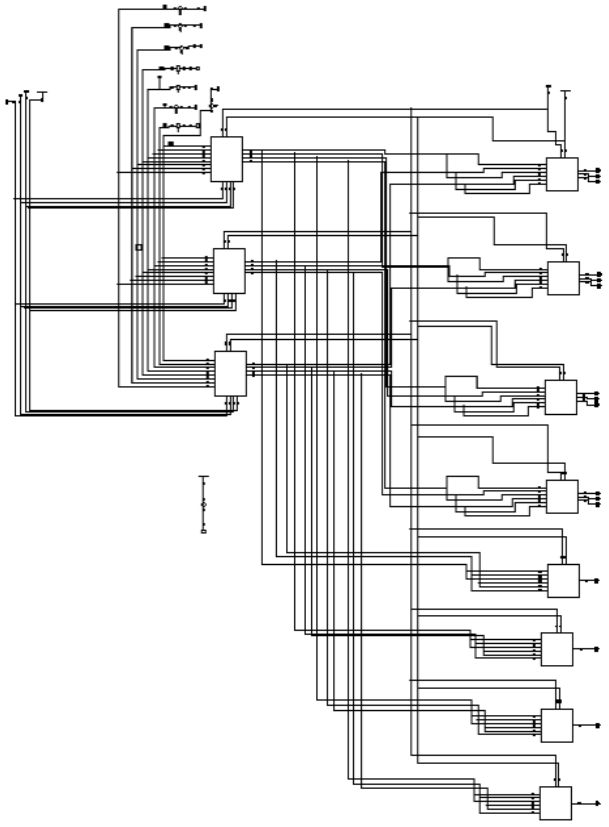

Figure 7. Fault Tolerant ALU system. The left three blocks are three ALUs combined the schematic of arithmetic unit and logical unit.

Table 2 demonstrates the conditions for the selection of correct ALU. A condition given under selection criteria should be fulfiled before voter logic gives any output. Table 3 describes the respective selection codes for various functions of designed ALU.

6. TESTING AND DIAGNOSE APPROACH USING TMR TECHNIQUE

A Fault skipping or masking technique, for instance triple modular redundancy (TMR), was embedded within the most critical and important Microprocessor's module like arithmetic and logical unit. The TMR technique [4] on line detects the operational faults and specially tolerates the transient faults well. However, TMR require extra area overhead, which can be accepted for the critical systems but it can also be a large trade off for regular systems where fault tolerance is not of much importance.

The concept of redundancy technique implies the addition of information, resources or time beyond what is required for regular system operation. The redundancy can take one of several forms, including software based, hardware based, information based and time redundancies. The physical replication of hardware components is perhaps the most common form of redundancy used in fault tolerant systems.

As semiconductor devices have become smaller and less expensive, the concept of hardware redundancy has become more common and more practical now a day. The fault tolerant technique concept is to hide the occurrence of random faults and prevent the faults from resulting in errors. The most common form of fault tolerance is the TMR. The basic concept of TMR is to triplicate the hardware and perform a majority vote to determine the output of the system. If one of the modules becomes faulty, the two remaining fault free modules will mask the results of the faulty one when the majority voting is performed. The primary difficulty with TMR is obviously the voter; if the voter fails, the complete system fails. The different fault tolerant techniques are given in Table 4.

Table 4. Comparison of Fault tolerant techniques

Study	Technique	Area (LUTs/FFs/Slices)	Overhead %	Reliability %
Temporal Redundancy with 36 bit Adder	None	109/70/73	NA	NA
	TSTMR	121/136/125	71	-66
	QTR	114/138/115	58	-68
	TMR	327/246/407	458	49
Quadded Logic (8 bit Adder)	None	3/0/2	NA	NA
	QL	37/0/21	1133	14.8
	TMR	10/0/6	233	27
Quadded Logic (8 bit Comparator)	None	3/0/2	NA	NA
	QL	29/0/16	866	15
	TMR	10/0/6	233	27.2

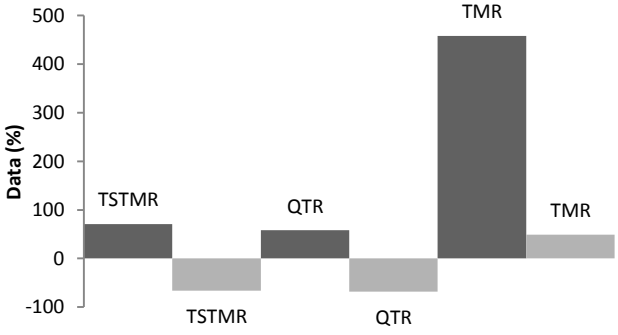

Figure 8. Comparison of fault tolerant techniques for temporal redundancy.

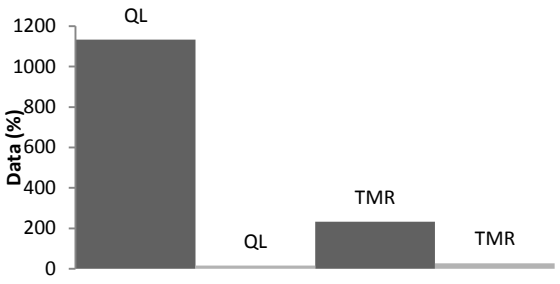

Figure 9. Comparison of fault tolerant techniques for 8-bit adder

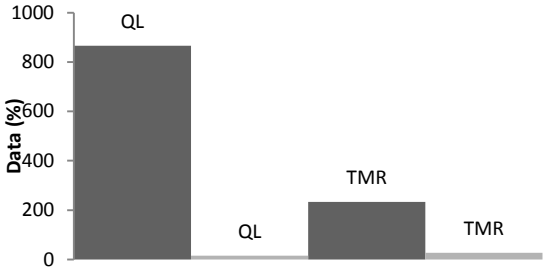

Figure 10. Comparison of fault tolerant techniques for comparator circuit.

7. RESULTS AND DISCUSSIONS

The schematic of an ALU design is shown in Fig. 5, the signal 'S' is used to select the required function, and 'A' and 'B' are input 4-bit data. As an example, if the system has to add 'A' = 1111 and 'B' = 1010 the 'S' signal should have value '0000' for addition operation assignment. The output signal 'F' leads to result '1001', with disagreement detector 'D' = 0-0 and 'D' = 4-2; providing high signal describes all the ALUs are working perfectly without any fault present. Fig. 4 shows the schematic of disagreement detector and Fig. 5 shows the designed arithmetic unit. Fig. 6 and Fig. 7, shows the Logical unit and Fault tolerant system design used in our system. Fig. 11 shows the output of system with all the ALUs working perfectly the transient response of four output blocks of system showing the system is working fault free. In our extensive testing of system with the help of simulations, the outcome was stable thus making this proposed system a fault tolerable and reliable system.

A. Power and Delay Analysis

The calculated output power required for the operation of circuit is 32.4 mW and delay is analyzed from the transient response of system. The delay calculated is 1.04 ns thus showing excellent performance while maintaining reliability of system. The power analysis of all the operations is given in Table 5.

B. Further Improvements using GDI Technique

GDI is a technique, which is suitable for design of fast, low power circuits using reduced number of transistors compared to traditional CMOS design and existing PTL techniques [10]. The basic GDI cell is shown in Fig. 12 and the basic functions performed by the GDI cell is shown in Table 6. Some changes in the Standard CMOS inverter derives to the basic GDI cell, where the sources of PMOS is not connected to the VDD and the sources of the NMOS is not connected to the GND. This feature gives the GDI cell two extra inputs to use, which makes GDI design more flexible than usual CMOS design [11]. The GDI technique is reported in [16] also.

Table 5. Power computation using CMOS 180nm

Sr.	Functions	Power Consumption (W)
1	NOT	2.89 uW
2	AND	8.73 uW
3	OR	9.89 uW
4	NOR	4.82 uW
5	NAND	5.68 uW
6	XOR	12.25 uW
7	XNOR	13.52 uW
8	Buffer	6.8 uW
9	4-bit Adder	45.5 uW
10	4-bit Subtraction	605.7 uW
11	4-bit Multiply	1.45 mW
12	4-bit Division	1.97 mW
13	Arithmetic Unit	24.14 mW
14	Logical Unit	8.32 mW
15	ALU	32.46 mW
16	Fault Tolerant ALU	92.4 mW

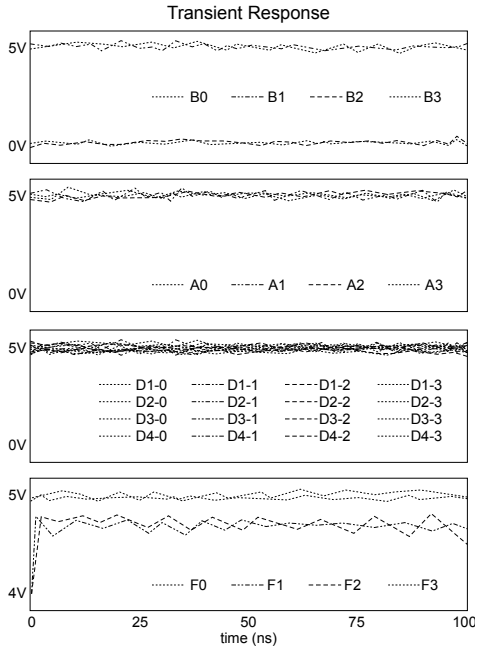

Figure 11. Transient response of Fault tolerant system. The response shows that the system has maintained the logic level of 5V and shows that all the functions are working perfectly.

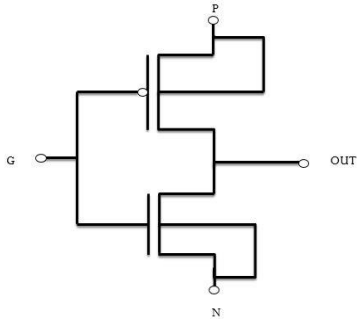

Figure 12. Basic GDI Cell that can be used to reduce the power consumption of the current fault tolerant system.

Table 6. Basic functions of GDI Technique

P	N	G	Output	Function
B	'1'	A	A+B	OR
'0'	B	A	A.B	AND
B	C	A	A(bar)B+AC	MUX
'1'	'0'	A	A(bar)	NOT
'1'	B(bar)	A	(A+B)bar	NOR
B(bar)	0	A	(A.B)bar	NAND
B	B(bar)	A	A(bar)B+A.B(bar)	EXOR
B(bar)	B	A	AB+AB(bar)	EXNOR

Table 7. Power consumption using GDI Technique

Sr.	Functions	Power Consumption (W)
1	NOT	2.89 uW
2	AND	78.73 nW
3	OR	100.7 nW
4	NOR	4.82 uW
5	NAND	5.68 uW
6	XOR	5.26 uW
7	XNOR	4.56 uW
8	Buffer	69.5 nW
13	Arithmetic Unit	4.82 mW
14	Logical Unit	12.7 uW
15	ALU	5.17 mW

The analysis shows that GDI is also a promising technique, we have analyzed till the basic behavior of ALU, there is a reduction in power but in fault tolerant operation, there are some glitches in delay. But in VLSI technology, for any single improvement, we have to pay off in some other parameter either delay/power or area overhead.

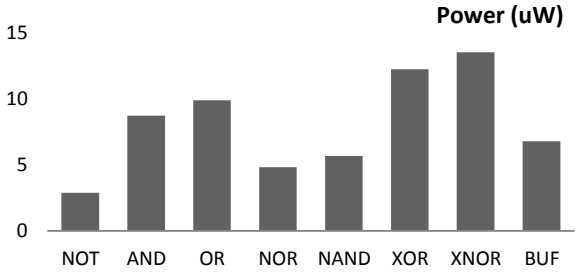

Figure 13. Power consumption of various gates designed using CMOS 180 nm process technology.

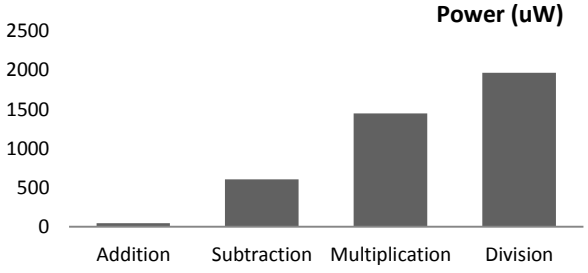

Figure 14. Power consumption of various arithmatic unit operations designed using CMOS 180 nm process technology.

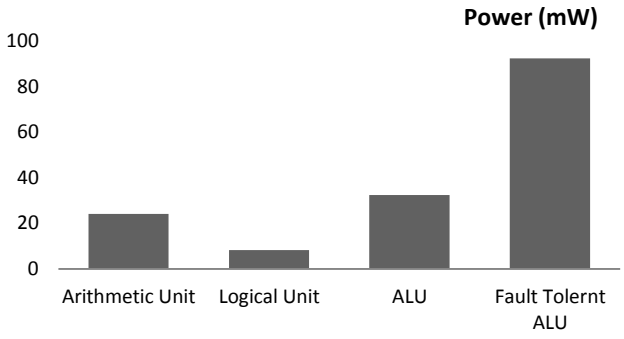

Figure 15. Power consumption of sub modules of ALU designed using CMOS 180 nm process technology.

The average power analysis is shown in Fig. 13 for all the basic gates designed using CMOS 180 nm process technology and Fig. 14 reports the power analysis of arithmetic operations viz. addition, subtraction, multiplication and division. It can be seen that addition consumes much less power followed by subtraction that needs more power than addition operation. Further, multiplication need more power than subtraction and division operation dissipated highest power among all operations.

Fig. 15 reports the average power dissipation of arithmetic unit and logical unit. The third column shows the dissipation of an ALU without any redundancy technique dissipates 32.46 uW of power while after applying TMR technique, the dissipation increased to 92.4 uW that also includes voting circuit and disagreement operation.

Although, triple modular redundancy seems promising but it consumes almost three times power than of regular system without any redundancy technique applied. Although, just to make the system reliable, first of all, more power consumption need to be accepted. To apply redundancy to critical systems, different low power techniques need to be applied to reduce the power consumption. GDI technique is shown for future work, with that the power consumption can be redunced.

Secondly, the use of redundancy technique increase area overhead of a chip. That means that for increased realiabily, almost more than 3x area is consumed due to continuous operation of system. Area can be reduced for TMR technique by using smaller process technology like 90 nm or 45 nm or ever beyond that but the trade off would be higher power dissipation and low performance. TMR technique althought serves the purpose of making reliable systems but still TMR technique fails to handle manufacturing defects or the type of defects that occurs in a system in which complete system can break down. For example, power surge can destroy the chip completely, hence TMR should be used to reduce transient faults only.

TMR relies on majority voting, hence if any two ALU report wrong output, majority output fails to deliver the correct response, however the chances are very slim that two ALU's give same wrong output, in that case, either all the three ALU's give correct output, or if the output of any ALU is faulty still majority voter report correct transient response of the system. This is practically not possible that two ALU's giving same wrong output, in that case the third output could be doubtful.

8. CONCLUSION

Fault tolerant and dependable ALU circuit is designed successfully, as there is a high need of reliable, accurate and fault free systems, triple modular redundancy approach is used to design a reliable system. Three ALUs are used simultaneously in the proposed design with provision to detect faulty output if any of ALU fails to operate while processing data. The circuit designed operates on low power and giving excellent performance. The proposed system being reliable and having fault tolerant capability needs only few mill watts of power for operation. This design can be used to implement computing systems used in medical equipment's, manufacturing units, and space systems where, accuracy and reliability are only factors as a matter of concern.

ACKNOWLEDGMENT

The authors would like to thank Lovely Professional University, PB for providing the work environment for testing the circuits in Cadence Design environment. Special thanks to Prof. P. Singh and Dr. K. Sehgal for their valuable discussion on fault tolerant systems.

REFERENCES

[1] V. S. Veeravalli, Fault Tolerance for Arithmetic and Logic Unit, *IEEE Southeastcon*, GA, Atlanta (2009) pp. 329-334.

[2] K. Matsumoto, M. Uehara and H. Mori, Evaluating the Fault Tolerance of Stateful TMR, in *13th International Conference on Network-based Information Systems*, Takayama (2010) pp. 332-336.

[3] J.Vial, A. Virazel, A. Bosio, P. Girard, C. Landrault, S. Pravossoudovitch, Is triple modular redundancy suitable for yield improvement?, *IET Computer Digital Technology* 3 (6) (2009) 581-592.

[4] N. M. Huu, B. Robisson, M. Agoyan and N. Drach, Low-cost fault tolerance on the ALU in simple pipelined processors, in *IEEE 13th International Symposium on Design and Diagnostics of Electronics Circuits and Systems*, Austria (2010) pp. 28-31.

[5] S. Subharani and R. Palaniappan, Fault tolerance for ALU - evaluated module redundancy technique, in *Proceedings of Singapore International Conference on Intelligent Control and Instrumentation*, India (1992), pp. 92-96.

[6] R. E. Lyons and W. Vanderkulk, The use of triple-modular redundancy to improve computer reliability, *IBM journal of research and development* 6 (2) (1962) 200-209.
[7] M. Hamamatsu, T. Tsuchiya and T. Kikuno" On the reliability of cascaded TMR systems, in *16th Pacific Rim International Symposium on Dependable Computing*, Tokyo (2010) pp. 184-190.
[8] P. Yin, Y. Chen, C. Lu, S. Shyu et al., A multi-state fault-tolerant multiplier with triple module redundancy (TMR) technique, in *IEEE 4th International Conference on Intelligent Systems Modelling and Simulation*, Bangkok (2013) pp. 636-641.
[9] V. Khorasani, B. V. Vahdat and M. Mortazavi, Analysing area penality of 32-bit fault tolerant ALU using BCH code, in *14th Euromicro Conference on Digital System Design*, Oulu (2011), pp. 409-413.
[10] D. Shinghal and D. Chandra, "Design and analysis of a fault tolerant microprocessor based on triple modular redundancy using VHDL," *International Journal of Advances in Engineering and Technology*, vol. 1, no. 1, pp. 21-27. 2011
[11] M. Dangeti, S. N. Singh, Minimization of transistor count and power in an embedded system using GDI technique, *Universal Journal of Applied Computer Science and Technology*, 2 (3) (2012) 308-313.
[12] P. Lee, C. Hsu and Y. Hung, Novel 10-T full adders realized by GDI structure, in *Proceedings of IEEE International Symposium on Integrated Circuits*, Singapore (2007) pp. 115-118.
[13] B. Batta, M. Choragudi, M. Varma, Energy efficient full-adder using GDI technique, *International Journal of Research in Computer and Communication Technology* 1 (6) (2012) 350-356.
[14] Arithmetic logic unit [*online*]: "http://en.wikipedia.org/wiki/Arithmetic_logic_unit"
[15] Fault tolerant system [*online*]: "http://www.answers.com/topic/fault-tolerant-system"
[16] ALU [*online*]: "http://www.techterms.com/definition/alu"
[17] Fred A. Bower, Daniel J. Sorin, and Sule Ozev (2005) ' A Mechanism for Online Diagnosis of Hard Faults in Microprocessors,' Appears in the *38th Annual International Symposium on Microarchitecture (MICRO*) Barcelona, Spain, November, 2005.
[18] T. Singh and R Kumar, Schematic design of fault tolerant and dependable ALU using gate diffusion input and triple modular redundancy technique, Elsevier's 2nd International Conference on Computing Sciences, PB, India, Nov 15-16, 2014. pp. 453-460.

Tejinder Singh received his Bachelor of Technology degree in Electronics and Communication Engineering from Lovely Professional University, PB, India in 2010 and Master of Technology degree in VLSI Design from Lovely Professional University, PB, India. He has received IEEE Asia–Pacific Region PG Paper Contest Award, IEEE M.V. Chauhan Award and various best paper awards. He is currently the reviewer of many reputed journals and currently serving on the technical program committee of various international conferences. He has authored many research papers in peer-reviewed journals and in international conferences. His research interests include designing and characterization of RF MEMS switches, Low-power VLSI circuits and Memristor/Memristive systems.

Farzaneh Pashaie received her Bachelor of Science degree in Electronics Engineering from Islamic Azad University, Saveh Branch, Tehran, Iran in 2010 and Master of Science degree in Mechatronic from Islamic Azad University, South Branch, Tehran, Iran in 2014. Her research interests include VLSI circuits and systems, Mechatronics and brain interfacing electrodes designing and characterization using Multiphysics environment.

Rajat Kumar received his Bachelor of Technology degree and Master of Technology degree from Lovely Professional University, PB in 2013. He is currently working as electronic systems hardware developer in Gingerbox InfoTech Pvt. Ltd., Chennai. He has co-authored research papers on fault tolerant systems. His research interests include designing and testing of reliability of computing systems.