Design of Various Adders Using Self Fault Detecting Full Adder

J.Dhanasekar, Aarthy Balan, R. Akshara, P. Anusuya, S.Deepa

ABSTRACT: ***To design various adders using self fault detecting full adder, since adders play an important role in computational operations. We have proposed a design of four different adders, which relatively requires less memory when compared to other self checking mechanisms like double modular redundancy and triple modular redundancy. Additionally the proposed design is more efficient in error detection. Even though an idle adder is being designed transient and permanent errors may occur due to noise, insufficient power supply problems etc., To overcome this issue we have designed a full adder that performs self-evaluation of its output in order to verify its operation, and also we have compared various limits for the proposed and existing mechanisms.***

Key words:- Double Modular Redundancy, Full Adder, Triple modular redundancy

I. INTRODUCTION:

Very large-scale integration (VLSI) is the method of designing an integrated circuit (IC) by combining thousands of devices or transistors into a single chip. This process reduces the area need of various complex circuits. One of the VLSI devices is the microprocessor. The VLSI design flow increases the life time and reliability of a circuit. There are various types of technology involved in VLSI. VLSI makes the electronic circuits to be compact and reliable. FPGA is one of the VLSI technologies which involve the gate level implementation of a design. Gate level implementation reduces the area and adds an advantage to the circuit that we design. Adder is one of an important basic device used to do various arithmetic and logic operations. Adders are used in most of the circuits where fast operations are carried out. There are various types of adders that are designed using some basic logic. Full adder is one type which is being used to design various other type of adders like ripple carry adder, carry skip adder, carry save adder etc., Full adders play a vital role in processors. Adders play a vital role in many real time applications, it is also used to do the same operations of a multiplier, divider etc., Adders are the basic building block of complex circuits and they do most of the operations at faster rate.

Revised Manuscript Received on March 04, 2019.

Mr. J.Dhanasekar, Assistant professor, Department of ECE, Sri Eshwar College of Engineering, Kinathukadavu

Ms. Aarthy Balan, UG students, Department of ECE, Sri Eshwar College of Engineering, Kinathukadavu

Ms. R.Akshara, UG students, Department of ECE, Sri Eshwar College of Engineering, Kinathukadavu

Ms.P.Anusuya' UG students, Department of ECE, Sri Eshwar College of Engineering, Kinathukadavu

Ms.S.Deepa, UG students, Department of ECE, Sri Eshwar College of Engineering, Kinathukadavu

The main aim of our project is to design a self fault checking full adder which performs efficient detection of errors. Also we design four different types of adders using the self fault checking full adder. The main aim of our project is to improve efficiency of diagnosing errors that occur in the adders, and to compare various parameters like power, delay, gate count, for the designed adders. This type of self checking mechanism is important where immediate human action is impossible, it can be even used in testing industries to verify the manufactured or designed adders. Error tolerance is one of the causes to be available in any type of operating circuits. Thus we have designed a fault checking full adder to improve the ideal feature of a conventional full adder.

II. EXISTING SELF –FAULT DETECTING FULL ADDER

There are various mechanisms that are involved in self checking like double modular redundancy check, triple modular redundancy checks etc.

Figure 1 shows that, fault is detected with the help of a full adder that is being connected to the original full adder. Redundancy is one of the common methods employed in self fault checking mechanisms. Additionally this mechanism requires more memory. This method also involves separate detection of errors in the sum as well as carry; this takes much time for detection process. It is also known as dual modular redundancy. Here in this type of self checking a duplicate adder that replicates the idea adder is being used to find the faults. There are many types of faults that occur in electronic circuits, of which most of the error occurs due to noise and power supply problems. This type of self fault detection is being implemented to avoid human work in testing.

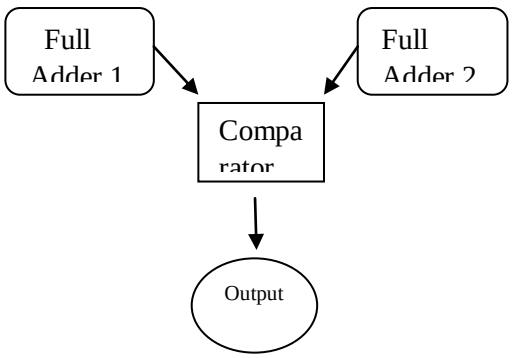


Figure 1: Double modular redundancy

From figure 2, it shows that it is same as the concept employed in double modular redundancy that is two redundant full adders are used with the original full adder. Those two redundant full adders process the provided inputs and produce the result to the comparator. The comparator then performs the comparison of outputs from the redundant full adders and the real full adder. The comparator checks for similarity in the outputs. If it predicts any error then it returns logic'1' and if no error or any dissimilarities occur with the compared output then it returns logic '0'. This is the basic operation of any redundancy checking methods.

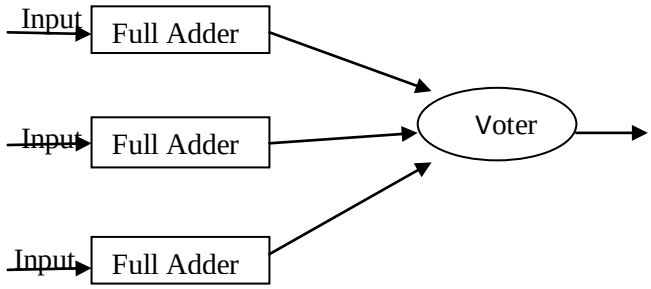


Figure 2: Triple modular redundancy

These types of mechanisms must sufficiently large hardware and it occupies more area in a chip. The efficiency of detecting faults by these methods is also less. In order to cut this drawback we have designed an efficient full adder that detects the faults that occurs in it. And for comparison of various limits between this redundancy method and the proposed method, we have designed four different types of full adder each with 16 bits.

III. PROPOSED FULL ADDER DESIGN

We have aimed at designing an adder that performs self detection in an efficient way. Here we have used a logical expression that is being implemented along with the original adder. The logical expressions process the inputs provided to the full adder. These logical expressions are implemented at gate level and it replaces the place of a redundant adder in the existing designs. And the logical block produces the outputs of its operation. By the time the original full adder also produces the output of sum and carry. Now the outputs from the logical block and the real full adder are compared for errors.

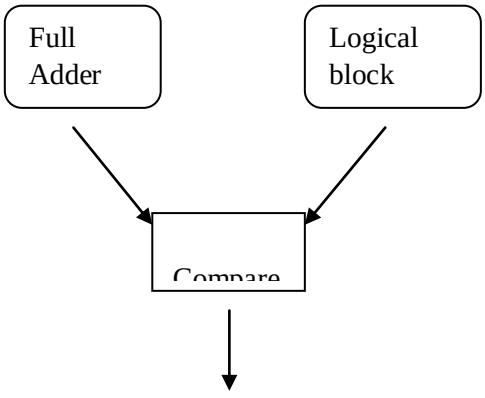


Figure 5: Self fault detecting full adder

The logical block in the above figure is being implemented using the logical expressions that perform certain operations of self detection.

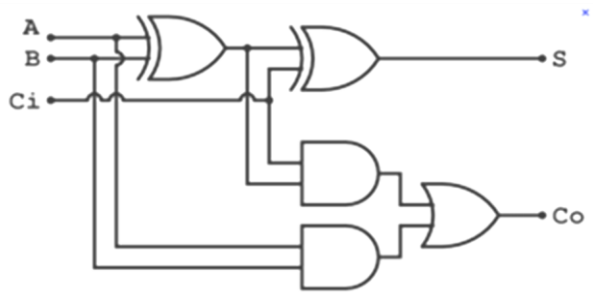


Figure 5: Full adder using two half adders

Sum and carry expression of a conventional full adder is; Output

$S = a \oplus b \oplus c$

$Co = a.b + c.(a + b)$

Table 1: Truth table of full adder

Truth Table			
Input		Output	
A	B	Sum	Carry
0	0	0	0
0	1	1	0
1	0	1	0
1	1	0	1

Here we have used this full adder to design four different adders such as ripple carry adder, carry skip adder, carry save adder, carry select adder. As these is the basic building block for most of the electronic devices. Speed in any device is implied on the performance of their integrated components , thus adder being one of the basic component used ,we have designed these adders in order to improve the overall performance of the circuits on the basis of their arithmetic and logic operations. There are various applications of full adder in the digital image processing, multimedia applications etc., on account of their applications increasing their performance would be the main demand. Using the proposed design we have designed four different adders each with 16 bits.

Outputs and Performance analysis: we have not only designed these adders but also we have compared various parameters between the existing and proposed designs. We have taken parameters like path delay, power, level of logic gates and memory.

All these parameters are measured and noted for the four adders that we have designed on the basis of both existing and proposed mechanisms. We have taken the RTL schematic and the output waveforms for all the four types of adders.

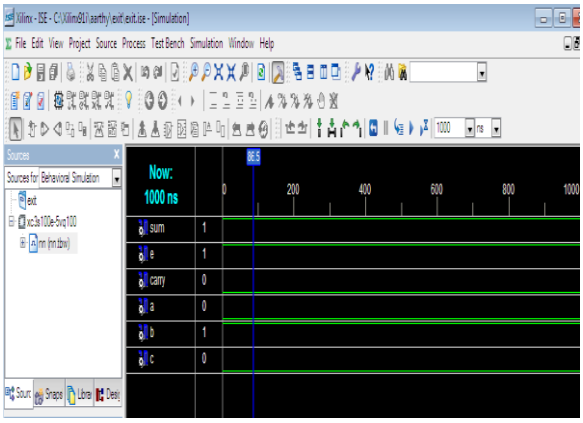


Figure 6: Output waveform for the self -fault detecting full adder using redundancy method

Using the existing full adders design we have designed ripple carry adder, carry skip adder, carry save adder, carry select adder. All the adders are designed for 16 bits. And their outputs are recorded.

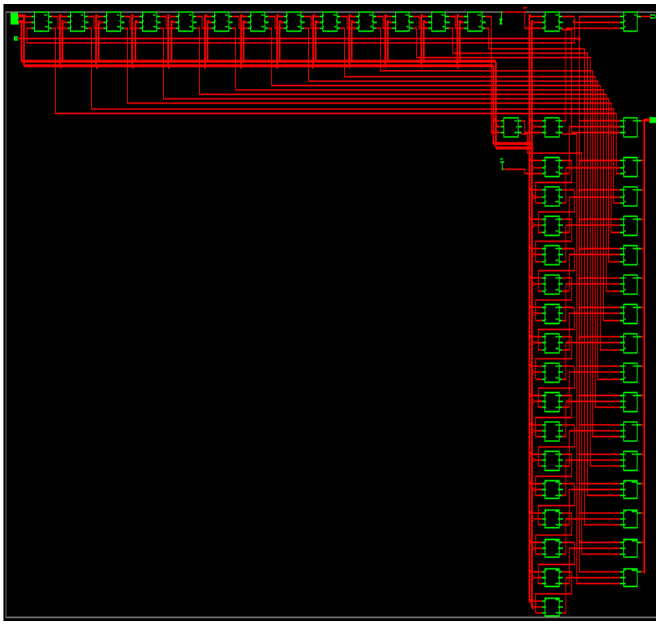
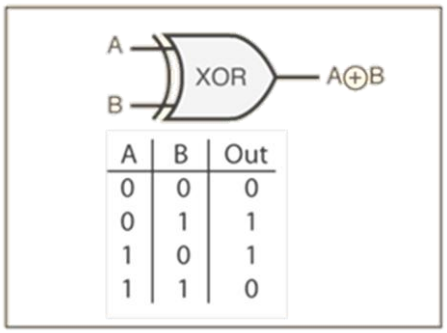


Figure 7: RTL schematic for 16 bit carry select adder designed using the existing redundancy method.

Using this self -fault detecting full adder we can also design multipliers and dividers that are being designed using the conventional full adders. In the proposed design we have little changes with the operations, that is here we have used some logical expressions that performs the operation of self detection. Finally comparison operation is done by a XOR gate.

Table 2: XOR gate and truth table



A	B	Out
0	0	0
0	1	1
1	0	1
1	1	0

Xor gate acts as the voter logic or as the comparator. The outputs from the logical block and the original full adder are fed to the Xor gate, which verifies and compares both the outputs for similarities. If it finds out any deviations in any one of the outputs, then it is considered to be an error. This method of detection requires less memory in a circuit, which is one of the important parameter to be considered.

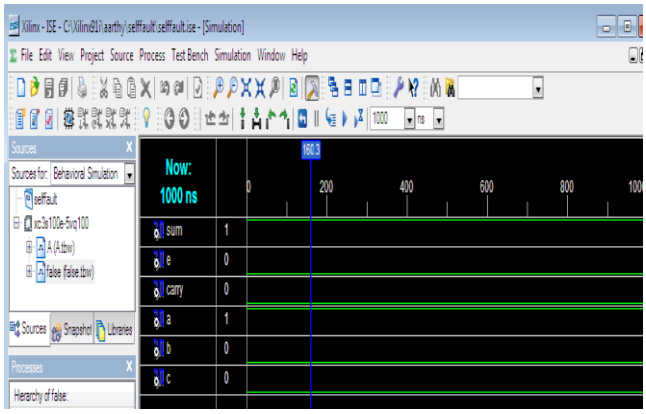


Figure 9: Output waveform of the proposed self fault detecting full adder.

The efficiency of both the methods is more or less similar, but still the memory required by the proposed design is subsequently less when compared to the memory occupied by the existing design. This adds an advantage to the adder that follows the logical processing method. Using the full adder that we have designed with fault detection mechanism, various other adders are designed... We have taken various reports like power report, delay reports to compare those parameters. All the other parameters than memory are similar for both the designs. Each and every parameter has been measured for all the four types of adders and they are compared.

IV. RESULTS AND DISCUSSION:

The output of various adders is obtained using Xilinx ISE 9.1 and the memory comparison in terms of Mega Bytes is shown in figure 10. Memory required for the proposed adders are reduced when compared with existing adders.

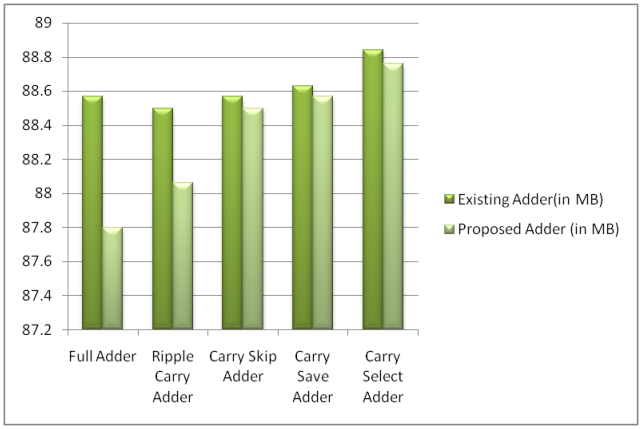


Figure 10 Memory Comparison

V. CONCLUSION AND FUTURE SCOPE:-

It is concluded that the self fault detecting mechanisms that are implemented requires less memory when compared to the existing design, it is an added advantage to the design. And it is easy to understand the operations in an effective way. This type of full adder will have a great impact in the growing technological world, where everything can be automated. Future implementation can be done based on the idea of self fault detection and corrections in an effective and in a fast way. These self-fault detection ideas can be used in testing industries to test the manufactured circuits that uses full adder.

REFERENCES:

1. Pankaj Kumar, Ranjender Kumar Sharma, "Real time fault tolerant full adder design for critical applications", volume 19, issue 3, September 2016. https://www.sciencedirect.com/science/article/pii/S2215098616300180
2. Abdelmonaem Ayachi, Belgacem Hamdi, "A fault tolerant full adder in double pass CMOS transistor", 2016. https://waset.org/publications/10003320/a-fault-tolerant-full-adder-in-double-pass-cmos-transistor
3. R. Matri Bai, G.Sahithi Reddy, "An efficient implementation of 16 bit carry select adder using Verilog HDL", august 2015. http://www.ijmetmr.com/olaugust2015/RMatriBai-GSahithiReddy-94.pdf
4. S. Muthulakshmi, "Detection of fault in self checking carry select adder", 2015. http://www.ripublication.com/irph/ijece/ijecev8n2_01.pdf
5. R.Pradhisha, N.Ishwarya, K.L.V. Gopinath Reddy, "FPGA implementation of self testing logic gates, adders and multipliers", September 2015. http://www.indjst.org/index.php/indjst/article/viewFile/79331/61641
6. 6. Ogun Turkyil, Fabien Clermidy," Self checking ripple carry adder with ambipolar silicon nanowire FET", 2013. https://ieeexplore.ieee.org/document/6572294
7. R.Rogini, G.Naresh,"Compression based self checking carry select adder design based on two rail encoding", April 2015. https://www.ijrcar.com/Volume_3_Issue_4/v3i414.pdf
8. Lavanya.K, Ramkumar Prabhu.M,"Performance analysis of low power high speed carry select adder", 2016.https://www.researchgate.net/publication/302579747_PERFORMANCE_ANALYSIS_OF_LOW_POWER_HIGH_SPEED_CARRY_SELECT_ADDER
9. Mudravoyina Srinivasa Rao, Bellam Varalakshmi,"Performance analysis of high speed CMOS full adder circuits for embedded systems", 2015.
10. https://www.ijcert.org/V2I1130.pdf
11. Allwin Deveraj,"Design of carry select adder with reduced area and power", 2015. https://www.researchgate.net/publication/311861717_Design_of_Carry_Select_Adder_with_Reduced_Area_and_Power
12. Balwinder Singh lakha, Simran Kaur, Jain D.K,"Design and performance analysis of various adders and multipliers using GDI technology", 2015. https://www.researchgate.net/publication/283811825_Design_and_Performance_Analysis_of_Various_Adders_and_Multipliers_Using_GDI_Technique
13. Vishalbharat R Hakki,"Design and implementation of 4 bit carry skip adder using NMOS pass transistor logic", Volume 6, issue 7, July 2017. https://ijcsmc.com/docs/papers/July2017/V6I7201748.pdf
14. David H.K. Hoe, L.P. Deepthi Bollepalli, Chris D.Martinez,"FPGA fault tolerant arithmetic logic: A case study using parallel prefix adder", Volume 2013.https://www.hindawi.com/journals/vlsi/2013/382682/