Design Efficient and Scalable Reversible Arithmetic Logic Unit for Fixed and Floating Point Data

Bhupendra Singh Ahirwar
Department of Ec/Vlsi
Beri, Bhopal (M.P) India

Prof. Sher Singh
Prof, Department Of Ec/Vlsi
Beri, Bhopal (M.P) India

Prof. Suresh S. Gawande
Hod, Department Of Ec/Vlsi
Beri, Bhopal (M.P) India

Abstract:-The advancement and improved of computerized world, for example, PC portable workstation adding machine and numerous computational gadgets utilized CPU. The CPU is heart of any computational and practical model. The fundamental centre some portion of CPU is ALU. The ALU play out the number-crunching operation for the working of advanced gadget. In this dissertation we design a very efficient ALU structure for the processing of high speed data and pattern analysis in the reversible mode. To improve the memory utilization and heat emission reduces the instruction cycle for the processing of data.

Keywords: -Arithmetic Logic Unit, Central Processing Unit, Ripple Carry Adder, Personal Digital Assistant.

I. INTRODUCTION

Arithmetic and Logic Unit (ALU) works as a data processing unit which is an important part in the central process unit (CPU) of any computer architecture. ALU is a multi-functional circuit that performs one of a few possible functions on two operands and which depends on the control inputs [7]. ALU needs to continually perform during the life-time of any computational devices such as a computer or a hand held device such as hand phone. Thus, reversible logic can be implemented in designing ALU to reduce the power dissipation and propagation delay in the circuits [1]. An arithmetic logic unit is a multi-functional circuit that conditionally performs one of several possible functions on two operands A and B depending on control inputs. It is nevertheless the main performer of any computing device. The ALU needs to continually perform during the life-time of any computational device such as a computer or a hand held device like PDA (Personal Digital Assistant) etc., Thus heat dissipation becomes a major issue in designing the ALU. Thus reversible logic can be aptly employed in designing the arithmetic logic unit. Also the ALU has to be resistant to the faults that may creep during the operation. Therefore it becomes more suitable that parity preserving reversible logic gates is used to design the ALU [3].

The basic component of the arithmetic circuit of the ALU is the parallel adder. This is the basic structure, though high speed adders such as Carry Skip Adder, Look Ahead Carry Adder, Carry Save Adder etc., can be used instead of the parallel adder. The parallel adder (also called Ripple Carry Adder) is constructed with a number of full adders connected in cascade. By controlling the data inputs to the parallel adder, it is possible to obtain different types of arithmetic operations.

A control unit is a circuit that directs operations within the computer's processor by directing the input and output of a computer system. A control unit consists of two decoders, a sequence counter, and a number of control logic gates. It fetches the instruction from instruction register (IR). For example, the block diagram of a 16-bit control unit is shown in Fig. 6. Here, the instruction register consists of 16 bits. The operation code (bit 12 to bit 14) is decoded by the 3-to-8 decoder. The outputs of the decoder are D0, D1. . . D7. Bit 0 to bit 11 and bit 15 are fed to the control logic gates. The 4-bit sequence counter counts from 0 to 15. The outputs of the counter rare decoded into 16 timing signals T0, T1. . .T15 by the 4-to-16 decoder.

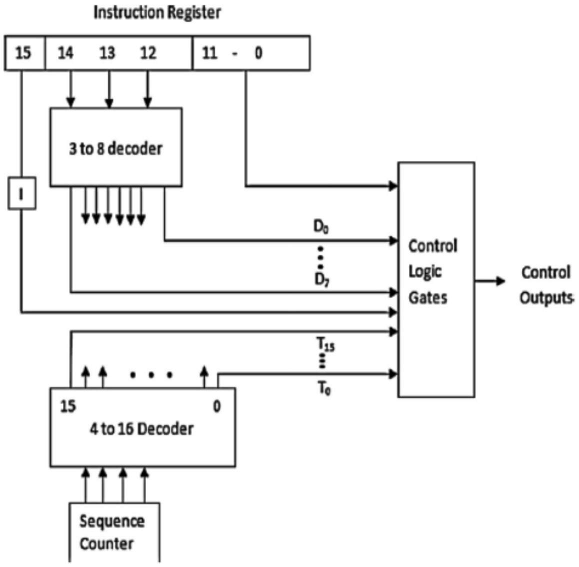

Fig. 1:- Block diagram of a 16-bit control unit.

The rest of this paper is organized as follows in section II we shows that the Parity Preserving Reversible Gate and Circuit Logic overview, in section III we discuss about the proposed methodology. In section IV we discuss about the Experimental result analysis for the proposed techniques. And finally in section V we conclude the paper and also define the future scope of this paper.

II. PARITY PRESERVING REVERSIBLE GATES AND CIRCUITS

A gate, a circuit or a function is reversible if and only if there is a one-to-one mapping between its input and output. Therefore, a reversible gate has an equal number of inputs and outputs. There is a number of commonly used reversible logic gate such as Feynman Gate, FG, Toffoli Gate, TG and Peres Gate, PG. Between reversible logic gates, those with their input parity being the same as their output parity are called “parity preserving reversible gates (circuits)” [8]. Most of arithmetic and other processing functions do not preserve the parity of the data. Parity checking is one of the most widely used methods for error detection in digital logic systems. Therefore it is important to construct parity preserving reversible gates and circuits. There are some problems using standard methods of error detection in reversible circuits, since fan-out is not allowed, and it may increase the number of gates being used along with the number of garbage outputs being produced. Given that reversible logic gates have the equal number of inputs and outputs, a sufficient requirement for parity preservation of a reversible circuit is that each gate to have a parity- preserving characteristics. Thus, a sufficient condition for having a parity preserving reversible logic gates is the implementation of the reversible circuit with each gate being parity preserving.

III. PROPOSED METHOD ARCHITECTURE

The advancement and improved of computerized world, for example, PC portable workstation adding machine and numerous computational gadgets utilized CPU. The CPU is heart of any computational and practical model. The fundamental centre some portion of CPU is ALU. The ALU play out the number-crunching operation for the working of advanced gadget. During the time spent outlining the ALU confronted more issue, for example, outflow of warmth and usage of more memory for the information operation. On the perspective of configuration issue utilized two methods for the handling of ALU outline rule one is chain structure and other is tree structure. In this paper outline an exceptionally effective ALU structure for the preparing of fast information and example examination. For the minimization of memory use and warmth outflow diminishes the direction cycle for the handling of information. In the coherence of section examine ALU configuration issue, chain structure of ALU configuration, Fixed Point, skimming point lastly talk about the proposed strategy and proposed display.

The chain structure useful parts are connected through a progression of multiplexers, every multiplexer takes a subset of utilitarian segments and pass the outcome to the yield. The Chain Structure of an ALU appeared underneath in below figure: The chain structure takes more levels of multiplexor transmission to the yield than Tree Structure. Despite the fact that the chain structure for the ALU spare region. In this paper, we examine the impact on way postponement of practical part arrangement in the chain structure we proposed a useful segment situation way to deal with decrease way delay for a given application.

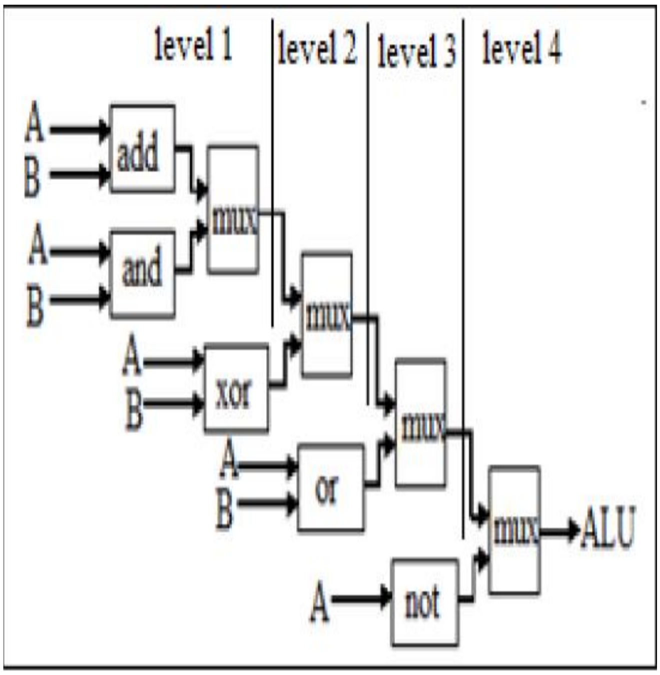

Fig. 2:- Shows that chin structure of ALU design.

In this area examine the 8-bit skimming point ALU outline with the end goal of force improvement and builds the life time of advanced circuit. The utilization of force builds the warm outflow in advanced circuit and circuit is harm and corrupted the execution of ALU. The planning of ALU is extremely basic issue in execution of PC and controller. The outline ALU perform well practices and utilitarian process for computational errand as far as number-crunching operation.

Every useful segment in the structure will work for any a count ask for, yet at most one of them executes the required estimation; different parts work to no end aside from practicing and expending power. We utilize execution to demonstrate a segment operation that reacts.

IV. EXPERIMENTAL RESULT ANALYSIS

In this section we discuss about the outline and reenactment of proposed model of ALU design for the preparing of drifting point number. The outline ALU reenact in Xilinx programming. The Xilinx programming form is 12.1.

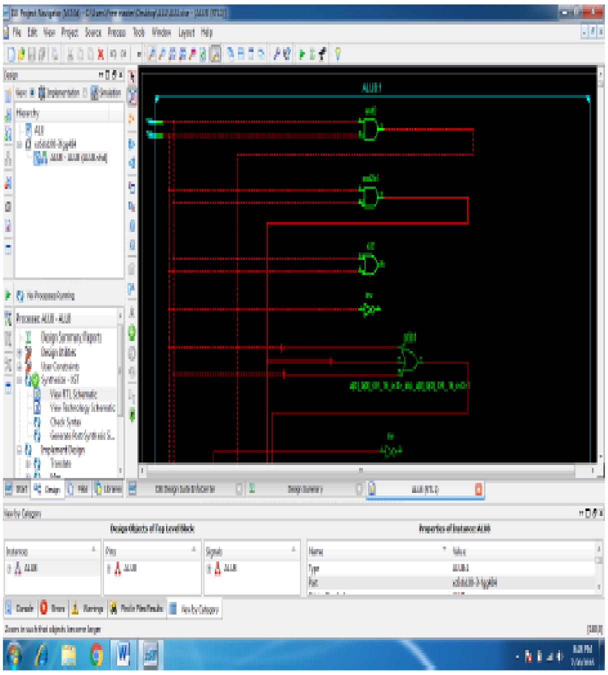

Fig. 3:- show that the window implementation RTL Schematic views in detail of ALU.

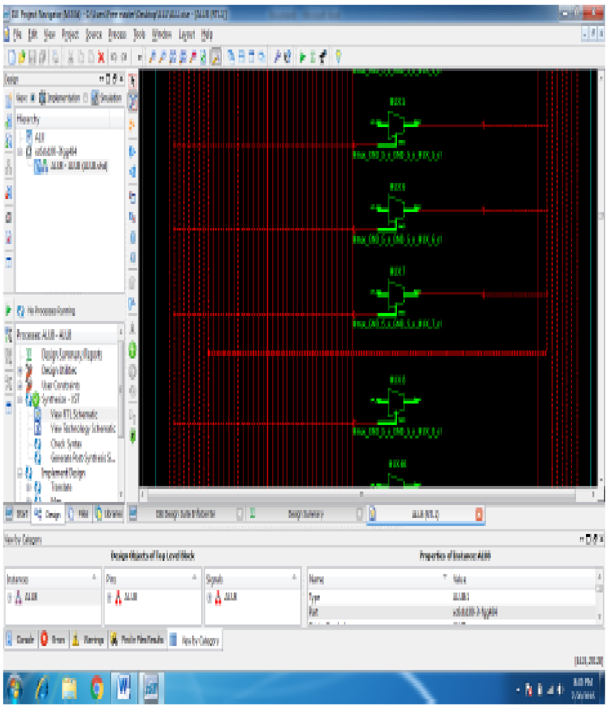

Fig. 4:- show that the window implementation RTL Schematic views in detail of ALU.

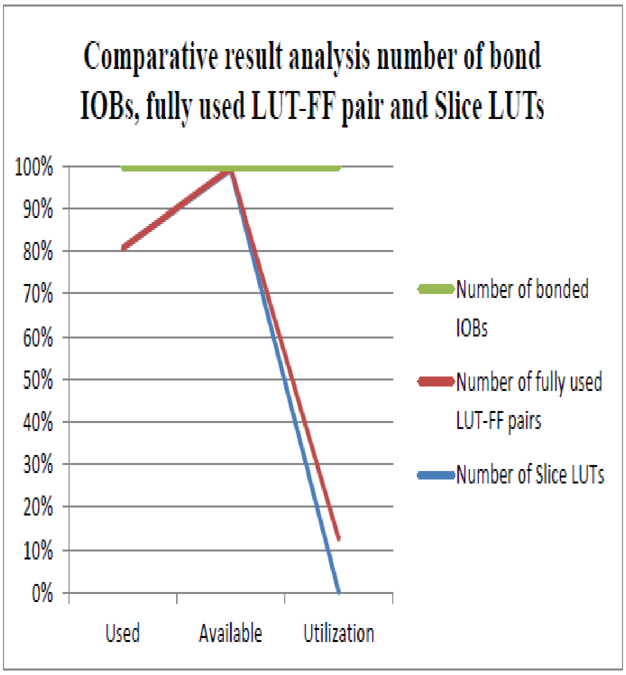

Fig. 5:- shows that comparative result analysis of number of bond IOBs, fully used LUT-FF pair and Slice LUTs in Logic Utilization.

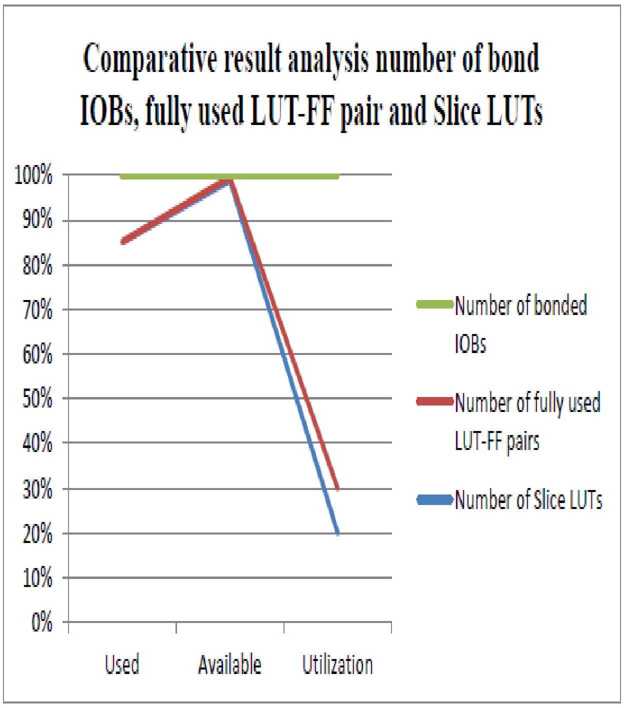

Fig. 6:- shows that comparative result analysis of number of bond IOBs, fully used LUT-FF pair and Slice LUTs in Logic Utilization.

V. CONCLUSION AND FUTURE WORK

Due of broad utilization of microchips and flag processors, execution of superior number juggling equipment has dependably remained an alluring outline issue. Number juggling and Logic Unit (ALU) is the workhorse of chip and decides the speed of operation of the processor. n thesis, proposed a gliding point ALU plan for low power impact. The impact of part arrangement on the chain structure plan. We found that the request of useful parts in the chain influences control utilization. To lessen control devoured by the chain, the as often as possible working segment ought to be situated near the yield of the chain. The throughput or execution of this processor can be enhanced propelled methods like pipelining.

REFERENCES

[1] Lenin Gopal, Nor Syahira Mohd Mahayadin, Adib Kabir Chowdhury, Alpha Agape Gopalai and Ashutosh Kumar Singh "Design and Synthesis of Reversible Arithmetic and Logic Unit (ALU)", IEEE, 2014, pp 289-293.

[2] Jeferson F. Chaves, Douglas S. Silva, Victor V. Camargos and Omar P. Vilela Neto "Towards reversible QCA computers: reversible gates and ALU" IEEE 2015. Pp 1-4.

[3] De Peng Kong, Qing Zhong Jia and Hong Xu "The Design of An Integrated Guidance and Control Computer System Based on Multi-core DSP and FPGA", CISP, 2015, pp 1625-1629.

[4] R. Anitha, Neha Deshmukh, arat Kumar Sahoo, S. Prabhakar Karthikeyan, Jacob Reglend "A 32 BIT MAC Unit Design Using Vedic Multiplier and Reversible Logic Gate" International Conference on Circuit, Power and Computing Technologies, IEEE 2015. pp 1-6.

[5] Shaik Shabeena, Jyotirmoy Pathak "Design and Verification of Reversible Logic Gates using Quantum Dot Cellular Automata" International Journal of Computer Applications, 2015. pp 1-4.

[6] Ravi Raj Singh, Sapna Upadhyay, Saranya S, Soumya, Jagannath KB, Hariprasad SA "Efficient Design of Arithmetic Logic Unit using Reversible Logic Gates" International Journal of Advanced Research in Computer Engineering & Technology, 2014. pp 1474-1477.

[7] Naman Sharma, Rajat Sachdeva, Upanshu Saraswat, Rajat Yadav, Gunjeet Kaur "Power Efficient Arithmetic Logic Unit Design using Reversible Logic" International Journal of Computer Applications, 2015. pp 36-41.

[8] Sayeeda Sultana, Katarzyna Radecka "Reversible Architecture of Computer Arithmetic" International Journal of Computer Applications, 2014. pp 6-14.

[9] Shefali Mamataj, Dibya Saha, Nahida Banu "A Review of Reversible Gates and its Application in Logic Design" American Journal of Engineering Research, 2014. pp 151-161.

[10] Goutam Paul, Anupam Chattopadhyay, Chander Chandak "Designing Parity Preserving Reversible Circuits" 2013. pp 1-9.

[11] Sadhu Suneel, L.M.L.Narayana Reddy "DSIGN OF A HIGH SPEED 8X8 UT MULTIPLIER USING REVERSIBLE LOGIC GATES" International Journal of Computer Science information and Engg., Technologies, 2014. pp 1-6.

[12] Lafifa Jamal, Hafiz Md. Hasan Babu "Design and Implementation of a Reversible Central Processing Unit" Computer Society Annual Symposium on VLSI, IEEE 2015. pp 187-190.

[13] Jenil Jain and Rahul Agrawal "Design and Development of Efficient Reversible Floating Point Arithmetic unit", Communication Systems and Network Technologies, 2015, Pp 811-815.

[14] Mozammel H A Khan "Classical Arithmetic Logic Unit Embedded on Reversible/Quantum Circuit", ICCITechn, 2012, Pp 1-6.