International Journal of Scientific Research in Engineering and Management (IJSREM)
Volume: 09 Issue: 04 | April - 2025 SJIF Rating: 8.586 ISSN: 2582-3930

64-Bit ALU Design Using Reversible Gates

Sri. C. MURALI MOHAN[1], T. SWATHI[2], G. APARNA[3], G. BANDHAVI[4], J. JAGADEESH[5],

R. KANNESH[6], M. SAINANDHAN REDDY[7]

[1]ASSOCIATE PROFESSOR, [2,3,4,5,6,7]STUDENT

[*1,2,3,4,5,6,7] Department of Electronics and Communication Engineering, SVIT, Anantapur, Andhra Pradesh, India.

Abstract-The implementation of reversible gates to build a 64-bit Arithmetic Logic Unit (ALU) is a promising contribution to the area of quantum computing as well as low-power digital circuit design. Reversible logic gates provide the benefit of not losing information throughout the process of computation (unlike traditional irreversible gates), which leads to energy being dissipated less frequently. The purpose of designing reversible logic is to decrease the power consumption and heat dissipation challenges present in modern Very Large Scale Integration (VLSI) design. The 64-bit ALU architecture has shown to improve computational efficiency by decreases in quantum cost, garbage outputs, and constant inputs, giving further credibility to it in the field of advanced computing systems in the future. The goal of this project is to execute arithmetic and logical operations through the use of reversible logic while still decreasing power consumption and delay. The constructed ALU utilizes reversible gates, while implementing reversible gates like a Fredkin gate, Feynman gate, and CNOT gate to implement favorable operations (addition, subtraction, and multiplication) including logical functions (AND, OR, XOR). The intent is to keep the number of garbage outputs and constant inputs to a minimum - garbage outputs and constant inputs are both important parameters of irreversible gates.

1. INTRODUCTION

With advancements in consumer electronics technology and communication systems, there is an increasing need for integrated circuit designs that use low power and high performance. One very important component of any processor design is the comparator, which determines the value of a signal compared to other signals. The comparator is important for proper digital signal processing and for multiple communications and networking protocols. Digital or Binary Comparators, developed through the combination of standard AND, NOR and NOT gates, will compare the digital signals that are presented to their input terminals and will produce an output based on the condition of the inputs. For example, we may want the ability to add and subtract binary numbers as well as could the number at input A is greater, less than, or equal to the number at input B. Currently, Reversible logic is one of the promising types of logics in different areas of application, including low power CMOS, quantum computing, nanotechnology, cryptography, optical computing, DNA computing, digital signal processing (DSP), quantum dot cellular automata, communication and computer graphics. Comparators currently in use include adder based comparator, priority encoder comparator, Bit-wise . Competition Logic comparator, etc. As with any new designs in digital circuits, speed is not only of the process fabric, low power.

2. LITERATURE SURVEY

[1] In a study conducted by R. Landauer (1961), it was shown that standard logic circuits consume a large amount of energy in the computation process because information is lost in the original computation. This crucial principle illustrated the necessity for reversible computation that could operate without losing information this would lessen a lot of energy from a non-reversible computation process.

[2] Bennett (1973) later showed that in reversible computation could theoretically be completed without energy loss thus enabling a broad range of research into constructing circuits composed of reversible logic gates such as Feynman, Toffoli, Fredkin, and Peres gates. The logic gates served as the foundations of many irreversible logic gate designs that we see today.

[3] Thapliyal and Ranganathan (2010) demonstrated a variety of reversible ALU architectures while using efficient combinations of logic gates. The key take away from their work focused on garbage outputs, constant inputs and quantum cost basically the design constraints that you will see used in a reversible logic system. Thapliyal and Ranganathan (2010) developed many

reversible ALU building blocks using Peres gates and Fredkin gates in their paper.

[4] A. Jain et al. (2017) introduced the RPLA (Reversible Programmable Logic Array) to implement a reversible 32-bit ALU in their findings. Their work indicated that there may be an optimized and scalable way to execute ALUs that would achieve lower power consumption. They also mentioned their circuit could be used as a basis for a 64-bit architecture as well.

3. PROPOSED METHODOLOGY

The methodology proposed here emphasizes developing a 64-bit Arithmetic Logic Unit (ALU) using reversible logic gates for low power and better computing performance. Reversible logic, defined as operating without information loss, is a key area of research in quantum computing and low power VLSI design.

REVERSIBLE GATES

3.1.Feynman Gate

The Feynman gate is a reversible 2x2 gate as seen in figure, with an input vector I(A,B) and output vector O(P,Q). The outputs are specified as P=A, Q=A B. The quantum cost of a Feynman gate is (1). The Feynman gate (FG) is useful as a copying gate since in reversible logic a fan-out is not possible, the FG gate allows an easy and quick way to make copies of the necessary outputs. The FG is a 2x2 gate and its logic circuit is depicted in the figure. The Feynman gate is another name for the Controlled Not (CNOT) gate. It has a quantum cost of (1), to be more specific, it is usually used for Fan Out. The 2x2 Feynman gate , also referred to as the controlled-not (CNOT) or “quantumEXOR”, realizes functions.

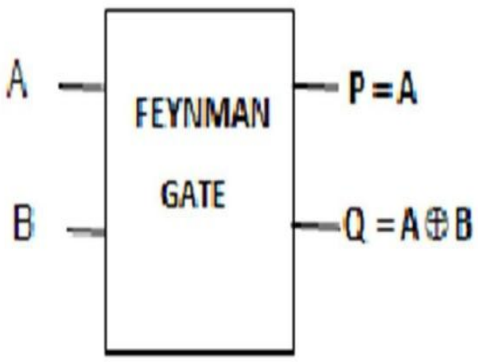

FIG 3.1 FEYNMAN GATE

Table 1 : Truth table of Feynman Gate

A	B	P	Q
0	0	0	0
0	1	0	1
1	0	1	1
1	1	1	0

3.2.Toffoli Gate

The figure below illustrates a 3*3 Toffoli gate. The input vector is I (A,B,C) and output vector is O(P,Q,R) where P=A, Q=B, R=AB xor C. The quantum cost of a Toffoli gate is 5. The 3*3 Toffoli gate is also called a 3*3 Feynman gate or controlled-controlled-not.It can be represented the equations below: P = AQ = BR = (AB) xor C.

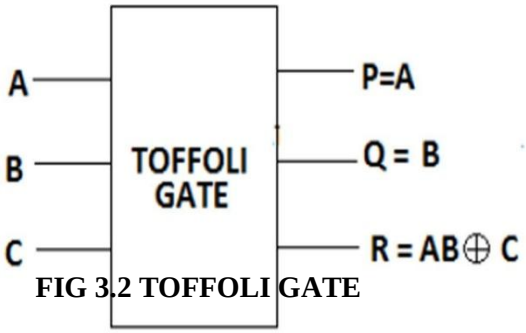

FIG 3.2 TOFFOLI GATE

A	B	C	P	Q	R
0	**0**	**0**	**0**	**0**	**0**
0	**0**	**1**	**0**	**0**	**1**
0	**1**	**0**	**0**	**1**	**0**
0	**1**	**1**	**0**	**1**	**1**
1	**0**	**0**	**1**	**0**	**0**
1	**0**	**1**	**1**	**0**	**1**
1	**1**	**0**	**1**	**1**	**1**
1	**1**	**1**	**1**	**1**	**0**

3.3.Fredkin Gate

The figure below depicts a 3x3 Fredkin gate; the input vector is I (A, B, C) and the output vector is O (P, Q, R). The output is defined as P=A, Q=(A'B)xor(AC), and R=(A′C)xor(AB). The quantum cost of a Fredkin gate is 5.It is a 3x3 gate with a circuit logic that is shown in the figure with a quantum cost of five. It can also be used as a multiplexer. Classically, this gate is just two multiplexers in the flipped (permuted) from the same control input A. The simplified notation of the circuit is shown in figure , with the truth table **.**

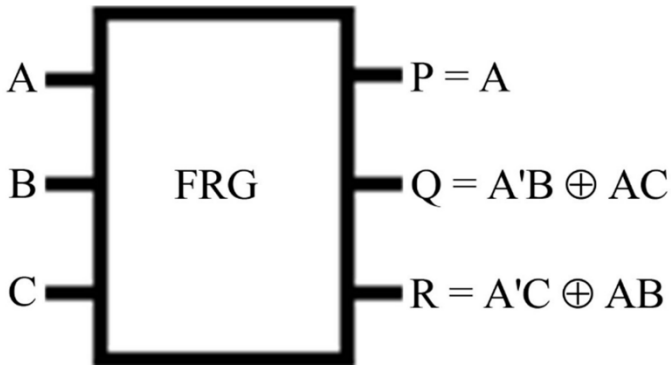

FIG 3.3 FREDKIN GATE

A	B	C	X	Y	Z
0	0	0	0	0	0
0	0	1	0	1	0
0	1	0	1	0	0
0	1	1	1	1	0
1	0	0	1	0	0
1	0	1	1	0	1
1	1	0	1	1	0
1	1	1	1	1	1+

Table 2:Truth table of Fredkin gate

4. SIMULATION RESULTS

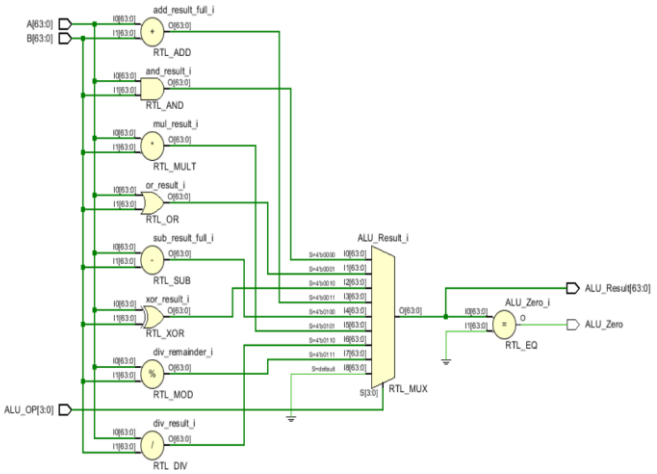

Figure 4.1:RTL Diagram of ALU

1. Inputs:

A [63:0] and B [63:0]: These are the two independent inputs of 64-bits for the ALU.
ALU_OP [3:0]: A 4-bit control input that selects the operation desired.

2. Operating Blocks (or Modules):

RTL_ADD - Addition.
RTL_SUB - Subtraction.
RTL_MULT - Multiplication.
RTL_DIV - Division.
RTL_MOD - Modulus.
RTL_AND, RTL_OR, RTL_XOR - Logical bitwise operations.
RTL_EQ - Compares the values of A and B and determines if they are equal.

3. Multiplexer Block:

RTL_MUX: All the output from the modules feeding the MUX block and the MUX selects one result to move forward.
The selection is based on the value of ALU_OP and the value selected is sent to the output ALU_Result[63:0].

4.Outputs:

ALU_Result [63:0]: The final result.
ALU_Zero : Indicates whether the result is zero.

Area Utilization :

Resource	Utilization	Available	Utilization %
LUT	8825	53200	16.59
DSP	10	220	4.55
IO	197	200	98.50

Figure 4.2: Area Utilization

Power :

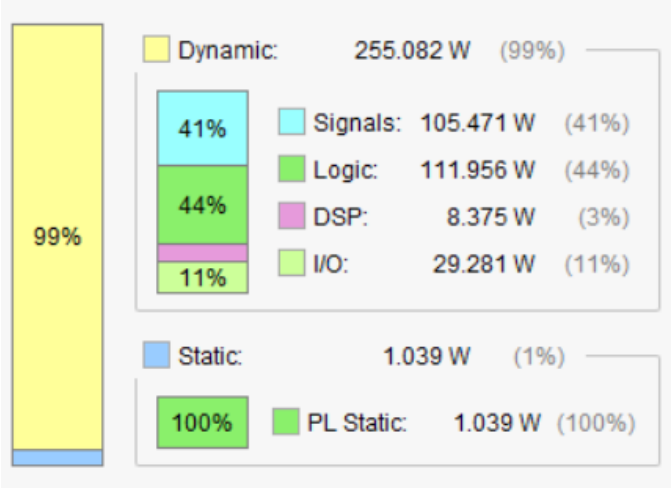

Figure 4.3: Power

Output :

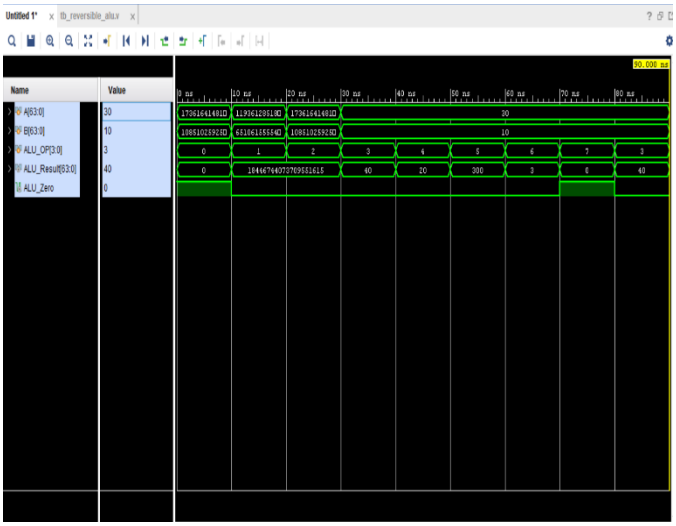

Figure 4.4: Output

5. CONCLUSION

Since then, the growth of computer hardware performance has been incredible! It is well known that most computational resources are energy, and this is directly related to the reversibility of the computation. The main purpose behind this project was to develop an understanding of Reversible Computation and to use this to build energy efficient devices that will last a long time. The multiplier is a primary arithmetic cell in computer arithmetic units; and when higher order multiplication is performed, it is assumed that it creates the most power. In this work, we synthesized a parity preserving reversible multiplier circuit taking advantage of the reversibility of the existing fault tolerant Fredkin, F2G and IG.

6. FUTURE SCOPE

Our discussion is only focus on logical reversibility - the inputs and outputs of reversible logic gates can be uniquely retrievable from each other. As such, there are many challenges in the way of researchers before reversible logic can actually be turned into technology that lives up to its practical competitive potential like.

1. Need for efficient synthesizing methods to build complex reversible units;
2. Need for optimizing algorithms to reduce for constants and garbage bits;
3. Testing and verifications tools when designing reversible logic. Additionally, the growth of the speed and complexity of the computing systems.

7. REFERENCES

[1] Ravish Aradhya H V, Muralidhara K N and Praveen Kumar B V, "Design of low power arithmetic unit based on reversible logic," in International Journal of VLSI and Signal Processing Applications, Vol. 1, pp. 30- 38, 2011

[2] Md. Saiful Islam "Fault Tolerant Variable Block Carry Skip Logic (VBCSL) Using Parity Preserving Reversible Gates" International Journal of Computer and Electrical Engineering, Vol. 3, pp1793-8163 February, 2011

[3] H.R.Bhagyalakshmi et al / International Journal of Engineering Science and Technology Vol. 2(8), pp 3838-3845, 2010

[4] Anindita Banerjee and Anirban Pathak, "Reversible multiplier circuit," in Third International Conference on

Emerging Trends in Engineering and Technology (ICETET), pp 781-786, 2010

[5] S. Offermann, R. Wille, G. W. Dueck, and R. Drechsler, "Synthesizing multiplier in reversible logic," in Symposium on Design and Diagnostics of Electronic Circuits and Systems, pp 35-340, 2010

[6] M. Ehsanpour, P. Moallem and A. Vafaei, "Design of a Novel Reversible Multiplier Circuit Using Modified Full Adder," in Proceeding International Conference on Computer Design And Appliations (ICCDA), Vol. 3, pp 230-234, 2010

[7] M. Nachtigal, H. Thapliyal, N. Ranganathan, "Design of a reversible single precision floating point multiplier based on operand decomposition", Proceedings of the 10th IEEE International Conference on Nanotechnology, Seoul, Korea, pp 233-237August 2010

[8] H.R.Bhagyalakshmi and M.K.Venkatesha, "An improved design of a multiplier using reversible logic gates