International Journal of Research in Engineering and Science (IJRES)
ISSN (Online): 2320-9364, ISSN (Print): 2320-9356
www.ijres.org Volume 10 Issue 3 ǁ 2022 ǁ PP. 53-65

Design and Analysis of Reversible Control Unit for Arithmetic and Logical Operations

Gaganashree B[(1)], K.B Ramesh[(2)]
(1)U G student,Department of Electronics and Instrumentation Engineering,R V College of Engineering,Bengaluru,India.
(2) Associate Professor,Department of Electronics and Instrumentation Engineering,R V College of Engineering,Bengaluru, India.

Abstract:-
In recent years, reversible logic has gained traction in fields such as low-power VLSI design, nanotechnology, and quantum computing. The proposed design performs 8 arithmetic and 7 logical operations.It allows for lower power usage and shorter quantum delays, resulting in faster processing. An ALU is that the most simple component of any digital logic programmable device or ADPS. the utilization of reversible computing techniques within the design of an ALU can considerably improve the performance and speed of digital systems. Furthermore, compared to existing traditional ALU designs, reversible logic based ALU consumes significantly less power. a longtime ALU design and a unique ALU design are contrasted and analyzed for various bit lengths during this study (1,8,16,32,64). The properties of reversible logic design, as well as power consumption metrics, are thoroughly examined. The proposed design outperforms the present design, it are often concluded.
Keywords: *Reversible logic, reversible ALU design, quantum cost, garbage output, low power VLSI design.*

Date of Submission: 28-02-2022 Date of acceptance: 09-03-2022

I. Introduction

Computers and other digital logic processing devices like mobile phones, calculators, and other gadgets have seen amazing growth over the previous few decades. the foremost difficult aspect of any computing unit's design is that the control unit, which has more important constraints. Power consumption may be a crucial factor in today's VLSI designs. Advances in VLSI designs, particularly portable device technologies, and increasingly high computing needs have enabled the development of faster, smaller, and more sophisticated electronic systems. With the introduction of multi-gigahertz CPUs and high-end electrical devices, system complexity, high-density packaging, and power consumption concerns have increased. Computations and operations became more efficient and rapid because to the incorporation of uncountable transistors into one chip. High transistor density, on the opposite hand, will lead to increased power dissipation. Existing technologies, like as CMOS, are expected to hit their limits within the near future, in keeping with today's researchers. the foremost significant issues with current technology are power dissipation and vulnerability to computing errors. In CMOS VLSI design, power optimization may be done at multiple abstraction levels

•At the Device (Technology) level, techniques like VT reduction, multi-threshold voltages, gate oxide thickness, and length and width variations are more common.

• At the circuit level, strategies such as using alternate devices and re-structuring networks; at the logic level, techniques such as using alternate logicvels ,styles,energy recovery methods are common

• Techniques like as parallel structures, pipelining, state machine encoding, different encoding methods, and others are increasingly frequent at the architecture (system) and algorithmic levels. The energy recovery technique, which uses reversible logic concepts at the circuit and logic level, is one example. Reversible Logic is assumed to be a big breakthrough that might address the shortcomings of current technologies. Because there's no energy or information loss in reversible circuits, they're mentioned as lossless circuits. These circuits are ideal for applications requiring very low power consumption or cooling, like communications, low-power VLSI (Very Large-Scale Integration) technology, DNA computing, and nanotechnology. Furthermore, in quantum computing, where quantum development is intrinsically reversible, reversible logic has proven to be particularly beneficial.

II. Definitions regarding reversible logic

a)Quantum Cost- In terms of the value of a primitive gate, quantum cost refers to the circuit's value. It is determined by determining how many simple reversible logic gates (1*1 or 2*2) are required to comprehend the circuit. Every reversible computer circuit's quantum cost is a critical optimization parameter.The quantum cost

of a circuit is the smallest number of 2*2 unitary gates required to represent the circuit while maintaining the output. The quantum cost of a 1*1 gate is 0 which of any 2*2 gate is that the same, which is 1.
b) Hardware Complexity -Hardware complexity refers to the entire number of logical calculation (TC) during a circuit counting the number of EX-OR operations, AND operations, NOT operations, and OR operations determines the hardware complexity (HC).Calculate the reversible circuits' hardware complexity, assuming:
α - no. of two input EX-OR gate operations
β - no. of two input AND gate operations
δ - no. of NOT gate operations
Ω - no. of OR operations
T - total logical operations.
Because the sum of AND, OR, EX-OR, and NOT computations, the entire logical calculation T lean.
c) Total On-Chip Power (TOCP) is the amount of power consumed by the FPGA internally, up to the sum of device static and dynamic power. Thermal Power is another name for it.
d) Device Static Power- the facility from transistor leakage on all connected voltage rails and therefore the circuits required for the FPGA to control normally, post configuration. Instrument Process, voltage, and temperature could all influence static power. This represents the device's intrinsic leakage in its steady state.
e) Dynamic Power-The power consumed when all the inputs are active. It is dependent on the voltage levels, as well as the logic and routing resources employed.
f) LUT-A LUT (Look-Up Table) could be a small asynchronous SRAM that's accustomed implement combinational logic. The content of LUTs is normally read-only, and it can only be altered during FPGA configuration.
g) Nets and Leaf Cells-A net is a set of interconnected pins and wires. Standard cells from an ASIC library, memory, macro cells, and IP, all of which would take up space in the core region, might be used as leaf cells. These are the foundation cells that will be used in subsequent design and layout.

III. Basic reversible logic gates

It is an n-input n-output logic function in which the inputs and outputs have a one-to-one correlation. The input vector can be uniquely obtained from the output vector as a consequence of this bijective mapping. Data loss, which is the principal source of power consumption in irreversible logic circuits, is avoided as a result. The following considerations must be made when building reversible logic circuits in order to attain the best results. The following considerations must be made when building reversible logic circuits in order to attain the best results.
They are

- Fan-out is not permitted.
- Loops or feedbacks are not permitted
- Garbage outputs must be minimum
- Minimum delay
- Minimum quantum cost.

The simplest reversible gates are 1 * 1 gates, not gates. An A 2 * 2 gate is an example of a controlled NOT (CNOT) gate. The 3 * 3 reversible gate samples include the F, TG, PG, and TR gates, to name a few. A 1*1 reversible gate has a quantum cost of zero, while a 2*2 reversible gate has a quantum cost of one.
To make all reversible gates, 1*1 NOT gates and 2*2 reversible gates, such as V, V+, and FG gate, which is the same as CNOT gate, are employed. Equations 1, 2 and 3 describe the property of the V and V+ Quantum gates.
V * V = NOT …… (1)
V * V+ = V+ * V = I …….. (2)
V+ * V+ = NOT ……. (3)
Counting the number of V, V+, and CNOTgates can be used to calculate the Quantum Cost of a Reversible Gate.

1.**Not gate**
This is a 1*1 reversible logic gate with only one input and output. As indicated in the picture, the input A is mapped to the output P=A'. NOT gate as a representationoutput P=A’ as shown below in the figure

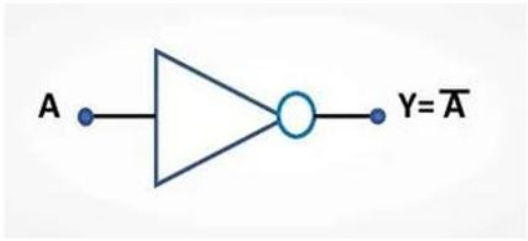

Fig 3.1 Representation of NOT gate

2.Feynman gate

Reversible gate 2 * 2 with inputs and outputs. Inputs (A, B) mapped to outputs (P = A, Q = A ^ B) are proven on the side of the figure.

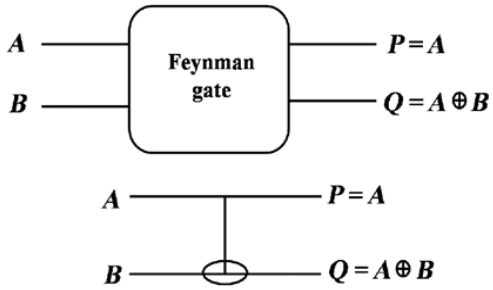

Fig 3.2 Representation of Feynman gate

3.Fredkin gate

This is a three-input, three-output reversible gate.The figure shows the inputs (A, B, C) mapped to the outputs (P = A, Q = A'B + AC, R = AB + A'.C).

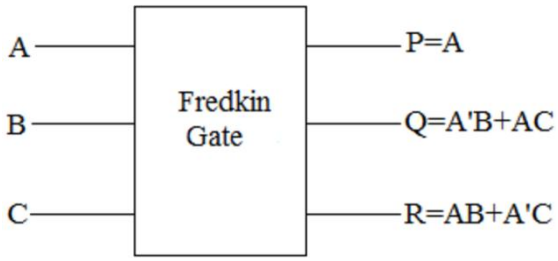

Fig 3.3 Representation of Fredkin gate

4.Toffoli gate

This is a 3 * 3 reversing valve with three inputs and three outputs. Inputs (A, B, C) are mapped to outputs (P = A, Q = B, R = A • B ⊕ C) as shown. Presentation of the Toffoli gate.

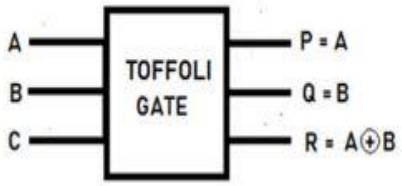

Fig 3.4 Representation of Toffoli gate

5.Peres gate

The Peres gate is a reversible gate with three inputs and three outputs (3x3) mapping (A, B, C) to (P=A, Q=A⊕B, R=(A.B)⊕C), where A, B, C are the inputs and P, Q, R are the outputs, respectively. Also known as the New Toffoli Gateway (NTG). It was built by Toffoli Gate and a Feynman one. It has a quantum cost of 4.

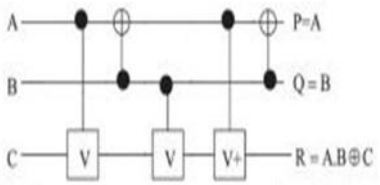

Fig 3.5 Representation of Peres gate

6.DKG gate

This is a 4*4 Reversible gate with four inputs and four outputs. The inputs (A, B, C, D) mapped to the outputs (P=B, Q=A'.C+ A.D' , R=A.^ B.C ^ D ^C.D, S=B^C*D) is shown in the Figure

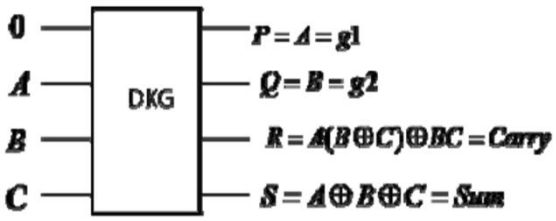

Fig 3.6 Representation of DKG gate

7.WG gate

This is a 4 * 4 reversible gate with 4 inputs and 4 outputs. The inputs (A, B, C, D) are mapped to the outputs (P = A, Q = A + B + D, R = A. + B + C, S = (A ⊕ D ⊕ B) (A ⊕ D ⊕ B). ⊕ D ⊕ C) ⊕ (A ⊕ D)) is shown in the figure.

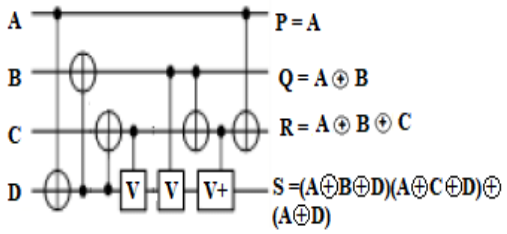

Fig 3.7 Representation of WG gate

8.HNG gate

This is a 4 * 4 reversible gate with four inputs and four outputs. The inputs (A, B, C, D) assigned to the outputs (P = A, Q = B, R = A ⊕ (B ⊕ C), S = A ⊕ B .C ⊕AB) are shown in the figure

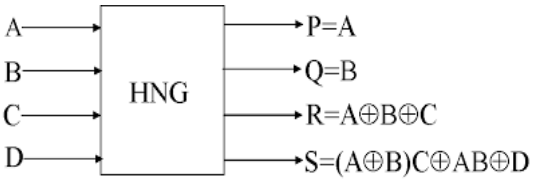

Fig 3.8 Representation of HNG gate

9.TR gate

It is a reversible valve with 3 inputs and 3 outputs. Inputs (A, B, C) are mapped to outputs (P = A, Q = A ⊕ B, R = (AB ') C).

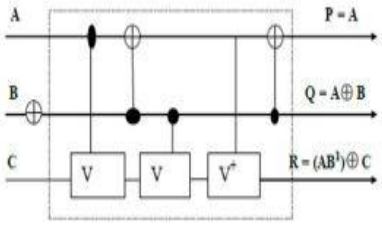

Fig 3.9 Representation of TR gate

GATE	QUANTUM COST	Hardware complexity
NOT	1	δ
FEYMAN	1	α
FREDKIN	5	4β+2 δ +2 Ω
TOFFOLI	5	α+β
PERES	4	2α+β
DKG	6	5α+4 β+2 δ+ Ω
HNG	6	4 α+2β
TR	6	2α+β+ δ
WG	7	6 α +4 Ω

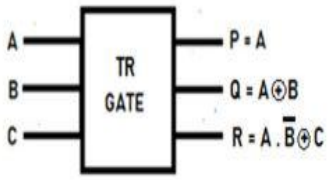

Fig 3.10 Reversible logic gates' quantum cost and amount of hardware complexity

IV. Design and Implementation

A parallel adder is the most fundamental component of an arithmetic logic unit (ALU). You may execute arithmetic operations like addition, subtraction, increment, and decrement, as well as logical operations like AND, OR, NOT, NOR, NAND, EXOR, and EXNOR, by controlling the inputs to the parallel adder.

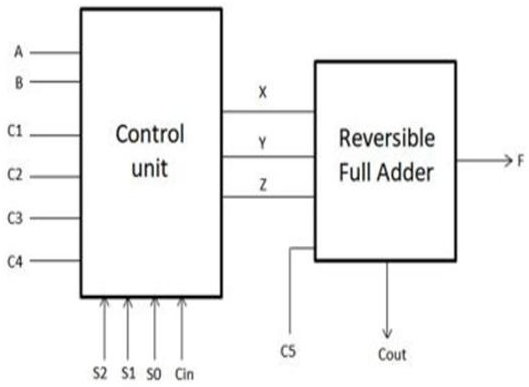

Fig 4.1 Block diagram of reversible ALU

By cascading one-bit arithmetic and logic units, we can obtain ALUs of different bit lengths(8 bit, 16 bit, and so on). The generalized block diagram of a reversible logic ALU is shown in figure 19 which consists of a control unit and a full adder unit both constructed using reversible gates. A and B are operands on which the ALU performs operations and the outputs are F and Cout(Output Carry). The inputs Cin, S0, S1, and S2 are used to select different arithmetic and logic operations that the ALU can perform. C1,C2,..., and C5 are either logic 1 or logic 0 elements that allow the circuit to conduct the desired operations while remaining constant throughout the design.

1.Existing design

In this design, the control unit is built using Fredkin gates, two Feynman gates and three Perez gates, and the DKG gate is used as a total adder. The S and R outputs of DKG, as shown in Figure 11, will output F and carry Cout. A one-bit ALU is shown in Figure 20. This design requires a quantum cost of 26. There are five fixed inputs and nine junk outputs. It can perform eight arithmetic operations and seven logical operations.

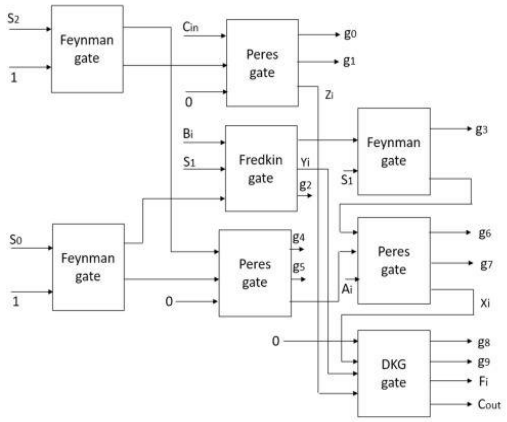

Fig 4.2 Circuit diagram of existing design reversible ALU

S2	S1	S0	Cin	Output	Function
0	0	0	0	F = A	Transfer A
0	0	0	1	F = A + 1	Increment A
0	0	1	0	F = A +B	Addition
0	0	1	1	F = A + B +1	Addition with carry
0	1	0	0	F = A - B - 1	Subtraction with Borrow
0	1	0	1	F = A – B	Subtraction
0	1	1	0	F = A – 1	Decrement A
1	0	0	0	F = A | B	OR
1	0	0	1	F = (A+B)'	NOR
1	0	1	0	F = A ⊕ B	XOR
1	0	1	1	F = (A⊕B)'	XNOR
1	1	0	0	F = A & B	AND
1	1	0	1	F = (A&B)'	NAND
1	1	1	0	F = A'	NOT A
1	1	1	1	F = A	Transfer A

Fig 4.3 Arithmetic and logical operations of Existing design reversible ALU

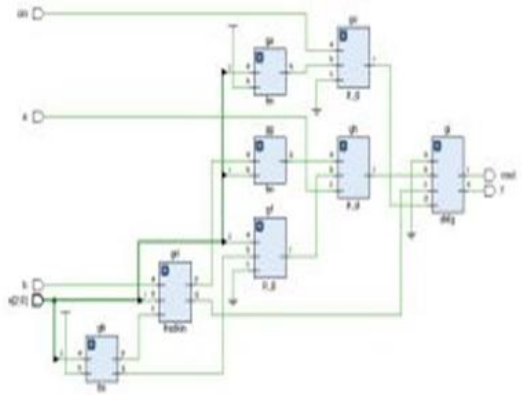

Fig 4.4 Schematic of 1 bit existing design ALU

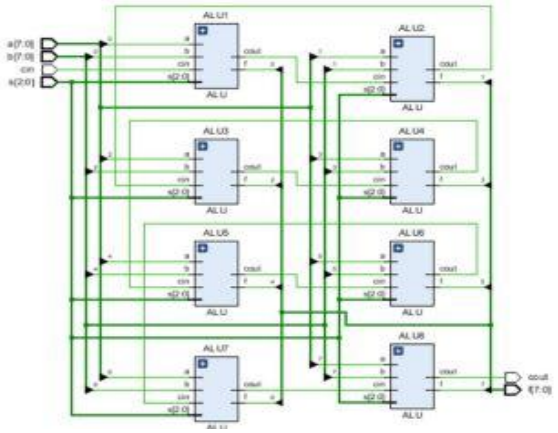

Fig 4.5 Schematic of 8 bit existing design ALU

2.Proposed design

In this design, the control unit consists of a Feynman gate, a 4 * 4 Toffoli gate, and two Fredkin gates. The HNG gate is used as a full adder unit. As shown in Figure 15, the outputs R and S of the HNG gate generate the output F and carry the Cout. The 1-bit ALU is shown in Figure 23. This design incurs a quantum cost of 22. It has 3 constant inputs and 7 garbage outputs. You can perform eight arithmetic operations and seven logical operations, as shown in Table 3. Arithmetic operations are selected by setting S2 to logical 0 and S2 to logical 1 and can perform logical operations.

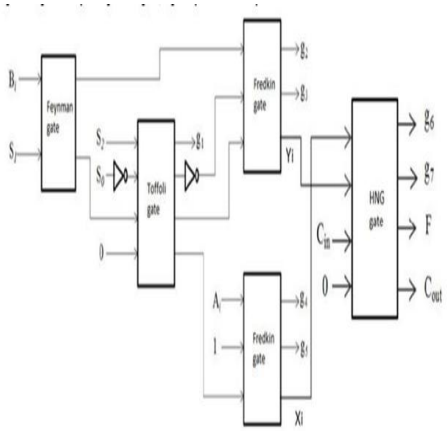

Fig 4.6 Circuit diagram of proposed design reversible ALU

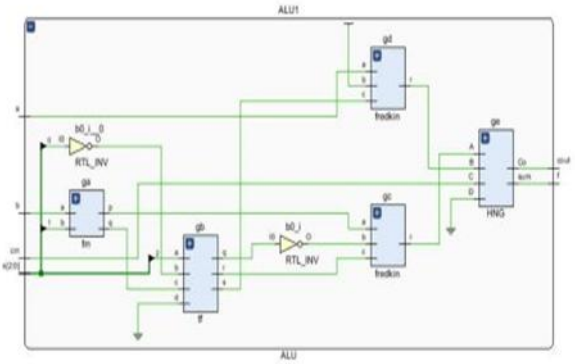

Fig 4.7 RTL Schematic of 1 bit existing design ALU

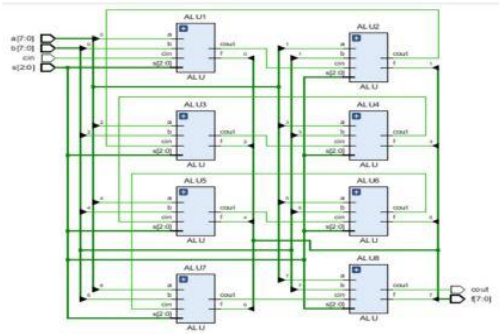

Fig 4.8 RTL Schematic of 8 bit existing design ALU

S2	S1	S0	Cin	Output	Function
0	0	0	0	F = A	Transfer A
0	0	0	1	F = A + 1	Increment A
0	0	1	0	F = A +B	Addition
0	0	1	1	F = A + B +1	Addition with carry
0	1	0	0	F = A - B - 1	Subtraction with Borrow
0	1	0	1	F = A – B	Subtraction
0	1	1	0	F = A – 1	Decrement A
0	1	1	1	F = A	Transfer A
1	0	0	0	F = A | B	OR
1	0	0	1	F = (A+B)'	NOR
1	0	1	0	F = A'	NOT A
1	0	1	1	F = A & B	AND
1	1	0	0	F = (A&B)'	NAND
1	1	0	1	F = A ⊕ B	EXOR
1	1	1	0	F = (A⊕B)'	EXNOR

Fig 4.9 Arithmetic and logical operations of the Proposed design of reversible ALU

V. Results and discussions

1.Existing design

	1 bit	8 bit	16 bit	32 bit	64 bit
Gate Count	8	64	128	256	512
Garbage Output	9	72	144	288	576
Quantum Cost	26	208	416	832	1664
Total on-chip Power (In W)	0.105	0.108	0.112	0.12	0.136
Static Power (in W)	0.104	0.104	0.104	0.105	0.105
Dynamic Power (in W)	0.001	0.004	0.008	0.016	0.031
Logic Power (in W)	<0.001	<0.001	<0.001	<0.001	<0.001
Delay (in ns)	1.56	6.685	12.967	25.518	50.062
Nets	16	77	149	293	581
Leaf Cells	10	57	113	225	449
LUTs	2	22	46	94	190
Bonded IOBs	8	29	53	101	197

Fig 5.1 Simulation results of different bit lengths of reversible of existing designs of ALU from Xilinx Vivado tools

Fig 5.2 Power analysis of a 32 bit ALU based on existing design

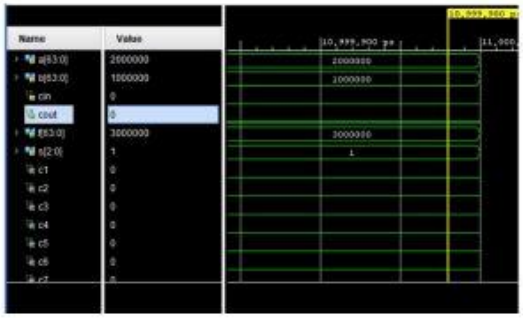

Fig 5.3 By setting S2=0, S1=0, S0=1 and Cin=0 addition operation is performed by 64 bit ALU on A and B

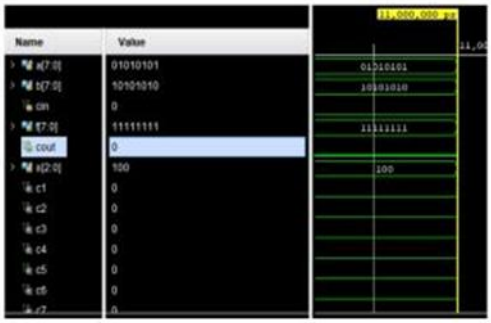

Fig 5.4 By setting S2=1, S1=0, S0=0 and Cin=0,OR operation on 64 bit inputs A and B is performed.

2.Proposed design

	1 bit	8 bit	16 bit	32 bit	64 bit
Gate Count	6	48	96	192	366
Garbage Output	6	48	96	192	366
Quantum Cost	22	176	352	704	1342
Total Onchip Power (in W)	0.105	0.107	0.109	0.114	0.122
Static Power (in W)	0.104	0.104	0.104	0.104	0.105
Dynamic Power (in W)	<0.001	0.002	0.004	0.009	0.017
Logic Power (in W)	<0.001	<0.001	<0.001	<0.001	<0.001
Delay (in ns)	1.197	6.2	11.858	22.133	45.197
Nets	16	83	162	299	594
Leaf Cells	10	63	126	231	462
LUTs	2	25	53	94	203
Bonded IOBs	8	29	53	101	197

Fig 5.5 The Xilinx Vivado tool was used to simulate alternative bit lengths of the proposed reversible ALU design.

Fig 5.6 Power consumption analysis of proposed design based 32 bit ALU

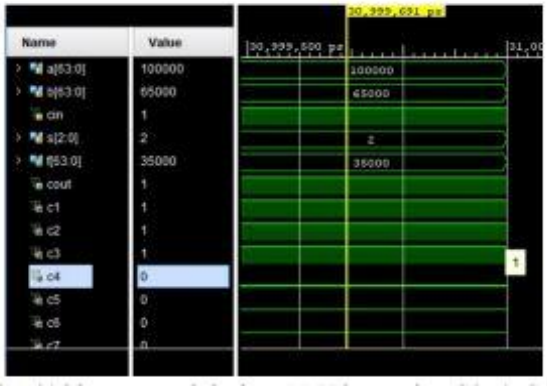

Fig 5.7 By setting S2=0, S1=1, S0=0 and Cin=1 subtraction by 64 bit proposed design ALU

Fig 5.8 By setting S2=1, S1=0, S0=1 and Cin=1 AND operation is carried out by 64 bit proposed design ALU

	Gate Count	Garbage outputs	Quantum cost
Existing design	512	576	1664
Proposed design	366	366	1342

Fig 5.9 Collation of gate count, number of garbage outputs and quantum cost of existing design and proposed design

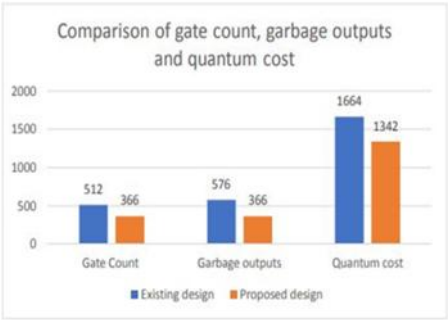

Fig 5.10 The table and graph above displays the comparison of the existing and proposed designs for the 64-bit reversible ALU. In comparison to the existing design, the proposed design requires fewer reversible logic gates, has fewer trash outputs, and incurs the lowest quantum cost.

	Dynamic power (in mW)	Total on-chip power (in mW)
Existing design	31	136
Proposed design	17	122

Fig 5.11 Collation of dynamic and total on-chip power of existing design and proposed design

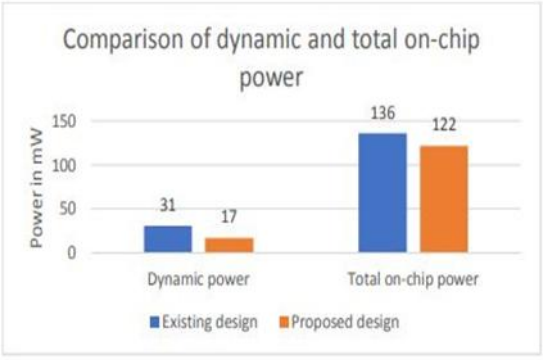

Fig 5.12 In terms of dynamic and total on-chip power consumed, the table compares the present and proposed designs of the 64 bit reversible ALU. It can be seen from tables that both static and logic power consumption of the designs remains fairly the same. It can be understood from the figure that proposed design has the least dynamic power consumption in comparison with the existing design, as it uses a fewer number of gates. It can also be delineated that the total on-chip power of the proposed design is comparatively less than the existing design.

	Nets	Leaf cells	LUTs
Existing design	581	449	190
Proposed design	594	462	203

Fig 5.13 Collation of numbers of LUTs, nets and leaf cells required by existing design and proposed design

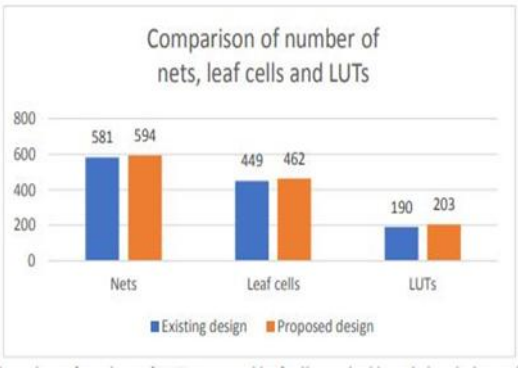

Fig 5.14 Table shows the collation between existing design and proposed design of the 64 bit reversible ALU in terms of the number of nets, leaf cells, and lookup tables required. It can be seen from Tables that existing design comparatively uses a lesser number of nets, leaf cells, and lookup tables. However, it is seen that the difference between the designs in the corresponding parameters is not so significant.

	1 bit	8 bit	64 bit
Existing design	1.56	6.875	50.062
Proposed design	1.197	6.2	45.197

Fig 5.15 Collation of delay incurred for 1,8,64 bit ALUs of existing design and proposed design

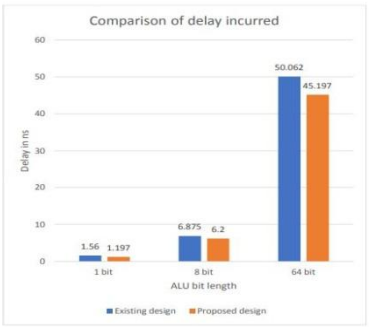

Fig 5.16 Collation chart of delay incurred for 1,8,64 bit ALUs of existing design and proposed design

The collation on delay between two designs of reversible ALU is made.In comparison to the proposed existing design, the proposed reversible ALU proposed design offers higher reductions in propagation delay.

VI. Conclusion

The use of a reversible control unit in an arithmetic, logical unit is proposed in this study. In terms of reversible gates, garbage outputs, quantum cost, quantum depth, constant inputs, logical and arithmetic functions, and hardware complexity, we compared these proposed architectures to current designs. Using the Xilinx Vivado tool, both designs were simulated, analysed, compared, and verified. The results of simulation show that the proposed reversible ALU performs better than the current design. The use of a reversible control unit in the arithmetic and logical unit has also shown to be a significant advance over previous designs. It features ten gates, eight garbage outputs, 29 quantum costs, and four constant inputs, all of which are significantly less than the present designA total of 16 arithmetic and logical operations are also available. In conclusion, the proposed reversible ALU design implementation in terms of number of gates, garbage outputs, and quantum cost can be used for low-power applications. With the help of proposed designs, we will be able to develop entire reversible computer architecture in the future. A future design of a fully reversible architecture using solely reversible logic units will include the reversible ALU as a central unit. More crucial pieces, such as a reversible control unit and a new approach to reversible memory, must be devised for a full architecture.

References

[1]. Nitin Krishna V, "Design and analysis of arithmetic logic units using reversible logic gates", International Journal for Research in Applied Science & Engineering Technology (IJRASET), ISSN: 2321-9653, Volume 8 Issue III Mar 2020.

[2]. Akanksha Dixit, Vinod Kapse, "Arithmetic and logic unit design using reversible logic design", International Journal of Engineering and Innovative Technology (IJEIT),Volume 1 Issue 6, June 2012.

[3]. Greeshma Arya,D S Chauhan, "Performance Analysis of Arithmetic Logic Unit with Reversible Logic", International Journal of Advanced Trends in Computer Science and Engineering,Volume 9,No.4 July-August 2020.

[4]. Lekshmi Viswanath, Ponni.M, "Design and Analysis of 16 Bit Reversible ALU", IOSR Journal of Computer Engineering (IOSRJCE) ISSN : 2278-0661 Volume 1, Issue 1 (May-June 2012), PP 46-53.

[5]. Vandana Shukla,O P Singh,G R Mishra,R K Tiwari, "Reversible Realization of N-bit Arithmetic circuit for low power loss ALU applications,6th International Conference on Smart computing and Communications,ICSCC 2017,7-8 December 2017.

[6]. Monika Rangari,Prof. Richa Saraswat, Dr. Rita Jain, "Design of Reversible Logic ALU using Reversible logic gates with Low Delay Profile", International Journal of Advanced Research in Computer and Communication Engineering ,Vol. 4, Issue 4, April 2015.