JETIR.ORG ISSN: 2349-5162 | ESTD Year : 2014 | Monthly Issue
JOURNAL OF EMERGING TECHNOLOGIES AND INNOVATIVE RESEARCH (JETIR)
An International Scholarly Open Access, Peer-reviewed, Refereed Journal

Low Power and High Speed Reversible Arithmetic Logic Unit based on Reversible Gate

Shubham Kumar Saurabh [1], Prof. Satyarth Tiwari[2], Prof. Suresh. S. Gawande[3]
M. Tech. Scholar, Department of Electronics and Communication, Bhabha Engineering Research Institute, Bhopal[1]
Guide, Department of Electronics and Communication, Bhabha Engineering Research Institute, Bhopal[2]
Co-guide, Department of Electronics and Communication, Bhabha Engineering Research Institute, Bhopal[3]

Abstract: **Reversible computing spans computational models that are both forward and backward deterministic. These models have applications in program inversion and bidirectional computing, and are also interesting as a study of theoretical properties. A reversible computation does, thus, not have to use energy, though this is impossible to avoid in practice, due to the way computers are build. It is, however, not always obvious how to implement reversible computing systems. The restriction to avoid information loss imposes new design criteria that need to be incorporated into the design; criteria that do not follow directly from conventional models.**
In this paper, investigate garbage-free reversible central processing unit computing systems to physical gate-level implementation. Arithmetic operations are a basis for many computing systems, so a proposed the design of adder, sub tractor, multiplexer, encoder and work towards a reversible circuit for general circuit are important new circuits. In all design implemented Xilinx software and simulated VHDL text bench.

Index Terms **– Filter Coefficient, Finite Impulse Response, Pass-band Frequency, Narrow Band Filter**

I. INTRODUCTION

Quantum Computation and Quantum Information is the study of the information processing tasks that can be accomplished using Quantum mechanical systems. Quantum mechanics is a mathematical framework or set of rules for the construction of physical theories. Quantum computation taught us to think physically about computation, and this approach yields many new and exciting capabilities for information processing and communication. In the broadest terms, any physical theory, not just Quantum mechanics, may be used as the basis for a theory of information processing and communication. One of the messages of Quantum computation and information is that new tools are available for those problems that are relatively more difficult or impossible to solve on Classical computers. Quantum computing believes that what is computable and what is not computable is limited by the Laws of physics [1]. Traditional computer science is based on Boolean logic and algorithms. Its basic variable is a bit with two possible values, 0 or 1. These values are represented in the computer as stable saturated states off or on. Quantum mechanics offer a new set of rules that go beyond this classical paragraph [2]. The basic variable is now a qubit, represented as a vector in a two dimensional complex Hilbert space. '0' and '1' form a basis in this space. The logic that can be implemented with such qubits is quite distinct from Boolean logic, and this is what has made Quantum computing exciting by opening new possibilities. A related historical strand contributing to the development of quantum computation and quantum information is the interest, dating to the 1970s, of obtaining complete control over single Quantum systems [3]. Applications of Quantum mechanics prior to the 1970s typically involved a gross level of control over a bulk sample containing an enormous 4 number of Quantum mechanical systems, none of them directly accessible. Since the 1970s many techniques for controlling single Quantum systems have been developed. For example, methods have been developed for trapping a single atom in an 'atom trap', isolating it from the rest of the world and allow us to probe many different aspects of its behavior with incredible precision [4]. Quantum computation and Quantum information provide a useful series of challenges at varied levels of difficulty for people devising methods to better manipulate single quantum systems, and simulate the development of new experimental techniques. The ability to control single quantum systems is essential if we are to harness the power of quantum mechanics for applications to Quantum computation and Quantum information. Despite this intense interest, efforts to build Quantum information processing systems have resulted in modest success to date. Small Quantum computers, capable of doing dozens of operations on a few qubits represent the state of the art in Quantum computation. However, it remains a great challenge to physicists and engineers to develop techniques for making large-scale Quantum information processing a reality [5].

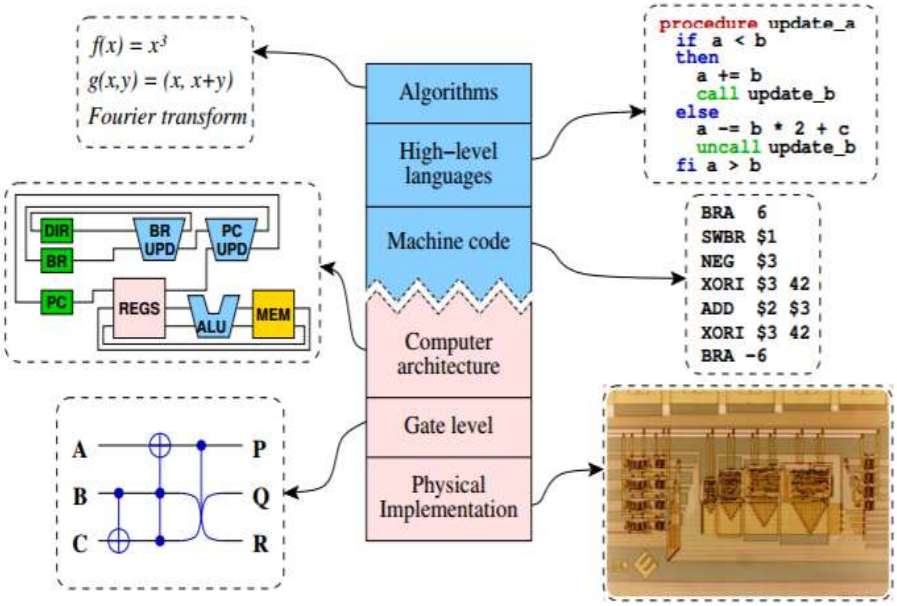

Figure 1: Tower of reversible computing system

In his 1961 paper, Landauer wrote that "we shall label a machine as being logical reversible, if and only if all its individual steps are logically reversible" [6]. This is a very grand challenge and we know from Bennett (and later work) that, theoretically, such machines do exist – even when we add the requirement that the final result must not include garbage. But is it possible to realize such machines in practice and can it is done with the fabrication technology that exists today? And will we actually be able to achieve the expected reduced heat dissipation? The Micro Power research project [7], which started in 2009, has as objective to develop a proof-of-concept reversible computing system and the computer science theory behind it. To do this all parts of the reversible computation tower (Figure 1.1) must be investigated. More specifically, the project investigates whether reversible computing can be applied in a power-limited application (specifically hearing aids) with the future hope to either reduce power consumption or increase functionality [8].

II. ARITHMETIC LOGIC UNIT

Arithmetic operations lie at the foundation of most computing systems and good logical implementations of these are important. Improvements to arithmetic circuits can result in improvements to the entire computing system. In a garbage-free reversible computing system it is especially important that the arithmetic circuits are also garbage-free, but how to do this is not always obvious, and history shows that rethinking our current knowledge can be necessary.

The adder that Feynman proposed [9] was a reversible embedding of the ripple-carry adder. Though addition is an injective function if one of the inputs is kept, the conventional ripple-carry structure is not reversible. The problem lies in the use of the full-adder circuit, because it is not possible to calculate both the sum and the carry without copying one of the inputs. You can say that there is an overlap in the information contained in the two results and these results in a garbage bit. Several reversible adder designs used this ripple structure; e.g. reversible binary-coded decimal adders have received some interest.

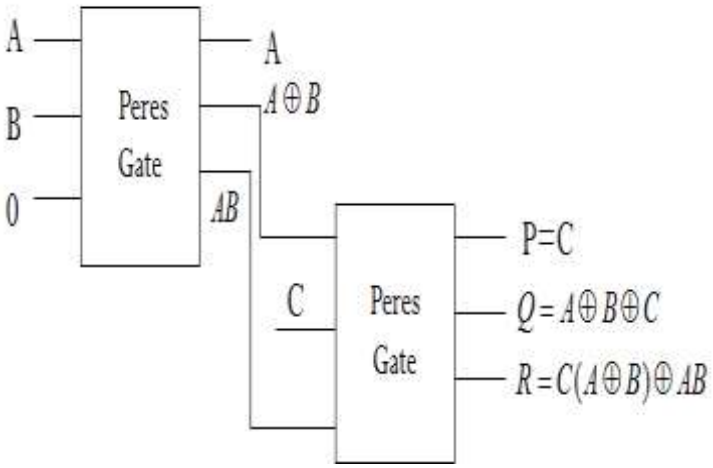

Figure 2: Block Diagram of 1-bit Full Adder

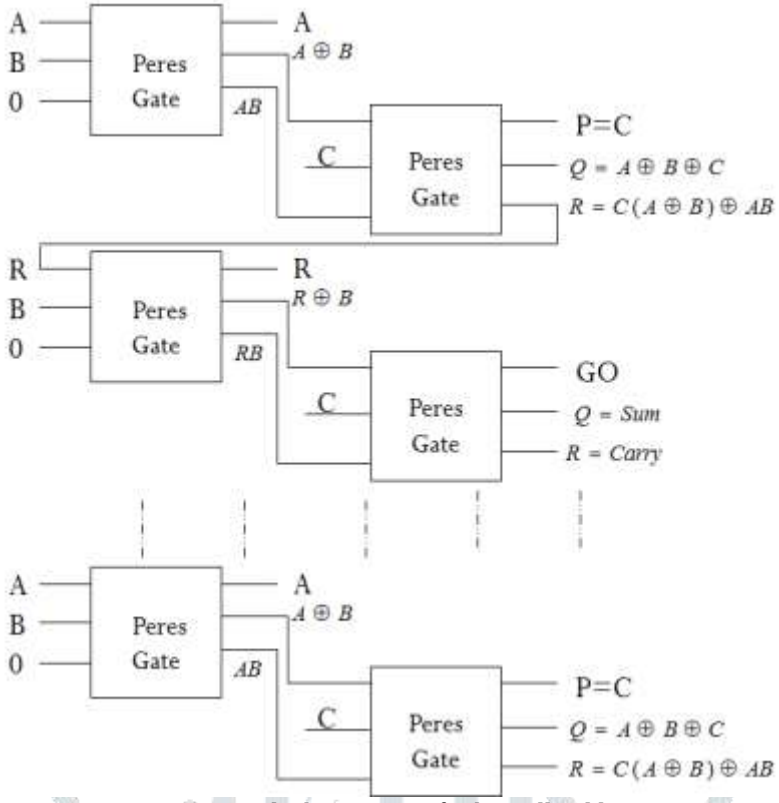

Figure 3: Block Diagram of n-bit Full Adder

Addition has an intuitive reversible formulation using modular arithmetic and reversible updates. Multiplication, on the other hand, is more difficult to define; mainly because the inverse operation is division. A simple way to define it is to take the embedding from Bennett and save both the multiplicand and the multiplier while still producing the product. This is the trick used by Kawada et al. in their (garbage-free) reversible logic implementation of the Karatsuba algorithm.

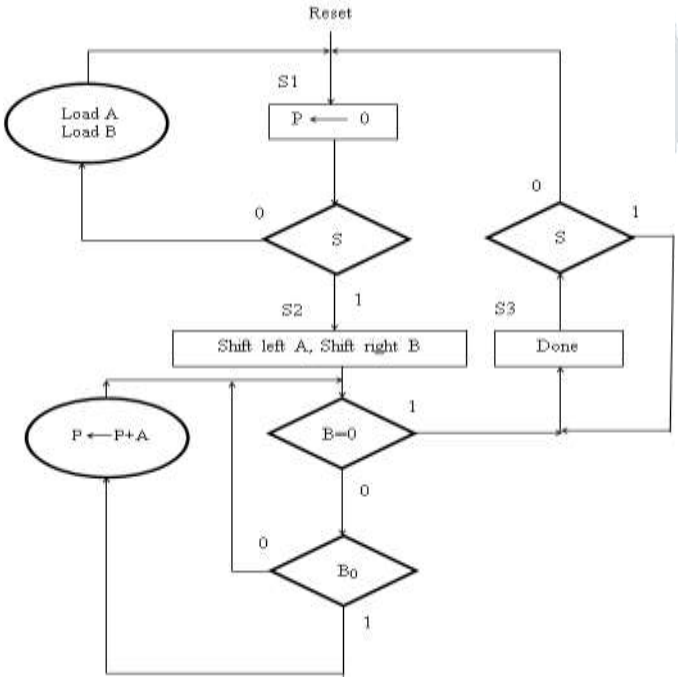

Figure 4: ASM Flow Chart of Multiplier

Multimedia transforms are an interesting application area for reversible circuits. In small battery-powered devices (e.g. smartphones and mp3-players) they are often included as part of an ASIC to reduce power consumption and a key property of many such transforms is that they are information-lossless (and thus invertible). There exist many application areas of such transforms and even the earliest quantum algorithms make use of a quantum implementation of the Fourier transform. Also a implementation of the fast Fourier transform in reversible logic has been investigated.

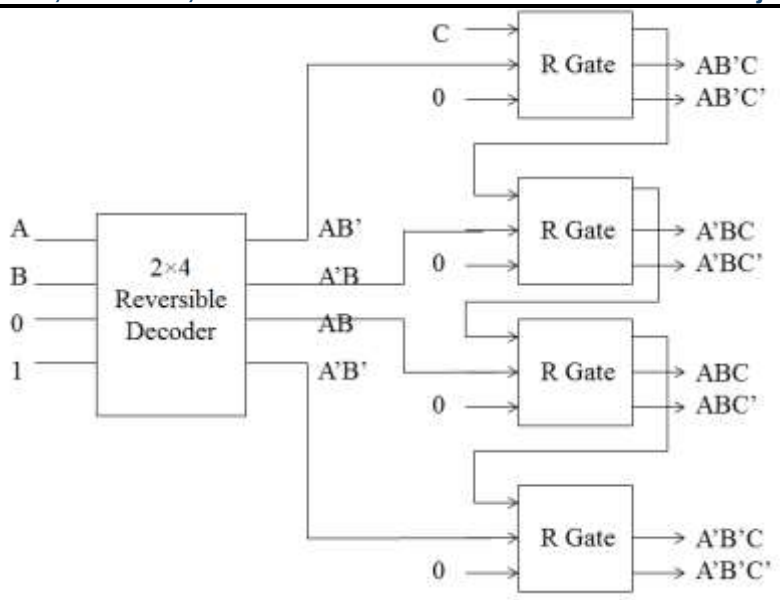

Figure 5: Proposed Diagram of 3×8 Decoder

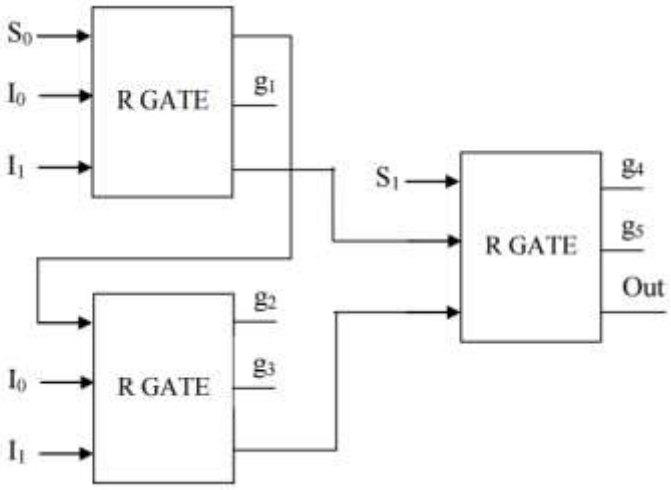

Figure 6: Proposed Diagram of 4×1 Multiplexer

The first approach to reversible logic synthesis is actually a very beautiful example of how mathematics can be related to reversible circuits. Based on work by Rayner and Newman, Storme and De Vos used that reversible gates and cascading of these by serial composition forms a group, with the result that it is possible to use the known methods from group theory. Specifically, one of these methods can be used to decompose an arbitrary reversible circuit into a cascade of simpler circuits that only updates one input wire.

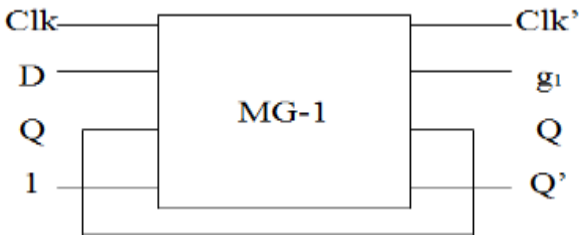

Figure 7: Proposed 1-bit Memory Store Element

Reversible logic synthesis have since been much researched and often with an interest to also apply the methods to quantum circuits. Many of the approaches is based on techniques known from Boolean logic synthesis. There exists many more, but common to these are that they are based on truth tables as input and that they actually try to solve an NP-hard problem.

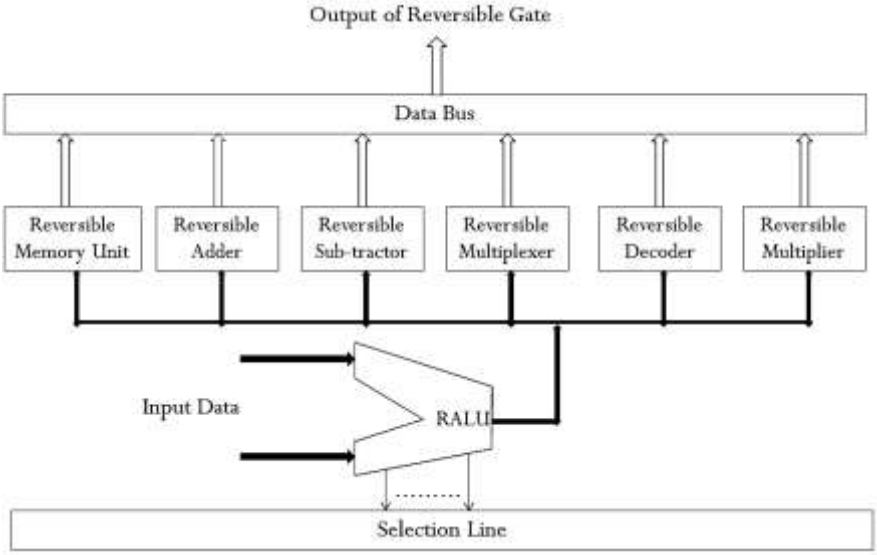

Figure 8: Flow Chart of Arithmetic Logic Unit

According to base paper, the use different technique used for central processing unit (CPU) using arithmetic logic unit (ALU) and different types of reversible logic gate. In base paper generally concentrate on the number of slice (area), number of look up table (LUT) and maximum combination path delay (MCPD) of the central processing unit. In this dissertation proposed central processing unit based TR gate and used to multiplier, adder, sub tractor, multiplexer and encoder. So step by step explain proposed design is below:

- Layout the data bus to handle all of the operations of the reversible ALU.
- Design the reversible realizations of the flip-flops.
- Design the reversible memory circuits (such as buffer registers and counter circuits) using the proposed reversible flip-flops of the previous step.
- Design the arithmetic circuits such as adder, multiplier, divider, comparator etc.
- Design the reversible realization of ALU.
- Design the reversible control unit of the processor by designing an efficient instruction decoder.

III. SIMULATION RESULTS

More specifically, we have developed new garbage-free circuits for addition and are working towards a general multiplication circuit. We have also combined multiple operations together to implement a reversible arithmetic logic unit. With these and other garbage-free arithmetic circuits it is possible to design larger reversible computing systems. As an example, we have implemented discrete lossless transforms by redesigning these with a lifting scheme. We have also shown the design of a reversible computing architecture and implemented this using only reversible logic gates. While, these are still small systems, with further development it should be possible to use similar strategies to implement even larger systems.

Reversible Gate Parameter:-

Gate Count (GC): The number of gates used to realize reversible circuit

Garbage Outputs (GO): The number of unused outputs in a reversible logic. The inputs regenerated at the outputs are not garbage outputs.

Ancilla Inputs (AI): The number of input kept constant at either 0 or 1.

Delay : It corresponds to number of primitive quantum gates in the critical path of the circuit.

Quantum Cost:- Quantum cost is defined, as the number of basic quantum gates like controlled-NOT, Controlled V+, Controlled V and NOT gate.

Table 1: Comparative Study of Parameters for Reversible Components

Reversible Designs		Quantum Cost	Garbage Output	Constant Input	Delay	Gate Count	Total Cost	Power
Reversible Multiplexer	Existing	15	5	0	10	21	41	2135.4
	Proposed	12	5	0	8	12	29	1707.6
Reversible Multiplier	Existing	341	61	46	50	45	447	31724.3
	Proposed	237	46	46	55	45	328	33725.1
Reversible Adder/ Subtractor	Existing	36	9	5	32	36	81	5122.8
	Proposed	30	10	5	24	30	70	4269.0
Reversible Comparator	Existing	26	15	10	10	34	75	1992.2
	Proposed	10	2	3	10	10	22	1565.3

Reversible ALU	Existing	418	90	61	102	136	644	19352.8
	Proposed	289	63	54	97	97	449	13803.1

From our own design experience, we know that designing logic gate-level circuits quickly becomes complicated when the functionality and number of wires involved are increased. To make the design process easier, we have developed two hardware description languages. Using examples from known reversible circuits, we have shown that circuits can be described reasonably concisely. These are, however, still small examples and we need to implement a larger system to show the usefulness of the languages.

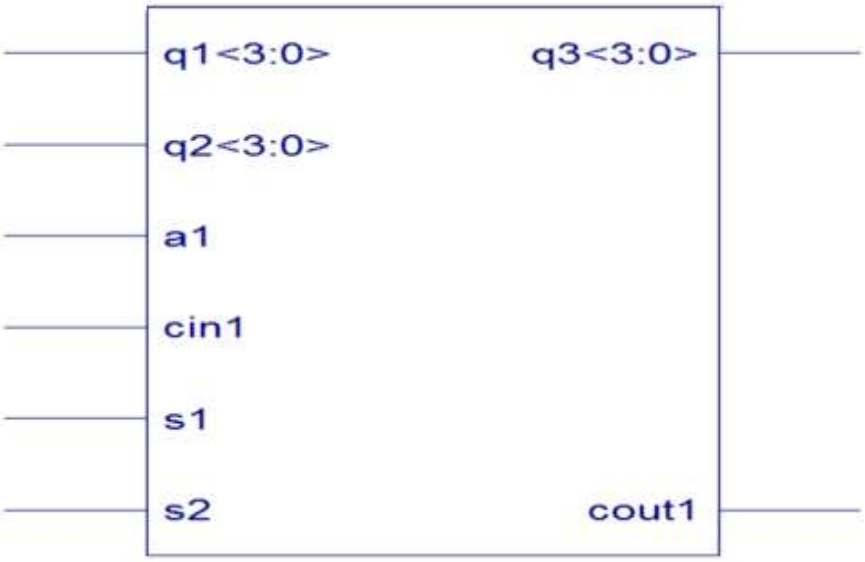

Figure 9: Technology Schematic of the 4-bit Reversible ALU

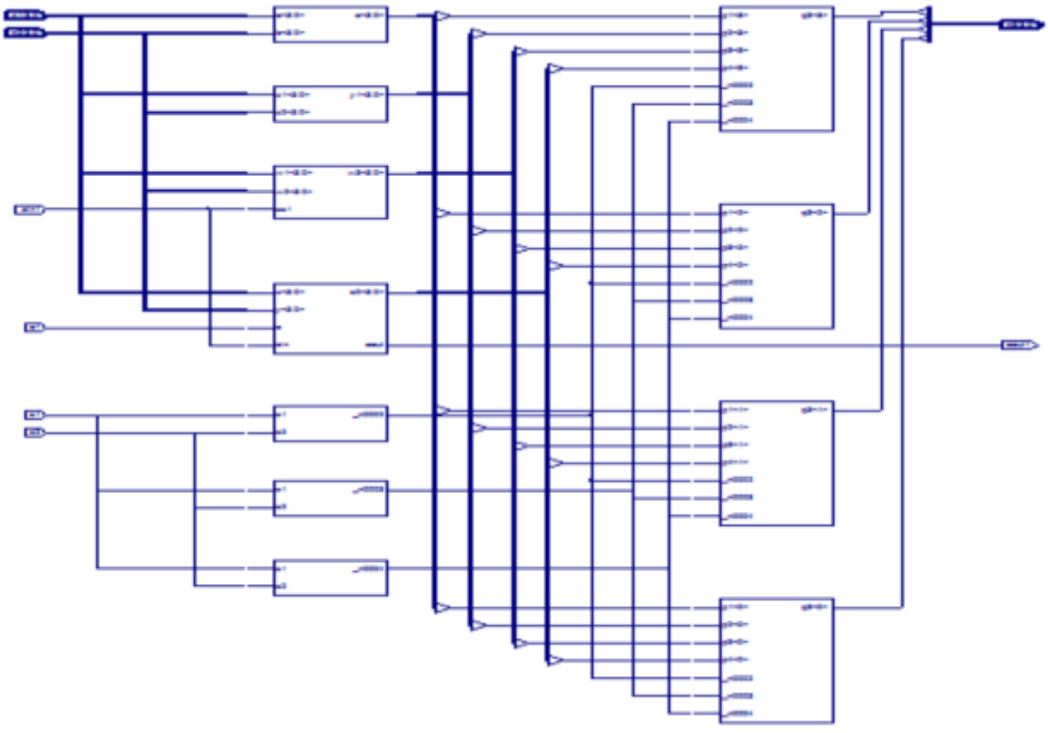

Figure 10: Register transfer level (RTL) using 4-bit RALU

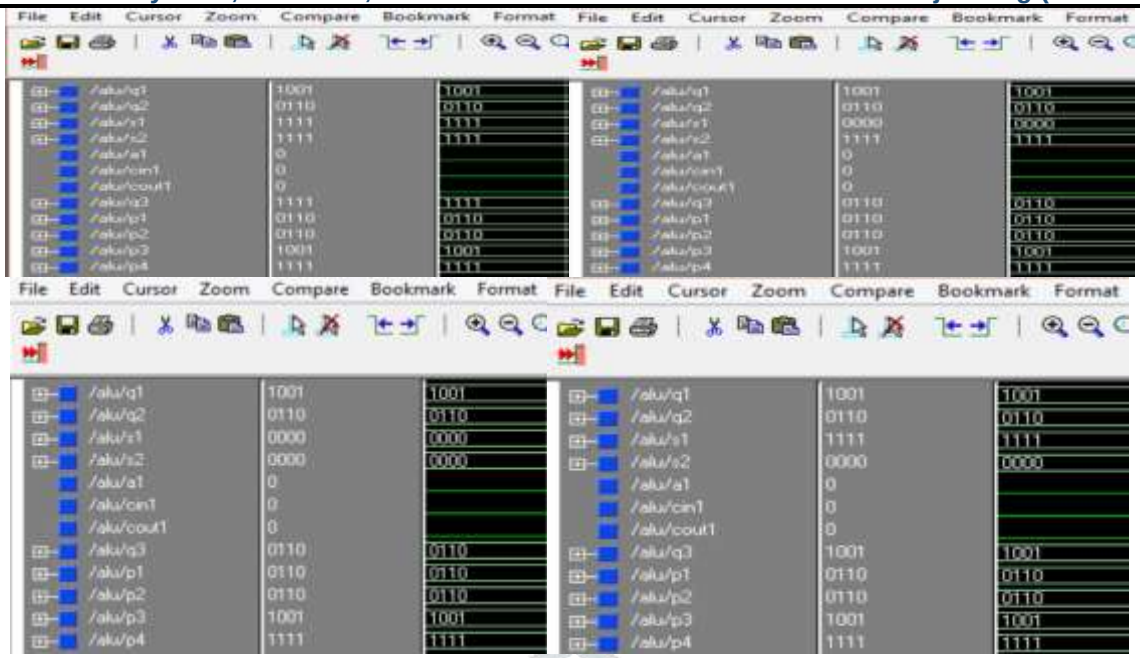

Figure 11: Output Waveform of 4-bit RALU

Table II: Comparative Results of Existing Algorithm and Proposed Algorithm in 4-bit

Parameter	4-bit Reversible Full Adder [1]	Reversible FA using Peres Gate	4-bit Reversible Full Sub tractor [1]	Reversible FS using TR Gate
AI	12	4	12	4
GO	8	8	8	8
QC	72	32	80	48
Delay	66	48	66	40
GC	24	20	32	28

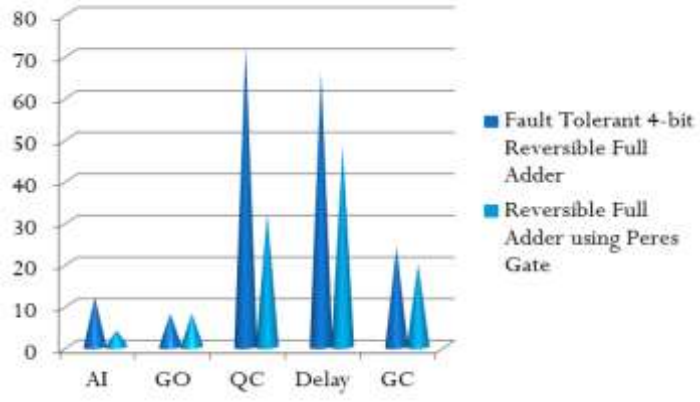

Figure 12: Bar Graph of the Previous and Proposed Reversible Full Adder

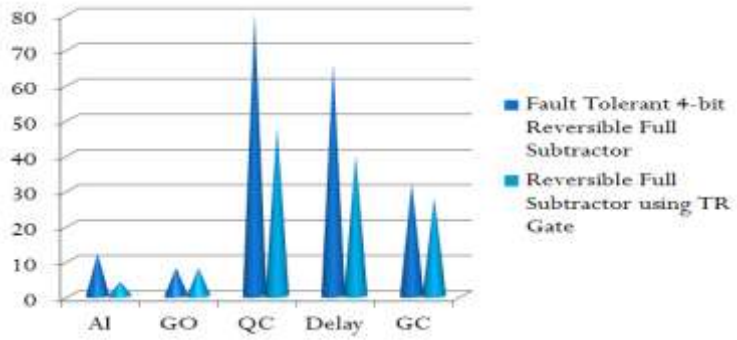

Figure 13: Bar Graph of the Previous and Proposed Reversible Full Adder

Table III: Comparative Results of Existing Algorithm and Proposed Algorithm in n-bit

parameter	Reversible 5-bit Sign Multiplier using DKG Gate	Reversible n-bit Sign Multiplier using DKG Gate
AI	45	9n
GO	41	8n+1
QC	237	47n+2
Delay	74	14n+4
GC	222	44n+2

IV. CONCLUSION

For conventional logic circuits there exists much research, even whole books, dedicated to the design and implementation of computer arithmetic. This is definitely not the case for reversible logic. The constraint that the circuits must be garbage-free is what makes it an interesting research problem, but most proposed designs (both hand-made and CAD generated) still implement the conventional algorithms with garbage. They use the reversible gates, but as their sole goal is to reduce logic size or number of garbage bits for a specific fixed-size circuit, very little knowledge is actually gained from this approach.

REFERENCES

[1] S. Nagaraj, B.Vamsi Krishna, Botta Chakradhar and Debanjan Sarkar, “Comparison of 32-bit ALU for Reversible Logic and Irreversible Logic”, Innovations in Power and Advanced Computing Technologies (i-PACT), IEEE 2021.

[2] Behrouz Safaiezadeh Ebrahim Mahdipour Majid Haghparast Samira Sayedsalehi and Mehdi Hosseinzadeh "Novel design and simulation of reversible ALU in quantum dot cellular automata" The Journal of Supercomputing 2021.

[3] Hari M. Gaur Ashutosh K. Singh and Umesh Ghanekar "Design Reversible Arithmetic Logic Unit with Built0in Testability" IEEE Design & Test 2019.

[4] Parth Khatter Neeta Pandey and Kirti Gupta "An Arithmetic and Logical Unit using Reversible Gates" 2018 International Conference on Computing Power and Communication Technologies (GUCON) 2018 September.

[5] Parth Khatter, Neeta Pandey and Kirti Gupta, “An Arithmetic and Logical Unit using Reversible Gates”, International Conference on Computing, Power and Communication Technologies (GUCON), IEEE 2018.

[6] S. M. Swamynathan and V. Banumathi "Design & analysis of FPGA based 32 bit ALU using Reversible gates" 2017 IEEE International Conference on Electrical Instrumentation and Communication Engineering (ICEICE) pp. 1-4 2017.

[7] Kamaraj and P. Marichamy "Design and implementation of arithmetic and logic unit (ALU) using novel reversible gates in quantum cellular automata" 2017 4th International Conference on Advanced Computing and Communication Systems (ICACCS -2017) pp. 1-8 2017.

[8] Gopi Chand Naguboina and K. Anusudha, “Design and Synthesis of Combinational Circuits Using Reversible Decoder In Xilinx”, IEEE International Conference on Computer, Communication, and Signal Processing (ICCCSP-2017).

[9] Hatkar A. P., Hatkar A. A. and Narkhede N. P., “ASIC Design of Reversible Multiplier Circuit”, International Conference on Electronic Systems, Signal Processing and Computing Technologies, pp. 01-05, 2014 IEEE.

[10] Matthew Morrison and Nagarajan Ranganathan, “Design of a Reversible ALU based on Novel Programmable Reversible Logic Gate Structures”, IEEE Computer Society Annual Symposium on VLSI, pp. 01-06, 2013 IEEE.

[11] Mr. Abhishek Gupta, Mr. Utsav Malviya and Prof. Vinod Kapse, “Design of Speed, Energy and Power Efficient Reversible Logic Based Vedic ALU for Digital Processors”, IEEE Computer Society Annual Symposium on VLSI, vol. 12, Issue 05, pp. 3456-3463, 2012.

[12] H. Thapliyal and N. Ranganathan, "Design of Efficient Reversible Binary Subtractors Based oNew Reversible Gate," Processing of the Computer Society Annual Symposium on VLSI, pp. 01-06, 2009.

[13] Matthew Morrison and Nagarajan Ranganathan, “Design of a Reversible ALU based on Novel Programmable Reversible Logic Gate Structures”, 2013 IEEE Computer Society Annual Symposium on VLSI.

[14] Lekshmi Viswanath and Ponni. M, “Design and Analysis of 16 Bit Reversible ALU”, ISSN: 2278-0661 Volume 1, Issue 1 (May-June 2012), PP 46-53

[15] Akanksha Dixit and Vinod Kapse, “Arithmetic & Logic Unit (ALU) Design using Reversible Control Unit”, International Journal of Engineering and Innovative Technology (IJEIT) Volume 1, Issue 6, June 2012.

[16] Mr. Abhishek Gupta, Mr. Utsav Malviya and Prof. Vinod Kapse, “Design of Speed, Energy and Power Efficient Reversible Logic Based Vedic ALU for Digital Processors”, 2012 IEEE Computer Society Annual Symposium on VLSI.

[17] H. Thapliyal and N. Ranganathan, "A New Reversible Design of BCD Adder," To appear in Proc. Design Automation and Test in Europe (DATE), 2011.

[18] H. Thapliyal and N. Ranganathan, "Design of Reversible Sequential Circuits Optimizing Quantum Cost, Delay, and Garbage Outputs," ACM Journal on Emerging Technologies in Computing Systems, 2010.