Single Bit Alu Using Reversible Logic Gates

Dasari Mahesh Kumar[1], Anas Ali Shafe[2], Mannala Divya[3],Kandi Srivyshnavi[4],Puli Sumanth[5],Gulla Rajkumar[6]

[1]Electronics & communication, Jyothishmathi institute of technology and science
[2]Electronics & communication, Jyothishmathi institute of technology and science
[3]Electronics & communication, Jyothishmathi institute of technology and science
[4]Electronics & communication, Jyothishmathi institute of technology and science
[5]Electronics & communication, Jyothishmathi institute of technology and science
[6]Electronics & communication , Jyothishmathi institute of technology and science

---***---

Abstract— In this digital world, technology depends on the operations of A.L.U to decide the system performance. The need for an Arithmetic Logic Unit (ALU) is as important as the computer, simply because ALU forms the fundamental part of any Central Processing Unit (CPU). And so the encryption of an ALU is highly mandatory for the safety of the device as there are hardly any device without an ALU. This paper deals with the design of an single-bit ALU using a hardware description language, HDL that is structurally modelled. The results are verified and synthesized through Xilinx.

Keywords- ALU, FPGA SPARTAN6, REVERSIBLE LOGIC, QUANTUM COST, FEYNMAN GATE.

I. INTRODUCTION

The present scenario of spectacular fusion of chip size reduction and increase in number of circuits on chips has given a tremendous growth in battery operated and power sensitive applications thus leading to the growth in the emerging field of Low Power Electronics. In our paper we are indulged in Static Power reduction at the Architectural level as in near future this area of power is going to rule the total amount of dissipated power in the SOCs (System On Chip). We have proposed of synthesizing the POWER GATING TECHNIQUE in specific the Fine-grained method in order to optimize the static power being dissipated. In this approach the inputs to the gates are blocked by using NMOS when not in use thus resulting in reduction of unnecessary utilization of input leading to significant amount of power reduction. Thus our whole paper revolves around the concept of reduction of static power at Architectural level starting with 1 bit and extending till 8 bit with corresponding decrease in the power consumption.

A processor is a main part of any digital system. And an ALU is one of the main components of a microprocessor. To give a simple analogy, CPU works as a brain to any system & and ALU works as a brain to CPU. So it’s a brain of computer’s brain. They consists of fast dynamic logic circuits and have carefully optimized structures. Of total power consumption in any processor, CPU accounts a significant portion of it. ALU also contribute to one of the highest power-density locations on the processor, as it is clocked at the highest speed and is busy mostly all the time which results in thermal hotspots and sharp temperature gradients within the execution core. Therefore, this motivate us strongly for a energy-efficient ALU designs that satisfy the high-performance requirements, while reducing peak and average power dissipation.

Arithmetic Unit

Employing fast and efficient adders in arithmetic logic unit will aid in the design of low power high performance system. Other operations such as subtraction and multiplication also employ addition in their operations, and their internal hardware is almost similar though not identical to addition hardware. Various adder families have been proposed in the past to trade-off power, area and speed for possible use in ALUs. The performance criticality of the ALU demands a dynamic adder implementation. Dynamic logic family of adders are the most efficient in terms of transistor-count, speed and power dissipation. This work covers the design of 3-bit adder using Complementary logic. This same Adder unit is used for the implementation of subtractor unit. This reuses the current hardware we made for adder and saves area.

Logic Unit

ALU can perform various logic operations like NOT, AND, OR, NAND, NOR XOR, XNOR etc. For these operations a special unit is made called as Logical Unit. This Logic Unit performs all logic operations asked to perform. A MUX operated by select lines, for which particular logic operation to perform, is used inside this logic block.

comparator & Shifter unit

A 1-bit comparator is made with the help of Complementary logic. It compare the two I/Ps and give three states of O/P for three different conditions. Also a 2:1 MUX is made which is used as a cell in the barrel shifter. Used for shifting and rotating operation

II.REVERSIBLE LOGIC GATES

A reversible logic gate is an n-input n-output logic device with one-to-one mapping. This helps to determine the outputs from the inputs and the inputs can be uniquely recovered from the outputs. Also in the synthesis of reversible circuits direct fan-Out is not allowed as one–to-many concept is not reversible. However, fan-out in reversible circuits is achieved using additional gates. A reversible circuit should be designed using minimum number of reversible logic gates. From the point of view of reversible circuit design, there are many parameters for determining the complexity and performance of circuits.

- The number of Reversible gates (N): The number of reversible gates used in circuit.
- The number of constant inputs (CI): This refers to the number of inputs that are to be maintained constant at either 0 or 1 in order to synthesize the given logical function.
- The number of garbage outputs (GO): This refers to the number of unused outputs present in a reversible logic.
- Circuit. One cannot avoid the garbage outputs as these are very essential to achieve reversibility.
- Quantum cost (QC): This refers to the cost of the circuit in terms of the cost of a primitive gate. It is calculated knowing the number of primitive reversible logic gates (1*1 or 2*2) required to realize the circuit.

BASIC REVERSIBLE LOGIC GATES

01. Feynman Gate

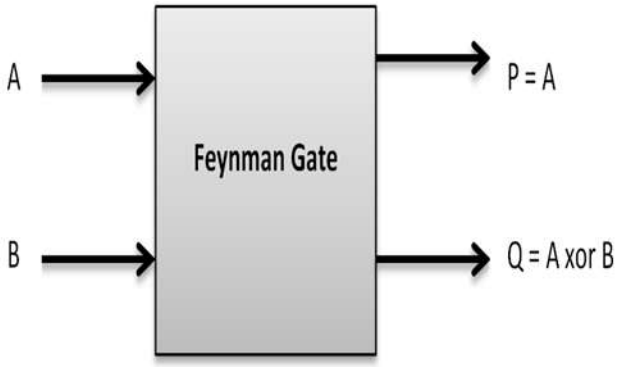
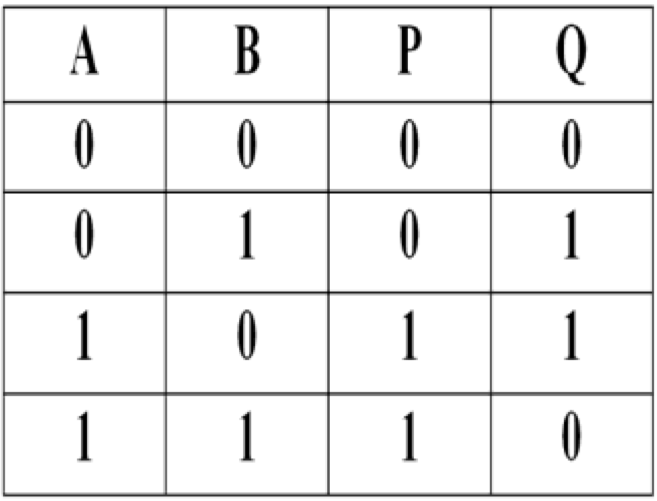


Fig 1: Feynman gate

Feynman gate is a 2*2 one through reversible gate. The input vector is I (A, B) and the output vector is O(P, Q). The outputs are defined by P=A, Q=A XOR B. Quantum cost of a Feynman gate is 1. Feynman Gate (FG)can be used as a copying gate. Since a fan-out is not allowed in reversible logic, this gate is useful for duplication of the required outputs.



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

Table 1: Truth table

02. Double Feynman Gate (F2G)

A 3*3 Double Feynman gate. The input vector is I (A, B, C) and the output vector is O (P, Q, R). The outputs are defined by P = A, Q=A XOR B, R=A XOR C. Quantum cost of double Feynman gate is 2.

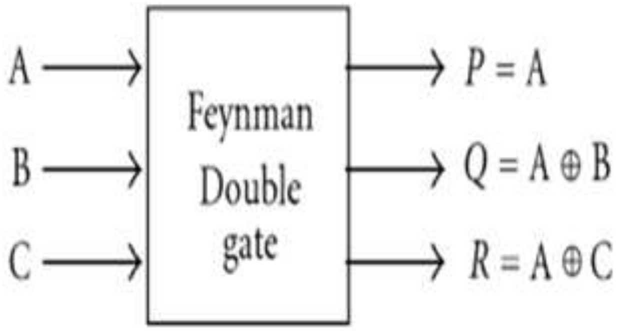


Fig 2: Double Feynman gate

03.Toffoli Gate:

A 3*3 Toffoli gate. The input vector is I (A, B, C) and the output vector is O(P, Q,R). The outputs are defined by P=A, Q=B, R=AB XOR C. Quantum cost of a Toffoli gate is 5.

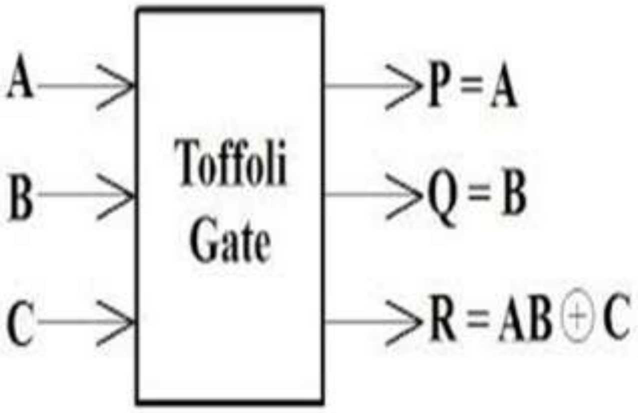
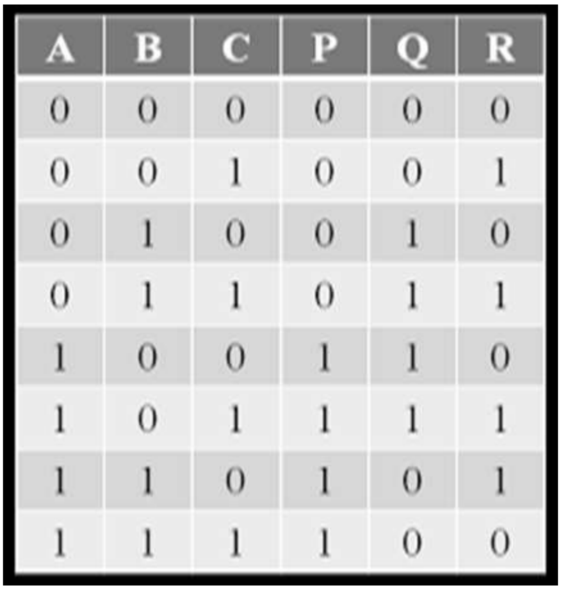


Fig 3: Toffoli gate



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	1	0
1	0	1	1	1	1
1	1	0	1	0	1
1	1	1	1	0	0

Table 2: Truth table of Toffoli gate

04. Fredkin Gate

A 3*3 Fredkin gate. The input vector is I (A, B, C) and the output vector is O (P, Q, R). The output is defined by P=A, Q=A′B XOR AC and R=A′C XOR AB. Quantum cost of a Fredkin gate is 5.

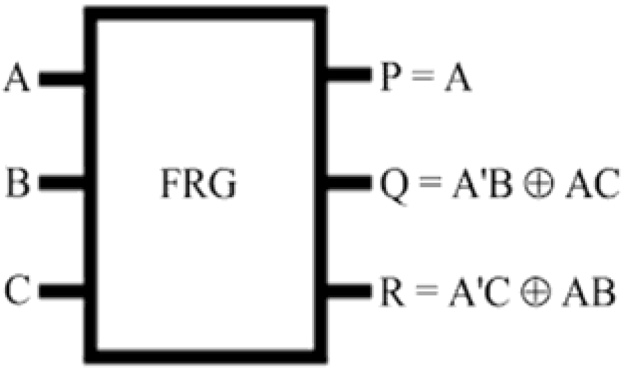


Fig 4(a): Fredkin gate

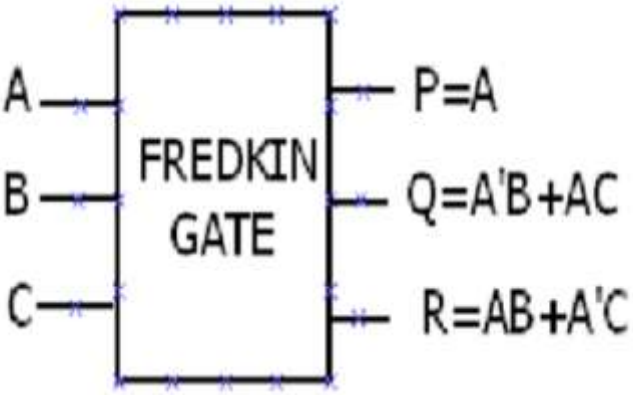
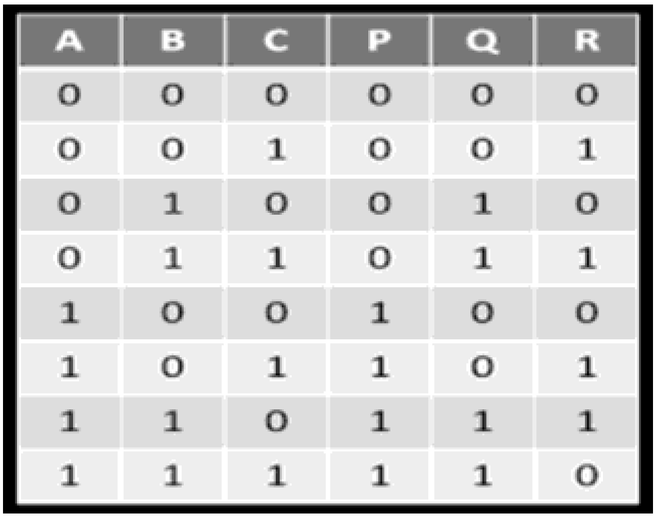


Fig 4(b): Fault tolerant Fredkin gate



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

Table 3: Truth table of Fredkin gate

05. Peres Gate

A 3*3 Peres gate. The input vector is I (A, B, C) and the output vector is O (P, Q, R). The output is defined by P = A, Q = A XOR B and R=AB XOR C. Quantum cost of a Peres gate is 4. In the proposed design Peres gate is used because of its lowest quantum cost.

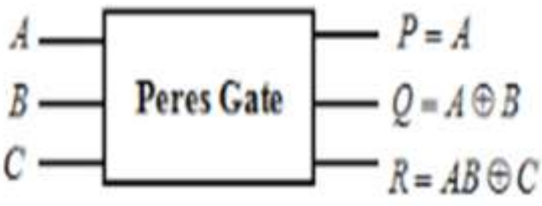
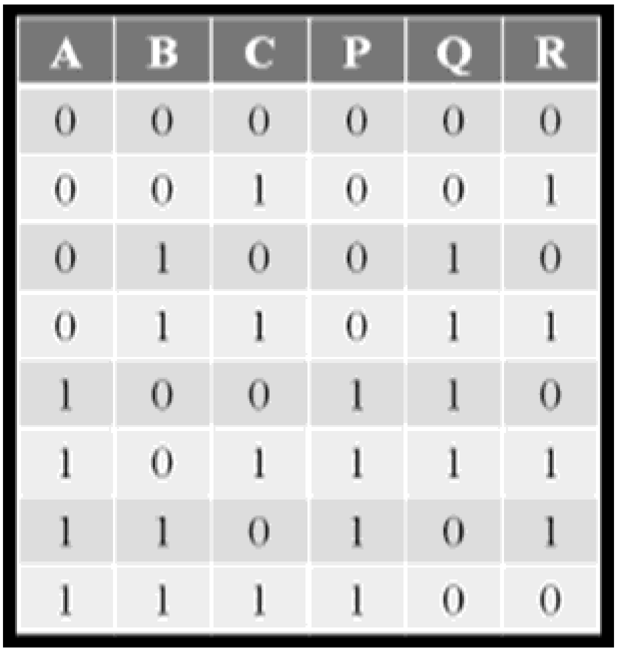


Fig 5: Peres Gate



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	1	0
1	0	1	1	1	1
1	1	0	1	0	1
1	1	1	1	0	0

Table 4: Truth table of Peres gate

06. Sayem gate

SG is a 1 trough 4x4 reversible gate. The input and output vector of this gate are, Iv = (A, B, C, D) and Ov = (A, A'B XOR AC, A'B XOR AC XOR D, AB XOR A'C XOR D). The block diagram of this gate.

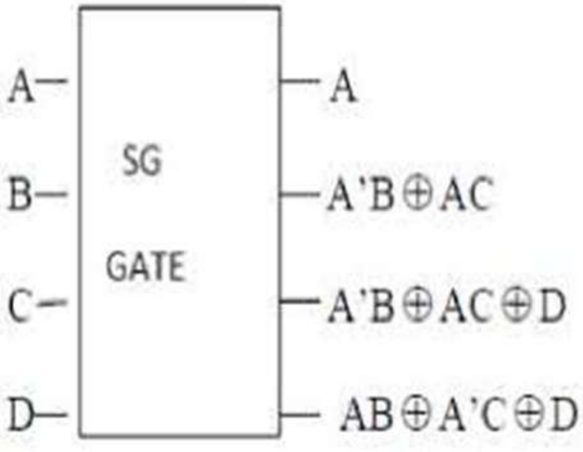


Fig 6: Sayem gate

APPLICATIONS

Reversible computing may have applications in computer security and transaction processing, but the main long-term benefit will be felt very well in those areas which require high energy efficiency, speed and performance. It includes the area like

1. Low power CMOS
2. Quantum computer
3. Nanotechnology
4. Optical computing

5. Design of low power arithmetic and data path for digital signal processing (DSP)
6. Field Programmable Gate Arrays (FPGAs) in CMOS technology for extremely low power, high testability and self-repair

We have presented an approach to the realize the multipurpose binary reversible gates. Such gates can be used in regular circuits realizing Boolean functions. In the same way it is possible to construct multiple-valued reversible gates having similar properties. The proposed asynchronous designs have the applications in digital circuits like a Timer/Counter, building reversible ALU, reversible processor etc. This work forms an important move in building large and complex reversible sequential circuits.

III. PROPOSED METHODOLOGY

Parallel adder is the basic part of arithmetic part of ALU. A parallel adder is built with full adders and 1 bit and 4-bit reversible ALU's. The reversible ALU uses DKG gates as full adder. Cin enters the full adder as carry input in its least important bit position and exists out as Cout . The output of the full adder is taken as sum. Since we are using arithmetic operation the arithmetic addition is realised when one set of inputs enters through A inputs and other set enters through B inputs and input carry is kept 0. When Cin =1 it is possible to add 1 to the sum in Functions, if we complement all the bits of B then we will get F= (A+B). And when CIN= 1 we will get F= (A-B). Similarly if all inputs of B is 0, we get the transfer A function. The proposed 1 and 4-bit ALUs is shown in figure 2.6. and logical outputs based on inputs opcodes. Proposed 1-Bit ALU Opcodes

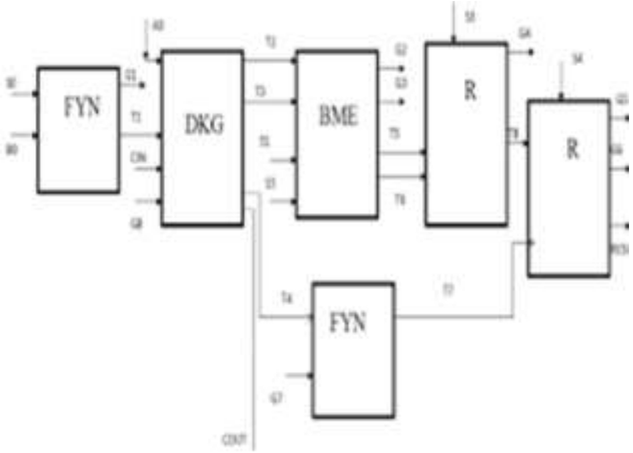


Fig 7: Reversible 1-bit ALU

The presented ALUs has two most important advantages. First, it produces less delay than the existing 1-Bit logic architecture. As a result, the proposed ALU has a better quantum delay for n-bit ALUs. Second, proposed ALU produces more arithmetic and logical operations. The proposed ALU is designed to be flexible so that any alterations required for implementation in an instruction set architecture will be easy.

IV. SIMULATION RESULTS

The simulation is the process which is termed as the final verification in respect to its working where as the schematic is the verification of the connections and blocks. The simulation window is launched as shifting from implantation to the simulation on the home screen of the tool, and the simulation window confines the output in the form of the wave forms. Here it has the flexibility of providing the different radix number systems.

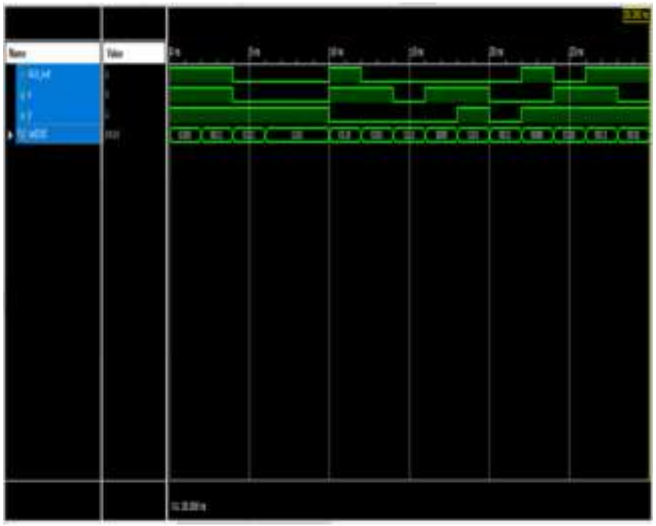


Fig 8: Simulated Waveforms of single bit ALU

The simulation is the process which is termed as the final verification in respect to its working whereas the schematic is the verification of the connections and blocks. The simulation window is launched as shifting from implantation to the simulation on the home screen of the tool ,and the simulation window confines the output in the form of the wave forms. Here it has the flexibility of providing the different radix number systems.

V. CONCLUSION

This paper has detailed the discussion on Reversible logic gates and the three parameters which are ancillary inputs, garbage outputs, and quantum cost . Existing reversible gates are discussed along with parity preserved Fredkin peers and CNOT gate. The ALU was designed and implemented using Verilog HDL to verify its functionality in Xilinx 14.7. The performance was analyzed and improved efficiency was obtained in all parameters considered. This design can be further used to design multiple bit ALUs with parallel computation.

VI. REFERENCES

[1] R. Landauer, “Irreversibility and heat generation in the computing process.” IBM Journal of Research and Development., vol. 5, no. 3, pp. 183–191, 1961.

[2] Parhami, Behrooz. "Fault-tolerant reversible circuits." In 2006 Fortieth Asilomar Conference on Signals, Systems and Computers, pp. 1726-1729. IEEE, 2006.

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

[4] Gaur, Hari M., Ashutosh K. Singh, and Umesh Ghanekar. "Design of Reversible Arithmetic Logic Unit with Built-in Testability." IEEE Design & Test (2019).

[5] Kole, Dipak Kumar, Jhuma Dutta, Arpita Kundu, Suravi Chatterjee, Suravi Agarwal, and Tanushri Kisku. "Generalized construction of quantum multiplexers and de-multiplexers using a proposed novel algorithm based on universal Fredkin gate." In 2016 Sixth International Symposium on Embedded Computing and System Design (ISED), pp. 82-86. IEEE, 2016.

[6] Khatter, Parth, Neeta Pandey, and Kirti Gupta. "An Arithmetic and Logical Unit using Reversible Gates." In 2018 International Conference on Computing, Power and Communication Technologies (GUCON), pp. 476-480. IEEE, 2018.

[7] R. Landauer, “Irreversibility and heat generation in the computing process.” IBM Journal of Research and Development., vol. 5, no. 3, pp. 183–191, 1961.

[8] C. H. Bennett, “Logical reversibility of computation.” IBM Journal of Research and Development., vol. 17, no. 6, pp. 525–532, 1973.

[9] Kamaraj, A., 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). IEEE, 2017.

[10] Kadar, M. Mohaideen Abdul, and A. V. Ananthalakshmi. "An energy efficient Montgomery modular multiplier for security systems using reversible gates." 2015 International Conference on Communications and Signal Processing (ICCSP). IEEE, 2015.

[11] Molahosseini, Amir Sabbagh, et al. "Towards efficient modular adders based on reversible circuits." 2018 IEEE International Symposium on Circuits and Systems (ISCAS). IEEE, 2018.

[12] Gassoumi, Ismail, Lamjed Touil, and Bouraoui Ouni. "Division circuit using reversible logic gates." 2018 International Conference on Advanced Systems and Electric Technologies (IC_ASET). IEEE, 2018.