iJRASET
International Journal For Research in Applied Science and Engineering Technology

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Volume: 8 **Issue:** II **Month of publication:** February 2020

DOI: http://doi.org/10.22214/ijraset.2020.2085

www.ijraset.com

Call: 08813907089 | E-mail ID: ijraset@gmail.com

International Journal for Research in Applied Science & Engineering Technology (IJRASET)
ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.177
Volume 8 Issue II Feb 2020- Available at www.ijraset.com

Truth Table Analysis of Logic Circuits using Reversible Gates

Soham Bhattacharya[1], Sourav Goswami[2]
[1, 2]*Electronics and Communication Engineering Department, Heritage Institute of Technology, Kolkata.*

Abstract: This paper determines the instance of the AND, OR, NAND, NOR and basic arithmetic functions, which can be used to build complex combinational digital logic circuits through Truth Table Analysis. This paper also illustrates a Boolean Expression using those truth table of the basic Universal Gates existence by Reversible Gates.
Keywords: Reversible logic, Logic circuits, Quantum cost, Universal Gates, Reversible Computing.

I. INTRODUCTION

Rolf Landauer first proposed a principle concerning about the lower theoretical limit of the energy consumption of the computation physically. This physical principle is termed as Landauer's Principle.
According to Landauer[1,2], the amount of energy dissipated for every irreversible bit operation is at least KTln2 joules, where K=1.3806505*10-23m2kg-2K-1 (joule/Kelvin-1) is the Boltzmann's constant and T is the absolute temperature at which operation is performed.. In 1973, Bennett[3] showed that energy dissipation problem is avoided with circuits built using reversible logic gates. Reversible circuits are essentially dissimilar from conventionally irreversible ones. In reversible logic, no information is lost. Application of reversible logic has received attention in the recent years for reducing the power dissipation with low power VLSI design, and it has a vast impact in low power CMOS, quantum computation and nanotechnology.
A circuit is reversible if the input is recoverable from the output. Reversible computing supports both forward and backward movement process as one generates inputs from the outputs [4].

II. REVERSIBLE LOGIC

Reversibility in computing implies that the information about computational states can never be lost and be used when needed. There are two types of Reversibility: 1. Logical and 2.Physical.
Logical reversibility is defined as the process in which any early stage can be recovered in computing backwards or un-computing the results. Physical reversibility refers to no heat dissipation. Logical reversibility is achieved after Physical Reversibility [5]. The generation of heat is gained when the voltage levels changes from low/high to high/low such that zero/one to one/zero bit change in digital logic of computers. Most of the energy is required in making the changes. As a result, reversible computing shall affect digital logic design.
'Reversible Computation' is defined as a model of computation where computational process at some extent, is reversible. It means that it can reserve the data as long as it requires and when needed. Properties of reversible logic are like it can recover the state of inputs from the outputs, it follows one to one mapping i.e. when 'n' number of inputs are taken, and one can get 'n' number of outputs from the gates and for each input pattern, there should be unique output pattern. Finally, the circuit obtained will be acyclic, i.e. feedback will be there but Fan-out will not be more than one.

III. BASIC REVERSIBLE GATES

There are some basic reversible gates used to design complex digital combinational circuits. They are:
NOT GATE, TOFFOLI GATE, FEYNMANN GATE, DOUBLE FEYNMANN GATE, PERES GATE, FREDKIN GATE, HNG GATE and TSG GATES.
The most economical implementation of basic gates and universal gates are shown in terms of truth table analysis. Quantum networks are required to be built from reversible logic components[6,7].
The block diagram of 1X1 NOT Gate is represented by Fig. 1(A).The quantum cost for 1X1 NOT Gate from Fig. 1(B) is 0. Total logical calculation is T=1δ. Here, δ represents a two input XOR gate calculation.

ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.177
Volume 8 Issue II Feb 2020- Available at www.ijraset.com

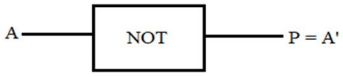


Fig. 1(A)

Fig. 1(B)

INPUT	OUTPUT
0	1
1	0

Fig. 1(C)

Fig. 1: (A) shows block diagram of 1X1 NOT gate. (B) shows its quantum representation. (C) shows its truth table.

The block diagram of 3X3 Toffoli Gate is represented by Fig. 2(A). The quantum cost of 3X3 Toffoli gate is 5 from Fig. 2(B). Total logical calculation is T = 1δ + 1β.[8]

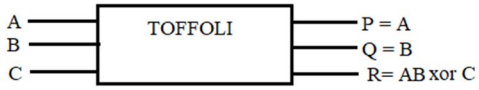


Fig. 2(A)

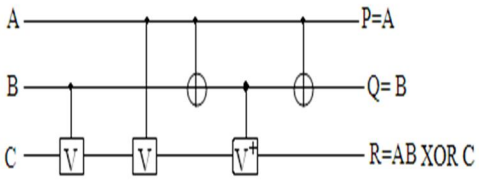


Fig. 2(B)

INPUT			OUTPUT		
A	B	C	P = A	Q = B	R = AB XOR C
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	0	1
1	1	1	1	1	0

Fig. 2(C)

Fig.2: (A) shows block diagram of 3X3 Toffoli gate. (B) shows its quantum representation. (C) shows its truth table.

The block diagram of 2X2 Feynmann gate is represented by Fig. 3(A). The quantum cost of 2X2 Feynmann gate is 1 from Fig. 3(B). Total logical calculation is T = 1α[9]. Here, α represents a NOT calculation.

ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.177
Volume 8 Issue II Feb 2020- Available at www.ijraset.com

Fig. 3(A)

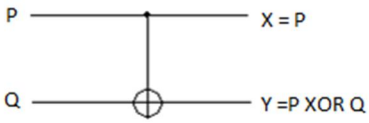


Fig. 3(B)

INPUT		OUTPUT	
P	Q	X = P	Y = P XOR Q
0	0	0	0
0	1	0	1
1	0	1	1
1	1	1	0

Fig. 3(C)

Fig.3: (A) shows block diagram of 2X2 Feynmann gate. (B) shows its quantum representation. (C) shows its truth table.

The block diagram of 3X3 Double Feynmann gate is represented by Fig. 4(A). The quantum cost is 2 from Fig. 4(B). Total Logical Calculation is T = 2α.

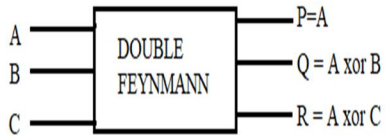


Fig. 4(A)

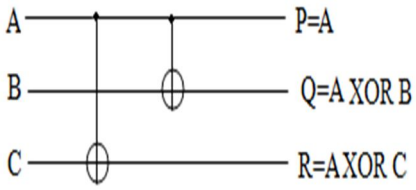


Fig. 4(B)

INPUT			OUTPUT		
A	B	C	P = A	Q = A XOR B	R = A XOR C
0	0	0	0	0	0
0	1	0	0	0	1
1	0	0	1	1	0
1	1	0	1	1	1
0	0	1	0	1	1
0	1	1	0	1	0
1	0	1	1	0	1
1	1	1	1	0	0

Fig. 4(C)

Fig.4: (A) shows block diagram of 3X3 Double Feynmann gate. (B) shows its quantum representation. (C) shows its truth table.

The block diagram of 3X3 Peres gate is represented by Fig. 5(A). The quantum cost is 4 from Fig. 5(B). Total logical calculation is T = 2α + 1β[10]. Here, α represents a NOT calculation and β represents a two input AND gate calculation.

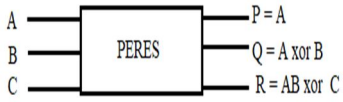


Fig. 5(A)

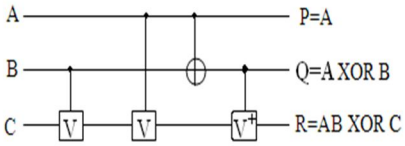


Fig. 5(B)

INPUT			OUTPUT		
A	B	C	P = A	Q = A XOR B	R = AB XOR C
0	0	0	0	0	0
0	1	0	0	0	0
1	0	0	1	1	0
1	1	0	1	1	1
0	0	1	0	1	1
0	1	1	0	1	1
1	0	1	1	0	1
1	1	1	1	0	0

Fig. 5(C)

Fig.5: (A) shows block diagram of 3X3 Peres gate. (B) shows its quantum representation. (C) shows its truth table.

The block diagram of 4X4 HNG gate is represented by Fig. 6(A). The quantum cost of 4X4 HNG gate is 6 from Fig. 6(B). Total logical calculation is T = 4α + 2β.[11]

Fig. 6(A)

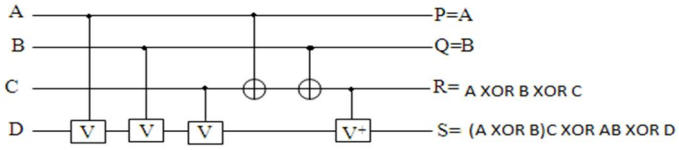


Fig. 6(B)

INPUT				OUTPUT			
A	B	C	D	P = A	Q = B	R = A XOR B XOR C	S = (A XOR B)C XOR AB XOR D
0	0	0	0	0	0	0	0
0	1	0	0	0	1	1	0
1	0	0	0	1	0	1	0
1	1	0	0	1	1	0	1
0	0	0	1	0	0	0	1
0	1	0	1	0	1	1	1
1	0	0	1	1	0	1	1
1	1	0	1	1	1	0	0
0	0	1	0	0	0	1	0
0	1	1	0	0	1	0	1
1	0	1	0	1	0	0	1
1	1	1	0	1	1	1	1
0	0	1	1	0	0	1	1
0	1	1	1	0	1	0	0
1	0	1	1	1	0	0	0
1	1	1	1	1	1	1	0

Fig. 6(C)

Fig.6: (A) shows block diagram of 4X4 HNG gate. (B) shows its quantum representation. (C) shows its truth table.

Every Boolean Function can be build using 3X3 Fredkin gate. The block diagram of 3X3 Fredkin gate is represented by Fig. 7(A). The quantum cost of Fredkin gate is 5 from Fig. 7(B). Total logical calculation is T = 2α+4β+1δ[12,13].

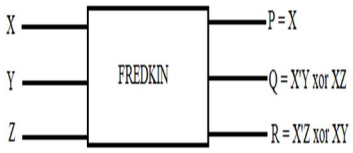


Fig. 7(A)

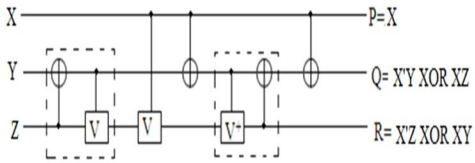


Fig. 7(B)

INPUT			OUTPUT		
X	Y	Z	P = X	Q = X'Y XOR XZ	R = X'Z XOR XY
0	0	0	0	0	0
0	1	0	0	1	0
1	0	0	1	0	0
1	1	0	1	0	1
0	0	1	0	0	1
0	1	1	0	1	1
1	0	1	1	1	0
1	1	1	1	1	1

Fig. 7(C)

Fig.7: (A) shows block diagram of 3X3 Fredkin gate. (B) shows its quantum representation. (C) shows its truth table.

The block diagram of 4X4 TSG gate is represented by Fig. 8(A). The quantum cost of 4X4 TSG gate is 17 from Fig.8(B). Total logical calculation is T= 4α+3β +3δ[14].

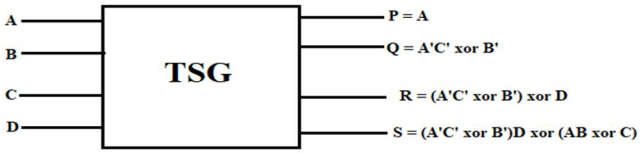


Fig. 8(A)

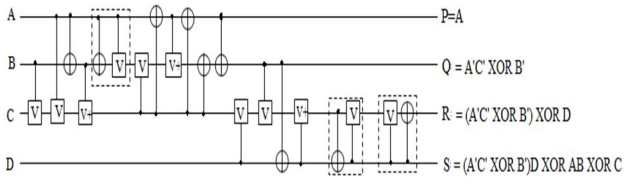


Fig. 8(B)

INPUT				OUTPUT			
A	B	C	D	P = A	Q = A'C' XOR B'	R = (A'C' XOR B') XOR D	S = (A'C' XOR B')D XOR (AB XOR C)
0	0	0	0	0	0	0	0
0	1	0	0	0	1	1	0
1	0	0	0	1	1	1	0
1	1	0	0	1	0	0	1
0	0	0	1	0	0	1	0
0	1	0	1	0	1	0	1
1	0	0	1	1	1	0	1
1	1	0	1	1	0	1	1
0	0	1	0	0	1	1	1
0	1	1	0	0	0	0	1
1	0	1	0	1	1	1	1
1	1	1	0	1	0	0	0
0	0	1	1	0	1	0	0
0	1	1	1	0	0	1	1
1	0	1	1	1	1	0	0
1	1	1	1	1	0	1	0

Fig. 8(C)

Fig.8: (A) shows block diagram of 4X4 TSG gate. (B) shows its Quantum representation. (C) shows its truth table.

ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.177
Volume 8 Issue II Feb 2020- Available at www.ijraset.com

IV. ANALYSIS OF TRUTH TABLES OF BASIC UNIVERSAL GATES USING REVERSIBLE GATES

Reversible are circuits (gates) that have one-to-one mapping between vectors of inputs and outputs; thus the vector of input states can be always reconstructed from the vector of output states.

A. And Gate

From the truth table in Fig. 9, the characteristics of two input AND gate operation is done, if C =0 and we will get the output from Port 'R', using Toffoli gate.

INPUT			OUTPUT		
A	B	C	P = A	Q = B	R = AB XOR C
0	0	0	0	0	0
0	1	0	0	1	0
1	0	0	1	0	0
1	1	0	1	1	1

Fig. 9: Implement a two input AND gate with inputs A & B and output R, using 3X3 Toffoli gate when C is grounded.

From the truth table in Fig. 10, characteristics of two input AND gate operation is done, if C = 0 and we will get the output from Port 'R', using Peres gate.

INPUT			OUTPUT		
A	B	C	P = A	Q = A XOR B	R = AB XOR C
0	0	0	0	0	0
0	1	0	0	0	0
1	0	0	1	1	0
1	1	0	1	1	1

Fig. 10: Implement a two input AND gate with inputs A & B and output R, using 3X3 Peres gate when C is grounded.

From the truth table in Fig.11, we will get the characteristics of two input AND gate operation when both C and D are grounded, and then, we will get the output through port 'S', using HNG gate.

INPUT				OUTPUT			
A	B	C	D	P = A	Q = B	R = A XOR B XOR C	S = (A XOR B)C XOR AB XOR D
0	0	0	0	0	0	0	0
0	1	0	0	0	1	1	0
1	0	0	0	1	0	1	0
1	1	0	0	1	1	0	1

Fig. 11: Implement a two input AND gate with inputs A & B and output R, using 4X4 HNG gate when C and D are grounded.

From the truth table in Fig.12, we will get the characteristics of two input AND gate output through Port 'R' when 'Z' is grounded, using Fredkin gate.

INPUT			OUTPUT		
X	Y	Z	P = X	Q = X'Y XOR XZ	R = X'Z XOR XY
0	0	0	0	0	0
0	1	0	0	1	0
1	0	0	1	0	0
1	1	0	1	0	1

Fig.12: Implement a two input AND Gate with inputs X & Y and output R, using 3X3 Fredkin gate when Z is grounded.

From the truth table in Fig.13, we will get the characteristics of two input AND gate output when C and D are taken as '0' and we will get the output through Port 'S', using TSG gate.

INPUT				OUTPUT			
A	B	C	D	P = A	Q = A'C' XOR B'	R = (A'C' XOR B') XOR D	S = (A'C' XOR B')D XOR (AB XOR C)
0	0	0	0	0	0	0	0
0	1	0	0	0	1	1	0
1	0	0	0	1	1	1	0
1	1	0	0	1	0	0	1

Fig. 13: Implement a two input AND gate with inputs A & B and output S, using 4X4 TSG gate when C and D are grounded.

B. Or Gate

From the truth table in Fig.14, characteristics of two input OR gate operation is done through Port 'Q' and when Z = 1, using Fredkin gate.

INPUT			OUTPUT	
X	Y	Z	P = X	Q = X'Y XOR XZ
0	0	1	0	0
0	1	1	0	1
1	0	1	1	1
1	1	1	1	1

Fig. 14: Implement a two input OR gate with inputs X & Y and output Q, using 3X3 Fredkin gate when Z is 1.

From the truth table in Fig. 15, characteristics of two input OR gate operation is done through Port S, when C = 1 and D is grounded, using HNG gate.

INPUT				OUTPUT			
A	B	C	D	P = A	Q = B	R = A XOR B XOR C	S = (A XOR B)C XOR AB XOR D
0	0	1	0	0	0	1	0
0	1	1	0	0	1	0	1
1	0	1	0	1	0	0	1
1	1	1	0	1	1	1	1

Fig. 15: A two input OR gate with inputs A & B and output S, using 4X4 HNG gate when C is 1 and D is grounded.

From the truth table in Fig. 16, characteristics of two input OR gate operation is done through Port S, when C is grounded and D = 1, using TSG gate.

INPUT				OUTPUT			
A	B	C	D	P = A	Q = A'C' XOR B'	R = (A'C' XOR B') XOR D	S = (A'C' XOR B')D XOR(AB XOR C)
0	0	0	1	0	0	1	0
0	1	0	1	0	1	0	1
1	0	0	1	1	1	0	1
1	1	0	1	1	0	1	1

Fig. 16: Implement a two input OR gate with inputs A & B and output S, using 4X4 TSG gate when C is grounded and D is 1.

C. Nand Gate

From the truth table in Fig. 17, characteristics of two input NAND gate operation is done through Port R, when C = 1, using Toffoli gate.

INPUT			OUTPUT		
A	B	C	P = A	Q = B	R = AB XOR C
0	0	1	0	0	1
0	1	1	0	1	1
1	0	1	1	0	1
1	1	1	1	1	0

Fig. 17: Implement a two input NAND gate with inputs A & B and output R, using 3X3 Toffoli gate when C is 1.

From the truth table in Fig. 18, two input NAND gate operation is done through Port R, when C = 1, using Peres gate.

INPUT			OUTPUT		
A	B	C	P = A	Q = A XOR B	R = AB XOR C
0	0	1	0	1	1
0	1	1	0	1	1
1	0	1	1	0	1
1	1	1	1	0	0

Fig. 18: Implement a two input NAND gate with inputs A & B and output R, using3X3 Peres gate when C is 1.

From the truth table in Fig. 19, characteristics of two input NAND gate operation is done through Port S, when C is Grounded and D = 1, using HNG gate.

INPUT				OUTPUT			
A	B	C	D	P= A	Q= B	R = A XOR B XOR C	S= (A XOR B)C XOR AB XOR D
0	0	0	1	0	0	0	1
0	1	0	1	0	1	1	1
1	0	0	1	1	0	1	1
1	1	0	1	1	1	0	0

Fig. 19: Implement a two input NAND gate with inputs A & B and output S, using 4X4 HNG gate when C is grounded and D is 1.

From the truth table in Fig. 20, characteristics of two input NAND gate operation is done through Port S, when C = 1 and D is Grounded, using TSG gate.

INPUT				OUTPUT			
A	B	C	D	P= A	Q= A'C' XOR B'	R= (A'C' XOR B') XOR D	S= (A'C' XOR B')D XOR (AB XOR C)
0	0	1	0	0	1	1	1
0	1	1	0	0	0	0	1
1	0	1	0	1	1	1	1
1	1	1	0	1	0	0	0

Fig. 20: Implement a two input NAND gate with inputs A & B and output S, using 4X4 TSG gate when C is 1 and D is grounded.

D. Nor Gate

From the truth table in Fig. 21, characteristics of two input NOR gate operation is done through Port S, when C and D, both are 1, using HNG gate.

INPUT				OUTPUT			
A	B	C	D	P=A	Q=B	R=A XOR B XOR C	S = (A XOR B)C XOR AB XOR D
0	0	1	1	0	0	1	1
0	1	1	1	0	1	0	0
1	0	1	1	1	0	0	0
1	1	1	1	1	1	1	0

Fig. 21: Implement a two input NOR gate with inputs A & B and output S, using a 4X4 HNG gate when C and D are 1.

V. ILLUSTRATION

fn(a, b, c, d) be a four valued Boolean function as shown in (1). Implementation of this function using reversible logic gates, with truth table is shown in Fig. 22.

$$fn(a, b, c, d) = (a + b)c + ad \ldots\ldots\ldots (1)$$

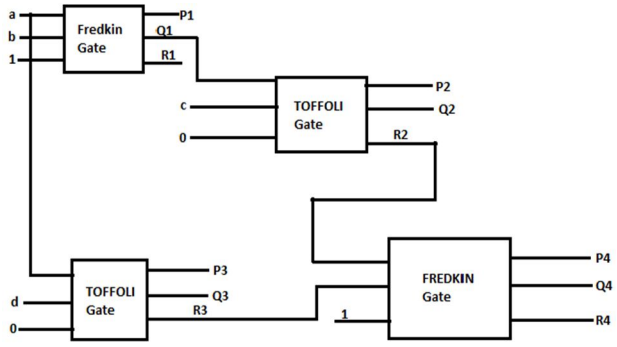


Fig. 22

[Here, P1 = a, Q1 = a'b xor b, R1 = a' xor ab, P2 = a'b xor b, Q2 = c, R2 = (a'b xor ab) and c, P3 = a, Q3 = d, R3 = a and d, P4 = (a'b xor ab) and c, Q4 = [((a'b xor ab) and c)'and ad] xor ad, R4 = (((a'b xor ab) and c)'xor [((a'b xor ab) and c] and ad)].

VI. CONCLUSION

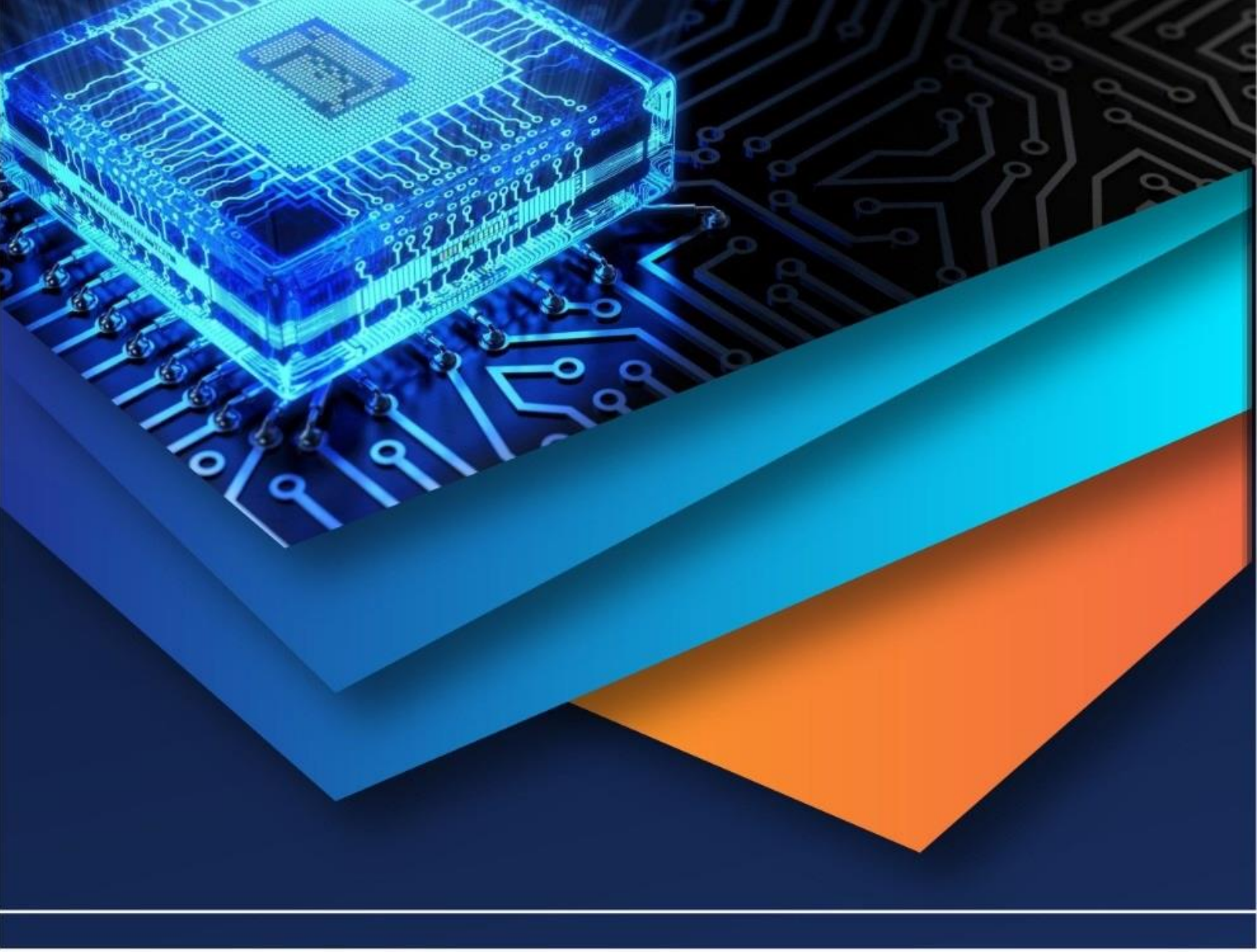
From the analyses, we can conclude that reversible gates can be used to implement all the basic as well as universal gates. Also, it can be used to implement any combinational digital operations using the truth tables, useful for those designs.

VII. ACKNOWLEDGEMENT

We would like to express our gratitude and respect to our Prof. Dr. Anindya Sen for his expert advice and encouragement which helped us in preparing this manuscript. We would also like to express our sincere thanks to our families for their enormous support in our work.

REFERENCES

[1] R. Landauer in “Irreversibility and Heat Generation in the Computing Process”. IBM J. Research and Development,5(3): pp. 183-191, 1961.
[2] R. Keyes, R. Landauer in “Minimal energy dissipation in logic”, IBM J. Res. Dev. 14(1970) 152–157.
[3] Charles H. Bennett , in "Logical Reversibility of computation", IBM Journal of Research and Development, vol. 17, no. 6, pp. 525-532, 1973.
[4] Soham Bhattacharya, Anindya Sen "Power and Delay Analysis of Logic Circuits Using Reversible Gates" International Journal of Latest Technology in Engineering, Management & Applied Science-IJLTEMAS vol.8 issue 12, December 2019, pp.54-63.
[5] P. Gupta; A. Agrawal; N.K. Jha in “An Algorithm for Synthesis of Reversible Logic Circuits”.
[6] M. A. Nielsen and I. L. Chuang, in “Quantum Computation and Quantum Information”, New York: Cambridge Univ. Press (2000)
[7] V.Vendral, A. Barenco, and A.Ekert,in “Quantum networks for elementary arithmetic operations”, Phys. Rev. A, vol. 54, no. 1, pp. 147-153, Jul1996.
[8] E. Fredkin and T. Toffoli in "Conservative Logic". International Theoretical Physics Vo121, pp.219-253, 1982.
[9] R. Feynman in “Quantum Mechanical Computers”, Optic News, Vol 11, pp 11-20 1985.
[10] A. Peres in “Reversible logic and quantum computers”, Phys.Rev. A 32 (1985) 3266-3276.
[11] Shams, M., M. Haghparast and K. Navi in “Novel reversible multiplier circuit in nanotechnology”, World Appl.Sci. J.,3(5): 806-810.
[12] Dmitri Maslov, Gerhard W. Dueck, and D. Michael Miller in “Synthesis of Fredkin–Toffoli Reversible Networks” in IEEE TRANSACTIONS ON VERY LARGE SCALE INTEGRATION (VLSI) SYSTEMS, VOL. 13, NO. 6, JUNE 2005.
[13] G. W. Dueck, D. Maslov, and D. M. Miller, “Transformation-based synthesis of networks of Toffoli/Fredkin gates,” in Proc. IEEE Canadian Conf. Electrical and Computer Engineering, May 2003, pp. 211–214.
[14] Himanshu Thapliyal and M.B. Srinivas in “A Novel Reversible TSG gate and Its Application for Designing Reversible Carry Look-Ahead and Other Adder Architectures” Center for VLSI and Embedded System Technologies, IIIT,Hyderabad, India, Asia-Pacific Conference on Advances in Computer Systems Architecture.



10.22214/IJRASET

45.98

IMPACT FACTOR: 7.129

IMPACT FACTOR: 7.429

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY