DESIGN AND FPGA IMPLEMENTATION OF REVERSIBLE PROGRAMMABLE READ ONLY MEMORY, ADDER & SUBTRACTER USING REVERSIBLE DECODER

Meghana PA M.Tech Student[1]
Dr. Ambedkar Institute of Technology
Bengaluru-560056

Dr. Ramesh S[2]
Professor, Dept of E&C
Dr. Ambedkar Institute of Technology
Bengaluru-560056

Abstract—

In this paper design & synthesize of Reversible PROM, half adder/subtracter & full adder/subtracter using reversible decoder logic is presented. Reversible logic is an emerging technology in the field of research in present era. The PROM Programmable Read Only Memory device which consists of fixed AND Gates and programmable OR gates array is one of the type of the simple PLD with n input and k output (referred as (n, k)) is said to be reversible if and only if the number of inputs is equal to number of outputs. The input pattern maps the output pattern uniquely. The reversible logic must run both forward and backward in such a way that the inputs can also be retrieved from outputs. The designed circuits are analyzed in terms of delay, quantum cost, garbage outputs and number of gates. The Circuit has been designed and simulated using Xilinx software and implemented on FPGA SPARTAN6.

KEYWORDS: PROM, Decoder, Adder/subtracter, Quantum Cost, Reversible Gates, Garbage Outputs, Number of gates, FPGA.

I. INTRODUCTION

In present VLSI technology, power Consumption has become a very important factor for consideration. By using Reversible logic, power consumption and heat dissipation can be minimized. Power consumption is very less in reversible logic circuits when compared to irreversible logic circuits. The heat dissipation in irreversible circuits is not because of the process involved in the operation, but it is due to the bits that were erased during the logical computation process Ralf Launder [1]. Launder's principle states that losing a single information bit in the circuit causes the smallest amount of heat in the computation which is of the order of to KTln2 joules where K is Boltzmann constant (approximately 1.38×10-23 J/K), T is Temperature and ln2 is natural algorithm of 2 (approximately 0.69315). The amount of heat dissipated in simple circuits is very small but it becomes large in the complex circuits. It is necessary to notice that there is a direct relationship between the numbers of information bits erased to the amount of heat dissipated in the circuit. The Power dissipation due

to the bit loss can be overcome if each and every computation in circuit was carried out in reversible manner C.H. Bennett [2]. As each gate perform a unitary operation, KTln2 Joules energy dissipation wouldn't occur if the computation is carried out in reversible manner. He argued that for zero heat dissipation, the computation must be done in reversible manner. But if reversible logic is utilized to do logical computation, the heat dissipation will be less than KTln2 for one information bit in contrast to Launder. Thus, computation done in reversible manner doesn't require erasing of bits.

Reversible Logic Gates

Reversible logic gate is said to reversible if and only if there is a one to one mapping between input vector lines and output vector lines. In reversible computation [2], the reversible gates are made to run both forward and backward directions. If it satisfies the above two conditions, then it obeys the second law of thermodynamics which preserves the information bits without being getting erased and guarantees that there is no heat dissipation. By using the output we can obtain the complete information of the inputs in reversible computing process.

The reversible logic gate consists of same number of inputs and outputs as shown in the Figure 1. The basic Reversible Logic Gates Fredkin Gate [12], Feynman Gate [4], Peres Gate [5], TR Gate [5] & NOT Gate are used. The gates with less quantum cost & suitable for design is selected. Certain constraints are to be considered while designing circuits based on reversible logic are

(i) Fan out is prohibited in reversible logic

(ii) Feedback is also prohibited in reversible logic.

Fan-out limitation can be overcome by using additional reversible logic gates, where the output lines are duplicated to necessary number of lines which in turn drives the inputs of consecutive device. Similarly delay elements are used for Feedback limitation.

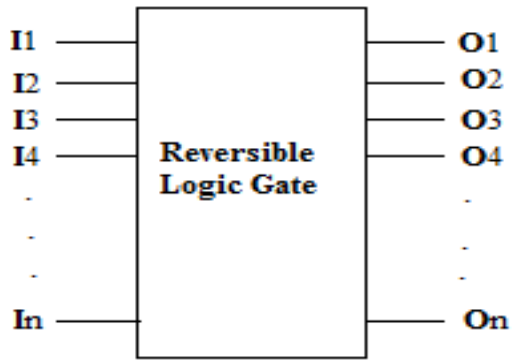


Figure 1: Simple Reversible Logic Gate

Basic Definitions

1. Garbage Outputs

Garbage outputs are the excess outputs which maintain reversibility by making number of inputs equal to number of outputs. They do not perform any type of operation but still has to be kept in the circuit as per the reversibility concept.

2. Number of Gates Count

Number of gates count is total number of reversible gates used in the circuit which becomes an important factor when cost case is considered. When new gates are to be formed by taking the number of gates it has not an appreciable good metric.

3. Quantum Cost

Quantum Cost depends on the number of basic reversible gates used to design the required reversible logic gate. It is nothing but the number

of reversible gates (1×1 or 2×2) required to construct the circuit. The quantum cost is very important to logical reversible computation. Total circuit area increases if the quantum cost increases & hence increases the propagation delay. But quantum cost doesn't impact heat dissipation. Delay is also an important when we consider the cost.

Table 1: Basic Reversible Logic Gates

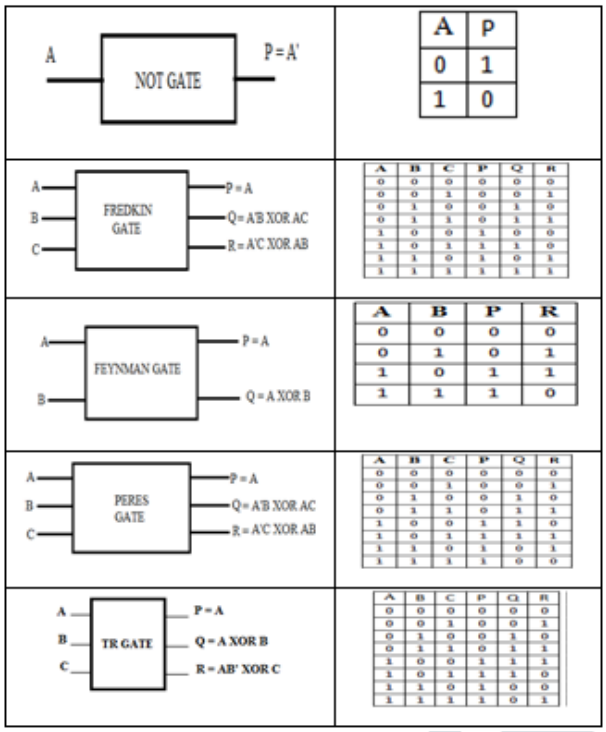


Table 1 shows the basic reversible logic gates used for designing the proposed system.

II. EXISTING SYSTEM

In existing system [13] Programmable Array Logic (PAL), Programmable Array Logic (PAL) & Gate Array Logic (GAL) are designed using the reversible logic. The programmable AND gate plane & OR gate plane are designed using reversible fuse while fixed connections by using the CNOT gate [12]. The time delays for reversible PAL, PLA & GAL are 5.847nsec, 5.847nsec & 5.847nsec. Time delay increases if the quantum cost is increased. The quantum cost increases with increase in length of Boolean equation. The Circuit has been designed and simulated using Xilinx 14.7 software and implemented on FPGA SPARTAN – 3E.

III. PROPOSED ALGORITHM

In the proposed system reversible PROM is designed using the reversible decoder i.e., by using 2*4, 3*8 & 4*16 decoder. By using the designed reversible PROM one of the combinational circuit's i.e. reversible half & full adder/Subtracter and also some Boolean functions is designed. The decoder acts as an AND array with fixed connections. 2×4 decoder with reversible gates like Peres gate, TR gate, NOT gate and CNOT gate. The whole 4*16 decoder design is done using Fredkin, NOT, CNOT, Peres gate & TR gate as shown in figure2. The Circuit is designed and simulated using Xilinx 14.7 software and implemented on FPGA SPARTAN – 6.

A decoder is a logic circuit that converts an N - bit binary input data into M bit binary output lines such that only one output line is activated for each one of the possible combinations of inputs. The decoder outputs for a particular input combination. The n-inputs are also given in the form of either 0 or 1; there are 2N possible input combinations. For each of input combination only one of the M outputs will be active (HIGH), all other outputs will remain inactive (LOW).

The 2:4 Decoder is designed using the Peres gate, TR gate, NOT gate and CNOT gate. If A & B are the inputs to the 2:4 decoder, then the four

possible outputs will be A'B', A'B, AB' and AB. The number of garbage output in this design is 3, quantum cost is 11 and the numbers of constant inputs are 3 as shown in the Figure 2.

The 3:8 Decoder is designed using the Peres gate, TR gate, NOT gate, Fredkin gate and CNOT gate. If A, B & C are the inputs to the 3:8 decoders, then the eight possible outputs will be A'B'C', A'B'C, A'BC', A'BC, AB'C', AB'C, ABC' and ABC. The number of garbage output in this design is 4, quantum cost is 31 and the numbers of constant inputs are 7 as shown in Figure 2

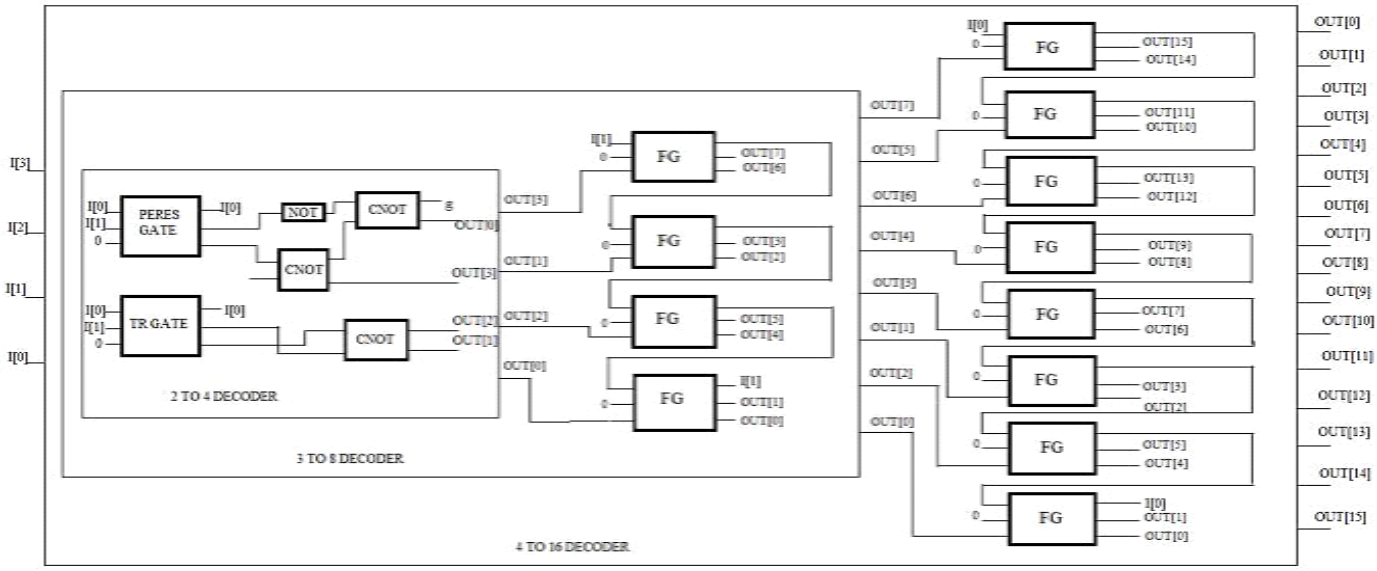
The 4:16 Decoder is designed using the Peres gate, TR gate, NOT gate, Fredkin gate and CNOT gate. If A, B, C & D are the inputs to the 4:16 decoder, then the sixteen possible outputs will be A'B'C'D', A'B'C'D, A'B'CD', A'B'CD, A'BC'D', A'BC'D, A'BCD', A'BCD, AB'C'D', AB'C'D, AB'CD', AB'CD, ABC'D', ABC'D, ABCD' and ABCD. The number of garbage output in this design is 5, quantum cost is 71 and the numbers of constant inputs are 15 as shown in the Figure2.

Figure 2: 4 to 16 Decoder

1. DESIGN OF PROM USING REVERSIBLE DECODER

The Programmable Read Only Memory (PROM) is designed by the use of reversible decoder that in turn uses the reversible logic. Here the AND array is fixed which can be built by the decoder. Duplicating a single output to the required number is done using Feynman gate due to the Fan-out limitation in reversible computation. The drawback with the standard conventional PROM is that the series of fuses present in it are burned to program the device can be programmed only once. It is an irreversible process where the fuses are burnt to program the chip.

In reversible PROM structure the fuses are replaced with a reversible fuse which is made of reversible Feynman gate and Fredkin gate as shown in the Figure 3(a). Here the Feynman reversible gate acts as a duplicating circuit. It duplicates the output line into two output lines in which one output line drives the next circuit & the other drives the second input of 2×1 reversible multiplexer. Reversible multiplexer's first input is grounded so



that it acts as an 'off' switch when the enable signal 'E' is low.

The reversible multiplexer [13] is made of Fredkin gate. The CNOT gates forms the fixed connection with the second input is set to '0' always. The CNOT gates give solution for two remedies i.e., it overcomes the feedback limitation and it acts as a fixed connection as shown in Figure 3b.

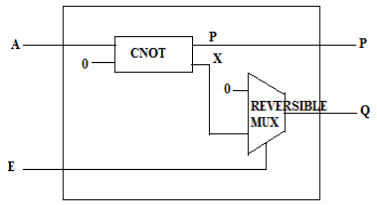


Figure 3a: Block Diagram of Reversible Fuse

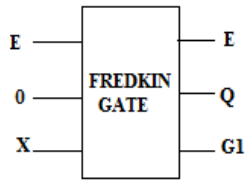


Figure 3b: Block Diagram of Reversible Mux

2. 8×3 REVERSIBLE PROM TO PERFORM FULL ADDER/FULL SUBTRACTOR OPERATION

The design of PROM made of reversible decoder which is programmed to perform the Full – Adder and Full – Subtracter operation is shown in the below Figure 4. Number of fuses used is 14 out of which 12 is used & 3 reversible OR gates are used to obtain sum/difference, carry & borrow respectively.

Sum/Difference = ∑ m (1, 2, 4, 7) …… (4)

Carry = ∑ m (3, 5, 6, 7) …… (5)

Borrow = ∑ m (1, 2, 3, 7) …….. (6)

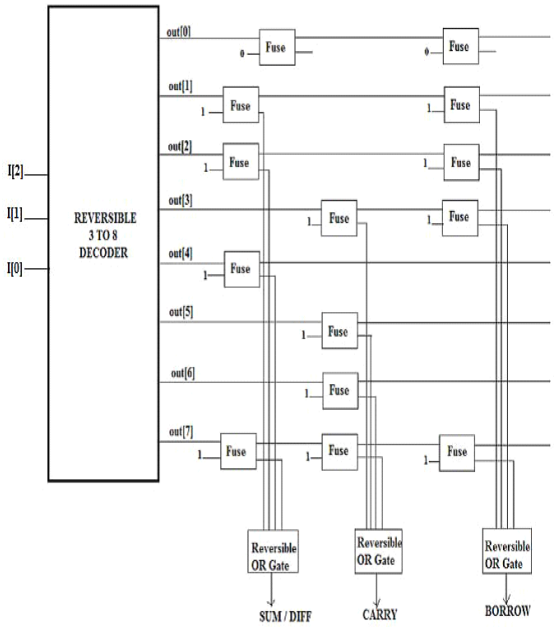


Figure 4: 8×3 Reversible PROM to Perform Full Adder/Full Subtracter Operation

3. 4×2 REVERSIBLE PROM TO PERFORM HALF ADDER/FULL SUBTRACTER OPERATION

The design of PROM made of reversible decoder which is programmed to perform the half – Adder and half – Subtracter operation is shown in the below Figure 5. Number of fuses used is 4 & 3 reversible OR gates are used to obtain sum/difference, carry & borrow respectively.

Sum/Difference = ∑ m (1, 2)…… (1)

Carry = ∑ m (3)…… (2)

Borrow = ∑ m (1) …… (3)

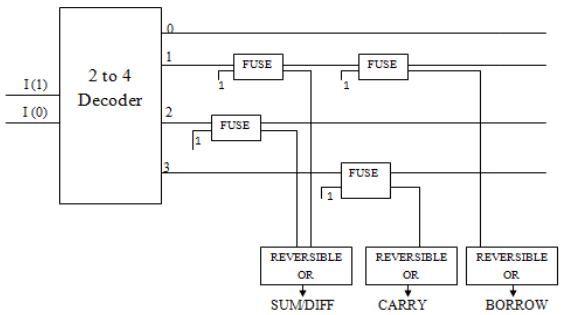


Figure 5: 4×2 Reversible PROM to Perform Half Adder/Full Subtracter Operation.

4. 16×5 REVERSIBLE PROM TO PERFORM BOOLEAN ALGEBRAIC FUNCTION OPERATION

The design of PROM made of reversible decoder which is programmed to perform the following Boolean functions is shown in the below Figure 6.

F1 = Σ m (0, 1, 10, 11) …….. (7)

F2 = Σ m (9, 11, 12, 13) …… (8)

F3 = Σ m (0, 2, 14, 15) …….. (9)

F4 = Σ m (3, 5, 6, 7) …… (10)

F5 = Σ m (5, 6, 8, 10) …….. (11)

The OR Gates present in the PROM are also reversible. The 'n' input OR gate consists of 'n' number of inputs. If any unusable ideal inputs are present they must be grounded (binary value '0') so as not to allow the high impedance value to OR gate which effects the proper functioning of OR gate. The fuses left without programming drives the value zero to the reversible OR gate which doesn't affect the operation of OR gate.

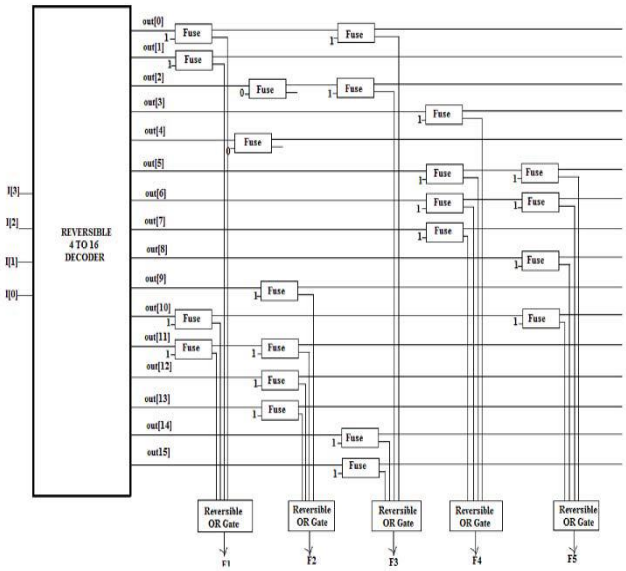


Figure 6: Circuit Diagram of 16×5 Reversible PROM to Perform Boolean Algebraic Functions Operation

ADVANTAGES:

- Low heat dissipation.
- Low power consumption.

APPLICATIONS:

- VLSI system applications.

IV. IMPLEMENTAION RESULTS, ANALYSIS AND COMPARISIONS

All the synthesis & simulation results of the proposed Reversible PROM, Half Adder/Subtracter & Full Adder/Subtracter are performed using Verilog HDL. The synthesis and simulation are performed on Xilinx ISE 14.7. The corresponding simulation results of the proposed Reversible PROM, Half Adder/Subtracter & Full Adder/Subtracter are shown below.

7. Fredkin Gate (FG)

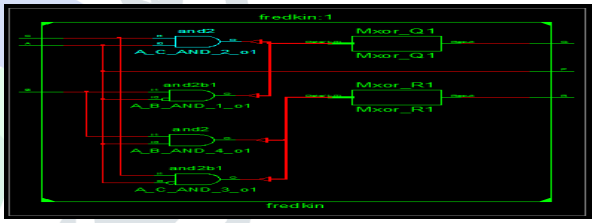


Figure 7a: RTL of Fredkin Gate

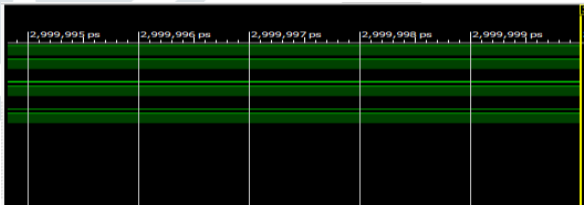


Figure 7b: Simulated Output of Fredkin Gate

8 Peres Gate (PR)

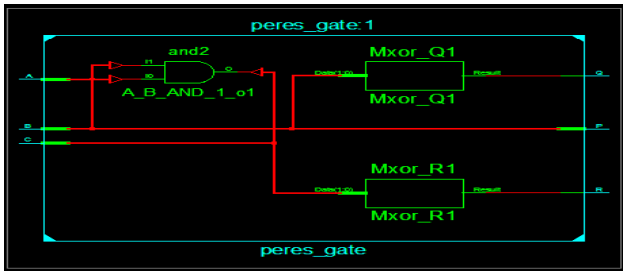


Figure 8a: RTL of Peres Gate

Figure 8b: Simulated Output of Peres Gate

9. TR Gate

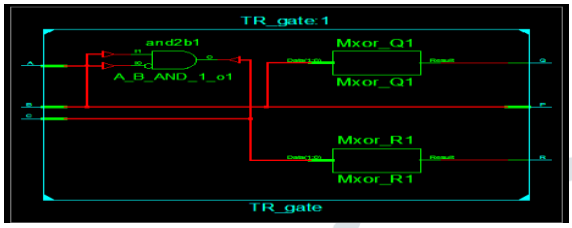


Figure 9a: RTL of TR Gate

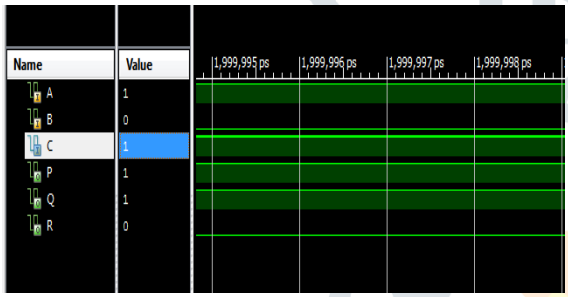


Figure 9b: Simulated Output of TR Gate

10 FEYNMAN GATE OUTPUT

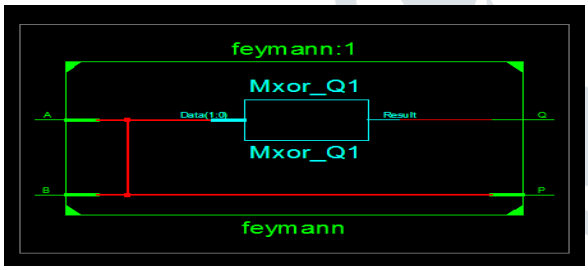


Figure 10a: RTL of Feynman gate

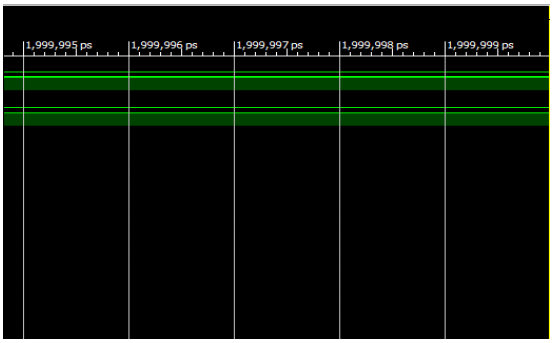


Figure 10b: Simulated Output of Feynman Gate

Figure 7a, 8a, 9a & 10a are the simulated RTL schematic of the Fredkin Gate, Peres Gate & Feynman Gate. Figure 7b, 8b, 9b & 10b are the simulated outputs.

11. REVERSIBLE FUSES

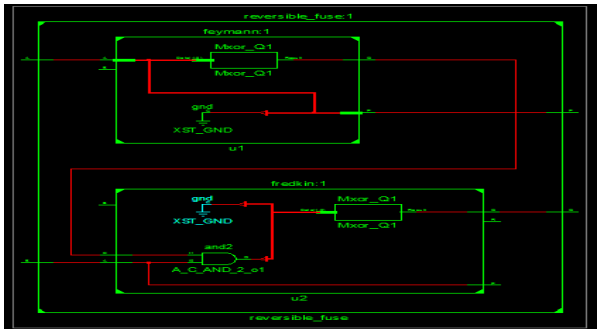


Figure 11: RTL of Reversible Fuse

Figure 11 shows the Simulated RTL of reversible fuse.

12. 2*4 Reversible Decoder

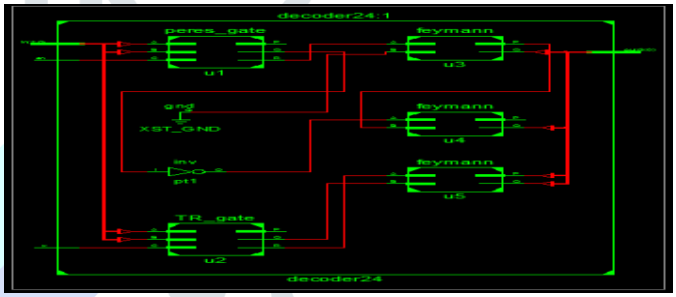


Figure 12a: RTL of 2*4Reversible Decoder

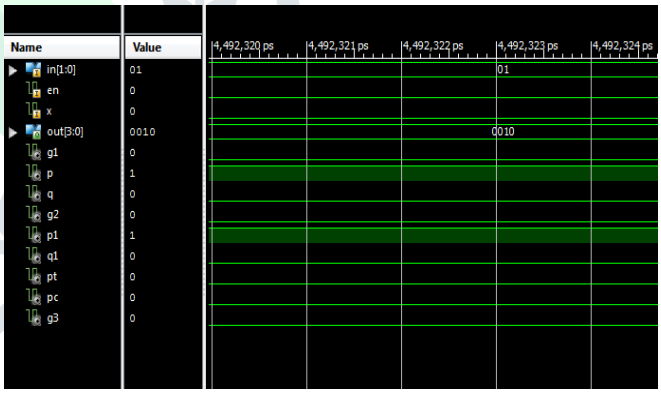


Figure 12b: Simulated Output of 2*4 Reversible Decoder

13. 3*8 Reversible Decoder

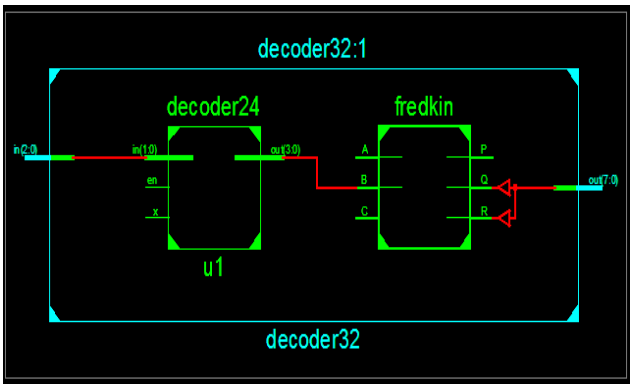


Figure 13a: RTL of 3*8Reversible Decoder

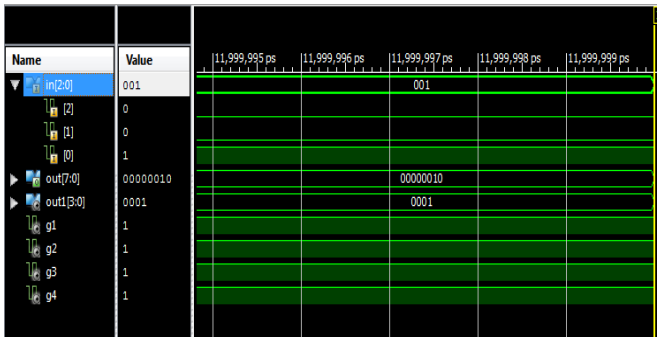


Figure 13b: Simulated Output of 3*8 Reversible Decoder

14. 4*16 Reversible Decoder

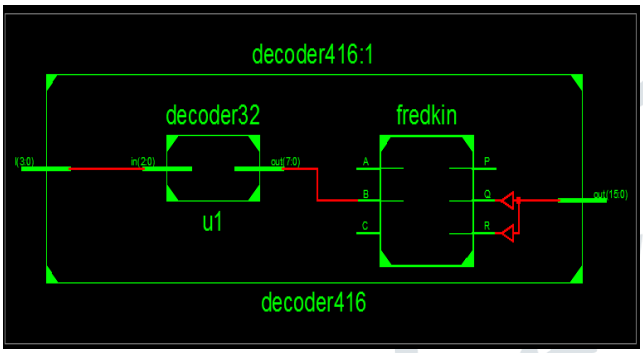


Figure 14a: RTL of 4*16Reversible Decoder

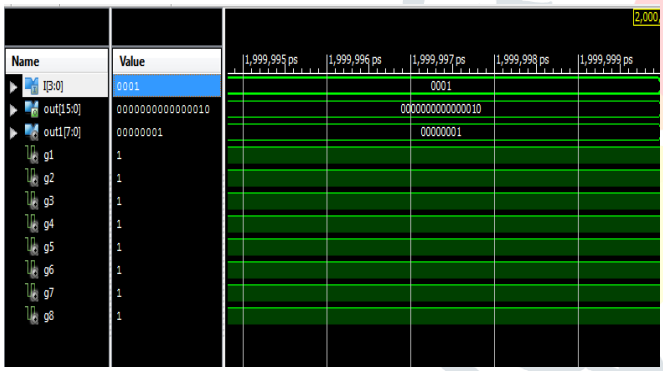


Figure 14b: Simulated Output of 4*16Reversible Decoder

Figure 12a, 13a & 14a shows the simulated RTL of reversible decoders. Figure 12b, 13b & 14b are the simulated output of the reversible decoders.

15. Reversible Half Adder/Subtracter Using 4* 3 Reversible PROM

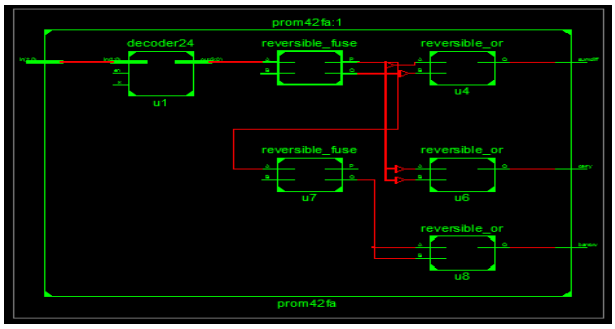


Figure 15a: RTL of Half Adder/ Subtracter

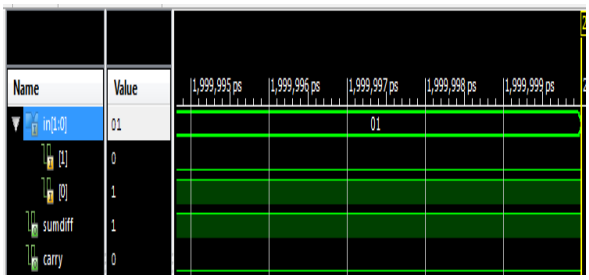


Figure 15b: Simulated Output of Half Adder/ Subtracter

16. Reversible Full Adder/Subtracter Using 8*3 Reversible PROM

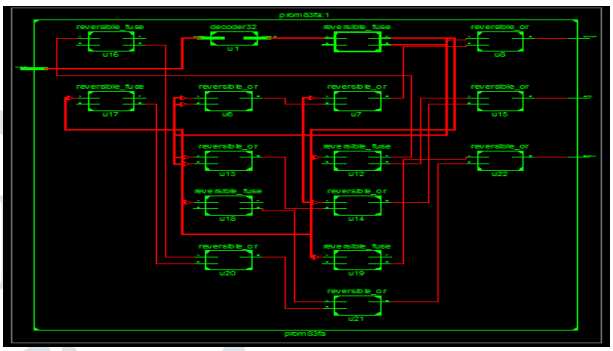


Figure 16a: RTL of Full Adder/Subtracter

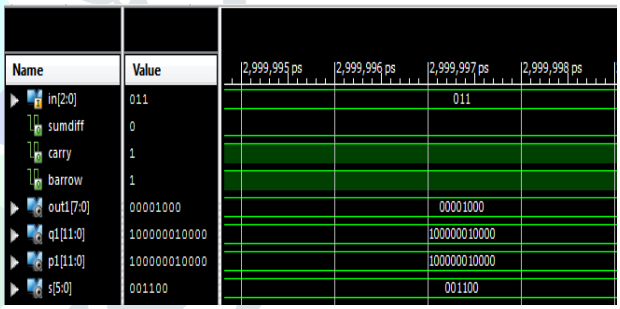


Figure 16b: Simulated Output of Full Adder/Subtracter

17. Boolean Functions Implementation Using Reversible 16*5 PROM

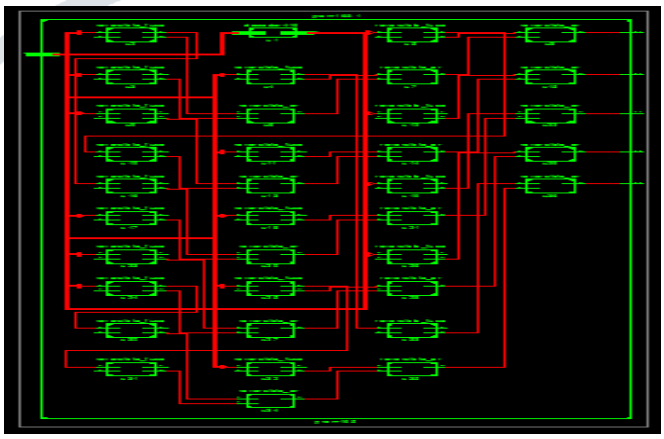


Figure 17a: RTL of Boolean Functions Implementation Using Reversible 16*5 PROM

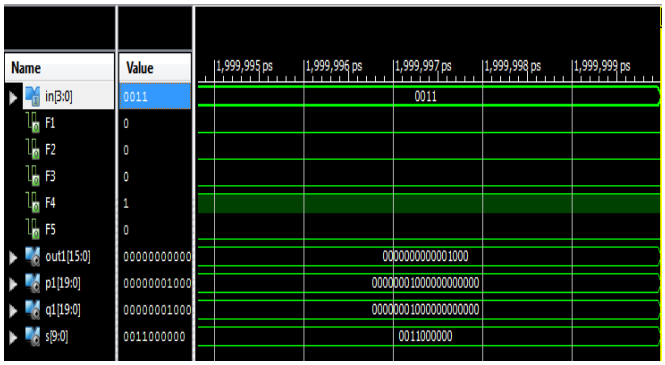


Figure 17b: Simulated Output of Boolean Functions Implementation Using Reversible 16*5 PROM

Figure 15a, 16a & 17a are the RTL schematic of Half Adder/Subtracter, Full Adder/Subtracter & Boolean function equations. Figure 15b, 16c &17c are the simulated outputs of it.

V. FPGA Implementation Table

Table 2: Reversible Half Adder/Subtracter

I (1)	I (0)	SUM/DIFF	CARRY	BORROW
0	0	0	0	0
0	1	1	0	1
1	0	1	0	0
1	1	0	1	0

Table 3: Reversible Full Adder/Subtracter

In[2]	In[1]	In[0]	Sum/diff	Carry	Borrow
0	0	0	0	0	0
0	0	1	1	0	1
0	1	0	1	0	1
0	1	1	0	1	1
1	0	0	1	0	0
1	0	1	0	1	0
1	1	0	0	1	0
1	1	1	1	1	1

Table 4: Boolean Algebraic Function Operation

In[3]	In[2]	In[1]	In[0]	F1	F2	F3	F4	F5
0	0	0	0	1	0	1	0	0
0	0	0	1	1	0	0	0	0
0	0	1	0	0	0	1	0	0
0	0	1	1	0	0	0	1	0
0	1	0	0	0	0	0	0	0
0	1	0	1	0	0	0	1	1
1	0	0	0	0	0	0	0	1
1	0	0	1	0	1	0	0	0

Once is design of the proposed circuit is completed i.e., synthesized then next step is to verify the design by using the above FPGA implementation truth table as shown in Table 2. This table is checked in both Xilinx simulation & FPGA hardware implementation. Similarly, it is also checked for full adder/subtracter (Table 3) & for the Boolean function operation (Table 4).

VI. Comparision Table

Table 5: Comparision Table

Circuit	Quantum Cost	Garbage output
2 * 4 Decoder	11	3
3 * 8 Decoder	31	4
4 * 16 Decoder	71	5
AND Gate	5	2
OR Gate	5	2
Fuse	6	2

Table 6: Comparision Table

Circuits	Delay	No. of slices	Quantum Cost	Garbage Output
PROM 16*5	5.537nsec	5	278	23
Half adder/subtracter (using Decoder)	5.330nsec	3	50	6
Full adder/subtracter (using Decoder)	5.456nsec	3	72	11
Half adder/subtracter (using Fuse)	5.33onsec	3	75	6
Full adder/subtracter	5.456nsec	3	161	11

Table 5 shows the quantum cost & garbage output of different sized reversible decoder, AND gate, OR gate & reversible fuse. From the above Comparision Table 6 it is clear that design of the full adder/subtracter, half adder/subtracter using the reversible decoder proves more advantageous since it has less quantum cost & garbage count when compared to design using reversible fuse.

VII. CONCLUSION

In the proposed method PROM, half adder/subtracter & full adder/subtracter is designed and analyzed in terms of quantum cost, garbage output & number of slices used. Delays obtained for the simulated output are 5.537ns, 5.330ns & 5.456ns respectively. When compared to the existing method it proves to be useful in terms of quantum cost. Main purpose of designing is to minimize quantum cost, to obtain minimum gate count and minimum garbage outputs. The propagation delay can be reduced if the quantum cost of the circuit is reduced. Using this method various other combinational circuits like multiplexer, de-multiplexer, n-bit adders, ALU etc, can be designed and hence this can be termed as the future scope of the design.

REFERENCES

[1] R. Landauer, "Irreversibility and Heat Generation in the Computational Process", IBM Journal of Research and Development, vol. 5, pp. 183- 191, July 1961.

[2] C.H. Bennett, "Logical Reversibility of Computation", IBM J.Research and Development, vol. 17, pp. 525-532, November 1973.

[3] C H Bennett, "Notes on the History of Reversible Computation", IBM Journal of Research and Development, vol. 32, pp. 16-23, January 1998.

[4] R. Feynman, "Quantum Mechanical Computers", Optic News, vol. 11, pp 11-20, February 1985.

[5] A. Peres, "Reversible Logic and Quantum Computers", phys. Rev. A, Gen. Phys., vol. 32, no. 6, pp. 32663276, Dec. 1985.

[6] H.G Rangaraju, U. Venugopal, K.N. Muralidhara, K. B. Raja, "Low Power Reversible Parallel Binary Adder/Subtracter", arXiv.org/1009.6218,2010.

[7] J.M. Rabaey and M. Pedram, "Low Power Design Methodologies", Kluwer Academic Springer, 1996.

[8] T Toffoli, "Reversible Computing", Technical memo MIT/LCS/TM-151, MIT Lab for Computer Science, 1980.

[9] Y. Syamala, and A. V. N. Tilak, "Reversible Arithmetic Logic Unit", Electronics Computer Technology (ICECT), 2011 3rd International, vol.5, pp.207-211,07 July 2011.

[10] Thapliyal H, Ranganathan N.," Design of Reversible Latches Optimized for Quantum Cost, Delay and Garbage Outputs", Centre for VLSI and Embedded .

[11] V.Rajmohan, V.Ranganathan, "Design of Counter Using Reversible Logic" 978-1-4244-8679-3/11/$26.00 ©2011 IEEE.

[12] V. V. Shende, A. K. Prasad, I. L. Markov, and J. P .Hayes," Synthesis of Reversible Logic Circuits", IEEE Transaction on computer-aided design of integrated circuits and systems, vol. 22, No. 6, June 2003.

[13] Gopi Chand Naguboina, K.Anusudha, "Design and Implementation of PAL and PLA Using Reversible Logic on FPGA Spartan 3E", 2017 4th International Conference on Signal

Processing, Communications and Networking (ICSCN -2017), March 2017.

[14] Mounika.N and V.Vijay Bhaskar, "Realization of Programmable Logic Array Using Compact Reversible Logic Gates", International journal & magazine of engineering, technology, management and research July 2017.

.