DESIGN AND IMPLEMENTATION OF PAL AND PLA USING REVERSIBLE LOGIC

D.V.DINESH KUMAR REDDY[1], P.HARI OBULESU[2]

[1]PG Student, Dept of ECE(VLSI-SD), SRIT, Proddatur, AP, India.

[2]Assistant Professor, Dept of ECE, SRIT, Proddatur, AP, India.

Abstract— **The aim of this paper is to design and synthesize a Programmable array Logic (PAL) and Programmable Logic array (PLA) using reversible logic with minimum quantum cost. The PAL is a Programmable Logic device which consists of programmable AND Gates and fixed OR gates array. The PLA is the PLD which contains programmable AND array and programmable OR array. The PLDs are the combinational circuits mainly used to realize Boolean functions on our interest. An n input and k output Boolean function f (a1, a2, a3, …., an) (referred as (n, k)) is said to be logically reversible if and only if, the number of inputs are equal to the number of outputs i.e., 'n' equals 'k' and the input pattern maps uniquely maps the output pattern. To overcome the Fan out limitation, the signals from required output lines are duplicated to desired lines using additional reversible combinational circuits. Reversible Logic owns its applications in various fields which include Quantum Computing, Optical Computing, Nano- technology, Computer Graphics, low power VLSI etc., Reversible logic is gaining its own importance in recent years largely due to its property of low power consumption and low heat dissipation. In this paper, the design of PAL and PLA which has less heat dissipation and low power consumption is proposed. The designed circuits are analyzed in terms of quantum cost, garbage outputs and number of gates. The Circuit has been designed and simulated using Xilinx software.**

KEYWORDS: **PAL, PLA, PLDS,** ***Quantum Cost, Reversible Gates, Garbage Outputs, Number of gates*****.**

I. INTRODUCTION

In the past few decades, reversible logic has become one of the most promising research areas. In modern technologies, power dissipation is an important issue and overheating is a serious concern for both manufacturer and consumer. When information loss occurs, energy is also lost. It happens when an input cannot recover its output and it has been proved by Landauer [1]. Healso expressed that, if a bit of information is lost, then*KTln2* joules of heat generate; Where *K* is Boltzman's Constant of1.38×10−23J/K and *T* is absolute temperature. To reduce energy waste, reversible circuit can be used and Bennet showed it [2]. Reversible logic follows one to one mapping system followed by input number and output number remains equal. Here no information loss and no energy dissipation occur. Miller proved that, if the number of gate increased, then it is not a good metric of optimization [3]. But reversible computing dissipates zero energy in terms of information loss and also it can detect errors of circuit by keeping unique input output mapping. Told that 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. He demonstrated Launder's principle which describes the lower theoretical limit of heat dissipation in logical computation. Launder's principle states that losing a single information bit in the circuit causes the smallest amount of heat in the computation which is equal 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 number of information bits erased to the amount of heat dissipated in the circuit. Later in 1973 C. H. Bennett [2] described that the Power dissipation due to the bit loss can be overcome if each and every computation in circuit was carried out in reversible manner. Quantum networks are designed of quantum logic gates. 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.

II. RESEARCH WORK

The Reversible Logic involves the use of Reversible Gates which consists of the same number of inputs and outputs i.e., there should be 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 the device obeys above two conditions, it satisfies the second law of thermodynamics which preserves the information bits without getting erased and guarantees that no heat is dissipated**.** Certain limitations are to be considered when designing circuits based on reversible logic (i) Fan out is not permitted in reversible logic and (ii) Feedback is also not permitted in reversible logic. In Reversible logic using outputs we can obtain full knowledge of inputs. To overcome the Fan-out limitation, by using additional reversible combinational circuits, the output lines are duplicated into required number of lines that are required to drive the inputs of consecutive device. Similarly for Feed- back limitation delay elements are used. Reversible logic conserves information. Some cost metrics [5][4] like Garbage outputs, Number of gates, Quantum cost, constant inputs are used to estimate the performance of reversible circuits. Garbage outputs are the extra outputs which help to make inputs and outputs equal in order to maintain reversibility. They are kept alone without performing any operations. Number of gates count is not a good metric since more number of gates can be taken together to form a new gate.

The quantum cost plays an important role in logical reversibility. If the quantum cost is more, then the area of the circuit increases, thereby increasing the propagation delay. But quantum cost doesn't impact heat dissipation. Delay is one of the important cost metrics. A Reversible circuit design can be modeled as a sequence of discrete time slices and depth is summation of total time slices. In Digital Electronics the binary decoder is a combinational logic circuit that converts the binary integer value to the associated output pattern. Various proposals are given to design of combinational and sequential circuits in the undergoing research.

In this paper the design of Programmable Array Logic (PAL) and Programmable Logic array (PLA) using reversible logic with minimum Quantum cost is proposed.

III. REVERSIBLE LOGIC GATES

The reversible logic gate consists of same number of inputs and outputs as shown in the figure1. The basic Reversible Logic Gates present in the literature are briefed below. The gates that are suitable for the design with optimum quantum cost can be selected.

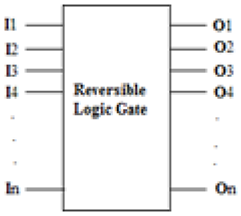
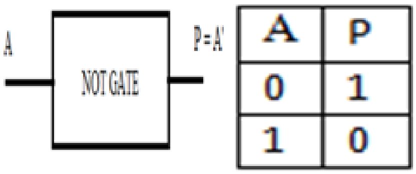


Fig.1 Simple reversible logic gate

1.NOT GATE:

The NOT GATE is the simple Reversible Logic gate. It is 1×1 Reversible Logic Gate with the quantum cost zero. The Not gate simply shifts the complementary of the input to output as shown in the figure2. It is the basic primitive gate which may involve in construction of reversible logic gate, thus owing its own importance in determining the quantum cost of designed Reversible logic gate.



A	P
0	1
1	0

Fig. 2 NOT Gate and its Truth Table

2.FEYNMAN GATE (FG):

Feynman gate [4] is a 2×2 reversible gate as shown in below figure3. The Feynman gate is also called as CNOT gate i.e., controlled NOT gate. The Feynman gate is used to duplicate of the required outputs since Fan-out is not allowed in reversible logic gates. The Quantum Cost of FG is 1. This is also the primitive gate owing its importance in determining quantum cost metric.

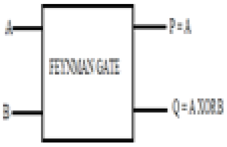
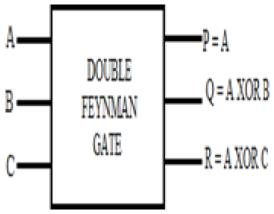


fig. 3 Feynman Gate and its Truth Table

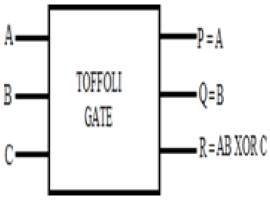
3.DOUBLE FEYNMAN GATE (F2G):
Double Feynman Gate [4] is a 3×3 reversible gate. The outputs P, Q, R are defined as the functions of inputs as shown in the figure4. The quantum cost of F2G is 2. This gate can also be used for duplicating outputs.



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

Fig.4 Double Feynman Gate and its Truth Table

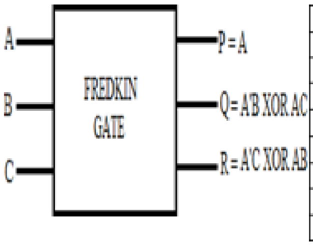
4.TOFFOLI GATE (TG):
Toffoli Gate [8] is 3×3 reversible gate. The outputs P, Q, R are defined as the functions of inputs as shown in the below figure5. The Quantum Cost of TG is 4.



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

Fig.5 Toffoli Gate and its Truth Table

5.FREDKIN GATE (FRG):
Fredkin Gate [12] is a 3×3 reversible gate. The outputs P, Q, R are defined as functions of inputs as shown in the below figure6. The Quantum Cost of FRG is 5. This paper mainly surrounds around 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	1	0
1	1	0	1	0	1
1	1	1	1	1	1

Fig.6 Functions of inputs

6.PERES GATE (PG): Peres Gate [5] is a 3×3 reversible gate. The outputs P, Q, R are defined as functions of inputs as shown in the below figure7. The Quantum Cost of PG is 4.

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

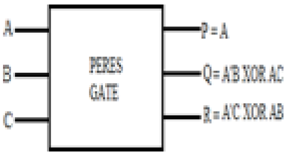
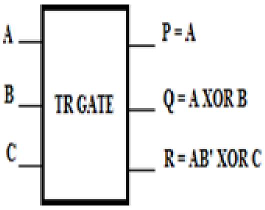


Fig.7 Peres Gate and its Truth Table

7.TR GATE: TR Gate [5] is a 3×3 reversible gate. The outputs P, Q, R are defined as functions of inputs as shown in the below figure8. The quantum cost of TRG gate is given by 4.



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

Fig.8 TR Gate and its Truth Table

IV. BASIC GATES USING REVERSIBLE GATES

Considering our circuit requirements we need to design AND gate and OR gate using reversible gates. Here fredkin gate is used to design AND and OR gates as shown in figure9. Importance is given to fredkin gate because it gives optimistic performance at less Quantum Cost for designing AND and OR gates.

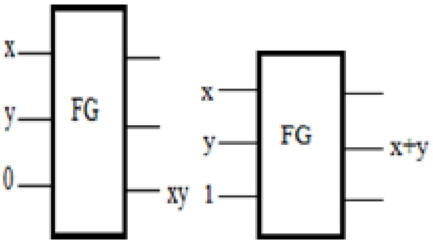


Fig.9 AND Gate using fredkin and OR Gate using fredkin

V. EXISTING METHOD

The Design of Combinational circuits [6][9][16] and Sequential Circuits [10][11] using reversible logic has been ongoing in research. Various proposals are given for the design of combinational circuits like adders, subtractors, multiplexers, decoders etc., in the existing method the author has given a novel design of 4x16 decoder [15] whose Quantum Cost is less than the previous design. Replacing fredkin gates for designing 2×4 decoder with reversible gates like Peres gate, TR gate, NOT gate and CNOT gate are used.

The whole design is done using Fredkin, CNOT, Peres gates which give better Quantum Cost when compared to the other reversible Logic gates. The number of gates required to design 4x16 decoder are 18 in which there are 12 fredkin gates, one peres gate, one TR gate, one NOT gate and 3 CNOT gates. The sum of all the quantum costs of each gate gives total quantum cost of 4x16 decoder. The sum of all the quantum costs of each gate used to design total circuit gives the quantum cost of total circuit.

VI. PROPOSED METHOD

The Programmable Array Logic (PAL), Programmable Logic array (PLA) is realized using reversible Fredkin gate and Feynman gate as shown in the figure11 and figure12 respectively. The concept of duplicating a single output to required number of outputs using Feynman gate is introduced where Fan-out was not allowed in reversible computation. The standard irreversible conventional PLDs can be programmed. The irreversible PLDs consist of a series of fuses which can be burned to program the device. By burning the fuses the chip can be programmed which is an irreversible process.

In reversible PLDs structure, the fuses are replaced with a reversible fuse which is made of reversible Feynman gate and fredkin gate as shown in the below figure10 (a). The Feynman reversible gate acts as a duplicating circuit. It duplicates the output line into two output lines out of which one output line drives the next circuit and the other drives the second input of 2×1 reversible multiplexer. The first input of reversible multiplexer is grounded so that when the enable signal 'E' is low it acts as an 'off' switch. The reversible multiplexer is made of Fredkin gate as shown in the below figure10 (b).

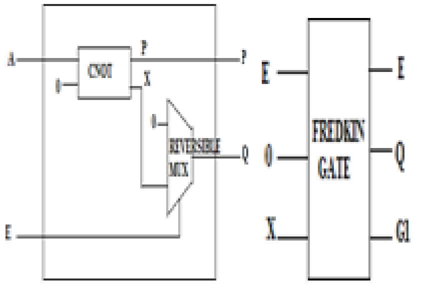


Fig.10 (a) Reversible Fuse Fig.10 (b) Reversible MUX

The fixed connections are replaced by CNOT gates in which the second input is set to '0' always. The CNOT gates give solution for two remedies. It overcomes the feedback limitation and it acts as a fixed connection. The Design of PAL made of reversible logic which is programmed to perform the operation of the below Boolean algebraic equations is shown in the below figure11. The PAL consists of fixed OR gates array and programmable AND gate array.

F1=I[1]I[2]+I[1]I[3]'+I[1]'I[2]'I[3] .... Eqn (1)
F2=I[1]I[2]+I[1]'I[2]'I[3]+I[1]I[3] ……..Eqn (2)
F3=I[1]I[3]'+I[1]I[2]I[3] …………… Eqn (3)

Contemporary to irreversible PAL, the fuses are replaced with programmable reversible fuses and the fixed connections are replaced with the CNOT gates as shown in the figure11. The 'P' output of fuse drives the subsequent fuse and the 'Q' output of fuse drives the input of AND gate as shown in figure13. The output of AND gate drives the fixed connections i.e., CNOT gate. The 'P' output of CNOT gate drives the next fixed connection and the 'Q' output of CNOT drives the input of OR gate.

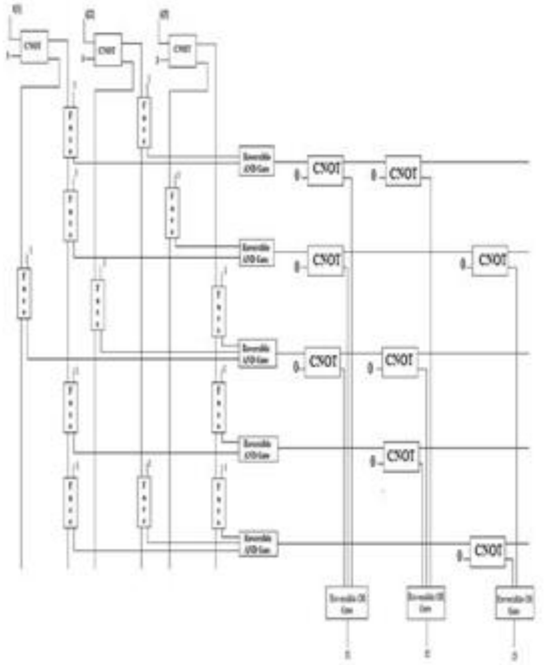


Fig.11 Circuit diagram of Reversible PAL to perform Boolean algebraic equations operation.

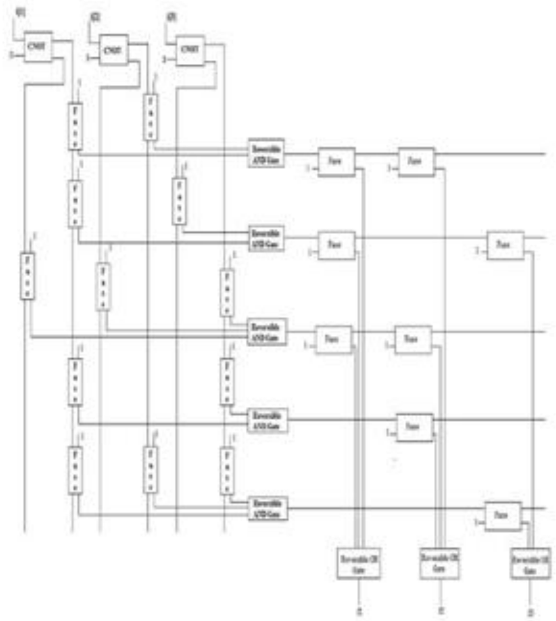


Fig.12 Circuit diagram of Reversible 3-5-3 PLA to perform Boolean Algebraic equations operation.

The OR Gates and AND gates used in PLDs are made of Reversible logic. The 'n' input OR gate consists of 'n' number of inputs. If any inputs are kept ideal without use they are to be driven with ground (binary value '0') so that the high impedance value is not driven to OR gate which effects the operation 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. Contrast to OR gate, for AND gate the left out inputs are driven with HIGH signal so that the operation of AND gate is not affected.

The circuit diagram of a Programmable Logic Array constructed using reversible logic to program Boolean equations Eqn (1), Eqn (2), Eqn (3) is shown in the above figure12. In PLA both the AND gate array and OR gate arrays are programmable. Hence reversible fuses are used to program the AND gate array and OR gate array of the device as shown in the figure12. Similarly the circuit diagram of a Generic Array Logic constructed using reversible logic to program Boolean equations Eqn(1), Eqn(2), Eqn(3) is shown in the below figure13. In GAL the outputs are also programmable along with the AND gate array and OR gate array. Hence it differs from PLA. Therefore reversible fuses are used to program the device as shown in the figure13.

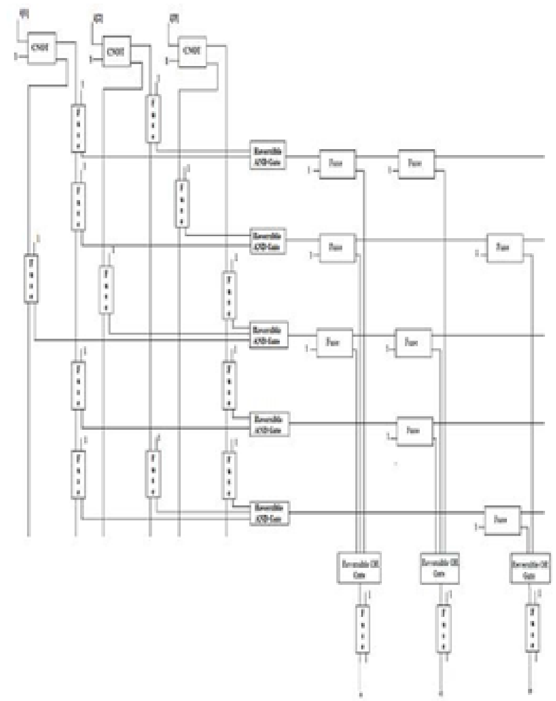


Fig.13 Circuit diagram of Reversible GAL to perform Boolean Algebraic equations operation.

VI. SIMULATION RESULTS AND COMPARISION

1. REVESIBLE PROGRAMMABLE ARRAY LOGIC

The equations Eqn 1 to Eqn 3 are implemented using reversible PAL. The RTL schematic and simulated output for Eqn1 to Eqn3 implemented using reversible PLA is shown in the figure14 and figure17 respectively.

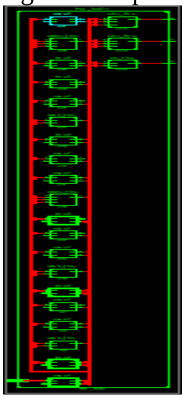


Fig.14 RTL Schematic of PAL implementing using reversible logic gates .

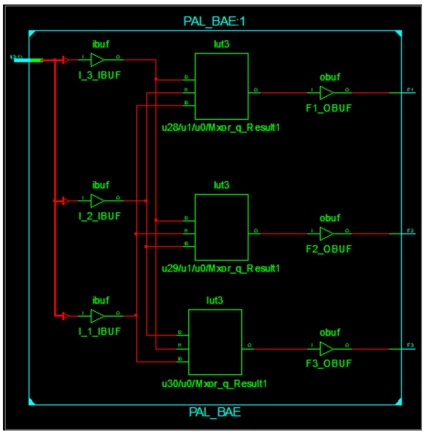
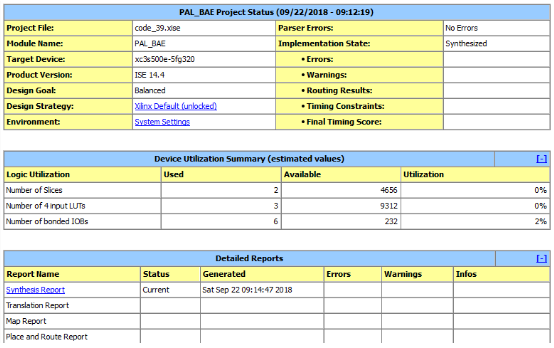


Fig 14.a Technology schematic of proposed PAL



PAL_BAE Project Status (09/22/2018 - 09:12:19)			
Project File:	code_39.xise	Parser Errors:	No Errors
Module Name:	PAL_BAE	Implementation State:	Synthesized
Target Device:	xc3s500e-5fg320	• Errors:	
Product Version:	ISE 14.4	• Warnings:	
Design Goal:	Balanced	• Routing Results:	
Design Strategy:	Xilinx Default (unlocked)	• Timing Constraints:	
Environment:	System Settings	• Final Timing Score:	

Device Utilization Summary (estimated values) [-]			
Logic Utilization	Used	Available	Utilization
Number of Slices	2	4656	0%
Number of 4 input LUTs	3	9312	0%
Number of bonded IOBs	6	232	2%

Detailed Reports [-]					
Report Name	Status	Generated	Errors	Warnings	Infos
Synthesis Report	Current	Sat Sep 22 09:14:47 2018			
Translation Report					
Map Report					
Place and Route Report					

Fig 14.b design utilization summary of proposed PAL

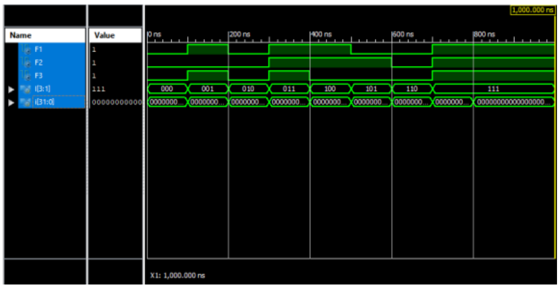


Fig 14.c simulation output of proposed PAL synthesized and simulated using XILINX 14.4 design suit.

2.REVERSIBLE PROGRAMMABLE LOGIC ARRAY

The equations Eqn 1 to Eqn 3 are implemented using reversible PLA. The RTL schematic implemented using reversible PLA is shown in the figure15 respectively.

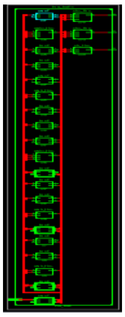


Fig 15 RTL Schematic of PLA implementing using reversible logic gates

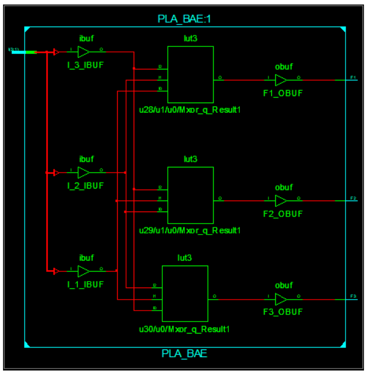
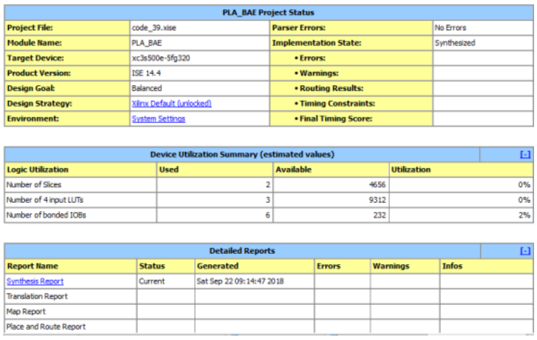


Fig 15.a Technology schematic of proposed PLA



PLA_BAE Project Status			
Project File:	code_39.xise	Parser Errors:	No Errors
Module Name:	PLA_BAE	Implementation State:	Synthesized
Target Device:	xc3s500e-5fg320	• Errors:	
Product Version:	ISE 14.4	• Warnings:	
Design Goal:	Balanced	• Routing Results:	
Design Strategy:	Xilinx Default (unlocked)	• Timing Constraints:	
Environment:	System Settings	• Final Timing Score:	

Device Utilization Summary (estimated values) [-]			
Logic Utilization	Used	Available	Utilization
Number of Slices	2	4656	0%
Number of 4 input LUTs	3	9312	0%
Number of bonded IOBs	6	232	2%

Detailed Reports [-]					
Report Name	Status	Generated	Errors	Warnings	Infos
Synthesis Report	Current	Sat Sep 22 09:14:47 2018			
Translation Report					
Map Report					
Place and Route Report					

Fig 15.b Design utilization summary of proposed PLA

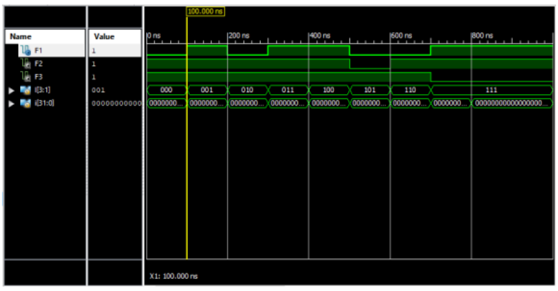


Fig 15.c Simulation output of proposed PLA

3.REVERSIBLE GENERIC ARRAY LOGIC

The equations Eqn 1 to Eqn 3 are implemented using reversible GAL. The RTL schematic and simulated output for Eqn1 to Eqn3

IJMTARC – VOLUME – VI – ISSUE – 24, OCT-DEC, 2018 ISSN: 2320-1363

implemented using GAL is shown in the figure16 and figure16.c respectively.

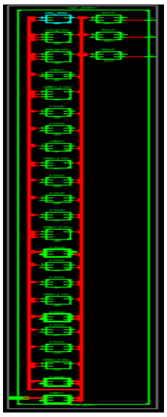


Fig.16 RTL schematic of reversible GAL

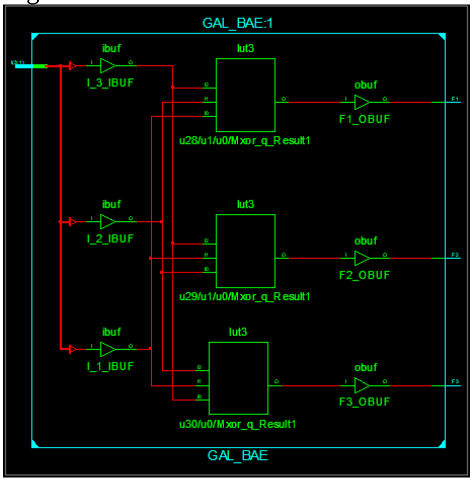
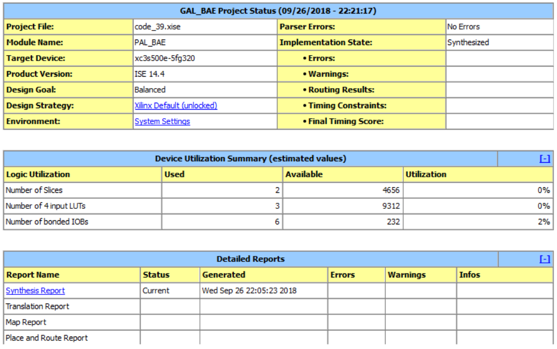


Fig 16.a Technology schematic of proposed GAL



GAL_BAE Project Status (09/26/2018 - 22:21:17)			
Project File:	code_39.xise	**Parser Errors:**	No Errors
Module Name:	PAL_BAE	**Implementation State:**	Synthesized
Target Device:	xc3s500e-5fg320	**• Errors:**	
Product Version:	ISE 14.4	**• Warnings:**	
Design Goal:	Balanced	**• Routing Results:**	
Design Strategy:	Xilinx Default (unlocked)	**• Timing Constraints:**	
Environment:	System Settings	**• Final Timing Score:**	

Device Utilization Summary (estimated values)			[-]
Logic Utilization	**Used**	**Available**	**Utilization**
Number of Slices	2	4656	0%
Number of 4 input LUTs	3	9312	0%
Number of bonded IOBs	6	232	2%

Detailed Reports					[-]
Report Name	**Status**	**Generated**	**Errors**	**Warnings**	**Infos**
Synthesis Report	Current	Wed Sep 26 22:05:23 2018			
Translation Report					
Map Report					
Place and Route Report					

Fig 16.b design summary of proposed GAL

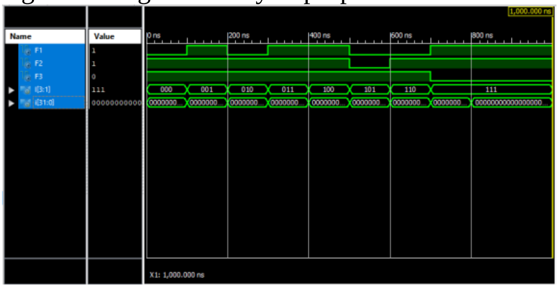


Fig 16.c Simulation output of proposed GAL

The simulated output will be same for all the three devices, which gives clear assistance that the circuits are operating with genuine performance.

The Reversible circuit elements used in designing PLDs are analyzed in terms of Quantum cost and Garbage outputs.

CONCLUSION

In this paper, the method of realizing different Boolean functions using reversible PLDs like reversible PAL, reversible PLA, reversible GAL are explained. These circuits are designed for minimum quantum cost and minimum garbage outputs. To overcome the fan-out limitation in reversible logic circuits the concept of duplicating the single output to required number of output lines is implemented by using additional reversible combinational circuits. The reversible PLDs are implemented using a reversible fuse and Feynman gate. The reversible fuse is made of CNOT gate and fredkin gate which is used for programmable connections. Feynman gate with second input made zero is used for fixed connections. The reversible circuits used for designing programmable connections and fixed connections give minimum heat dissipation. By using this reversible PLDs eighty percentage of efficiency can be acquired in terms of heat dissipation. The time delay increases a little when compared to irreversible PLDs which can be termed as a disadvantage but it is negligible. The time delays for reversible PAL, PLA and GAL are 5.847nsec, 5.847nsec and 5.847nsec respectively. The time delay depends upon the Boolean expressions considered to program on the device. The time delay is the function of quantum cost. The quantum cost increases with increase in length of Boolean function, because the number of programmable reversible fuses and Feynman gates (fixed connectors) increases with increase in length of Boolean function. If quantum cost increases, the time delay also increases. The reversible PAL, GAL finds more advantages when compared to the reversible PAL and PROM, since both OR array and AND array are programmable. Because of using reversible decoder in PROM the quantum cost becomes less when compared to the remaining PLDs. The propagation delay can be reduced if the quantum cost of the circuit is reduced further more. This can be termed as future scope for this paper.

REFERENCES

1) R. Landauer, “Irreversibility and Heat

Generation in the Computational Process", IBM Journal of Research and Development, 5, pp. 183-191, 1961.
2) C.H. Bennett, "Logical Reversibility of Computation", IBM J.Research and Development, 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, 1998.
4) R. Feynman," quantum mechanical computers:, Optic News, vol. 11,pp 11-20, 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/subtractor" arXiv.org/1009.6218,2010. J.M. Rabaey and M. Pedram, "Low Power Design Methodologies," Kluwer Academic Publisher, 1997.
7) T. Toffoli., "Reversible Computing", Tech memo MIT/LCS/TM-151, MIT Lab for Computer Science 1980.
8) 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.
9) Thapliyal H, Ranganathan N.," Design of Reversible Latches Optimized for Quantum Cost, Delay and Garbage Outputs" Centre for VLSI and Embedded .