IOSR Journal of Electronics and Communication Engineering (IOSR-JECE)
e-ISSN: 2278-2834,p- ISSN: 2278-8735.Volume 9, Issue 3, Ver. VI (May - Jun. 2014), PP 43-50
www.iosrjournals.org

Reversible Multiplier with Peres Gate and Full Adder.

[1]Prof. V M Sakode, [2]Prof A D Morankar
[1]*Government Polytechnic, Gadchiroli (INDIA)*
[2]*Visvesveraya National Institute of Technology,Nagpur (INDIA)*

Abstract: *Low Power dissipation and smaller area are one of the important factors while designing multipliers for digital circuits. As multipliers used in digital circuits dissipate large amount of heat whenever there is a transition of bits. Reversible Logic has emerged as a promising technology in reducing power dissipation. It has application in various fields such as low power VLSI, Quantum computing and Nano-technology. Hence, In this paper we try to design a reversible multiplier using Peres gate and Full adder. We show that, proposed multiplier is efficient in terms area, power and delay. Furthermore, it requires fewer garbage outputs and constant inputs. Proposed reversible multiplier is implemented using VHDL, simulated and synthesized on Xilinx 13.1 tool.*

Keywords: *Constant input, Digital circuits, Full adder, Garbage output, Multipliers, Reversible Logic, Peres gate. VHDL, Xilinx.*

I. Introduction

Multiplication is heavily used algorithm in many computational units. It is vital for processors to have low power multipliers. Multipliers play an important role in many high performance systems such as Microprocessors, Digital Signal Processing, Fir filters etc. Performance of any system is generally determined by the performance of multiplication unit. Because multiplication is a most power consuming and slowest process in any computational unit. Hence designing a efficient multiplier in terms of area, power and delay is a major objective in this work.

Multipliers designed with reversible logic are one of the best solution to reduce power dissipation. As demonstrated by R.Landauer in the early 1960, irreversible hardware computation regardless of its realization technique, results in energy dissipation due to the information loss [1].It is proved that the loss of each one bit of information dissipates at least KTln2 joules of energy (heat), where K=Boltzmann's constant and T=absolute temperature at which operation is performed[1]. In .1973, Bennet showed that in order to avoid KTln2 joules of energy dissipation in a circuit, it must be built using reversible logic gates [2]. In this paper we try to design a reversible multiplier with Peres gate and Full adder.

II. Basic Definitions Related To Reversible Logic

An n input n output function is said to be reversible if there is one to one correspondence (mapping) between inputs and outputs. Hence, in any reversible system number of inputs is always equal to number of outputs. Also the input vector can always be determined from its output vector [4-6]

A) **Reversible Function** :

A multiple output Boolean function (x1: x2: xn) of n Boolean variables is called as reversible if

1) The number of outputs is equal to number of inputs.
2) If reversible function performs permutations of set of input vectors.

B) **Reversible Logic Gate** :

Reversible Logic gates are circuits in which number of inputs is equal to number of outputs and there is one to one mapping between inputs and outputs[3-5].It not only helps us to determine output vectors from input vectors but also we can uniquely determine input vectors from its output vectors[4-6].

C) **Constant Inputs** :

This refers to number of inputs that has to be maintained constant at 1 or 0 in order to synthesize the given logical function [7].

D) **Garbage Outputs** :

The output that is added to an n x k function to make it reversible is called as garbage output[6].

The following simple formula shows the relation between constant inputs and garbage outputs :

(Input + Constant inputs) = (Output + Garbage Outputs).

E) **Quantum Cost** :

Quantum cost refers to the cost of the circuit in terms of the cost of primitive gate. It is calculated knowing the number of primitive reversible logic gates (1*1 or 2*2) required to realize the circuit. The quantum cost of a circuit is the minimum number of 2*2 unitary gates to represent the circuit keeping the output unchanged. The quantum cost of a 1*1 gate is 0 and that of 2*2 gates is unity [8].

F) **Flexibility** :

Flexibility refers to the universal nature of a reversible logic gate in realizing more functions [9].

G) **Gate Level** :

It refers to number of levels required to realize the given function.

H) **Hardware Complexity** :

This refers to total number of logic operations in a circuit, i.e. total number of AND,OR EXOR operations in a circuit[10-11].

I) **Design Constraints for Reversible Logic Circuits** :

The following are the important design constraints for reversible logic circuits:

a) Reversible Logic circuits do not allow fan-outs.
b) Should use minimum number of constant inputs.
c) Should use minimum number of garbage outputs.
d) Should use minimum gate levels or logic depth.
e) Should have minimum quantum cost.

.

III. Reversible Logic Gates

There are many number of reversible logic gates that exist at present. The quantum cost of each reversible logic gate is an important optimization parameter [12]. The quantum cost of a 1x1 reversible gate is assumed to be zero while the quantum cost of a 2x2 reversible logic gate is taken as unity. The quantum cost of other reversible gates is calculated by counting the number of V, V+ and CNOT gates present in their circuit. V is the square root of NOT gate and V+ is its Hermit an.

An n x n reversible gate can be represented as:

Iv = (I1, I2, I3…In)
Ov = (O1, O2, O3….On)

Where Iv and Ov are input and output vectors respectively.

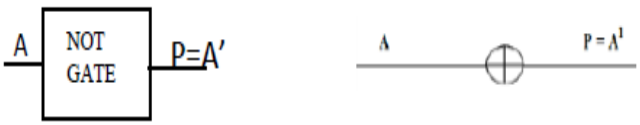
The V and V+ quantum gates have the following properties:

V * V = NOT ……. (1)
V * V+ = V+ * V = 1 ……. (2)
V+ * V+ = NOT ……. (3)

Some of the important reversible logic gates are,

1) **NOT Gate**

The simplest Reversible gate is NOT gate and is a 1*1 gate [13]. The Reversible 1*1 gate is NOT Gate with zero Quantum Cost is as shown in the Figure 1.



(a) Block diagram (b) symbol

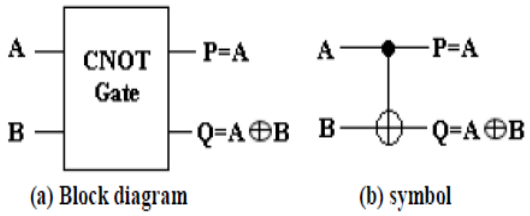
Figure1. NOT gate

2) **CNOT GATE**

CNOT gate is also known as controlled-not gate. It is a 2*2 reversible gate. The CNOT gate can be described as:

Iv = (A, B) ; Ov = (P= A, Q= A B)

Iv and Ov are input and output vectors respectively. Quantum cost of CNOT gate is 1[14]. Figure 2 shows a 2*2 CNOT gate and its symbol.

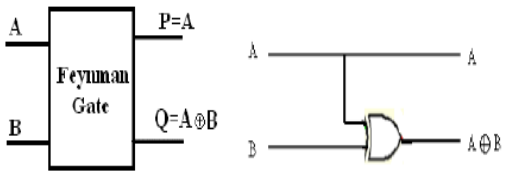


(a) Block diagram (b) symbol

Figure 2: CNOT Gate

3) **FEYNMAN Gate**

The Feynman gate which is a 2*2 gate and is also called as Controlled NOT and it is widely used for fan-out purposes. The inputs (A, B) and outputs P=A, Q= A XOR B. It has Quantum cost one [15]



(a) Block diagram (b) Logic circuit

Figure 3: FEYNMAN Gate

4) **Double Feynman Gate (F2G**)

It is a 3*3 Double Feynman gate [16].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=AÅB, R=AÅC. Quantum cost of Double Feynman gate is 2.

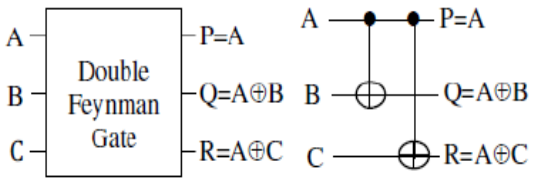
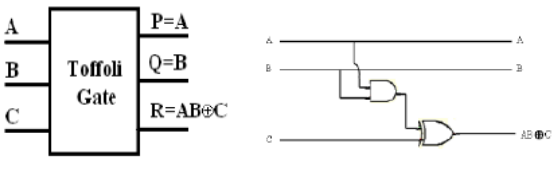


Figure 4: Double Feynman gate.

5) **TOFFOLI Gate:**

TOFFOLI gate which is a 3*3 gate with inputs (A, B, C) and outputs P=A,Q=B, R=AB XOR C. It has Quantum cost five [17].

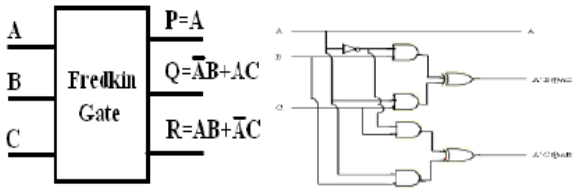


(a) Block diagram (b) Logic circuit

Figure 5: Toffoli Gate

6) **FREDKIN Gate:**

Fredkin gate which is a 3*3 gate with inputs (A, B, C) and outputs P=A, Q=A'B+AC, R=AB+A'C. It has Quantum cost five [18].

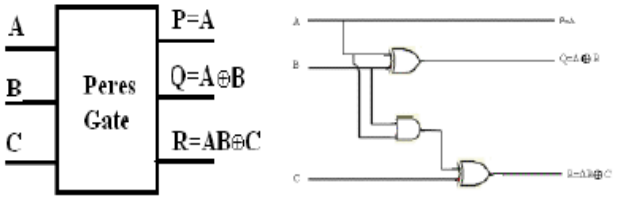


(a) Block diagram (b) Logic circuit

Figure 6: Fredkin Gate

7) **PERES Gate:**

Peres gate which is a 3*3 gate having inputs (A, B, C) and outputs P = A; Q = A XOR B; R = AB XOR C. It has Quantum cost four [19].



(a) Block diagram (b) Logic circuit

Figure 7: Peres Gate

IV. Proposed Reversible Multiplier

General multiplier is basically based on two steps:

a) Generation of partial products

b) Addition of partial products.

Partial Product Generation				x	x_3 y_3	x_2 y_2	x_1 y_1	x_0 y_0
					x_3y_0	x_2y_0	x_1y_0	x_0y_0
				x_3y_1	x_2y_1	x_1y_1	x_0y_1	
Multi Operand Addition			x_3y_2	x_2y_2	x_1y_2	x_0y_2		
		x_3y_3	x_2y_3	x_1y_3	x_0y_3			
	P_7	P_6	P_5	P_4	P_3	P_2	P_1	P_0

The operation of the partial products can be generated in parallel using 16 Peres gates as shown in Fig.8 because of lower hardware complexity, we use Peres gate instead of other reversible gates. This structure is proposed in [20].

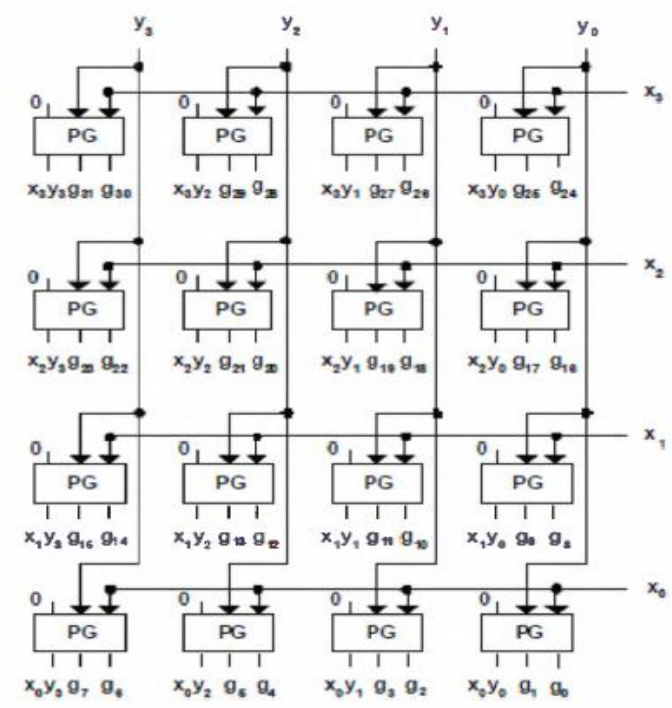


Fig.8 Partial Products Generation (PPG) using Peres gate

The purpose of this paper is the design of reversible multiplier using VHDL, with the aim of optimizing its hardware complexity to make it more economical in terms of number of garbage Therefore we have used a Full adder having zero garbage outputs and zero constant inputs. The proposed full adder will be of great help in reducing the garbage outputs and constant input parameters.

Output variable Sum in a Full Adder can be implemented using simple XOR gates as shown in fig 9.

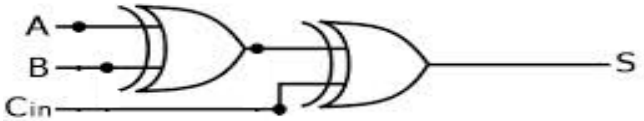


Fig.9 Gate Implementation of SUM in Full Adder.

The carry of the next stage in Full Adder can be expressed as:

$Q = \overline{(\overline{AB})\ (\overline{BC})\ (\overline{AC})}$ as shown in fig 10.

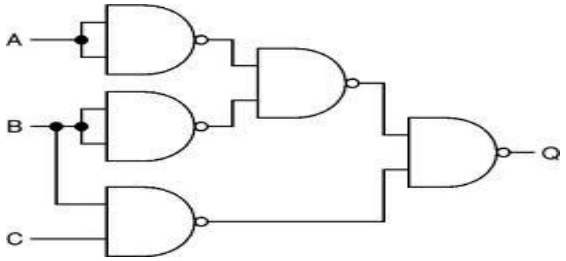


Fig.10 Gate Implementation of Output Carry in Full Adder.

NAND gates are used for faster carry propagation and help us to improve speed.

Proposed reversible uses eight full adders that they produce zero garbage outputs and zero constant inputs. In addition, it needs four reversible adders. We use Peres gate as reversible half adder, because it has less hardware complexity. Addition of partial products requires 8 full adders and 4 half adders as shown in fig 11

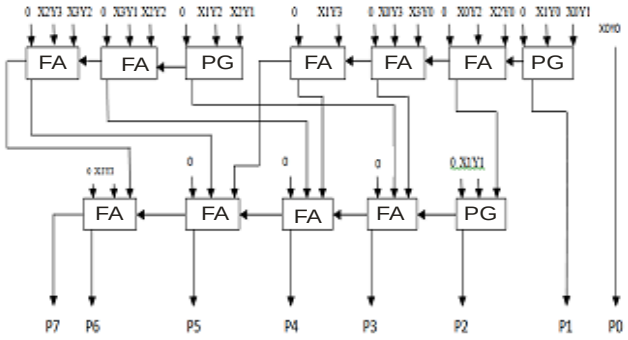


Fig.11 Reversible multiplier architecture (4bit) where outputs of PPG are input of full adder and Peres gate (half adder)

Proposed 4 bit reversible multiplier can be generalized for 8 bit and higher order multiplication using following algorithm (fig 12) .

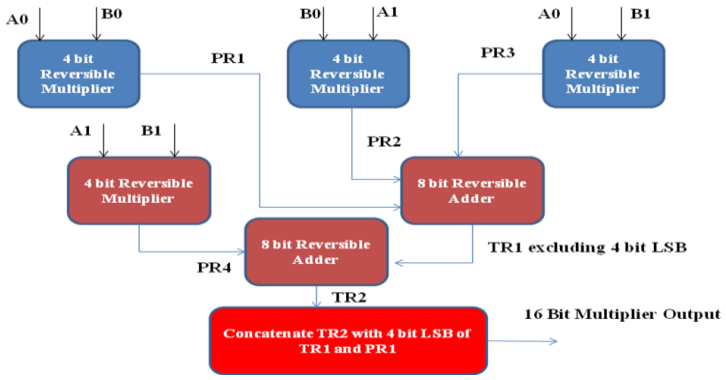


Fig.12 8 bit multiplier using 4 bit multiplier.

V. Results.

Complete code of reversible multiplier (4bit, 8bit and 16bit) is
Written using VHDL, simulated using Xilinx ISE 13.1
Simulator and synthesized using Xilinx synthesis tool (XST)
13.1. Power analysis performed using Xilinx X Power
Analyzer. Target FPGA used belongs to SPARTAN 3 families, XC3S50 device, VQ100 package, speed grade - 5.
Fig. 13, 14, 15 shows the simulation results, device utilization
Summary and power analysis respectively.

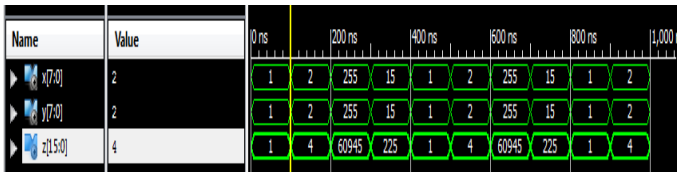
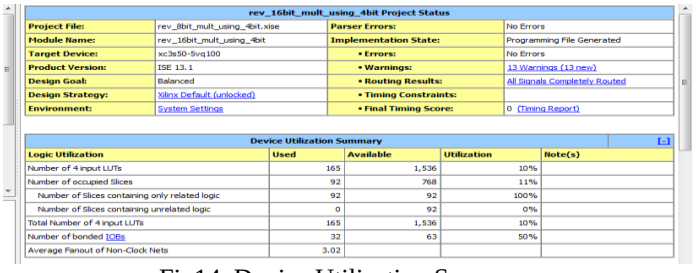


Fig13.Simulation Results



rev_16bit_mult_using_4bit Project Status

Project File:	rev_8bit_mult_using_4bit.xise	Parser Errors:	No Errors
Module Name:	rev_16bit_mult_using_4bit	Implementation State:	Programming File Generated
Target Device:	xc3s50-5vq100	• Errors:	No Errors
Product Version:	ISE 13.1	• Warnings:	13 Warnings (13 new)
Design Goal:	Balanced	• Routing Results:	All Signals Completely Routed
Design Strategy:	Xilinx Default (unlocked)	• Timing Constraints:	
Environment:	System Settings	• Final Timing Score:	0 (Timing Report)

Device Utilization Summary

Logic Utilization	Used	Available	Utilization	Note(s)
Number of 4 input LUTs	165	1,536	10%	
Number of occupied Slices	92	768	11%	
Number of Slices containing only related logic	92	92	100%	
Number of Slices containing unrelated logic	0	92	0%	
Total Number of 4 input LUTs	165	1,536	10%	
Number of bonded IOBs	32	63	50%	
Average Fanout of Non-Clock Nets	3.02			

Fig14. Device Utilization Summary

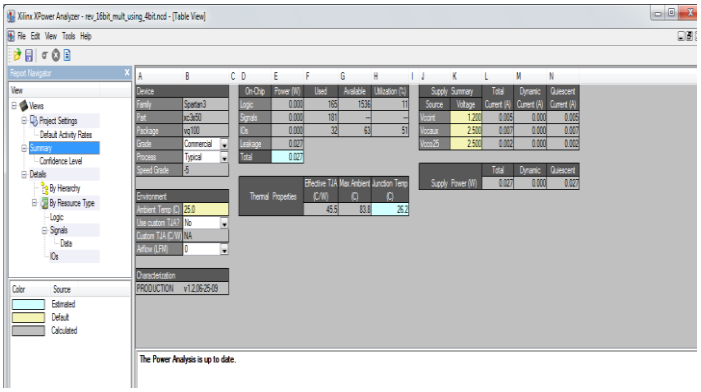


Fig15. Power Analysis

Table 1: Results of 4bit, 8bit and 16bit reversible multipliers.					
Parameter	**Delay**	**Power**	**Slices Used**	**4 Input LUT**	**Bonded IOBs**
4 Bit	13.65ns	16mw	18	29	16
8 Bit	29.23ns	29mw	95	165	32
16 Bit	42.45ns	38.20mw	127	254	46.25
Table 2: Comparison of Reversible and Conventional Multipliers.					
Parameter	**Delay**	**Power**	**Slices Used**	**4 Input LUT**	**Bonded IOBs**
4 Bit Rev. Multiplier	13.65ns	16mw	18	29	16
4 Bit Conv. Multiplier	16.32ns	27mw	18	32	16

Table 3: Comparison of different 4bit reversible multipliers						
Paper No.	**NoG**	**Gout**	**Gin**	**QC**	**No. of Logical Calculations**	**CC**
This Work	28	20	20	80	16×(2a+1b)(for PG)+ 8×(2a+5b+5c) (for PFAG)+4×(2a+1b) (for PG)	56a + 60b +40c
[21] First Design	28	28	28	128	16×(2a+1b)(for PG)+ 8×(5a+2b) (for PFAG)+4×(2a+1b) (for PG)	80a+36b
[21] Second Design	28	28	28	153	9×(1a+1b)(for TG)+7×(2a+1b)(for PG)+8×(5a+ 2b)(for PFAG)+4× (2a+1b) (for PG)	71a+36b
[22]	28	28	28	137	9×(1a+1b)(for TG)+7 × (2a+1b)(for PG)+8× (5a+2b)(for HNG)+4× (2a+1b)(for PG)	71a+36b
[23]	44	52	44	160	(80+16)a+36b	96a+36b
[24]	44	52	44	160	(80+16)a+36b	96a+36b
[25]	44	56	48	236	(92+16)a+52b+36d	108a+52b+36d
[26]	45	58	50	278	(110+16)a+103b+71d	126a+103b+71d
[27]	56	56	47	220	(80+16)a+100b+68d	96a+100b+68d

We define:
a=A two input XOR calculation.
b=A two input AND gate calculation.
c=A NOT gate calculation.
NoG = Number of Logic Gates.
Gout = Number of Garbage outputs.
Gin = Number of Garbage Inputs.
QC = Quantum Cost.
CC = Circuit Cost.

As shown in table3, our proposed design requires only 28 reversible gates and is also optimized in terms of number of garbage outputs, number of constant inputs and hardware complexity.

VI. Conclusion.

Table 2 shows that proposed reversible multiplier using Peres gate and Full Adder is better in terms of area power and delay as compared with conventional multiplier. Furthermore, Table 3 shows that it is efficient in terms of number of garbage outputs, constant inputs and quantum cost as compared with other reversible multipliers. However, we need synthesizing methods to minimize number of garbage outputs and constant inputs. We are short of simulation, synthesis, testing and verifying tools for designing reversible logic. But researchers around the world have been doing exciting research and making progress in this direction. Soon, reversible computing will become promising technology in near future.

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] Kerntopf, P., M.A. Perkowski and M.H.A. Khan, 2004. On universality of general reversible multiple valued logic gates, IEEE Proceeding of the 34th international symposium on multiple valued logic (ISMVL‟04), pp: 68-73.

[4] Perkowski, M., A. Al-Rabadi, P. Kerntopf, A.Buller, M. Chrzanowska-Jeske, A. Mishchenko, M. Azad Khan, A. Coppola, S. Yanushkevich, V. Shmerko and L. Jozwiak, 2001. A general decomposition for reversible logic, Proc. RM‟2001, Starkville, pp: 119-138.

[5] Perkowski, M. and P. Kerntopf, 2001. Reversible Logic. Invited tutorial, Proc. EURO-MICRO, Sept 2001, Warsaw, Poland. G. Eason, B. Noble, and I. N. Sneddon, "On certain integrals of Lipschitz-Hankel type involving products of Bessel functions," Phil. Trans. Roy. Soc. London, vol. A247, pp. 529–551, April 1955. (references)

[6] Thapliyal Himanshu, and M.B. Srinivas, 2005.Novel reversible TSG gate and its application for designing reversible carry look ahead adder and other adder architectures, Proceedings of the 10th Asia-Pacific Computer Systems Architecture Conference (ACSAC 05). Lecture Notes of Computer Science, Springer-Verlag, 3740: 775-786.

[7] Saiful Islam, M.D. and M.D. Rafiqul Islam, 2005.Minimization of reversible adder circuits. Asian J. Inform. Tech., 4 (12): 1146-1151.

[8] J.Smoline and David P.DiVincenzo, ―Five two-qubit gates are sufficient to implement the quantum fredkin gate‖, Physics Review A, vol. 53, no.4, pp. 2855-2856,1996.

[9] H.R.Bhagyalakshmi, M.K.Venkatesha,‖ Optimized Reversible BCD adder using new Reversible Logic Gates‖, Journal of Computing, Vol 2, Issue 2, February 2010.

[10] M. Haghparast,K. Navi, ‖A Novel Reversible BCD Adder For Nanotechnology Based Systems‖, American Journal of Applied Sciences 5 (3): 282-288, 2008,ISSN 1546-9239.

[11] Abu Sadat Md. Sayem, Masashi Ueda.‖ Optimization of reversible sequential circuits‖. Journal ofComputing, Volume 2, Issue 6, June 2010, ISSN 2151-9617.

[12] M. Mohammadi and M. Eshghi. On figures of merit in reversible and quantum logic designs. Quantum Information Processing, 8(4):297–318, Aug. 2009.

[13] Himanshu Thapliyal, Nagarajan Ranganathan, ―A New Reversible Design of BCD Adder‖ 978-3-9810801-7-9/DATE11/@2011 EDAA

[14] R. Feynman,‖ Quantum Mechanical Computers‖, Optic News, Vol 11, pp 11-20 1985.

[15] Md. Belayet Ali , Md. Mosharof Hossin and Md. Eneyat Ullah, ―Design of Reversible Sequential Circuit UsingReversible Logic Synthesis‖ International Journal of VLSI design & Communication Systems (VLSICS) Vol.2, No.4, December 2011

[16] T.Toffoli,‖Reversible Computing‖ Tech memoMIT/LCS/TM-151, MIT Lab for Computer Science 1980.

[17] Fredkin E. Fredkin and T. Toffoli,‖Conservative Logic‖, Int'l J. Theoretical Physics Vol 21, pp.219-253, 1982

[18] Peres, ―Reversible Logic and Quantum Computers‖, Physical review A, 32:3266- 3276, 1985.

[19] Rakshith Saligram, Rakshith T.R, ―Novel Code Converter Employing Reversible logic‖,Volume 52– No.18, August 2012.

[20] M. Shams, M. Haghparast and K. Navi, "Novel reversible multiplier circuit in nanotechnology", World App!. Sci. 1., 3 (2008) 806.

[21] P. Moallem*a, M. Ehsanpourb "A Novel Design of Reversible Multiplier Circuit.

[22] Mohammadi, M., Navi, K. and Eshghi, M., "Optimized reversible multiplier circuit", Journal of Circuits, Systems and Computers, Vol. 18, No. 2, (2009), 311-323.

[23] Islam, M. S., Rahman, M. M., Begum, Z. and Hafiz, M. Z., "Low cost quantum realization of reversible multiplier circuit",**Information Technology Journal**, Vol. 8, No. 2, (2009), 208-213.

[24] Haghparast"Design of a novel reversible multiplier circuit using HNG gate in nanotechnology", World Applied Sciences Journal, Vol. 3, No. 6, (2008), 974-978).

[25] Shams, M., Haghparast, M., Navi, K., "Novel reversible multiplier circuit in nanotechnology" World Appllied Science Journal, Vol. 3, No. 5, (2008), 806-810.213

[26] Thaplyal, H. and Srinivas, M. B., "Novel reversible multiplier architecture using reversible TSG gate", IEEE International Conferecne on Computer Systems and Applications, (2006), 100-103.

[27] Thaplyal, H., Srinivas, M. B. and Arabnia, H. R., "A reversible version of 4´4 bit array multiplier with minimum gates and garbage outputs", International Conference on Embedded System and Applications (ESA'05), Las Vegag, USA, (2005), 106-114