International Journal of Ethics in Engineering & Management Education
Website: www.ijeee.in (ISSN: 2348-4748, Volume 2, Issue 8, August 2015)

A Novel Analysis of Sequential Circuits Design Using Reversible Logic

P Vijaya Lakshmi
M.Tech Student (VLSI-SD)
Dept.of.ECE, GNIT
vijayalakshmi.p21@gmail.com

Mr. P. Ranjith
Assistant Professor
Dept.of.ECE, GNIT
ranjith22kits@gmail.com

Mr. M Suman Kumar
Associate Professor
Dept.of.ECE, GNIT
sumankumar.gnit@gmail.com

Prof B Kedarnath
HOD-ECE
Dept.of.ECE, GNIT
hodece.gnit@gniindia.org

***Abstract*: Reversible logic design attracts more interest due to its low power consumption. A lot of research has been done in combinational as well as sequential design of reversible circuits. As far as it is known, this is the first attempt to apply reversible logic to synchronous counters design using T flip-flop. In this paper we have also proposed a new reversible gate which can be used as copying gate. We have proposed a reversible T-Flip-flop which is better than the existing designs by reducing number of gates, garbage outputs and power dissipation. We hope this paper will initiate a new area of research in the field of complex reversible sequential circuits for quantum computers.**

***Keywords:* reversible logic, low power gate, sequential circuits, GPUs**

1. INTRODUCTION

With the small and effective transistor at their hands, electrical engineers of the 50s saw the possibilities of constructing far more advanced circuits than before. However, as the complexity of the circuits grew, problems started arising. Another problem was the size of the circuits. A complex circuit, like a computer, was dependent on speed. If the components of the computer were too large or the wires interconnecting them too long, the electric signals couldn't travel fast enough through the circuit, thus making the computer too slow to be effective.

Jack Kilby at Texas Instruments found a solution to this problem in 1958. Kilby's idea was to make all the components and the chip out of the same block (monolith) of semiconductor material. When the rest of the workers returned from vacation, Kilby presented his new idea to his superiors. He was allowed to build a test version of his circuit. In September 1958, he had his first integrated circuit ready. Although the first integrated circuit was pretty crude and had some problems, the idea was groundbreaking. By making all the parts out of the same block of material and adding the metal needed to connect them as a layer on top of it, there was no more need for individual discrete components. No more wires and components had to be assembled manually. The circuits could be made smaller and the manufacturing process could be automated. From here the idea of integrating all components on a single silicon wafer came into existence and which led to development in Small Scale Integration(SSI) in early 1960s, Medium Scale Integration(MSI) in late 1960s, Large Scale Integration(LSI) and in early 1980s VLSI 10,000s of transistors on a chip (later 100,000s & now 1,000,000s).

The first semiconductor chips held two transistors each. Subsequent advances added more and more transistors, and, as a consequence, more individual functions or systems were integrated over time. The first integrated circuits held only a few devices, perhaps as many as ten diodes, transistors, resistors and capacitors, making it possible to fabricate one or more logic gates on a single device. Now known retrospectively as small-scale integration (SSI), improvements in technique led to devices with hundreds of logic gates, known as medium-scale integration (MSI). Further improvements led to large-scale integration (LSI), i.e. systems with at least a thousand logic gates. Current technology has moved far past this mark and today's microprocessors have many millions of gates and billions of individual transistors.

At one time, there was an effort to name and calibrate various levels of large-scale integration above VLSI. Terms like ultra-large-scale integration (ULSI) were used. But the huge number of gates and transistors available on common devices has rendered such fine distinctions moot. Terms suggesting greater than VLSI levels of integration are no longer in widespread use.

As of early 2008, billion-transistor processors are commercially available. This is expected to become more commonplace as semiconductor fabrication moves from the current generation of 65 nm processes to the next 45 nm generations (while experiencing new challenges such as increased variation across process corners). A notable example is Nvidia's 280 series GPU. This GPU is unique in the fact that almost all of its 1.4 billion transistors are used for logic, in contrast to the Itanium, whose large transistor count is largely due to its 24 MB L3 cache. Current designs, unlike the earliest devices, use extensive design automation and automated logic synthesis to lay out the transistors, enabling higher levels of complexity in the resulting logic functionality. Certain high-performance logic blocks like the SRAM (Static Random Access Memory) cell, however, are still designed by hand to ensure the highest efficiency (sometimes by bending or breaking established design rules to obtain the last bit of

International Journal of Ethics in Engineering & Management Education
Website: www.ijeee.in (ISSN: 2348-4748, Volume 2, Issue 8, August 2015)

performance by trading stability)[citation needed]. VLSI technology is moving towards radical level miniaturization with introduction of NEMS technology. A lot of problems need to be sorted out before the transition is actually made.

2. REVERSIBLE COMPUTING

Reversible computing is a model of computing where the computational process to some extent is reversible, i.e., time-invertible. A necessary condition for reversibility of a computational model is that the relation of the mapping states of transition functions to their successors should at all times be one-to-one. Reversible computing is generally considered an unconventional form of computing.

There are two major, closely related, types of reversibility that are of particular interest for this purpose: physical reversibility and logical reversibility.

A process is said to be physically reversible if it results in no increase in physical entropy; it is isentropic. These circuits are also referred to as charge recovery logic or adiabatic computing. Although in practice no non stationary physical process can be exactly physically reversible or isentropic, there is no known limit to the closeness with which we can approach perfect reversibility, in systems that are sufficiently well-isolated from interactions with unknown external environments, when the laws of physics describing the system's evolution are precisely known.

Probably the largest motivation for the study of technologies aimed at actually implementing reversible computing is that they offer what is predicted to be the only potential way to improve the energy efficiency of computers beyond the fundamental von Neumann-Landauer limit[2] of kT ln(2) energy dissipated per irreversible bit operation.

As was first argued by Rolf Landauer of IBM, in order for a computational process to be physically reversible, it must also be logically reversible. Landauer's principle is the loosely formulated notion that the erasure of n bits of information must always incur a cost of nk ln(2) in thermodynamic entropy. A discrete, deterministic computational process is said to be logically reversible if the transition function that maps old computational states to new ones is a one-to-one function; i.e. the output logical states uniquely defines the input logical states of the computational operation.

For computational processes that are nondeterministic (in the sense of being probabilistic or random), the relation between old and new states is not a single-valued function, and the requirement needed to obtain physical reversibility becomes a slightly weaker condition, namely that the size of a given ensemble of possible initial computational states does not decrease, on average, as the computation proceeds forwards.

The reversibility of physics and reversible computing Landauer's principle (and indeed, the second law of thermodynamics itself) can also be understood to be a direct logical consequence of the underlying reversibility of physics, as is reflected in the general Hamiltonian formulation of mechanics, and in the unitary time-evolution operator of quantum mechanics more specifically.

In the context of reversible physics, the phenomenon of entropy increase (and the observed arrow of time) can be understood to be consequences of the fact that our evolved predictive capabilities are rather limited, and cannot keep perfect track of the exact reversible evolution of complex physical systems, especially since these systems are never perfectly isolated from an unknown external environment, and even the laws of physics themselves are still not known with complete precision. Thus, we (and physical observers generally) always accumulate some uncertainty about the state of physical systems, even if the system's true underlying dynamics is a perfectly reversible one that is subject to no entropy increase if viewed from a hypothetical omniscient perspective in which the dynamical laws are precisely known.The implementation of reversible computing thus amounts to learning how to characterize and control the physical dynamics of mechanisms to carry out desired computational operations so precisely that we can accumulate a negligible total amount of uncertainty regarding the complete physical state of the mechanism, per each logic operation that is performed. In other words, we would need to precisely track the state of the active energy that is involved in carrying out computational operations within the machine, and design the machine in such a way that the majority of this energy is recovered in an organized form that can be reused for subsequent operations, rather than being permitted to dissipate into the form of heat.

Although achieving this goal presents a significant challenge for the design, manufacturing, and characterization of ultra-precise new physical mechanisms for computing, there is at present no fundamental reason to think that this goal cannot eventually be accomplished, allowing us to someday build computers that generate much less than 1 bit's worth of physical entropy (and dissipate much less than kT ln 2 energy to heat) for each useful logical operation that they carry out internally.

The motivation behind much of the research that has been done in reversible computing was the first seminal paper on the topic, which was published by Charles H. Bennett of IBM research in 1973. Today, the field has a substantial body of academic literature behind it. A wide variety of reversible device concepts, logic gates, electronic circuits, processor architectures, programming languages, and application algorithms have been designed and analyzed by physicists, electrical engineers, and computer scientists.

This field of research awaits the detailed development of a high-quality, cost-effective, nearly reversible logic device technology, one that includes highly energy-efficient clocking and synchronization mechanisms. This sort of solid engineering progress will be needed before

International Journal of Ethics in Engineering & Management Education
Website: www.ijeee.in (ISSN: 2348-4748, Volume 2, Issue 8, August 2015)

the large body of theoretical research on reversible computing can find practical application in enabling real computer technology to circumvent the various near-term barriers to its energy efficiency, including the von Neumann-Landauer bound. This may only be circumvented by the use of logically reversible computing, due to the Second Law of Thermodynamics.

This section describes the reversible logic gates that are being used in the design. Fig. 1 shows a Feynman Gate. Feynman Gate (FG) can be used as a copying gate. Since a fan-out is not allowed in reversible logic, this gate is useful for duplication of the required outputs. Fig. 2 shows Sayem Gate . A single Sayem Gate (SG) can be used to realize the function of D-Latch. Fig. 3 shows the Fredkin Gate (FRG) .This is the most widely used reversible gate. Fig. 4 shows a Peres Gate (PG). It is also known as New Toffoli Gate (NTG). Functionally Peres Gate is equal with the transformation produced by a Toffoli Gate followed by Feynman Gate.

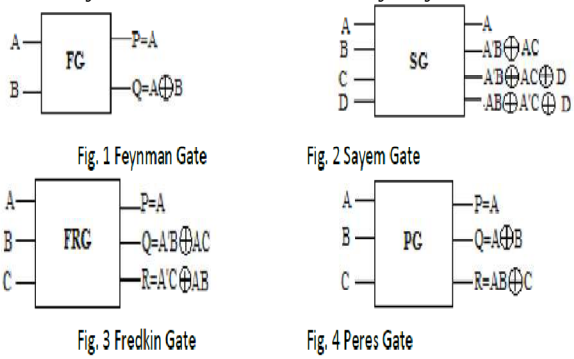


Fig.1. Reversible Logic Gates

3. REVERSIBLE CIRCUITS

To implement reversible computation, estimate its cost, and to judge its limits, it is formalized it in terms of gate-level circuits. For example, the inverter (logic gate) (NOT) gate is reversible because it can be undone. The exclusive or (XOR) gate is irreversible because its inputs cannot be unambiguously reconstructed from an output value. However, a reversible version of the XOR gate—the controlled NOT gate (CNOT)—can be defined by preserving one of the inputs. The three-input variant of the CNOT gate is called the Toffoli gate. It preserves two of its inputs a,b and replaces the third c by c\oplus (a\cdot b). With c=0, this gives the AND function, and with a\cdot b=1 this gives the NOT function. Thus, the Toffoli gate is universal and can implement any reversible Boolean function (given enough zero-initialized ancillary bits). More generally, reversible gates have the same number of inputs and outputs. A reversible circuit connects reversible gates without fanouts and loops. Therefore, such circuits contain equal numbers of input and output wires, each going through an entire circuit.

Reversible logic circuits have been first motivated in the 1960s by theoretical considerations of zero-energy computation as well as practical improvement of bit-manipulation transforms in cryptography and computer graphics. Since the 1980s, reversible circuits have attracted interest as components of quantum algorithms, and more recently in photonic and nano-computing technologies where some switching devices offer no signal gain.

Surveys of reversible circuits, their construction and optimization as well as recent research challenges is available

4. SOME OF REVERSIBLE GATES:

FEYNMAN GATE: Feynman gate is also known as CNOT (Controlled Not) gate. The two key reasons to use this gate in reversible circuit are: i)make the copy of an input (putting any of the input a constant 0) and ii)to invert an input bit (putting any of the input a constant 1).

FREDKIN GATE: The Fredkin gate (also CSWAP gate) is a computational circuit suitable for reversible computing, invented by Ed Fredkin. It is universal, which means that any logical or arithmetic operation can be constructed entirely of Fredkin gates. The Fredkin gate is the three-bit gate that swaps the last two bits if the first bit is 1.

5. PROPOSED REVERSIBLE GATE

We have proposed a new conservative reversible gate named RSJ Gate. This is a 2 through 4x4 reversible gates. The block diagram of the proposed gate is shown in Fig. 5. Its corresponding Truth Table is shown in Table I. From this truth table we can verify that the input and output vectors are unique which satisfies the condition of reversibility.

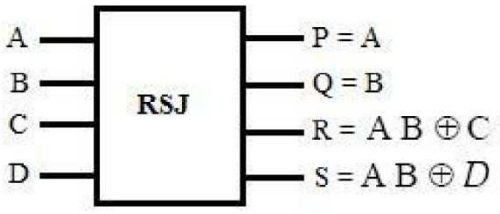


Fig.2. Proposed Rsj Gate

Table 3.1 Truth Table Of The Proposed Reversible Gate

A	B	C	D	P	Q
0	0	0	0	0	0
0	0	0	1	0	0
0	0	1	0	0	0
0	0	1	1	0	0
0	1	0	0	0	1
0	1	0	1	0	1

International Journal of Ethics in Engineering & Management Education
Website: www.ijeee.in (ISSN: 2348-4748, Volume 2, Issue 8, August 2015)

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

6. PROPOSED REVERSIBLE POSITIVE EDGE TRIGGERED

T -FLIPFLOP

A flip-flop is a bi-stable electronic circuit that has two stable states and can be used as a one-bit memory device. In this section we propose the construction of a Master-Slave T Flip-flop using reversible gates. The truth table of the T Flip-flop is given in Table II. The reversible design is shown in Fig. 6 and the corresponding block diagram is shown in Fig. 7. The reversible realization of T Flip-flop has two SG gates and one Feynman Gate. And it has two constant inputs and it produces three garbage outputs. The comparison of the proposed design with the existing designs is given in

Table Positive Edge Triggered T Flip-Flop

CLK	T	Q_{t-1}	Q
0	0	0	0
1	0	0	0
0	0	1	1
1	0	1	1
0	1	0	0
1	1	0	1
0	1	1	1
1	1	1	0

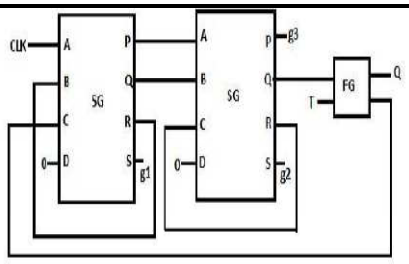


Fig. 6 Reversible Positive Edge Triggered T Flip-flop

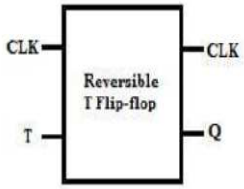


Fig. 7 Block Diagram

Fig.3. Reversible Positive Edge Triggered T -Flipflop

Comparison of Different T Flip-Flops with Only Q Output

	No. of Gates	Garbage Outputs	Constant Inputs
Existing	10	12	10
Existing	5	3	2
Existing	10	10	10
Proposed design	3	3	2

In the synchronous counters, the count pulses are applied directly to the control/CLK inputs of all the Flip-flops. Synchronous counters have regular pattern and can be constructed using flip-flops and gates.

7. PROPOSED 4-BIT SYNCHRONOUS DOWN-COUNTER

A conventional 4-bit Synchronous down Counter with count enable function can be realized as shown in Fig.8. The reversible design of the above 4-bit Synchronous down Counter is shown in Fig. 9. The proposed RSJ gates are used to produce the copy of the Q output of the T Flip-flops. The Peres gate is used to realize the AND function. The proposed reversible synchronous counter design contains 15 reversible gates, 13 constant inputs and produces 12 garbage outputs. The conventional 4-bit Synchronous Up/Down-Counter is shown in Fig.10. The reversible design of this 4-bit Synchronous Up/Down Counter is shown in Fig. 11. The proposed RSJ gates are used to produce the copy of the Q output of the T Flip-flops. The Peres gate is used to realize the AND function. The proposed reversible synchronous counter design contains 18 reversible gates, 18 constant inputs and produces 16 garbage outputs.

International Journal of Ethics in Engineering & Management Education
Website: www.ijeee.in (ISSN: 2348-4748, Volume 2, Issue 8, August 2015)

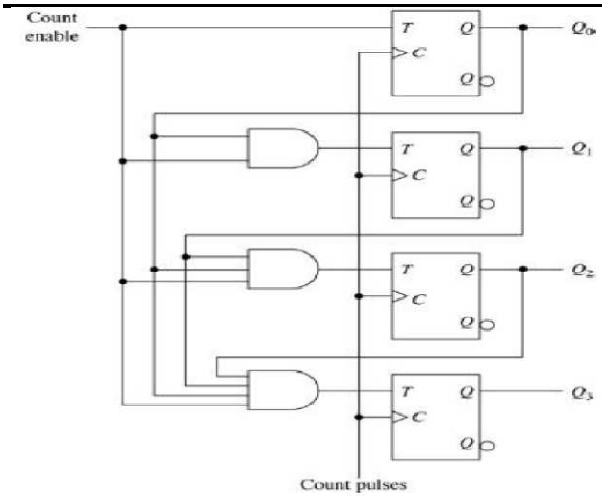


Fig. 8 Conventional 4-bit Synchronous Down-Counter

Conventional 4-Bit Synchronous Down-Counter

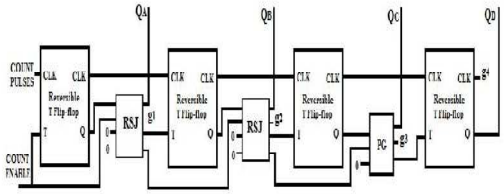


Fig. 9 Proposed 4-bit reversible synchronous down counter

Proposed 4-Bit Synchronous Up/Down Counter

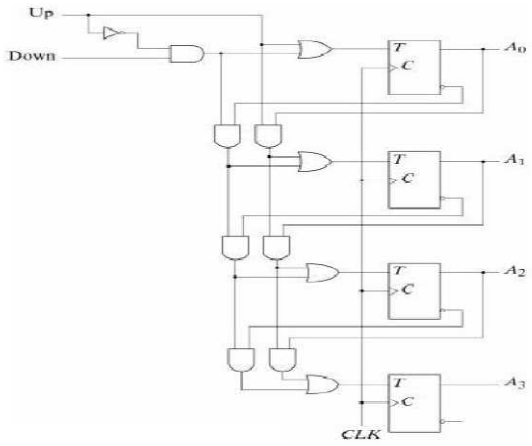


Fig. 10 Conventional 4-bit Synchronous Up/Down Counter

Result analysis:

RTL Top Level Output File Name : syncounter.ngr

Top Level Output File Name : syncounter

Output Format : NGC

Optimization Goal :Speed

Keep Hierarchy :NO

DESIGN STATISTICS

# IOs	:	10

Cell Usage:

# BELS	:	15
# LUT2	:	6
# LUT3	:	5
# LUT4	:	3
# MUXF5	:	1
# IO Buffers	:	10
# IBUF	:	2
# OBUF	:	8

Device utilization summary

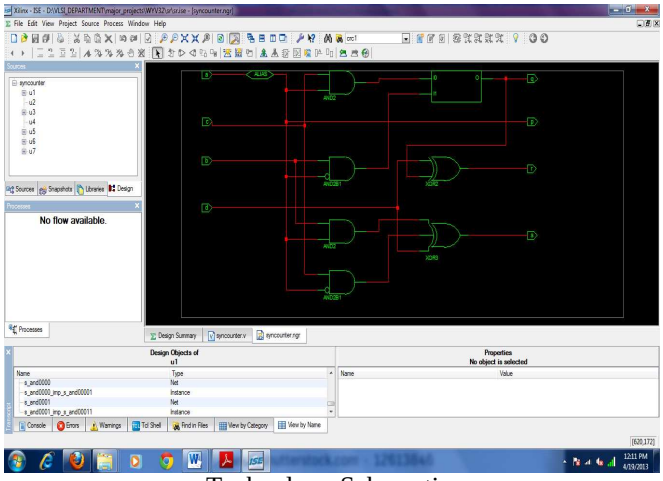
Selected Device : 3s200ft256-4

Number of Slices : 8 out of 1920 0%

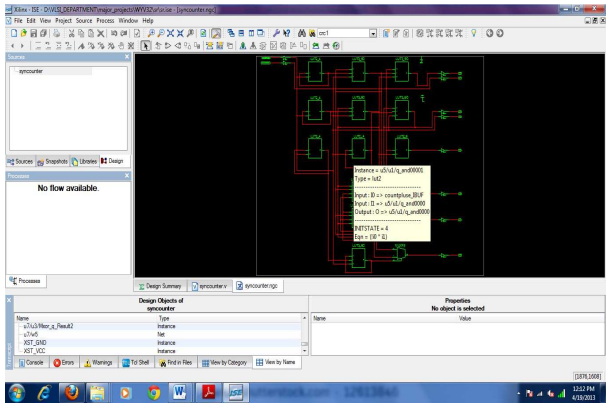
Number of 4 input LUTs : 14 out of 3840 0%

Number of IOs : 10

Number of bonded IOBs : 10 out of 173 5%

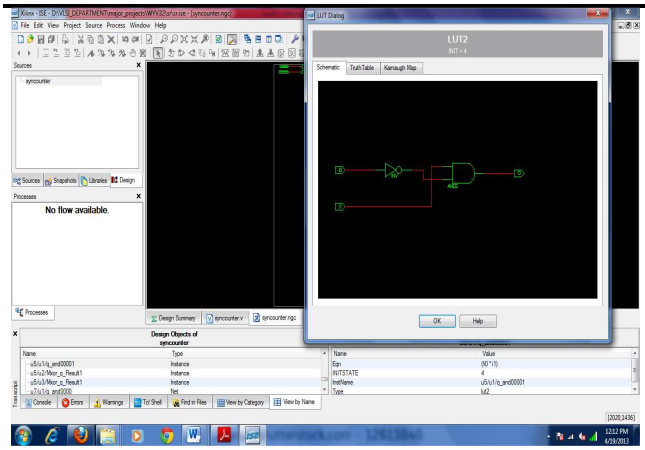


Technology Schematic



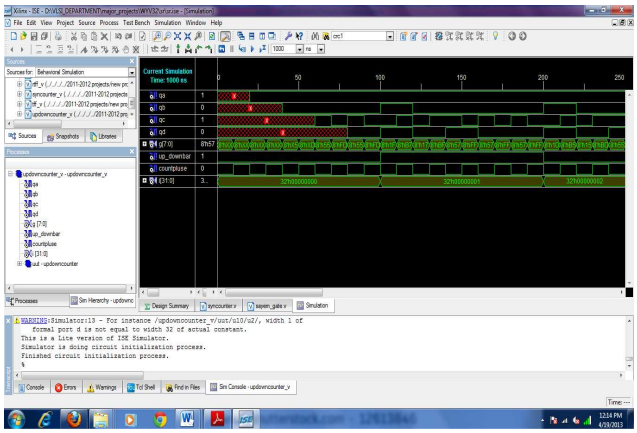
Technology detailed Schematic

International Journal of Ethics in Engineering & Management Education
Website: www.ijeee.in (ISSN: 2348-4748, Volume 2, Issue 8, August 2015)



Technology detailed Schematic with LUTS

SIMULATION



8. CONCLUSION

The key contribution of this paper is the reversible realization of 4-bit synchronous counters by using proposed reversible gates and the existing one. As far as it is known, this is the first attempt to apply reversible logic to synchronous counter design. We also have proposed a new conservative reversible gate. This gate can be used to produce multiple copies of a signal.

The proposed synchronous counter designs have the applications in building reversible ALU, reversible processor etc. This work forms an important move in building large and complex reversible sequential circuits for quantum computers. The future work could be to develop efficient reversible counters and reversible controller circuits.

REFERENCES

[1]. Landauer, R., "Irreversibility and heat generation in the computing process", IBM J. Research and Development, 5(3): pp. 183-191, 1961.
[2]. Bennett, C.H., "Logical reversibility of Computation", IBM J. Research and Development, 17: pp. 525-532, 1973.
[3]. G. Schrom, "Ultra-low-power CMOS Technology", PhD thesis, technschen Universitat Wien, June, 1998.
[4]. P. Picton, "Multi-valued sequential logic design using Fredkin gates," Multiple-Valued Logic Journal, vol. 1, pp. 241-251, 1996.
[5]. M. Z. Moghadam and K. Navi, "Ultra-area-efficient reversible multiplier," Microelectronics Journal, Vol. 43, no. 6, 377-385, 2012.
[6]. T. D. Nguyen, and R. Van Meter, "A Space-Efficient Design for Reversible Floating Point Adder in Quantum Computing", arXiv preprint arXiv:1306.3760, 2013.
[7]. H. Thapliyal, N. Ranganathan and S.Kotiyal, "Reversible Logic Based Design and Test of Field Coupled Nanocomputing Circuits", To Appear Springer Lecture Notes on Computer Science State-of the-Art-Survey Series Special Volume on Field-Coupled Nanocomputing, 2014.
[8]. H. Thapliyal, N. Ranganathan and S. Kotiyal, "Design of Testable Reversible Sequential Circuits ", IEEE Transactions on VLSI, vol. 21, no.7, pp.1201-1209, July 2013.
[9]. A. Peres, "Reversible logic and quantum computers," Phys. Rev. A, Gen. Phys., vol. 32, no. 6, pp. 3266–3276, Dec. 1985.
[10]. W. N. Hung, X. Song, G.Yang, J.Yang, and M. Perkowski, "Optimal synthesis of multiple output boolean functions using a set of quantum gates by symbolic reachability analysis," IEEE Trans. Computer-Aided Design, vol. 25, no. 9, pp. 1652–1663, Sept. 2006.
[11]. S. Kotiyal, H. Thapliyal, and N. Ranganathan, "Design of a reversible bidirectional barrel shifter," Proceedings of the 11th IEEE NANO, Portland, Aug. 2011, pp. 463–468.
[12]. E. Fredkin and T. Toffoli, "Conservative logic," International J. Theor.Physics, vol. 21, pp. 219–253, 1982.
[13]. R. Feynman, "Quantum Mechanical Computers," Optical News, pp. 11- 20, 1985.
[14]. M. A. Nielsen and I. L. Chuang, Quantum Computation and Quantum Information. New York: Cambridge Univ. Press, 2000.
[15]. J. A. Smolin and D. P. DiVincenzo, "Five two-bit quantum gates are sufficient to implement the quantum fredkin gate," Physical Review A, 53: pp.2855–2856, 1996.
[16]. H. R. Bhagyalakshmi and M. K. Venkatesha, "An improved design of a multiplier using reversible logic gates," Int. J. Eng. Sci. Technol. 2 (8), pp. 3838–3845, 2010.
[17]. M. Haghparast, M. Mohammadi, K. Navi, and M. Eshghi, "Optimized reversible multiplier circuit." Journal of Circuits, Systems & Computers, vol. 18, no. 2, pp. 311 – 323, 2009.
[18]. M. Haghparast, S. Jassbi, K. Navi, and O. Hashemipour, "Design of a novel reversible multiplier circuit using HNG gate in nanotechnology,"World Applied Sciences Journal , vol. 3, no. 6, pp. 974–978, 2008.