Designing of 2-Bit ALU using Reversible Gates on FPGA

Praveen Kumar[1], Prof. Puran Gour[2], Prof. Braj Bihari Soni[3]
[1]M-Tech Research Scholar, [2]Research Guide, [3]Co-Guide, Deptt. of E & C,
NRI Institute of Information Science & Technology, Bhopal

Abstract - Reversible logic is optimization of logic design on chip to minimize the delay in calculation and reduce the quantum cost so that the energy consumption of the chip should be minimized. The optimization of the design is the requirement of the VLSI design to increase the response of the signals and operations. To achieve this we are optimizing the use of various reversible gates to calculate various logic operations. In this paper we have designed architecture for 2-bits and compared with the 1-bit architecture and calculated the delay and found the 2-bit architecture having less delay.

Keywords- Reversible Gates, 2-bit architecture, delay and quantum cost .

I. INTRODUCTION

Reversible computing is a model of computing where the computational process to some extent is reversible, that is, time-invertible. In a computational model that uses transitions from one state of the abstract machine to another, a essential condition for reversibility is that the relation of the mapping from states to their successors must be each and every one. Reversible computing is normally considered an unconventional form of computing. There are two major, nearly associated, there are types of reversibility which 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 called as charge recovery logic, adiabatic computing or adiabatic circuits. Although in practice no non-stationary physical process can be exactly physically reversible or isentropic, there is no identified limit to the closeness with which we can approach perfect reversibility, in architecture that are adequately well-isolated from interactions with unknown external environments, as the laws of physics telling 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.

In most computing tasks, the number of output bits is relatively small compared to the number of input bits. As, in a decision problem, the output is only one bit (yes or no) and the input can be as large as desired. Though, computational tasks in digital signal processing, communication model, computer graphics, and cryptography necessitate that all of the information encoded in the input be preserved in the output. A number of those tasks are important enough to justify adding new microprocessor instructions to the HP PA-RISC (MAX and MAX-2), Sun SPARC (VIS), PowerPC (AltiVec), IA-32 and IA-64 (MMX) instruction sets [18, 13]. In particular, new bit-permutation instructions were shown to vastly improve performance of several standard methodologies, including matrix transposition and DES, in addition to two recent cryptographic algorithms Twofish and Serpent [13]. Bit permutations are a special case of reversible functions, functions that permute the set of possible input values.

It is a key element of Fast Fourier Transform algorithms and has been used in application-specific Xtensa processors from Tensilica. One may expect to get more speed-ups by adding instructions to allow computation of an arbitrary reversible function. The problem of chaining such instructions together provides one motivation for studying reversible computation and reversible logic circuits, logic circuits composed of gates computing reversible functions. Reversible circuits are too attractive because the loss of information associated with irreversibility implies energy loss [2]. Younis and Knight [22] showed that some reversible circuits can be made asymptotically energy-lossless as their delay is allowed to grow arbitrarily large.

Currently, energy losses due to irreversibility are dwarfed by the overall power dissipation, but this might be changed if power dissipation improves. Specifically, reversibility is

important for nanotechnologies where switching devices with gain are very difficult to design. Finally, reversible circuits can be viewed as a special case of quantum circuits because quantum evolution must be reversible [14]. Classical (non-quantum) reversible gates are subject to the same "circuit rules," whether they work on classical bits or quantum states. In detail, popular universal gate libraries for quantum computation often contain as subsets universal gate libraries for classical reversible computation. While the speed-ups that make quantum computing attractive are not available without purely quantum gate, the logic synthesis for classical reversible circuits is a first step toward synthesis of quantum circuits.

We review existing work on classical reversible circuits. Toffoli [20] gives constructions for an arbitrary reversible or irreversible function in terms of a certain gate library. However, his method makes use of a large number of temporary storage channels, i.e. input-output wire-pairs other than those on which the function is computed (also known as ancilla bits). Sasao and Kinoshita show that any conservative function (f (x) is conservative if x and f (x) always contain the same number of 1s in their binary expansions) has an implementation with only three temporary storage channels using a certain fixed library of conservative gates, although no explicit construction is given [16]. Kerntopf uses exhaustive search methods to examine small-scale synthesis problems and related theoretical questions about reversible circuit synthesis [9]. There has also been much recent work on synthesizing reversible circuits that implement non-reversible Boolean functions on some of their outputs, with the aim of providing the quantum phase shift operators needed by Grover's quantum search algorithm [8, 12, 21]. Some work on local optimization of such circuits via equivalences has also been done [12, 8]. In a different direction, group theory has recently been employed as a tool to analyze reversible logic gates [19] and investigate generators of the group of reversible gates [5].

II. Reversible Gates and Circuits

If arbitrary signals are allowed on the inputs, a required condition for reversibility is that the gate have the same number of input and output wires. If it has k inputs & output wires, it is known a k_k gate, or a gate on k wires. We would think of the mth input wire and the mth output wire as really being the similar wire. Numerous gates satisfying these conditions have been examined in the literature [15]. We will consider a specific set defined by Toffoli [20].

another definition is Clearly the k-CNOT gates are all reversible. The first three of these have unusual names. The 0- CNOT is just an inverter or NOT gate, and is signified by N. It performs the operation (x) !(x1), where denotes XOR. The 1-CNOT, which performs the operation (y;x)!(y;x_y) is referred to as a Controlled-NOT [7], or CNOT (C). The 2-CNOT is normally called a TOFFOLI (T) gate, and performs the operation (z;y;x) (z;y;x_yz). We will also be using another reversible gate, called the SWAP (S) gate. It is a 2_2 gate which exchanges the inputs; that is, (x;y) ! (y;x). One reason for choosing these particular gates is that they appear often in the quantum computing context, where no physical "wires" exist, and swapping two values needs non-trivial effort. [14]. We will be working with circuits from the given, limited-gate library. Generally, this will be the CNTS given gate library, consisting of CNOT, NOT Gate and TOFFOLI, and SWAP gates.

As with reversible gates, a reversible circuit has the same number of input and output wires; again we will call a reversible circuit with n inputs an n_n circuit, or a circuit on n wires. We draw reversible circuits as arrays of horizontal lines presenting wires. Gates are presented by vertically-oriented symbols. Such as, in Figure 1, we see a reversible circuit drawn in the notation introduced by Feynman [7]. The $\oplus$ symbols represent inverters and the $\bullet$ symbols represent controls.

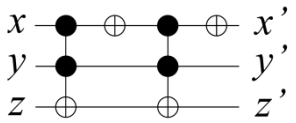

Fig. 2.1: 3x3 reversible circuit with two T gates and two N gates.

x	y	z	x'	y'	z'
0	0	0	0	0	0
0	0	1	0	0	1
0	1	0	0	1	1
0	1	1	0	1	0
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. 2.2: Truth table for the circuit in Fig. 2.2

A vertical line connecting a control to an inverter means that the inverter is only applied if the wire on which the control is set carries 1 signal. Therefore, the gates used are, from left to right, TOFFOLI, TOFFOLI NOT, , and NOT.

III. PROPOSED METHODOLOGY

The design of proposed architecture is shown in figure below. The architecture is designed for 2-bit ALU using Fynman gate, HNG gate, PAOG Gate and Fredkin gate. In the Fig. 3.1 A, B and C are inputs g1, g2, g3, g4, g5, g6, g7, g8, g9, g10, g11, g12, g13 and g14 are outputs of gates. t1, t2, t3, t4, t5, t6, t7, t8 , t9, t10, t11, t12, t13, t14, t15 and t16 are signals. S0, S1, S2, S3 and S4 are selection lines.

Table 1: Selection Line Codes and Respective Results for Fig. 3.1

S4	S3	S2	S1	S0	Result
0	0	0	0	0	AND
0	0	0	1	0	NAND
0	1	0	0	0	OR
0	1	1	0	0	NOR
1	0	0	0	0	Addition
1	0	0	0	1	Subtraction

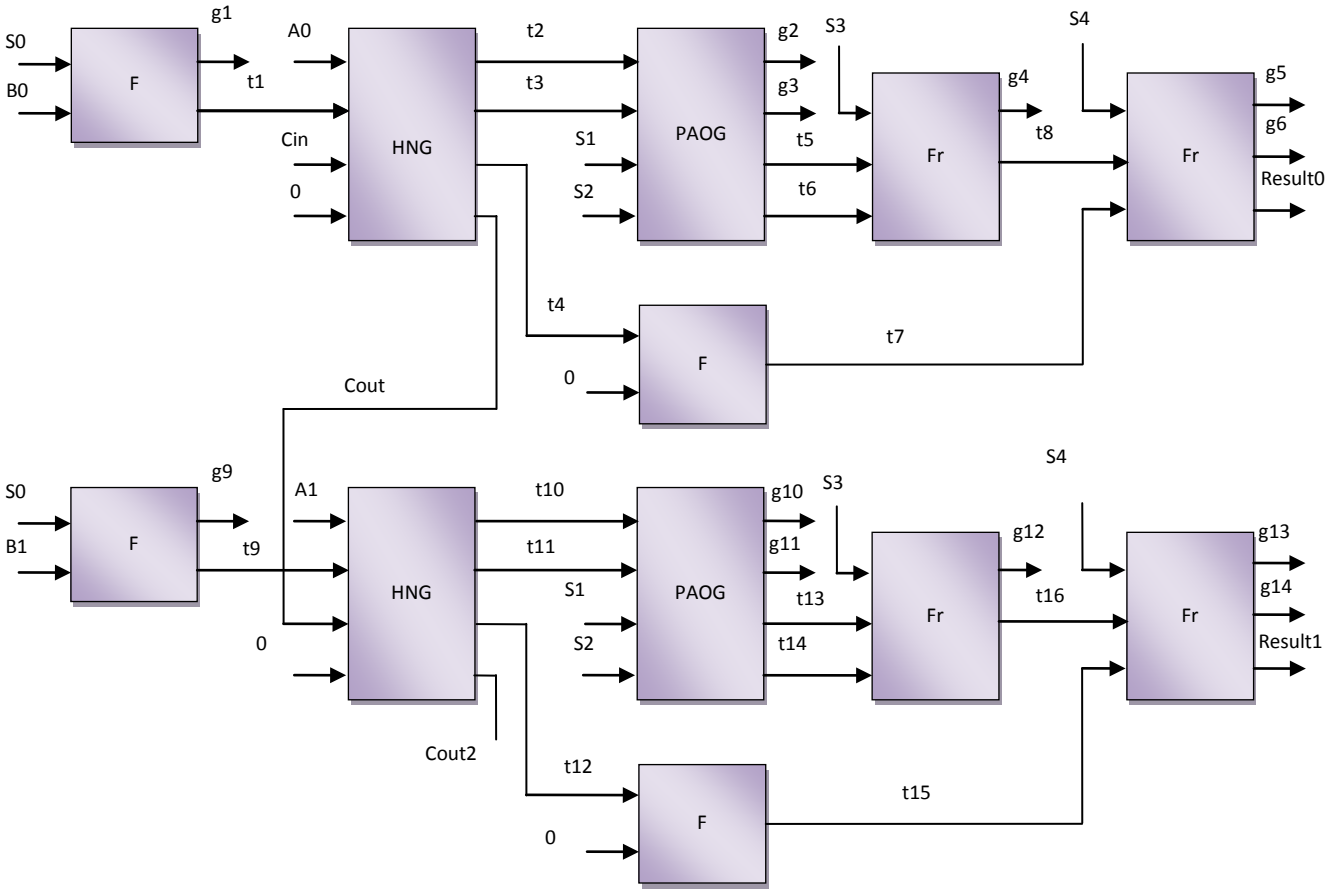

Fig. 3.1 Block Diagram of Proposed Architecture of 2-Bit ALU using Reversible Gates

IV. SIMULATION RESULTS

The simulation of proposed architecture is implemented and simulated and verified. The delay calculated for 2-bits is 11.24ns and for 1-bit is 10.72ns means there is no significant delay introduced by the proposed architecture compared to 1-bit architecture. This is achieved because of the parallel arrangement of the reversible gates for separate bits. The timing details generated by simulation tools are given in below figures.

```
Timing Detail:
--------------
All values displayed in nanoseconds (ns)

=========================================================================
Timing constraint: Default path analysis
  Total number of paths / destination ports: 34 / 9
-------------------------------------------------------------------------
Delay:               10.726ns (Levels of Logic = 4)
  Source:            a<0> (PAD)
  Destination:       sum0 (PAD)

  Data Path: a<0> to sum0
                                Gate     Net
    Cell:in->out      fanout   Delay   Delay  Logical Name (Net Name)
    ----------------------------------------  ------------
     IBUF:I->O             3   0.821   1.246  a_0_IBUF (a_0_IBUF)
     LUT3:I0->O            4   0.551   1.112  g31 (g3_OBUF)
     LUT3:I1->O            1   0.551   0.801  f2/Mxor_q_Result1 (sum0_OBUF)
     OBUF:I->O                 5.644          sum0_OBUF (sum0)
    ----------------------------------------
    Total                     10.726ns (7.567ns logic, 3.159ns route)
                                       (70.5% logic, 29.5% route)

=========================================================================
```

Fig. 4.1: Timing Details of the 1-Bit ALU architecture using reversible gates.

```
Timing Detail:
--------------
All values displayed in nanoseconds (ns)

=========================================================================
Timing constraint: Default path analysis
  Total number of paths / destination ports: 82 / 18
-------------------------------------------------------------------------
Delay:               11.240ns (Levels of Logic = 5)
  Source:            a<1> (PAD)
  Destination:       ornor1 (PAD)

  Data Path: a<1> to ornor1
                                Gate     Net
    Cell:in->out      fanout   Delay   Delay  Logical Name (Net Name)
    ----------------------------------------  ------------
     IBUF:I->O             4   0.821   1.256  a_1_IBUF (a_1_IBUF)
     LUT3:I0->O            4   0.551   1.256  g111 (g11_OBUF)
     LUT4:I0->O            1   0.551   0.000  fy4/Mxor_r_Result11 (fy4/Mxor_r_Result1)
     MUXF5:I1->O           1   0.360   0.801  fy4/Mxor_r_Result1_f5 (ornor1_OBUF)
     OBUF:I->O                 5.644          ornor1_OBUF (ornor1)
    ----------------------------------------
    Total                     11.240ns (7.927ns logic, 3.313ns route)
                                       (70.5% logic, 29.5% route)

=========================================================================
```

Fig. 4.2: Timing Details of the 2-Bit ALU architecture using reversible gates.

The test bench waveform is shown in the Fig. 4.3 and 4.4 for 1-bit and 2-bit architecture respectively.

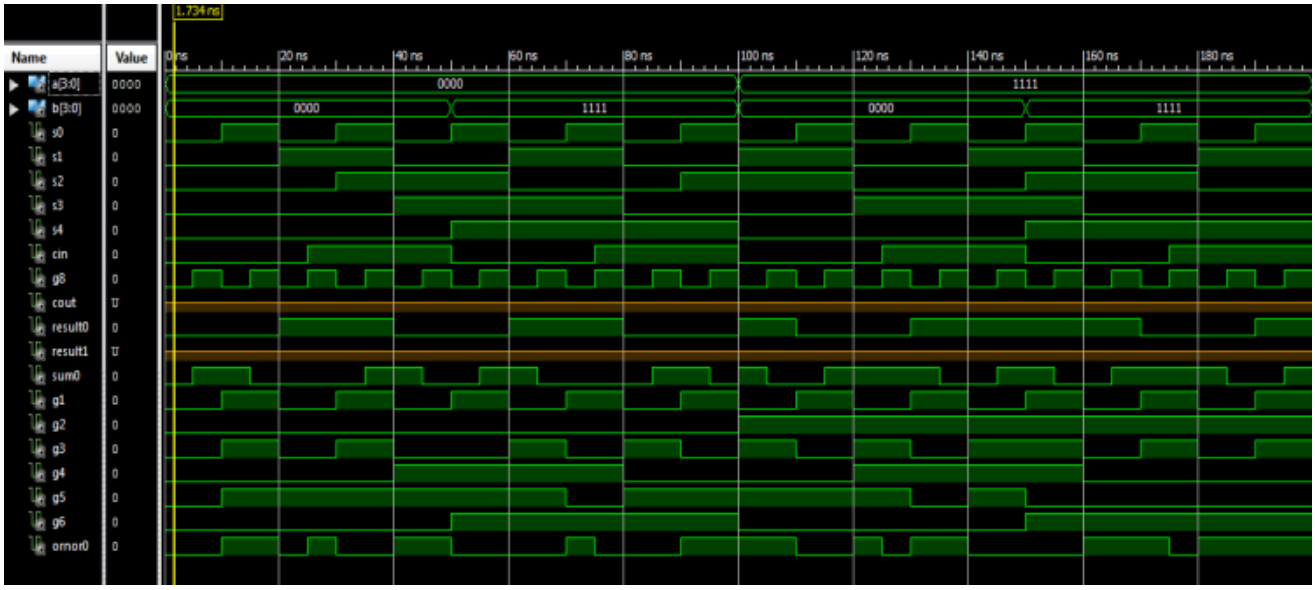

Fig.4.3: Test Bench Waveforms for Single Bit Reversible Gate ALU

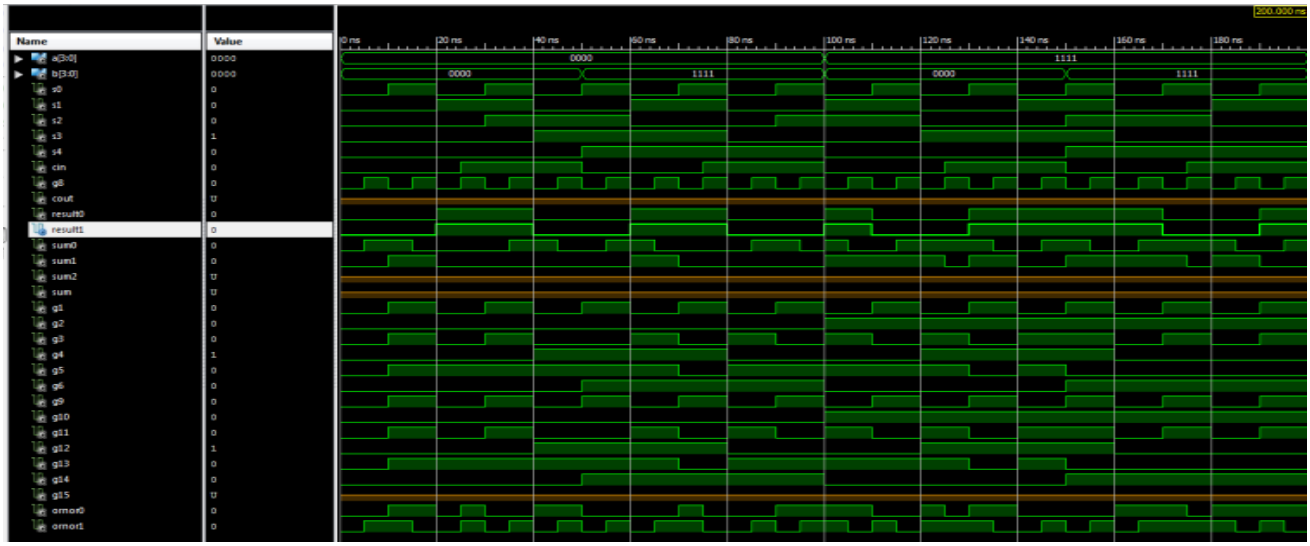

Fig. 4.4: Test Bench Waveforms for 2- Bit Reversible Gate ALU

V. Conclusion and Future Scope

From the simulation results it is clear that the delay of the ALU for 1-bit is reduced by using the architecture in parallel with the 1-bit and the proposed parallel architecture produces the delay as compared to the previous 1-bit architecture. Now if we consider the previous architecture to design 32-ALU than the total delay is about 10.72ns*32 = 343.04ns, but by using proposed 2-bit architecture it will be 11.24ns*16 = 179.84ns only means there is the proposed architecture complete any 32-bit arithmetic operation near in half time of previous architecture.

In future the proposed architecture will be made with lesser delay using more parallel architectures e.g. 4-bit or 8-bit.

References

[1] A. Barenco et al., “Elementary Gates For Quantum Computation,” *Physical Review A*, 52, 1995, pp. 3457-3467.
[2] C. Bennett, “Logical Reversibility of Computation,” *IBM J. of R.&D.*, 17, ‘73, pp. 525-532.
[3] T. Beth and M. R¨otteler, ”Quantum Algorithms: Applicable Algebra and Quantum Physics,” Springer Tracts in Modern Physics, 173, 2001, pp. 96-50.
[4] S. Bettelli, L. Serafini and T. Calarco, “Toward an Architecture for Quantum Programming,” Nov. 2001, http://arxiv.org/abs/cs.PL/0103009.
[5] A. De Vos et al., ”Generating the Group of Reversible Logic Gates,” *Journal of Physics A:Mathematical and General*, 35, 2002, pp. 7063-7078.
[6] L. K. Grover, “A Framework For Fast Quantum Mechanical Algorithms,” *Proc. Symp. on Theory of Computing*, 1998.
[7] R. Feynman, ”Quantum Mechanical Computers,” *Optics News*, 11, 1985, pp. 11-20.
[8] K. Iwama et al., ”Transformation Rules For Designing CNOT-based Quantum Circuits,” Proc. DAC 2002, pp. 419-425.
[9] P. Kerntopf, “A Comparison of Logical Efficiency of Reversible and Conventional Gates,” *IWLS 2000*, pp. 261-269.
[10] R. Korf, “Artificial Intelligence Search Algorithms”, *Algorithms and Theory of Computation Handbook*, CRC Press, 1999.
[11] E. Lawler, “An Approach to Multilevel Boolean Minimization,” *JACM*, 11, ‘64, pp. 283-295.
[12] J. Lee et al., ”A Practical Method of Constructing Quantum Combinational Logic Circuits,” http://arxiv.org/abs/cs.PL/9911053.
[13] J. P. McGregor and R. B. Lee, “Architectural Enhancements for Fast Subword Permutations with Repetitions in Cryptographic Applications,” *ICCD*, 2001, pp. 453-461.
[14] M. Nielsen and I. Chuang, *Quantum Computation and Quantum Information*, Cambridge Univ. Press, 2000.
[15] M. Perkowski et al., “A General Decomposition For Reversible Logic,” *Reed-Muller Workshop*, Aug. 2001.
[16] T. Sasao and K. Kinoshita, “Conservative Logic Elements and Their Universality,” *IEEE Trans. on Computers*, 28, 1979, pp. 682-685.
[17] Z. Shi and R. Lee, “Bit Permutation Instructions for Accelerating Software Cryptography,” *IEEE Intl. Conf. on App.-specific Systems, Architectures, and Processors*, 2000, pp. 138-148.
[18] L. Storme et al., ”Group Theoretical Aspects of Reversible Logic Gates,” *Journal of Universal Computer Science*, 5, 1999, pp. 307-321.
[19] T. Toffoli, “Reversible Computing,” *Tech. Memo MIT/LCS/TM-151*, MIT Lab for CS, ‘80.
[20] B. Travaglione et al. “ROM-based computation: Quantum Versus Classical,” 2001.
[21] http://arxiv.org/abs/quant-ph/0109016
[22] S. Younis and T. Knight, “Asymptotically Zero Energy Split-Level Charge Recovery Logic,” *Workshop on Low Power Design*, 1994.
[23] Y. Syamala and A. V. N. Tilak *“Reversible Arithmetic Logic Unit”,2011 IEEE*
[24] R. Feynman, "Quantum Mechanical Computers," *Foundations of Physics*, vol. 16,iss. 6, 1986.
[25] M. Morrison, M. Lewandowski, R. Meana and N. Ranganathan, "Design of Static and Dynamic RAM Arrays Using a Novel Reversible Logic Gate and Decoder," *To Appear in the IEEE 11th International Conference on Nanotechnology*, 2011.
[26] M. Morrison and N. Ranganathan, "Design of a Moore Finite State Machine using a Novel Reversible Logic Gate, Decoder and Synchronous Up-Counter," *To Appear in the IEEE 11th International Conference on Nanotechnology*, 2011.
[27] R. Feynman, "Simulating Physics with Computers," *International Journal of Theoretical Physics*, 1982.
[28] J. Smolin and D. Divincenzo, "Five Two-bit Quantum sGates Are Sufficient to Implement the Quantum Fredkin Gate," *Physical Review A*, vol. 53, 1996, pp 2855-6.
[29] Saurabh Kotiyal, Himanshu Thapliyal and Nagarajan Ranganathan, “Mach-Zehnder Interferometer Based Design of All Optical Reversible Binary Adder” 2012 IEEE.
[30] Bibhash Sen, Manojit Dutta, Debajyoty Banik, Dipak K Singh, Biplab K Sikdar, “Design of Fault Tolerant Reversible Arithmetic Logic Unit in QCA” 2012 International Symposium on Electronic System Design.
[31] Y. Syamala, A. V. N. Tilak, “Reversible Arithmetic Logic Unit”, 2011 IEEE.
[32] Matthew Morrison, Matthew Lewandowski, Richard Meana and Nagarajan Ranganathan, “Design of a Novel Reversible ALU using an Enhanced Carry Look- Ahead Adder” , 2011 11th IEEE International Conference on Nanotechnology Portland Marriott August 15-18, 2011, Portland, Oregon, USA.
[33] Anindita Banerjee , “Reversible cryptographic hardware with optimized quantum cost and delay” 2010 Annual IEEE India Conference (INDICON).
[34] H. Thapliyal and M. Srinivas, "Novel Reversible 'TSG' Gate and Its Application for Designing Components of Primitive Reversible/Quantum ALU," *Tenth Asia-Pacific Computer Systems Architecture Conference*, 2005.
[35] M. Shams, M. Haghparast and K. Navi, “Novel Reversible Multiplier Circuit in Nanotechnology”, *World Applied Sci. J.*, Vol. 3, pp. 806- 810, 2008.

[36] M. Haghparast, S. J. Jassbi, K. Navi and O. Hashemipour, "Design of a Novel Reversible Multiplier Circuit using HNG Gate in Nanotechnology ", *World Applied Sci. J.*, Vol. 3, pp. 974-978, 2008.
[37] S. Islam, M. Rahman, Z. Begum, and M. Hafiz, "Realization of Novel Fault Tolerant Reversible Full Adder Circuit in Nanotechnology," *The International Arab Journal of Information Technology*, vol. 7, iss. 3, 2010, pp 317-333.

Author's Profile

Praveen Kumar is research scholar at at NRI Research Technology & Science, Bhopal, under Rajiv Gandhi Proudyogiki Vishwavidyalaya, Bhopal. He is pursuing M. Tech. in VLSI Design. He has keen to work on ALU Designing using Reversible Logic Gates.