

A Survey on Design Approaches towards Quantum ALU using Reversible Logic Structures

Vijay G. Roy, Prof. P. R. Indurkar, Prof. Mrs. D. M. Khatri

Abstract— Reversible logic is emerging as a promising computing paradigm in the recent years due to its ability to reduce power dissipation. It has numerous applications in low power VLSI design such as quantum computing, quantum dot cellular automata, optical information processing, DNA computing and nanotechnology. Fundamental reversible gates used for reversible logic synthesis are Feynman Gate, Fredkin Gate, Toffoli Gate, Peres Gate etc. Many researchers have investigated the design of reversible logic elements with different approaches. In this paper, we describe the significant contributions by various researchers in their literature towards the design of reversible logic structures and arithmetic logical units. In these literatures, reversible circuits has been designed that do not lose information and reversible computation in a system can be performed only when the system comprises of reversible gates. The efficiency of such circuits can be realized by comparing them in terms of quantum cost, garbage outputs, constant inputs, quantum delay and complexity of gates.

Index Terms— Reversible Logic, Qubit, Quantum cost, Garbage Outputs,

1 INTRODUCTION

IN thermodynamics, physical reversibility is a process that dissipates no energy to heat. Reversibility in computing means that no information about the computational states can ever be lost, so we can recover any earlier stage by computing backwards. This is termed as logical reversibility. Reversible logic supports the process of running the system both forward and backward. It means that reversible computations can generate inputs from outputs and can stop and go back to any point in the computation history. Reversible circuits can generate unique output vector from each input vector and vice versa, that is, there is one-to-one correspondence between the input and output assignments.

According to Landauer's research, the amount of energy dissipated for every irreversible bit operation is at least $KT \ln 2$ joules, where $K=1.38 \times 10^{-23} \text{m}^2 \text{kg}^{-2} \text{K}^{-1}$ (joule/Kelvin) is the Boltzmann's constant and T is the temperature at which operation is performed [2]. The heat generated due to the loss of one bit of information is very small at room temperature but when the number of bits is more as in the case of high speed computational works the heat dissipated by them will be so large that it affects the performance and results in reduction of lifetime of the components. In 1973, Bennett showed that $KT \ln 2$ energy would not dissipate from a system as long as the system allows the recovery of inputs from the observed outputs. Thus energy dissipation can be reduced or even eliminated if computation becomes information-lossless [3].

Reversible computing will also lead to improvement in energy efficiency. Energy efficiency will fundamentally affect the

speed of circuits such as nanocircuits and therefore the speed of most computing applications. To increase the portability of devices again reversible computing is required. It will let circuit element sizes to reduce to atomic size limits and hence devices will become more portable. Although the hardware design costs incurred in near future may be high but the power cost and performance being more dominant than logic hardware cost in this era, the need of reversible computing cannot be ignored. It has been proved that, "losing information in a circuit causes losing power. Information lost when the input vector cannot be uniquely recovered from the output vector of a combinational circuit."

2 REVERSIBLE LOGIC

As the basic elements of any logic circuits, logic gates are used to realize any Boolean functions. Similarly by combining reversible logic gates, we can construct reversible logic circuits that can perform complex logical and arithmetic operations. For a gate to be reversible, the logic function it realizes has to be bijective. That is, there must be one-to-one mapping between inputs and outputs [5],[7]. According to the definition, any reversible gate has an inverse (also called as "dual"). Also reversible circuits are called as lossless circuits, as there is neither energy loss nor information loss [1].

A reversible logic gate a k -input, k -output ($k \times k$) device that maps each possible input pattern into a unique output pattern as shown in fig.1 where $I_0, I_1, I_2, \dots, I_{K-1}$ are the inputs and $O_0, O_1, O_2, \dots, O_{K-1}$ are the outputs.

- Vijay G. Roy is currently pursuing masters degree program in VLSI from B.D.C.O.E., Sevagram in RTMNU, Nagpur University, India, PH-9372425983. E-mail: vgroy15@yahoo.com
- Prof.P. R. Indurkar is currently working as Associate Professor in B.D.C.O.E., Sevagram, PH.9422116333. E-mail: prashantindurkar@rediffmail.com
- Prof.Mrs.D. M. Khatri is currently working as Assistant Professor in B.D.C.O.E.,Sevagram,PH.9422842999. E-mail: deepali.85@rediffmail.com

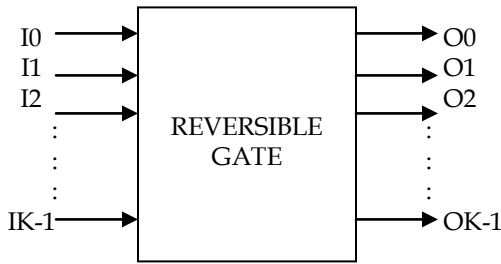


Fig 1. General Structure of Reversible Gate

2.1 Basic Reversible Gates

Basic reversible logic gates[8],[9] used in quantum computing are also called as qubit gates. Some of fundamental qubit gates are as follows:

Feynman Gate: Feynman gate is a 2*2 one through reversible gate. The input vector is I (A, B) and the output vector is O (P, Q). The outputs are defined by $P=A$ and $Q=A \text{ XOR } B$. Quantum cost of a Feynman gate is 1. 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.

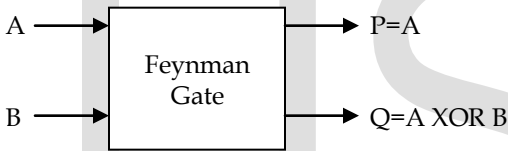


Fig 2. Block Diagram of Feynman Gate

TABLE 1
 Truth Table of Feynman Gate

A	B	P	Q
0	0	0	0
0	1	0	1
1	0	1	1
1	1	1	0

Toffoli Gate: A 3*3 Toffoli gate whose input vector is I (A, B, C) and the output vector is O (P, Q, R). The outputs are defined by $P=A$, $Q=B$ and $R=AB \text{ XOR } C$. Quantum cost of a Toffoli gate is 5.

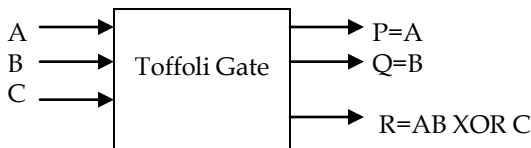


Fig 3. Block Diagram of Toffoli Gate

TABLE 2
 Truth Table of Toffoli 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	0	1
1	1	0	1	1	1
1	1	1	1	1	0

Fredkin Gate: A 3*3 Fredkin gate whose input vector is I (A, B, C) and the output vector is O (P, Q, R). The outputs are defined by $P=A$, $Q=A'B \text{ XOR } AC$ and $R=A'C \text{ XOR } AB$. Quantum cost of a Fredkin gate is 5.

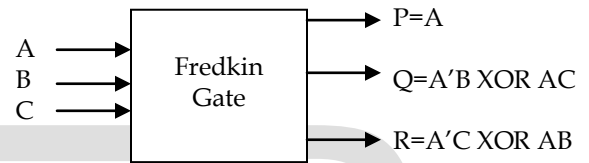


Fig 4. Block Diagram of Fredkin Gate

TABLE 3
 Truth Table of 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

Peres Gate: A 3*3 Peres gate whose input vector is I (A, B, C) and the output vector is O (P, Q, R). The output is defined by $P=A$, $Q=AB$ and $R=AB \text{ XOR } C$. Quantum cost of a Peres gate is 4.

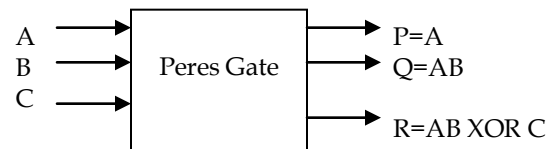


Fig 5. Block Diagram of Peres Gate

TABLE 4
Truth Table of Peres Gate

A	B	C	P	Q	R
0	0	0	0	0	0
0	0	1	0	0	1
0	1	0	0	0	0
0	1	1	0	0	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

These reversible gates are compared in terms of their quantum cost and worst case delay as shown in table below:

TABLE 5
Quantum Cost and Delay Comparison of Gates

Reversible Gate	Quantum Cost	Worst Case Delay
Feynman Gate	1	1
Toffoli Gate	5	5
Fredkin Gate	5	5
Peres Gate	4	4

2.2 Advanced Reversible Gates

In irreversible logic gates, some logic gates such as XOR, XNOR, NAND and NOR are derived from basic gates such as AND, OR and NOT. Similarly in reversible case, more advanced gates[9] can be designed using fundamental reversible gates. Some of those advanced gates are as follows:

New Gate: A 3*3 New gate whose input vector is I (A, B, C) and the output vector is O (P, Q, R). The outputs are defined by $P=A$, $Q=AB \text{ XOR } C$ and $R=A'C \text{ XOR } B'$.

TR Gate: A 3*3 TR gate whose input vector is I (A, B, C) and the output vector is O (P, Q, R). The outputs are defined by $P=A$, $Q=A \text{ XOR } B$ and $R=AB' \text{ XOR } C$.

MRG Gate: A 4*4 Morrison-Ranganathan gate whose input vector is I (A, B, C, D) and the output vector is O (P, Q, R, S). The outputs are defined by $P=A$, $Q=A \text{ XOR } B$, $R=(A \text{ XOR } B) \text{ XOR } C$ and $S=(AB \text{ XOR } D) \text{ XOR } ((A \text{ XOR } B) \text{ XOR } C)$. MRG gate has quantum cost of 6 and worst case delay of 4. With two select inputs it can produce four logical calculation such as OR, NOR, XOR and XNOR.

PAOG Gate: A 4*4 Peres AND-OR gate whose input vector is I (A, B, C, D) and the output vector is O (P, Q, R, S). The outputs are defined by $P=A$, $Q=A \text{ XOR } B$, $R=AB \text{ XOR } C$ and $S=((A \text{ XOR } B) \text{ XOR } D) \text{ XOR } (AB \text{ XOR } C)$. The cost and delay are identical to MRG gate. With two select inputs it will calculate four logical operations such as OR, NOR, AND and NAND.

PFAG Gate: A 4*4 Peres Full Adder gate whose input vector is

I (A, B, C, D) and the output vector is O (P, Q, R, S). The outputs are defined by $P=A$, $Q=A \text{ XOR } B$, $R=(A \text{ XOR } B) \text{ XOR } C$ and $S=(A \text{ XOR } B)C \text{ XOR } (AB \text{ XOR } D)$. The quantum cost and delay of PFAG gate are 8 and 6 respectively. The logical operations are similar to MRG gate.

HNG Gate: A 4*4 Hagparast-Navi gate whose input vector is I (A, B, C, D) and the output vector is O (P, Q, R, S). The outputs are defined by $P=A$, $Q=A \text{ XOR } B$, $R=(A \text{ XOR } B) \text{ XOR } C$ and $S=(AB \text{ XOR } D) \text{ XOR } ((A \text{ XOR } B) \text{ XOR } C)$. The quantum cost and delay of HNG gate are 6 and 4 respectively. The logical operations are similar to MRG gate.

2.3 Important Terminologies

Basic idioms[6] used in reversible logic computation are as follows:

Gate count: Number of reversible gates used in the circuit.

Garbage outputs: The outputs that are not used for further computations.

Quantum cost: The number of 1x1 or 2x2 gates that are used in the circuit.

Logical calculation: Related to hardware complexity indicating the number of NOT and two input XOR and AND gates required to implement the logic of the circuit.

Quantum depth: Quantum cost of the longest path from input to output.

Consant inputs: The inputs which are to be maintained constant at 1 or 0 throughout the circuit operation depending on the function required from the gate or circuit.

2.4 Significant Theorems

Theorem 1: Any reversible logic gate must have same number of inputs and outputs. Hence any programmable reversible logic gate with 'k' inputs and outputs has a quantity of fixed select inputs 'm', fixed select outputs 'n', data inputs 'd' and propagated outputs 'p' such that $|d-p| = |m-n|$.

Theorem 2: A programmable reversible logic gate with 'm' select inputs may produce at maximum $n \cdot 2^m$ logical calculations on the n logical outputs.

Theorem 3: While constructing reversible circuits with the help of reversible gates, some restrictions should be strictly maintained such as

- i) Fan-out is not permitted.
- ii) Loops are not permitted.

2.4 Realization of Irreversible Gates using Reversible Gates

By using basic reversible gates, we can construct basic logic gates such as AND gate, OR gate, NOT gate etc. For example, we can construct the following logic gates by pre-setting some inputs of the reversible gates[1].

The Fredkin gate is a universal gate such that it can be “programmed” as different basic building blocks of virtually any reversible logic circuits as follows:

AND gate: If input $B=0$, then $P=A$, $Q=AC$ and $R=A'C$.

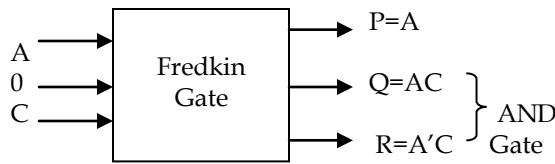


Fig 6. AND Gate Realization using Fredkin Gate

OR gate: If input $A=1$, then $P=A$, $Q= A+B$ and $R= A'+B$.

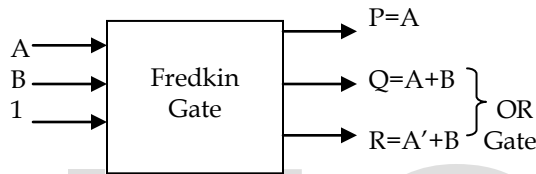


Fig 7. OR Gate Realization using Fredkin Gate

NOT gate/FANOUT: If inputs $A=1$ and $B=0$, then $P=A$, $Q=A$ and $R= A'$.

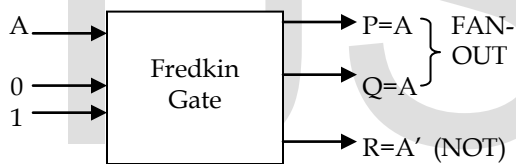


Fig 8. NOT Gate Realization using Fredkin Gate

2.5 Reversible Gate Implementation of Single Bit Full Adder

The combinational logic circuits can be made using reversible logic gates. Fig.9 illustrates a full adder [1] that is built with Fredkin gates, where A and B are single-bit inputs, and Cin denotes the single-bit “carry input.” The single-bit output for the sum is: $Sum = (Cin'(A \text{ XOR } B)) + (Cin (A \text{ XNOR } B))$, and the single-bit output for the “carry output” is: $Cout = Cin (A + B) + (AB)$.

This full adder requires six fredkin gates thus having quantum cost of $5*6=30$ and worst case delay of $5*3=15$.

In this adder example, it should be noted that the reversible logic circuit is quite different from conventional irreversible logic circuits in that the circuit creates some redundant outputs such as $g1, g2, g3, g4$ and $g5$ (garbage outputs). These outputs appear to be “useless” in terms of realizing the desired logic functions (e.g., sum and carry out in the case of the adder).

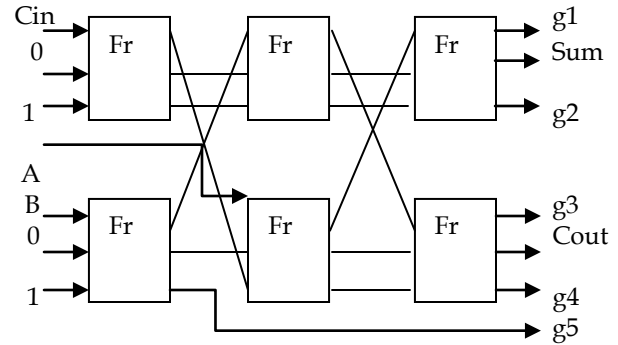


Fig 9. Full Adder using Fredkin Gate

Another problem is that we have to fix four input lines with constant values. Constant inputs are also called as ancillary inputs which reduces flexibility in the design.

2.6 Reversible ALU Implementation

Fig.10 illustrates reversible ALU which utilizes the MRG gate and HNG gate to produce six logical calculations: ADD, SUB, XOR, XNOR, OR and NOR. The ALU has 8 inputs and 8 outputs. The inputs consist of three data inputs (A, B and C) and five fixed input select lines (S0, S1, S2, S3 and S4).

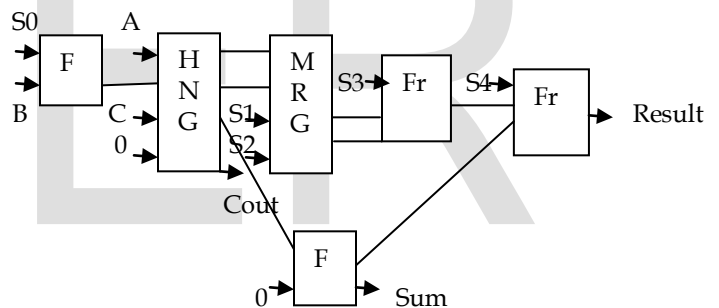


Fig 10. ALU Realization using Reversible Gates

The cost of this 1-bit ALU is 24, and the worst-case delay is 16. The quantum cost of arithmetic logic unit can be evaluated by adding costs of individual reversible gates. Similarly the worst case delay can be obtained by following the longest path from input towards output.

The combinations of select inputs are utilized as the 5-bit length operation code for given arithmetic logic unit.

TABLE 6
Op-codes of Reversible ALU

S4	S3	S2	S1	S0	Result
0	0	0	0	0	XOR
0	0	0	1	0	XNOR
0	1	0	0	0	OR
0	1	1	0	0	NOR
1	0	0	0	0	ADD
1	0	0	0	1	SUB

3 APPLICATIONS

Reversible logic is potentially applicable in various fields such as:

- i) Nanocomputing
- ii) Bio Molecular Computations
- iii) Laptop/Handheld/Wearable Computers
- iv) Spacecraft
- v) Implanted Medical Devices
- vi) Wallet "smart cards"
- vii) " Smart tags" on inventory

4 CONCLUSION

Any arithmetic logic unit must be able to produce a variety of logical outputs such as AND, OR, and XOR, based on inputs determined by the programmer for implementation in an instruction set architecture. Therefore, a reversible gate used for this purpose must be able to maximize the types of logical operations it can calculate while minimizing the number of select lines and logical output lines, cost and delay. To this end, a programmable reversible logic gate is defined here as a logic structure which possesses a bijection between input and output states and an equal number of inputs and outputs wherein a subset of the inputs are fixed select lines, and a fixed subset of the outputs produce guaranteed logical calculations.

ACKNOWLEDGMENT

I wish to thank Prof.P.R.Indurkar and Prof.Mrs.D.M.Khatri for their guidance and support.

REFERENCES

- [1] W.David Pan and Mahesh Nalasani, "Reversible logic," 2005 IEEE potentials pp.38-41.
- [2] Himanshu Thapliyal and M. B. Shrinivas, "Novel Reversible 'TSG' Gate and Its Application for Designing Components of Primitive Reversible /Quantum ALU,"2005 IEEE ICICS pp. 1425-1429.
- [3] Mingming Zhang, Shuguang Zhao, Xu Wang. Poor, "Multi-objective Reversible Logic Gate-level Evolutionary Synthesis Using Multi-objective Adaptive Discrete Differential Evolution,"2009 Third International Symposium on Intelligent Information Technology Application pp. 533-536.
- [4] Himanshu Thapliyal and Nagarajan Ranganathan, "Reversible Logic Based Concurrent Error Detection Methodology for Emerging Nanocircuits," 2010 IEEE proceedings of 10th International Conference on nanotechnology pp. 217-222.
- [5] Mahapatro, Sisira Kanta Panda, Jagannath Satpathy, Meraj Saheel, "Design of Arithmetic Circuits Using Reversible Logic Gates and Power Dissipation Calculation," 2010 International Symposium on Electronic System pp. 85-90.
- [6] Prashant R. Yelekar and Prof. Sujata S. Chiwande," Introduction to Reversible Logic Gates & its Application,"2nd National Conference on Information and Communication Technology (NCICT) 2011 pp. 5-9.
- [7] Zhijin Guan, Wenjuan Li, Weiping Ding, Yueqin Hang Lihui Ni,"An Arithmetic Logic Unit Design Based on Reversible Logic Gates," 2011 IEEE pp. 925-931.
- [8] Y. Syamala and A.V. N. Tilak,"Reversible Arithmetic Logic Unit," IEEE 2011 pp. 207-211.
- [9] Matthew Morrison and Nagarajan Ranganathan,"Design of a reversible ALU based on Novel Programmable Reversible Logic Gate Structures,"2011 IEEE Computer Society Annual Symposium on VLSI pp. 126-131.
- [10] Himanshu Thapliyal and Nagarajan Ranganathan,"Reversible Logic: Fundamentals and Applications in Ultra-low power, Fault testing and Emerging Nanotechnologies, and Challenges in future," Tutorial T2 2012 25th International Conference on VLSI Design pp.13-15.
- [11] Ismo Hanninen and Jarmo Takala,"Irreversibility Induced Density Limits and logical Reversibility in Nanocircuits," 2012 IEEE/ACM NANOARCH pp.50-54.
- [12] Rakshith T. R. and Rakshith Saligram,"Parity Preserving Logic based Fault Tolerant Reversible ALU," 2013 IEEE Conference on Information and Communication Technologies (ICT 2013) pp. 485-490.