An Optimized Design of Binary Comparator Circuit in Quantum Computing

Ankur Sarker, M. Shamiul Amin, Avishek Bose and Nafisah Islam
Department of Computer Science & Engineering
University of Dhaka, Dhaka-1000, Bangladesh.
E-mail: ankur.sum@gmail.com, samiul_cse@yahoo.com, avishek.csedu.res@gmail.com, nafisah.islam@gmail.com

***Abstract*— Reversible logic; transforms logic signal in a way that allows the original input signals to be recovered from the produced outputs, has attracted great attention because of its application in diverse areas such as quantum computing, low power computing, nanotechnology, DNA computing, quantum dot cellular automata, optical computing. In this paper, we design low power binary comparators using reversible logic gates. Firstly, single bit binary reversible comparator circuits are designed using different reversible gates along with proposed gate named Newly Proposed Gate. Then, these procedures are generalized for constructing binary *n*-bit reversible comparator circuit. The design synthesis consists of two parts: Comparator Cell and Propagator Cell. An algorithm, based on our proposed design, shows that proposed circuit reduces overall cost and it outperforms than existing sequential comparator circuits. Also, comparing with existing tree-based comparator circuit, proposed design reduces quantum cost, garbage output and gate count in a significance level which means better improvement as cost of any quantum circuit is directly associated with quantum cost, garbage output and gate count.**

Keywords- reversible logic gates, binary comparator, quantum computing, low power computing.

I. INTRODUCTION

Researchers like Landauer have shown that, for irreversible logic computations, each bit of information lost generates $kT\times ln2$ joules of heat energy, where k is the Boltzmann's constant and T is the absolute temperature at which the computation is per-formed [1]. Bennett showed that $kT\times ln2$ energy dissipation would not occur if a computation is carried out in a reversible way [2]. The amount of energy dissipated in a system bears a direct relationship to the number of bits erased during computation. Reversible circuits are those circuits that do not lose information. Reversible computation in a system can be performed only when the system comprises of reversible gates. These circuits can generate a unique output vector from each input vector, and vice-versa. In a reversible circuit, there is a one-to-one mapping between input and output vectors. Bennett's theorem [2] about heat dissipation is only a necessary and not sufficient condition, but its extreme importance lies in the fact that every future technology would have to use reversible gates to reduce power. One of the main constraints in reversible logic is that the minimization of the number of reversible gates and garbage output. Garbage output refers to the output that is not used for further computations. In reversible logic, the circuits also ensure that the fan-out is always 1. There are a number of existing reversible gates such as *Fredkin gate* [3-5], *Toffoli gate* [3], [4], *New gate* [6].

A reversible logic gate must maintain the one-to-one mapping between input and output vectors. So, extra outputs are added to the output vector to make the logic reversible. One important constraint in reversible logic is designing the restriction of fan-out. A reversible circuit should be designed using the minimum number of reversible logic gates with the possible lowest number of garbage outputs and constant inputs [3], [4]. The comparison of binary numbers has diverse range of applications form processor design to data mining to biological computations. In traditional binary comparators, traditional logic gates are used and it consumes little more power. Purposes of designing reversible binary comparators are to dissipate less power, efficient computations with minimum number of garbage, constant inputs and quantum cost. Due to its significance, several reversible comparator circuits have already been proposed [7-9], but existing comparators have higher garbage output and quantum cost which has impact in overall cost of the comparator circuit. The motivation of our research is to develop an algorithm for *n*-bit reversible comparator circuit with better improvements in quantum cost, garbage output, gate count, and delay/clock cycles.

This paper has following outline: in Section 2, necessary background studies are discussed. We introduce our proposed design and algorithm for developing *n*-bit comparator in Section 3. Section 4 depicts benchmark study and comparison of proposed design with existing designs. Finally, conclusion is given in Section 5.

II. BACKGROUND STUDIES

In this section, we present the basic definitions and ideas related to reversible logic.

Definition 2.1: Let the input vector be I_V and output vector be O_V, where $I_V = (I_1, I_2, ...,I_k)$, $O_V = (O_1, O_2, ... O_k)$ and $I_V \leftrightarrow O_V$. A $k\times k$ reversible gate is a k-input, k-output circuit that produces a unique output pattern for each possible input combination [11].

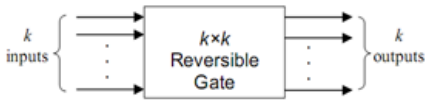

Fig. 1. $k\times k$ Reversible Gate..

Example 2.1: Fig. 1 shows a $k\times k$ reversible gate. One bit inverter is a 1×1 reversible gate with one input and one output. Similarly, 2×2 *Feynman* Gate (FG) [shown in Fig. 2] is also a 2×2 reversible gate with two inputs and two outputs [10].

978-1-4799-5180-2/14/$31.00 ©2014 IEEE

Other popular reversible gates are *3×3 Toffoli* gate and *3×3 Fredkin* gate [3, 4].

Definition 2.2: Unused outputs of a reversible gate are known as *garbage outputs*. Those outputs are only used to maintain the reversibility.

Example 2.2: When a *Feynman gate* is used for *EX-OR* operation of two inputs, extra one output is generated at the output part in addition to the *EX-OR* output. This additional output is known as garbage output [4].

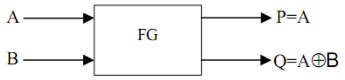

Fig. 2. 2×2 *Feynman* Gate.

Definition 2.3: 3×3 *Peres gate* realizes $P = A$, $Q = A \oplus B$ and $R = AB \oplus C$, where A, B and C are inputs and P, Q and R are outputs [11].

Example 2.3: A *3×3 Peres gate* is shown in Fig. 3.

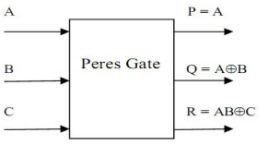

Fig. 3. *3×3 Peres* Gate.

Definition 2.4: Fan-out is a term that defines the maximum number of inputs in which the output of a single logic gate can be fed.

Example 2.4: The fan-out of any reversible gate is 1.

Definition 2.5: An *n-bit binary comparator* takes two *n*-bit binary numbers and produces three outputs that represents whether two inputs are equal or first one is greater than second one. It is also possible that second one is greater than the first one.

*Example 2.5:*1-bit binary comparator takes two inputs A and B, produces three outputs- *(A>B) = AB', (A<B) =A'B* and *(A=B) =(A⊕B)'*.

Definition 2.6: Quantum cost refers to the cost of the circuit in terms of primitive gate in quantum logic. It realizes the circuit by primitive gates and number of required gate is equal to the cost of that circuit.

Example 2.6: Quantum cost of *Feynman* gate is 2, *Toffoli* gate is 5 etc.

III. Design of Reversible Comparator Circuit

In this Section, we describe the overall procedures of our proposed comparator circuits. Firstly, we design binary single bit comparator circuits. Then, we use these comparators as building blocks to construct an *n*-bit reversible binary comparator circuit.

A. Design of 1-bit Comparator Circuit using EX-OR Gates

We know, *A⊕B = A'B+AB'* ; equivalently to *(A<B) and (A>B) respectively.* Thus, an *EX-OR* operation contains the output of a single bit comparator. In irreversible logic, output is high for both *(A<B)* and *(A>B)*. But by using reversible *EX-OR* gate, we can easily figure out whether *(A<B)* or *(A>B)* is high. We use first input *A* as a control signal to find out, are two inputs equal or not. Then, we realize which input is greater. Fig. 4 draws the overall construction of a *1*-bit reversible comparator circuit using *EX-OR* gates. Here, we use one *3×3* extended type of *Feynman gate,* (where outputs are- *P=A, Q=A⊕B, R=A⊕B⊕C*)*;* other three *Feynman* gates are used. This comparator takes two numbers which to be compared with one constant input and produces *(A>B)*, *(A<B)* and one garbage. Quantum cost of this comparator is 5.

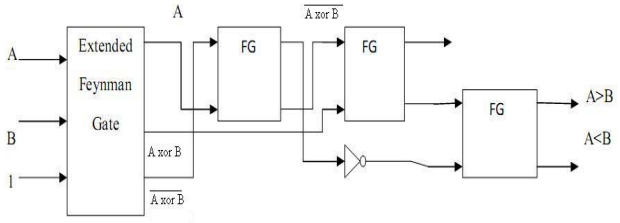

Fig. 4. *1*-bit Comparator Circuit Using Reversible EX-OR Gates.

B. Design of 1-bit Comparator circuit using Existing Reversible gates

Using one *Peres* and one *Feynman gate*, we can easily construct another *1*-bit binary comparator circuit. Fig. 5 shows the overall construction of this *1*-bit binary comparator. This circuit has also three inputs (with one constant low) and two outputs (with one garbage output). Here, *(A>B)* = AB' and *(A<B) =AB'⊕(A⊕B')'*

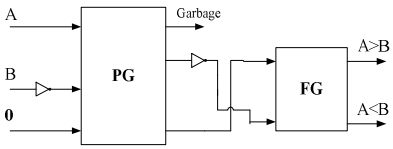

Fig. 5. *1*-bit Comparator Circuit Using Existing Reversible Gates.

C. Design of 1-bit Comparator Using Gates

In this sub-section, we present one novel reversible gate which is capable to perform comparison between two binary numbers.

a. Newly Proposed Gate

We design a *4×4* reversible logic gate named as *Newly Proposed gate* (*NPG*) which has four inputs (*A, B, C* and *D*) and four Outputs *(P = A, Q = A⊕B', R = AB'⊕C and S = (A⊕B')⊕AB'⊕C⊕D)*. Quantum cost of *NPG* is 6. Fig. 6 shows our proposed *4×4 NPG* and its equivalent quantum realization.

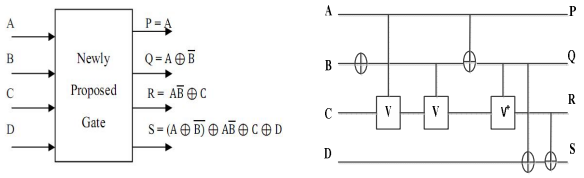

Fig. 6. (a) *4×4 NPG*; **(b)** Quantum realization of *NPG*.

978-1-4799-5180-2/14/$31.00 ©2014 IEEE

b. NPG as a 1-bit Comparator Circuit

We can use *NPG* to construct a *1*-bit comparator. Two numbers are fed into input *A*, *B*. Input *C, D* serve as constants (*C=0, D=1)*. Output *Q, R, S* produce *A⊕B', AB', (AB⊕B')'*, respectively with one garbage *P*. Fig. 7 depicts the overall representation.

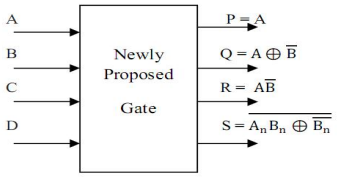

Fig. 7. *4×4 NPG* as single bit comparator circuit.

D. Design of n-bit Comparator Circuit

Using *n*-bit comparator circuit, we usually compare from most significant bit (*MSB*) to least significant bit (*LSB*). So, we can divide the *n*-bit comparator circuit into two parts: *Comparator Cell* (to compare *MSB*) and *Propagator Cell* (to propagate comparing results from less to least significant bit). In the *Comparator Cell*, we generate Q_n, R_n and S_n by using our proposed *NPG*. On the other hand, *Propagator Cell* propagates those Q_n, R_n and S_n with help of $(A>B)$ and $(A<B)$, (where *A*, *B* are two binary comparator input) to calculate Q_{n-1}, R_{n-1}, S_{n-1}.Similar procedures carry on to generate Q_1, R_1, S_1. Since it is binary *n*-bit comparator we need one *Comparator Cell* and *n-1 Propagator Cell*s.

a. Comparator Cell: *Newly Proposed* gate is used to construct *Comparator Cell*, where- $Q_n = A_n \oplus B_n'$, $R_n = A_nB_n'$ and $S_n = (A_nB_n \oplus B_n')'$.

b. Propagator Cell: First generates followings: $(A_{n-1}>B_{n-1})$ and $(A_{n-1}<B_{n-1})$, where- $(A_{n-1}>B_{n-1})=(A_{n-1}B_{n-1}')$ and $(A_{n-1}<B_{n-1})= (A_{n-1}B_{n-1}' \oplus (A_{n-1} \oplus B_{n-1}')')$.Then, using above bits, it generates- $Q_{n-1}= (R_{n-1} \oplus S_{n-1})'$, $R_{n-1} = (A_{n-1} > B_{n-1})Q_n \oplus R_n$ and $S_{n-1} = (A_{n-1} < B_{n-1})Q_n \oplus S_n$.

Similar procedures are continued to the *LSB*.

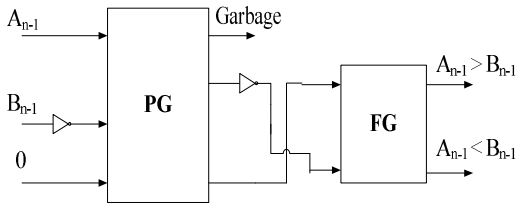

Fig. 8. Propagator Cell (part one).

Fig. 8 and Fig. 9, together represent the construction of *Propagator Cell.* Fig. 10 shows the overall representation of *n*-bit comparator circuit.

Lemma 3.1: A circuit formation of two cells for the highest order bit and the lower order bits should be sufficient to manifest the *2*-bit to *n*-bit circuits, where *n* is any positive integer greater than two *(n≥2)*.

Explanation: A *2*-bit circuit has two bits- *most significant* and *least significant*. For greater than two bit circuit, it has at least two bits- *MSB* and *LSB*. A circuit comprised of two cells (one for the *most significant bit* and other for the *least significant bits*), can be used to construct *n*-bit circuits where *n*≥2. So, it verifies the above statement.

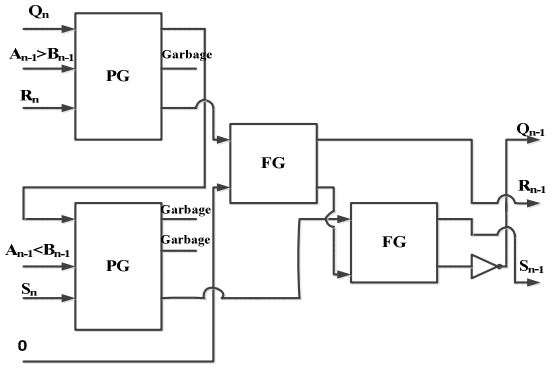

Fig. 9. Propagator Cell (part two).

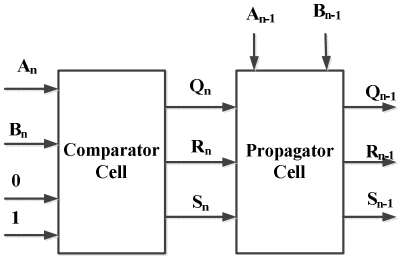

Fig. 10. *n*-bit comparator circuit.

a. Algorithm of n-bit comparator circuit

Input: Two numbers which will be used as comparator's input- $A_1, A_2 \ldots A_n$; $B_1, B_2 \ldots Bn$.

Output: Final output Q_i (for *A* and *B* are equal), R_i (for *A* is greater than *B*), S_i (for *A* is less than *B*).

Initialize: Initialize variables $(A_i > B_i)$, $(A_i < B_i)$, Q_{i+1}, Q_i, R_{i+1}, R_i, S_{i+1}, S_i.

Step 1: For n^{th} bit of two numbers do *Step 2* to *Step 4*
Step 2: $Q_{i+1} \leftarrow A_n \oplus B_n'$;
Step 3: $R_{i+1} \leftarrow A_nB_n'$;
Step 4: $S_{i+1} \leftarrow (A_nB_n \oplus B_n')'$;
Step 5: *for* each *i*th input bit equals to *(n-1)* to *1*, repeat *Step 6* to *Step 13*

Step 6: $(A_i > B_i) \leftarrow A_iB_i'$;
Step 5: $(A_i < B_i) \leftarrow A_iB_i' \oplus (A_i \oplus B_i')'$;
Step 6: $R_i \leftarrow (A_i > B_i)Q_{i+1} \oplus R_{i+1}$;
Step 7: $S_i \leftarrow (A_i < B_i)Q_{i+1} \oplus S_{i+1}$;
Step 8: $Q_i \leftarrow (R_i \oplus S_i)'$;
Step 9: *if* $i \neq 1$ do *Step 10* to *Step 12*
Step 10: $Q_{i+1} \leftarrow Q_i$;
Step 11: $R_{i+1} \leftarrow R_i$;
Step 12: $S_{i+1} \leftarrow S_i$;
Step 13: End of *for* loop
Step 14: Print the value Q_i, R_i, S_i;

The above algorithm conveys procedures to construct our proposed *n*-bit comparator circuit. For *n*-bit comparator, the run time complexity of this algorithm is *O(n)*.

978-1-4799-5180-2/14/$31.00 ©2014 IEEE

IV. BENCHMARK STUDIES

Followings are the overall analyses of our proposed comparator circuits in terms of quantum cost, garbage outputs, number of gates. Table 1 summarizes the overall parameters of proposed binary comparator circuits in terms of gate count, quantum cost, garbage output, constant input and required clock cycles. As an *n*-bit comparator consists of two cells, calculating complexity can be divided into two parts, for *Comparator Cell* and *Propagator cell*. So, for an *n*-bit comparator circuit-

I. Total Number of garbage = Comparator Cell's garbage + (n-1) X Propagator Cell's garbage = 1 + 4(n-1).

II. Total Constant input = Comparator Cell's constant input + (n-1) X Propagator Cell's constant input = 2 + 2(n-1).

III. Total Clock Cycles = Maximum (Comparator Cell, Propagator Cell (part one)) + (n-1) X Propagator Cell (Part two) = 4 + 4(n-1) = 4n.

IV. Total Gate count = Comparator Cell's gate count + (n-1) X Propagator Cell's gate count = 1 + 9(n-1).

Table 1. Summarization of Proposed Binary Comparators.

Binary Comparator Circuits	Quantum Cost	Garbage Output	Constant Input	Clock Cycle	Gate Count
1-bit comparator using Ex-OR gates	5	1	1	4	5
1-bit comparator using reversible gates	5	1	1	4	4
1-bit comparator using NPG	6	1	2	1	1
2-bit comparator	21	5	4	8	10
3-bit comparator	36	9	6	12	19
8-bit comparator	111	29	16	32	64

[a]A clock cycle denotes the time to perform one basic instruction. [b]It has been derived using Section 4.4 equations.

A. Comparisons

In this sub-section, we depict the overall comparison between proposed circuits and other existing circuits [7-9].

Table 2. Comparison of Different Comparators.

Different Comparators		Quantum Cost	Garbage Output	Clock Cycle
8-bit	Existing design [7]	135	42	61
	Existing (Min/Max) design [8]	450	84	62
	Existing design [9]	321	64	199
	Proposed design	111	29	32
	Improvements *w.r.t.* [7], [8], [9]	17.78%, 75.33%, 65.42%	30.95%, 65.48%, 54.69%	47.54%, 48.39%, 83.92%
32-bit	Existing (sequential) design [8]	532	160	315
	Existing design [9]	1257	256	775
	Proposed design	471	125	128
	Improvements *w.r.t.* [8], [9]	11.47%, 78.44%	21.88%, 51.17%	59.37%, 83.49%
64-bit	Existing design [7]	1143	378	115
	Existing design [9]	2505	512	1543
	Proposed design	951	253	256
	Improvements *w.r.t.* [7], [9]	16.8%, 62.04%	33.07%, 50.59%	63.19%

For considering sequential design, our design outperforms than existing one. For existing tree-based comparators, the comparison shows that proposed design minimizes the overall cost of those circuits in terms of quantum cost and garbage output.

V. CONCLUSION

We have proposed a optimized approach to design a low power binary comparator circuit using reversible logic gates. This proposed design can perform bit by bit binary comparison. Firstly, we have designed single bit comparator circuits in different approaches with different combinations of reversible gates. A new reversible gate has also been designed and used for comparator circuits. Then, higher bit comparator circuits have been demonstrated by dividing comparator into two cells for higher order bits and single bit comparator circuits are used as a building blocks of *n*-bit comparator circuit. We have also presented an algorithm which is associated with proposed design. Our proposed circuits achieve the considerable reduction in quantum cost, clock cycles and require fewer gates. We have also verified our design efficiency by comparing with existing methods, both sequential and tree-based designs. As reversible logic is applicable to a diverse application area including Nano scale computing, low power computing modules and quantum computing, DNA computing; our proposed design could also be applicable in those fields.

978-1-4799-5180-2/14/$31.00 ©2014 IEEE

REFERENCES

[1] Landauer, Rolf and James W. F. Woo. "Minimal energy dissipation and maximal error for the computational process", in *Journal of Applied Physics* 42, no. 6 (1971): 2301-2308.

[2] Bennett, Charles H. "Logical reversibility of computation", in *IBM journal of Research and Development* 17, no. 6 (1973): 525-532.

[3] FREDKIN, E. "Conservative logic" in *Int. J. Theoretical Physics* 21 (1982): 219-253.

[4] Toffoli, Tommaso. "*Reversible computing*", in Springer Berlin Heidelberg, 1980.

[5] Leporati, Alberto, Claudio Zandron, and Giancarlo Mauri. "Simulating the Fredkin Gate with Energy-Based P Systems" in *J. UCS* 10, no. 5 (2004): 600-619.

[6] Khan, Md. M. H. Azad. "Design of full-adder with reversible gates", in *International Conference on Computer and Information Technology, Dhaka, Bangladesh*, pp. 515-519. 2002.

[7] Thapliyal, Himanshu, Nagarajan Ranganathan, and Ryan Ferreira. "Design of a comparator tree based on reversible logic", in *Nanotechnology (IEEE-NANO), 2010 10th IEEE Conference on*, pp. 1113-1116. IEEE, 2010.

[8] Morrison, Matthew, Matthew Lewandowski, and Nagarajan Ranganathan. "Design of a Tree-Based Comparator and Memory Unit Based on a Novel Reversible Logic Structure", in *VLSI (ISVLSI), 2012 IEEE Computer Society Annual Symposium on*, pp. 231-236. IEEE, 2012.

[9] Al-Rabadi, Anas N. "Closed-system quantum logic network implementation of the viterbi algorithm", in *Facta universitatis-series: Electronics and Energetics* 22, no. 1 (2009): 1-33.

[10] Feynman, Richard P. "Quantum mechanical computers", in *Foundations of physics* 16, no. 6 (1986): 507-531.

[11] Peres, Asher. "Reversible logic and quantum computers", in *Physical review A* 32, no. 6 (1985): 3266.

978-1-4799-5180-2/14/$31.00 ©2014 IEEE