Realization of BCD adder using Reversible Logic

X. Susan Christina, M.Sangeetha Justine, K.Rekha, U.Subha and R.Sumathi

Abstract - **This paper proposes novel carry select and carry look-ahead BCD adders using reversible logic. Reversible logic gates are widely known to be compatible with future computing technologies which virtually dissipate zero heat. Adders are fundamental building blocks in many computational units. For this reason, we have simulated several adder circuits using the reversible gates. Among all the other adders, the main virtue of BCD adders is that it allows easy conversion to decimal digits for printing or display and faster decimal calculations. These reversible BCD circuits are basis of the decimal ALU of primitive Quantum CPU. The proposed BCD adders have been simulated in VLSI and static timing report was analyzed.**

Key Words - **Reversible logic gates, BCD adder , quantum computing.**

I. Introduction

Everyday new technology which is faster, smaller and more complex than its predecessor is being developed. The increase in clock frequency to achieve greater speed and increase in number of transistors packed onto a chip to achieve complexity of a conventional system results in increased power consumption. Almost all the millions of gates used to perform logical operations in a conventional computer are irreversible. That is, every time a logical operation is performed some information about the input is erased or lost and is dissipated as heat. As per Landauer [1], for irreversible logic, each bit of information lost generates kTln2 Joules of heat energy, where k is Boltzmann's constant and T is absolute temperature at which the computation is performed. For room temperature T, the amount of heat dissipated for one bit is small i.e. 2.9×10-21 J [2]. The current processors, first of all dissipate 500 times this amount of heat every time a bit is lost. Secondly, assuming every transistor out of more than 4×107 dissipates heat at the processor frequency of 2GHz, the figure becomes 4×1019*kTln2 J/sec. Although, it is around 0.1W at 400 degree Kelvin, Moore's law predicts exponential growth of heat generated due to information loss which will be an intolerable amount in the next decade. This heat dissipation dramatically reduces the performance and lifetime of the circuits. The solution is to use revolutionary technology which enables extremely low power consumption and heat dissipation in computing.

Reversible logic, which is an actively researched area, is considered as an apt alternative. The employment of reversible logic operations, which do not erase information, virtually dissipates zero heat. Bennett concluded that no energy would dissipate from a system if it would be able to return to its initial state from its final state regardless of what occurred in between [3]. This ability allows us to reproduce the inputs from the outputs. Reversible computing uses a physical mechanism that is thermodynamically as well as logically reversible. They are adiabatic systems that recycle their energy and thus emit very little heat. For a gate to be reversible, the logic function it realizes has to be bijective. That is, there is a one-to-one mapping between the input and output vectors. These circuits can generate unique output vector from each input vector, and vice versa. In theory, every reversible gate has an "inverse" or "dual". By cascading a reversible logic gate with its dual, one can run the logical operations backwards. This technology finds its application in wide range of areas which use extremely low power consumption or heat dissipation.

II. Basic reversible gates

A. Feynman Gate

The most well known (2, 2) reversible gate is the Feynman gate[4]. The logical functions performed by a Feynman gate with input vector (A, B) and output vector (P, Q) are shown in Fig.1.

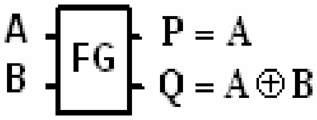

Fig.1.Feynman gate.

In Feynman gate, one of the input bits act as control signal (A). That is, if A= 0 then the output Q follows the input B. If A = 1 then the input B is flipped at the output Q. So it is called as controlled NOT (1-NOT) and also called as quantum XOR because of its popularity in the field of quantum computing. This gate is one-through gate which means that one input variable is also output. Feynman gate acts as copying gate when the second input is zero by duplicating the first input at the output.

B. Toffoli Gate

Toffoli gate[5] is one of the example for (3, 3) reversible gates. Fig.2 shows the Toffoli gate.

X. Susan Christina, M.Sangeetha Justine, K.Rekha, U.Subha and R.Sumathi, Department of Electronics and Communication Engineering Mookambigai College of Engineering , Pudukottai-60, India. fab_jesu@yahoo.co.in

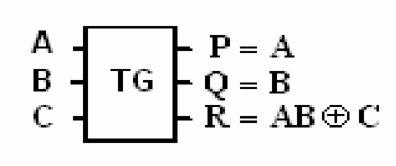

Fig.2.Toffoli gate.

This gate is two through gate because two of its outputs are identical with its inputs. Because of this, Toffoli gate is also known as two controlled NOT (2-CNOT). If the first two input bits are one, then the third output bit is the inverse of third input bit i.e., A = B = 1, then R = C. Toffoli gate performs basic AND operation when zero is given as its third input (C = 0; R = AB). As discussed earlier, any reversible gate has an inverse or dual. The dual of Toffoli gate is also a Toffoli gate and so it is self-reversible.

C. Fredkin Gate

Fredkin gate[6], shown in Fig.3, is a (3, 3) reversible gate which realizes P=A, Q=$\overline{A}$B⊕AC and R=$\overline{AC}$⊕AB where (A, B, C) is the input vector and (P, Q, R) is the output vector. Fredkin gate is also self-reversible as it is its own inverse. It is a conservative gate because the hamming weight (number of logical ones) of an input is same as its output.

It uses ‘A’ as its control input: if A = 0, then the outputs are simply duplicates of the inputs; otherwise if A = 1, then the two input lines (B and C) is interchanged at the output.

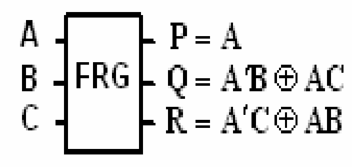

Fig.3. Fredkin gate.

Fredkin gate is a universal gate, that is, we can construct the basic blocks such as AND, OR, NOT and other gates from this Fredkin gate by pre-setting some of its inputs.

D. New Gate

The new gate[7] shown in Fig.4 is another gate which implements all the basic operations like a universal gate.

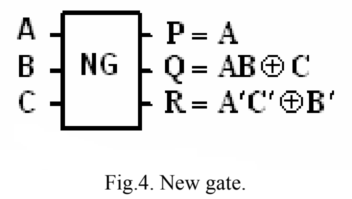

Fig.4. New gate.

E. Peres Gate

Peres gate is another important gate [8] which has a low quantum cost as compared to other gates. It is shown in Fig.5. A single Peres gate can give generate and propagate outputs when the third input C = 0. Because of this, it is quite useful while designing carry look-ahead adders. Two Peres gates can be combined to form a full adder as in Fig.6.

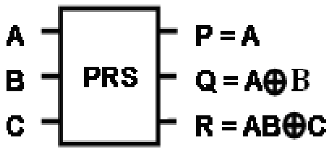

Fig.5.Peres gate.

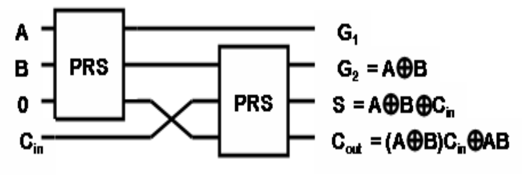

Fig.6. Peres gate as full adder.

F. TSG Gate

The TSG gate [9] is a (4, 4) reversible gate. The most significant aspect of this gate is that it can work singly as a reversible full adder, that is, a reversible full adder can be implemented using a single gate only. This gate is better than its existing counterparts in terms of number of gates and garbage outputs. Minimizing the number of reversible gates and garbage output is one of the major concerns in reversible logic and it can be observed that TSG gate achieves this leading to high speed and low power reversible circuits. The TSG gate is shown in Fig.7 and its working as a full adder is shown in Fig.8.

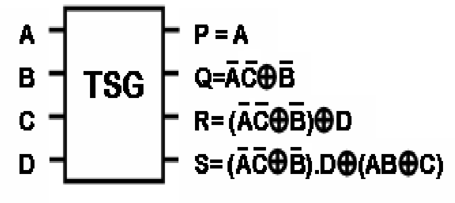
A
B
C
D
TSG
P = A
Q=$\overline{A}\,\overline{C}$⊕$\overline{B}$
R= ($\overline{A}\,\overline{C}$⊕$\overline{B}$)⊕D
S= ($\overline{A}\,\overline{C}$⊕$\overline{B}$).D⊕(AB⊕C)

Fig.7. TSG gate.

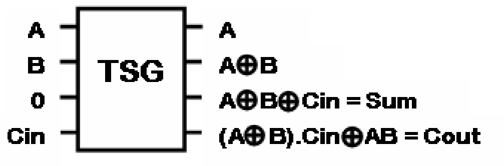

Fig.8.TSG gate working as a reversible full adder.

III. Proposed Circuits

The reversible logic implementation of conventional BCD and carry skip BCD adder has already been proposed by Himanshu et al [10]. In this paper, two adder schemes which are faster in computing carry output are proposed. The implemented reversible circuits are carry select BCD adder and carry look-ahead BCD adder are discussed as follows.

A. Carry Select Scheme

The carry select scheme is better than the carry skip scheme as it is faster in processing the result. In carry skip scheme, carry is skipped only for a particular condition, i.e. when either of the input is one. Therefore, the carry output does not wait for each stage carry to propagate in ripple fashion. The carry select scheme further reduces the delay by selecting a pre-computed sum and carry outputs depending on the carry-in. The carry select concept can be better understood from the Fig.9 [11].

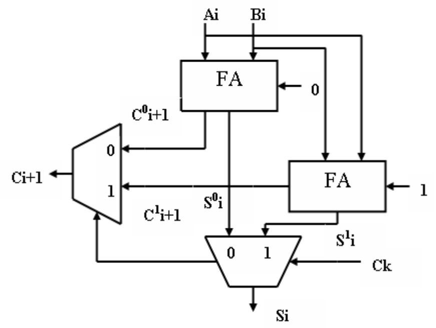

Fig.9.Carry Select scheme.

The basic problem faced in speeding up carry propagation is the fast processing of late carry input. Since this carry-in can have only two values (0 and 1), there is going to be only two possible results; one for carry being zero end the other for carry being one (S0i, C0i+1 and S1i, C1i+1) respectively. These sum and carry can be pre-computed and selected afterwards depending on the late carry-in (Ck).

Selection of the final sum and carry is done by using the following equations;

$$S_i = C_k.S^0{}_i + C_k.S^1{}_i \tag{1}$$

$$C_{i+1} = C_k.C^0{}_{i+1} + C_k.C^1{}_{i+1} \tag{2}$$

The resulting carry select addition scheme requires two full adders (FA), one with Ck = 0 and the other with Ck = 1 and two 2-to-1 multiplexer for each sum bit and carry-out selection.

B. Reversible Logic Implementation of Carry- Select BCD Adder

In this BCD adder, the carry select concept is implemented using five reversible gates which form a block. Two TSG gates which act as full adders are used to compute sum and carry for the two possibilities of carry-in. As fan-out is not allowed in reversible logic, one of the input bits is given to a Feynman gate to duplicate it. The two multiplexers are replaced with Fredkin gates, to select the appropriate sum and carry depending on the carry-in. Reversible carry select BCD adder is given in Fig.10.

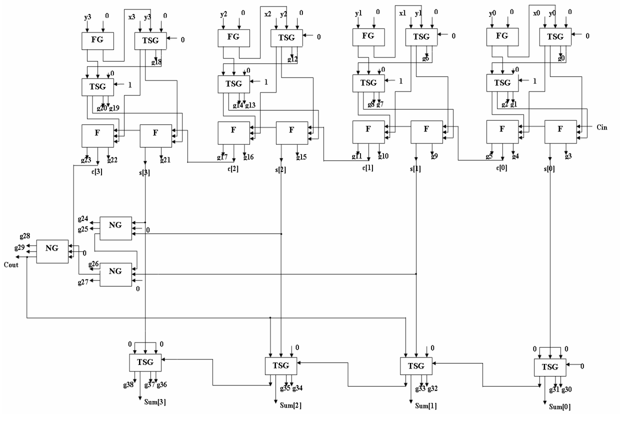

Fig.10.Reversible logic implementation of proposed carry- select BCD adder.

In this BCD adder, four of the described blocks are designed and the usual BCD operation is performed using TSG and New gates. The number of reversible gates used in the proposed reversible logic implementation of carry select BCD adder is twenty seven and the garbage outputs produced is thirty nine.

The resulting simulation output and RTL schematic diagram got after coding using Xilinx ISE 9.2i software.

C. Carry Look-Ahead Concept:

Carry look-ahead adders represent the best possible performance in terms of computing the most significant carries more quickly. However, carry look-ahead adder achieve their speed through parallel carry computation, thus employing a large number of gates [12].

The carry-out function of a full adder using carry look-ahead for ith stage can be realized as,

$$C_{i+1} = g_i + p_i.C_i \tag{3}$$

where,

$$g_i = X_i.Y_i \tag{4}$$

$$p_i = X_i \oplus Y_i \tag{5}$$

The function 'gi' is equal to 1 when both inputs Xi and Yi are equal to 1, regardless of the value incoming carry, Ci. Since in this case ith stage is guaranteed to generate a carry-out, 'g' is called generate function. The function pi is equal to 1 when at least one of the inputs is equal to 1. In this case, a carry output is produced if Ci = 1. The effect is that the carry-in of 1 is propagated through ith stage. Hence 'pi' is called the propagate function.

The first two stages of the carry look-ahead adder which is shown in Fig.11 use the following expressions to implement the carry-out functions;

$$C_1 = g_0 + p_0C_0 \tag{6}$$

$$C_2 = g_1 + p_1g_0 + p_1p_0C_0 \tag{7}$$

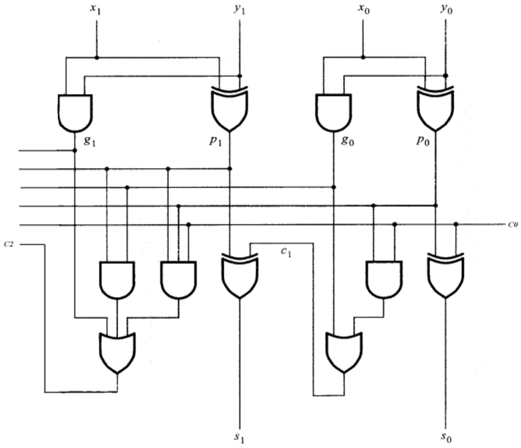

Fig.11 The first two stages of carry look-ahead adder.

In this circuit, C2 is produced just as quickly as C1. This can be extended to next stages and a BCD adder computes sum and carry for bits, we extend it to two more stages so as to achieve the proposed reversible carry look-ahead adder.

D. Reversible Logic Implementation Of Carry Look-Ahead Adder

As seen earlier, a single Peres gate can be used to get the generate and propagate outputs for the input bits of X and Y. Toffoli gates are used to perform the AND function and Fredkin gates are used for OR operation. Feynman gates are used both as copying gates to avoid fan-out and also to perform XOR function to find the sum outputs of each bits.

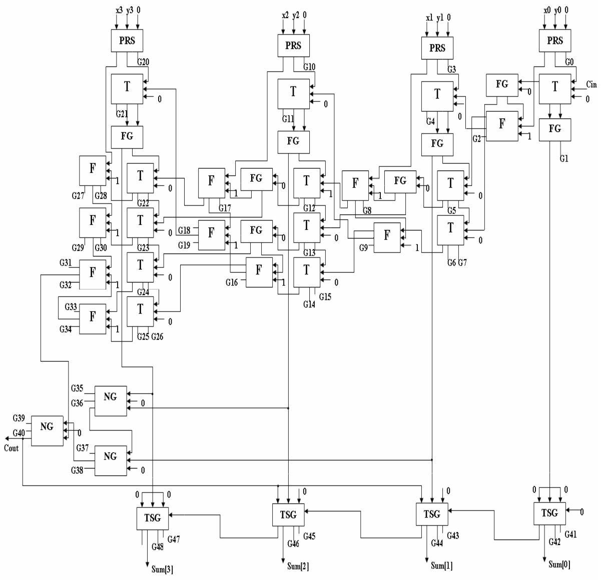

Fig.12 Carry look-ahead BCD adder using reversible gates.

The expressions for the carry-out of each stage are given below.

$$C_1 = g_0 + p_0C_0 \quad (8)$$

$$C_2 = g_1 + p_1g_0 + p_1p_0C_0 \quad (9)$$

$$C_3 = g_2 + p_2g_1 + p_2p_1g_0 + p_2p_1p_0C_0 \quad (10)$$

$$C_4 = g_3 + p_3g_2 + p_3p_2g_1 + p_3p_2p_1g_0 + p_3p_2p_1p_0C_0 \quad (11)$$

The BCD addition is performed by using TSG and New gates. Thus, the total number of gates used in the reversible implementation of carry look-ahead adder is 42 and the number of garbage outputs produced is 49. The reversible implementation of carry look-ahead adder is given in Fig.12.

IV. Results and Discussion

The RTL schematic diagram for carry select BCD adder using reversible gate is shown in Fig.13 and its Xilinx ISE 9.2i simulated output is shown in Fig.14. The schematic diagram for carry look-ahead BCD adder using reversible gate is shown in Fig.15 and its Xilinx ISE 9.2i simulated output is shown in Fig.16.

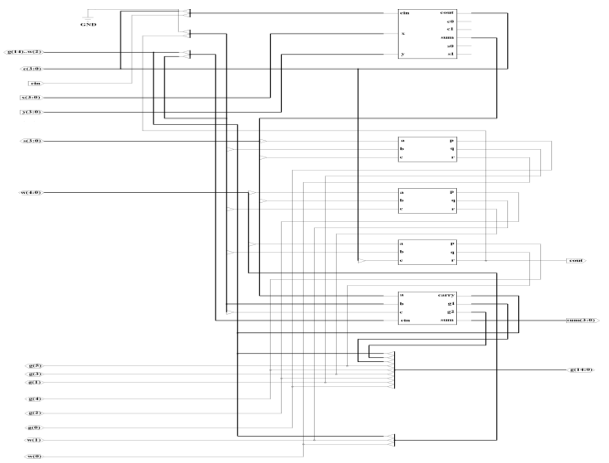

Fig.13 RTL Schematic diagram of reversible carry select BCD adder.

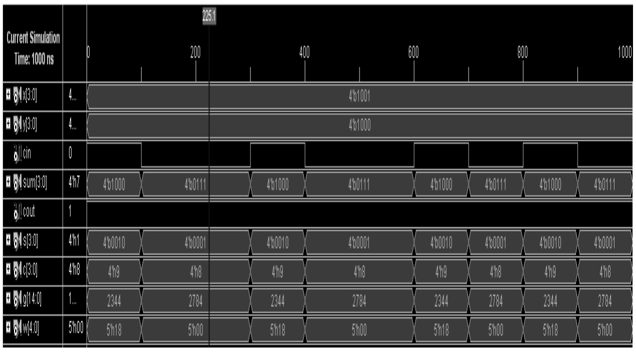

Fig.14 Simulation output for reversible carry select BCD adder

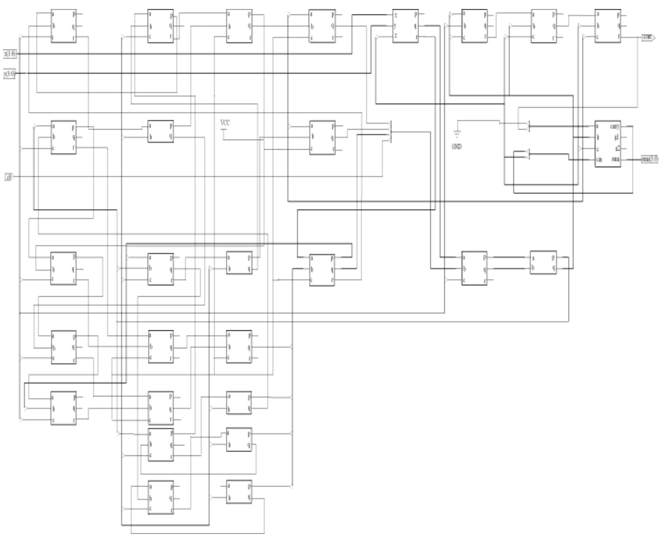

Fig.15 RTL Schematic diagram of reversible carry look-ahead BCD adder.

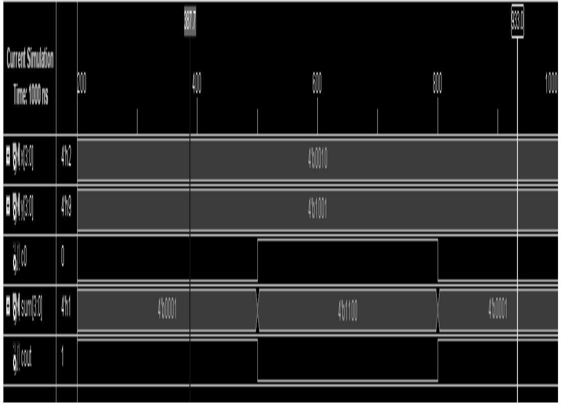

Fig.16 Simulation output for reversible carry look-ahead BCD adder.

The speed up characteristics of various BCD adders are compared and tabulated in table 1.

TABLE I. SPEED-UP CHARACTERISTICS OF ADDITION SCHEMES

Adder Scheme	Description	Delay	Speed-up paths
Ripple adder	Each carry-out waits for its carry-in at every stage.	547.048 ns	Delay is considerable
Carry Skip	If the bits of two inputs are inverse of each other, it skips the carry.	520.048 ns	Cin → Cout
Carry Select	It selects the precomputed sum & carry depending on carry-in.	476.223 ns	Cin → Cout Cin → Si
Carry Look-ahead	All the carries are precomputed i.e. looked ahead for calculation of sum.	456.774 ns	Cin → Cout Cin → Si Xi,Yi → Cout Xi, Yi → Si

V. CONCLUSION

Computers today terribly waste energy and storage capacity. They throw away millions of bits, billions of times every second. These are based on irreversible logic devices, which have been recognized as being fundamentally energy-inefficient for several decades. Truly, the only way we might ever get around this limit is by using reversible computing, which uncomputes bits that are no longer needed, rather than overwriting them. Un-computing bits allows their energy to be recovered and recycled for use in later operation. Before the computer industry reaches the fundamental brick wall of performance and energy constraints of computing devices, reversible computing needs to be fully developed.

This paper proposed novel designs of reversible BCD carry select and carry look-ahead adders. The architecture is designed specially to make them suitable for reversible logic synthesis. The simulation of these circuits has been done and they are ready to be used for designing large reversible systems which is the necessary requirement of quantum computers. These provide a base to build more complex system like the BCD ALU of a primitive quantum CPU.

REFERENCES

[1] R.Landauer, “Irreversibility and Heat Generation in the Computational Process”, IBM Journal of Research and Development, 5,pp. 183-191, 1961.

[2] Dmitri Maslov, “Reversible logic synthesis”, University of New Brunswick, September 2003.

[3] C.H.Bennett, “Logical Reversibility of Computation”, IBM J.Research and Development, pp.525-532, November1973.

[4] Feynman R., 1985. Quantum mechanical computers, Optics News, 11: 11-20.

[5] Toffoli T., 1980. Reversible computing, Tech Memo MIT/LCS/TM-151,
MIT Lab for Computer Science.

[6] Fredkin E. and T. Toffoli, 1982. Conservative logic. Int’l J. Theoretical Physics, 21: 219-253.

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

[8] Peres, A. 1985. Reversible logic and quantum computers. Physical Review A, 32: 3266-3276.

[9] Himanshu Thapliyal, and M.B Srinivas,“A Novel reversible TSG Gate and its Application for Designing Reversible Carry Look Ahead and other Adder Architecture” Accepted in Tenth Asia –Pacific Computer Architecture Conference , Singapore, October 24-26,2005.

[10] Himanshu Thapliyal, Saurabh Kotiyal, M.B Srinivas,“Novel BCD adders and their reversible logic implementation for IEEE 754r format”, 19th International Conference on VLSI Design 2006.

[11] Reto Zimmermann, “Binary adder architectures for cell based VLSI and their synthesis”, Swiss Federal Institute of Technology, Zurich, 1997.

[12] Stephan Brown, Zvonko Vranesic, “Fundamentals of digital logic design with VHDL”, Second edition, 2004.