VLSI Implementation of Reversible Binary Adders in Quantum-Dot Cellular Automata

1 KAVYA SRI.T.
2 SURESH KUMAR.K.V
1 M.TECH., VLSI
2 ASSISTANT PROFESSOR
1,2 NARASIMHAREDDY ENGINEERING COLLEGE

ABSTRACT:

Nowadays exponential advancement in reversible computation has lead to better fabrication and integration process. It has become very popular over the last few years since reversible logic circuits dramatically reduce energy loss. It consumes less power by recovering bit loss from its unique input-output mapping. This paper presents two new gates called RC-I and RC-II to design an n-bit signed binary comparator where simulation results show that the proposed circuit works correctly and gives significantly better performance than the existing counterparts. An algorithm has been presented in this paper for constructing an optimized reversible n-bit signed comparator circuit. Moreover some lower bounds have been proposed on the quantum cost, the numbers of gates used and the number of garbage outputs generated for designing a low cost reversible signed comparator. The comparative study shows that the proposed design exhibits superior performance considering all the efficiency parameters of reversible logic design which includes number of gates used, quantum cost, garbage output and constant inputs. This proposed design has certainly outperformed all the other existing approaches.

Keywords: Reversible Comparator, Quantum Computing, Signed Arithmetic, Low Power, Binary comparators, majority gates.

I. INTRODUCTION

In recent time every sort of computation is getting complex and researchers are facing enormous challenges over billions of arithmetic operations per second. According to Gordon Moore, the transistor count and performance of logic circuits is doubled every two years, this process will continue until semiconductor circuits reach to its physical limit. Thus achieving ultra-speed computation leads us to various kinds of computing technology such as Quantum Information Processing [2], DNA Computing [3], etc. But we live in a world of classical or irreversible architecture where unused energy is dissipated due to power loss. This is because, Landauer [4] proved erasure of each bit of information dissipates at least KT × ln2 joules of energy where K is the Boltzamann's constant and T is the absolute temperature at which the operation is being performed. In 1973, Bennet [2, 5] had shown that energy dissipation problem of VLSI circuits can be overcome by using reversible logic. This is so because reversible computation does not erase any bit of information and consequently it does not dissipate any energy for computation. Generally, reversible logic performs Boolean operations having equal number of inputs and outputs where input states are uniquely mapped to individual output states or vice versa [6]. Also reversible logic should never permit feedbacks. Reversible logic does multiple operations per cycle without losing any input bits. As a result zero power
dissipation would be achieved if a logic circuit consists of reversible gates. Reversible logic can easily be manipulated for Fault Testing [5], Embedded Devices [7], Digital Signal Processing [8], Quantum Dot Cellular Automata [9-10], etc.

Comparison of two binary numbers finds its wide application in general purpose microprocessors, communication systems, encryption devices, sorting networks, etc [11]. This paper presents an n bit signed comparator which has less number of gates, produces less garbage outputs and quantum cost. In the process of designing this architecture, two very efficient reversible gates RC-I and RC-II have also been proposed. Quantum realization of these two gates shows that they significantly improve the overall cost of the proposed n-bit signed binary comparator. With the help of theorems and lemmas the efficiency of reversible logic synthesis of n-bit signed comparator has also been proved in this paper. This design is based on the reversible comparator gate RC-I which produces 2 outputs q= xLy(xless than y)= x'y and r= xGy(x greater than y)= xy' and RC-II gate which produces 3 outputs r= xEy(xequals y)= (x⊕y)', q= xLy(xless than y)= x'y and p= xGy(xgreater than y)= xy'. RC-I has been used to compare two single bits whereas RC-II compares two signed bits therefore they are named as Reversible Comparator gate RC-I and RCII.

Cellular automata

The information storage and transport on quantum-dot cellular automata is not based on the flow of electrical particle current, but on the local position of the charged particles inside a small section of the circuit, called a cellular automaton. This QCA cell has a limited number of quantum-dots, which the particles can occupy, and these dots are arranged such that the cell can have only two polarizations (two degenerate quantum mechanical ground states), representing binary value zero or one. A cell can switch between the two states by letting the charged particles tunnel between the dots quantum mechanically.

The cells exchange information by classical Coulombic interaction. An input cell forced to a polarization drives the next cell into the same polarization, since this combination of states has minimum energy in the electric field between the charged particles in neighboring cells. Information is copied and propagated in a wire consisting of the cell automata. Figure 1 shows the available two cell types, which are orthogonal and have minimal interaction with each other, enabling the coplanar wire crossing, where the wires consist of different cell types and can operate independently on the same fabrication layer. A traditional multi-layer crossing can be constructed with either cell type, but the technology has not been demonstrated yet.

II. Literature Survey

Before going into the detail of reversible n-bit signed comparator, some preliminaries on reversible computations and related topics have been discussed in this section.

Reversible Gate

Reversible Gate is an n × n data stripe block which uniquely maps between input vector Iv= (I0, I1, . . . In) and output vector Ov= (O0, O1, . . . On) denoted as Iv↔Ov
The basic element of a nanostructure based on QCA is a square cell with four quantum dots and two free electrons. The latter can tunnel through the dots within the cell, but, owing to Coulombic repulsion, they will always reside in opposite corners, thus leading to only two possible stable states, also named polarizations. Locations of the electrons in the cell are associated with the binary states 1 and 0.

Adjacent cells interact through electrostatic forces and tend to align their polarizations. However, QCA cells do not have intrinsic data flow directionality. Therefore, to achieve controllable data directions, the cells within a QCA design are partitioned into the so-called clock zones that are progressively associated with four clock signals, each phase shifted by 90°. This clock scheme, named the zone clocking scheme, makes the QCA designs intrinsically pipelined, since each clock zone behaves like a D-latch.

QCA cells are used for both logic structures and interconnections that can exploit either the coplanar cross or the bridge technique. The fundamental logic gates inherently available within the QCA technology are the inverter and the majority gate (MG). Given three inputs a, b, and c, the MG performs the logic function reported in (1) provided that all input cells are associated with the same clock signal clk_x (with x ranging from 0 to 3), whereas the remaining cells of the MG are associated with the clock signal clk_{x+1}:

\[ M(a, b, c) = a \cdot b + a \cdot c + b \cdot c \]
There are several QCA designs of comparators in the literature. A 1-bit binary comparator receives two bits a and b as inputs and establishes whether they are equal, less than or greater than each other. These possible states are represented through three output signals, here named A eq B, A big B, B big A, that are asserted, respectively, when a=b, a>b, and a<b. Full comparators are those that can separately identify all the above cases, whereas non-full comparators recognize just one or two of them. As an example, the comparator designed in and depicted in Fig. 1(a) can verify only whether a=b. Conversely, the circuits shown in Fig. 1(b) and (c), are full comparators. The latter also exploits two 1-bit registers D to process n-bit operands serially from the least significant bit to the most significant one.

With the main objective of reducing the number of wire crossings, which is still a big challenge of QCA designs, the universal logic gate (ULG) \( f(y_1,y_2,y_3)=M(M(y_1,y_2,0),M(y_1,y_3,1),1) \) was proposed and then used to implement the comparator illustrated in Fig. 1(d). It is worth noting that, two n-bit numbers \( A_{(n-1:0)}=a_{n-1}...a_0 \) and \( B_{(n-1:0)}=b_{n-1}...b_0 \) can be processed by cascading n instances of the 1-bit comparator. Each instance receives as inputs the ith bits ai and bi (with i=n−1,...,0) of the operands and the signals \( A_{\text{big}}B_{(i-1:0)} \) and \( B_{\text{big}}A_{(i-1:0)} \). The former is asserted when the sub word \( A_{(i-1:0)}=a_{i-1}...a_0 \) represents a binary number greater than \( B_{(i-1:0)}=b_{i-1}...b_0 \). In a similar way, \( B_{\text{big}}A_{(i-1:0)} \) is set to 1 when \( A_{(i-1:0)} < B_{(i-1:0)} \). The outputs \( A_{\text{big}}B_{(1:0)} \) and \( B_{\text{big}}A_{(1:0)} \) directly feed the next stage. It can be seen that this circuit does not identify the case in which A=B, therefore it cannot be classified as a full-comparator.

The design described exploits a tree-based (TB) architecture and exhibits a delay that in theory logarithmically increases with n. The 2-bit version of such designed comparator is illustrated in Fig. 1(e).

Also the full comparator exploits a TB architecture to achieve high speed. As shown in Fig. 1(f), where 4-bit operands are assumed, one instance of the 1-bit comparator presented for each bit position. The intermediate results obtained in this way are then further processed through a proper number of cascaded 2-input OR and AND gates implemented by means of MGs having one input permanently set to 1 and 0, respectively.

Analyzing existing QCA implementations of binary comparators it can be observed that they were designed directly mapping the basic Boolean functions consolidated for the CMOS logic designs to MGs and inverters, or ULGs. Unfortunately, in this way the computational capability offered by each MG could be underutilized. As a consequence, both the complexity and the overall delay of the resulting QCA designs could be increased in vain.

New Formulations for QCA Implementations of N-Bit full comparators

In this section, four original theorems and two corollaries are enunciated that can significantly increase the speed performances of QCA-based designs of full comparators and can significantly reduce the number of used MGs and inverters with respect to existing comparators, thus reducing also the number of used cells and the overall active area. The Appendix at the end of the paper provides the proofs of the novel theorems and corollaries.

The novel formulations can be exploited in the design of n-bit full comparators splitting the operands \( A_{(n-1:0)}=a_{n-1}...a_0 \) and \( B_{(n-1:0)}=b_{n-1}...b_0 \) into a proper number of 2-bit and 3-bit sub words that can be compared applying Theorems 1 and 2. The intermediate results obtained in this way can be then further processed by applying Theorems 3 and 4 together with Corollaries 1 and 2.
Theorem 1: If \( A_{(k-2:k-1)} = a_{k-1}a_{k-2} \) and \( B_{(k-2:k-1)} = b_{k-1}b_{k-2} \), with \( k=2, 4, ..., n-2 \), \( n \) are two 2-bit sub words of then-bit numbers \( A_{(n-1:0)} \) and \( B_{(n-1:0)} \) respectively, then \( \text{A big B}_{(k-2:k-1)} \) as defined in (2) is equal to 1 if and only if \( A_{(k-2:k-1)} > B_{(k-2:k-1)} \), \( \text{B big A}_{(k-2:k-1)} \) as defined in (3) is equal to 0 if and only if \( A_{(k-2:k-1)} < B_{(k-2:k-1)} \).

\[
\text{A big B}_{(k-2:k-1)} = M \left( a_{k-1}, \overline{b_{k-1}}, a_{k-2} \right) \cdot M \left( a_{k-1}, b_{k-1}, \overline{b_{k-2}} \right) \\
\text{B big A}_{(k-2:k-1)} = M \left( a_{k-1}, \overline{b_{k-1}}, \overline{b_{k-2}} \right) + M \left( a_{k-1}, b_{k-1}, a_{k-2} \right)
\]

Theorem 2: If \( A_{(k-3:k-1)} = a_{k-1}a_{k-2}a_{k-3} \) and \( B_{(k-3:k-1)} = b_{k-1}b_{k-2}b_{k-3} \), with \( k=3, 6, ..., n-3 \), \( n \) are 3-bit sub words of then-bit numbers \( A_{(n-1:0)} \) and \( B_{(n-1:0)} \) respectively, then \( \text{A big B}_{(k-3:k-1)} \) as defined in (4) is equal to 1 if and only if \( A_{(k-3:k-1)} > B_{(k-3:k-1)} \), \( \text{B big A}_{(k-3:k-1)} \) as given in (5) is equal to 0 if and only if \( A_{(k-3:k-1)} < B_{(k-3:k-1)} \).

\[
\text{A big B}_{(k-3:k-1)} = M \left( a_{k-1}, \overline{b_{k-1}}, a_{k-2}, \overline{a_{k-3}} \right) \cdot M \left( a_{k-1}, \overline{b_{k-1}}, b_{k-2}, a_{k-3} \right) \\
\text{B big A}_{(k-3:k-1)} = M \left( a_{k-1}, \overline{b_{k-1}}, \overline{b_{k-2}}, a_{k-3} \right) + M \left( a_{k-1}, b_{k-1}, a_{k-2}, \overline{a_{k-3}} \right)
\]

III. PROPOSED METHOD

A QCA is a nano-structure having as its basic cell a square four quantum dots structure charged with two free electrons able to tunnel through the dots inside the cell. Because of Coulombic repulsion, the two electrons will forever reside in opposite corners. The locations of the electrons in the cell determine two possible stable states that can be associated to the binary state 1 and 0. Although adjacent cells interact through electrostatic forces and tend to arrange in a line their polarizations, QCA cells do not have intrinsic data flow directionality. To achieve controllable data directions, the cells inside a QCA design are partitioned into the so-called clock zones that are progressively associated to four clock signals, each phase shifted by 90°. This clock system named the zone clocking scheme, makes the QCA designs intrinsically pipelined, as each clock zone behaves like a D-latch. QCA cells are used for both logic designs and interconnections that can exploit either the coplanar cross or bridge technique. The fundamental logic gates inherently available within the QCA technology are the inverter and the MG. Given three inputs \( a, b, \) and \( c \), the MG perform the logic function reported in (1) provided that all input cells are associated to the same clock signal \( \text{clkx} \) (with \( x \) ranging from 0 to 3), whereas the remaining cells of the MG are linked to the clock signal \( \text{clkx+1} \).

Fig 2 Novel n-bit adder

Several designs of adders in QCA exist in literature. The RCA [11], [13] and the CFA [12] process \( n \)-bit operands by cascading \( n \) full-adders (FAs). Even although these addition circuits use different topologies of the generic FA, they include a carry-in to carry-out path consisting of one MG, and a carry-in to sum bit path contain two MGs plus one inverter. As a importance, the worst case
computational paths of the n-bit RCA and then-bit CFA consist of (n+2) MGs and one inverter. A CLA design formed by 4-bit slices was also presented. In particular, the auxiliary propagate and generate signals, namely are computed for each bit of the operands and then they are grouped four by four. Such a designed n-bit CLA has a computational path composed of 7+4×(log₂ n) cascade MGs and one inverter. This can be easily verified by observing that, given propagate and generate signals, to compute grouped propagate and grouped generate signals; four cascade MGs are introduced in the computational path. In addition, to calculate the carry signals, one level of the CLA logic is required for each factor of four in the operands word-length. This means that process n bit addends, log₂ n levels of CLA logic are required, every contributing to the computational path with four cascaded MGs. Finally, the calculation of sum bits introduces two further cascaded MGs and one inverter. Apart from the level required to compute propagate and generate signals, the prefix tree consists of 2 × log₂ (n/8) + 2 × log₂ (n/8) − 1 MGs and one inverter. The abovementioned approach can be applied also to design the BKA. In this case the overall area is reduced with respect to, but maintaining the same computational path. By applying the decomposition method demonstrated, the computational paths of the CLA and the CFA are reduced to 7 + 2 × log₂ (n/8) MGs and one inverter and to (n/2) + 3 MGs and one inverter, respectively.

IV. Novel QCA Adder

To introduce the novel architecture proposed for implementing ripple adders in QCA, let consider two n-bit addends A = an−1,...,a0 and B = bn−1,...,b0 and suppose that for the ith bit position (with i = n−1,...,0) the auxiliary propagate and generate signals, namely pi = ai + bi and gi = ai· bi, are computed. ci being the carry produced at the generic (i−1)th bit position, the carry signal ci+2, furnished at the (i+1)th bit position, can be computed using the conventional CLA logic reported in (2). The latter can be rewritten as given in (3), by exploiting Theorems 1 and 2 demonstrated in [15]. In this way, the RCA action, needed to propagate the carry ci through the two subsequent bit positions, requires only one MG. Conversely, conventional circuits operating in the RCA fashion, namely the RCA and the CFA, require two cascaded MGs to perform the same operation. In other words, an RCA adder designed as proposed has a worst case path almost halved with respect to the conventional RCA and CFA.
Equation (3) is exploited in the design of the novel 2-bit module shown in Fig. 1 that also shows the computation of the carry \( c_{i+1} = M(p_igici) \). The proposed n-bit adder is then implemented by cascading \( n/2 \) 2-bit modules as shown in Fig. 2(a). Having assumed that the carry-in of the adder is \( c_{in} = 0 \), the signal \( p_0 \) is not required and the 2-bit module used at the least significant bit position is simplified. The sum bits are finally computed as shown in Fig. 2(b).

It must be noted that the time critical addition is performed when a carry is generated at the least significant bit position (i.e., \( g_0 = 1 \)) and then it is propagated through the subsequent bit positions to the most significant one. In this case, the first 2-bit module computes \( c_2 \), contributing to the worst case computational path with two cascaded MGs. The subsequent 2-bit modules contribute with only one MG each, thus introducing a total number of cascaded MGs equal to \( (n-2)/2 \). Considering that further two MGs and one inverter are required to compute the sum bits, the worst case path of the novel adder consists of \( (n/2) + 3 \) MGs and one inverter.

\[
\begin{align*}
    c_{i+2} &= g_{i+1} + p_{i+1} \cdot g_i + p_{i+1} \cdot p_i \cdot c_i \\
    c_{i+2} &= M(M(a_{i+1}, b_{i+1}, g_i) M(a_{i+1}, b_{i+1}, p_i) c_i).
\end{align*}
\]

V. Simulation and Comparison

The proposed designs of reversible binary comparators have been functionally verified through simulations in DSCH. The simulation results show that the comparators give perfect output for all possible combinations of inputs. The simulation results for 1-bit, 2-bit, 3-bit and 4-bit reversible binary comparator are shown in Fig
VI. Conclusion

This paper presents a systematic approach to design an n-bit signed comparator in reversible mode. With the help of comparisons and theorems it has been shown that the paper exhibits its efficiency over all the existing designs in terms of all the performance parameters. Quantum cost minimization is the strength of the proposed architecture. Since comparison of two numbers can be useful in many operations inside the microprocessor, communication systems, encryption devices, sorting networks and many more, it can be expected that the optimized low cost design will surely bring more efficiency and scalability in the world of reversible computing. Currently, studies are being performed to extend this design in a form of tree based architecture to bring more features of reversibility.

References