Efficient Design of Nano Scale Adder and Subtractor Circuits using Quantum dot Cellular Automata

Javeed Iqbal Reshi*, M. Tariq Banday†

Department of Electronics and Inst. Technology, University of Kashmir, Srinagar, India.
*javeedreshi@gmail.com, †sgrmtb@yahoo.com

Keywords: Quantum dot Cellular Automata; QCA; CMOS Technology; QCADesigner; Adder; Subtractor.

Abstract

Low power dissipation, high packaging density and operation at Tera-hertz frequencies are highly desired features in the design and fabrication of logic circuits. Promising realization of these highly desired features have made Quantum dot Cellular Automata (QCA) a prospective replacement to the conventional CMOS technology for the fabrication of logic circuits in nano scale range. The traditional technologies including CMOS use voltage levels to represent binary states, whereas Quantum dot Cellular Automata uses polarization of electrons to represent binary states. In any arithmetic design, modulo-2 addition is required and most suitable gate for obtaining this function is an XOR gate. An efficient XOR gate can facilitate the design of arithmetic circuits with less number of cells and thus can save area of occupation on the chip. This paper proposes various designs for adder and subtractor circuits in QCA that require reduced number of cells and therefore can save chip area to a larger extent. The proposed design is suitable for applications in arithmetic systems in nano scale range. The experimentation with the proposed designs have been successfully verified through the QCADesigner tool.

1 Introduction

Over several decades Quantum dot Cellular Automata has gained a lot of attention as it offers extremely low power, operation at very high frequency (usually in Tera-Hertz) and high density for implementing any digital logical circuit. The main motivation of scaling is to provide high performance devices which can operate at very high frequency ranges usually in Tera-Hertz and have low power consumption. The conventional CMOS technologies have become resistant to scaling, because of fabrication difficulties, sub-threshold leakage, high leakage current, and gate oxide tunneling leakage. In order to overcome these difficulties, the ITRS report [1] proposes several possible solutions and the QCA emerges as a suitable alternative candidate for CMOS technology. In QCA the information storing procedure is different, as position of electrons decide the logic states. Dr. Craig Lent first proposed this novel idea. In the QCA technology the primitive structure is a QCA cell [2]. In this paradigm the QCA cell consists of four quantum dots at the four corners of a square geometry where mobile free electrons are pushed and are confined to any of two quantum dots [3] at the opposite diagonals. In QCA the dataflow from one point to another is accomplished by means of columbic interaction of the electrons of neighbouring cells [4]. Polarization of one QCA cell affects the polarization of its neighboring cells, therefore, a specific arrangement of QCA cells results in desired implementation of a Boolean function.

The remaining paper is organized as follows; section II gives a brief review of basic QCA based computational elements. In section III, an evaluation of previously related designs of QCA X0R gate, adder and subtractor circuits has been presented. Section IV proposes new designs of QCA based XOR gate, adder and subtractor circuits along with experimentation results and comparison with previous related designs.

2 Background study

The basic computational element of Quantum dot cellular automata technology is the QCA cell which consists of four quantum dots positioned at four corners of cell and two mobile electrons confined within the cell [5]. A QCA cell stores the logic states as the polarization of electrons. The two possible stable polarization of electrons within a QCA cell are representing two binary logic states. By proper arrangement of QCA cells various primitives QCA devices can be designed. The major QCA elements are majority gate, QCA inverter, and the QCA wire. The majority gate is obtained by employing 5 QCA cells. The simplest type of QCA structure is an inverter which can be realized by using only two quantum cells. QCA wire is another important component in QCA technology which is formed by 45° or 90° cascaded arrangement of QCA cells. A unique feature of QCA technology is that it allows two wires to cross in a single plane. This can be accomplished by using coplanar or multilayer crossover. In QCA circuits the synchronization and information flow are accomplished by using the QCA clock. QCA clock works on four phase clocking mechanism. These phases are switch, hold, release and relax. During these phases the inter-dot barrier is varied such that the cells under the influence of clock either remain in the ground state or change to some fixed polarization state.

3 Related WORK

In this section, previously reported designs of XOR gate, adder and subtractor circuits are discussed and reviewed.
Several designs for XOR gate have been proposed which are different from one another in terms of number of cells used, total area of occupancy, latency, etc. Shah proposed an XOR gate employing 5 cells with cell area of 0.011µm² and the total design area of 0.060 µm² [19]. The latency in this design is 4 clock phases. Jagarlamudi implemented an XOR gate, which required 58 cells with cell area of 0.018µm² and total area of 0.090µm² [12]. The latency of this design is 3 clock phases. Lakshmi and Mustafa proposed two designs of XOR gate requiring 64 and 41 cells [14, 15]. The cell area in these designs are 0.020µm² and 0.013µm². Both of these implementations of XOR gates have latency of 4 clock phases. Roy and Teja proposed XOR gates, but both of these designs require large number of cells for their implementation [15, 16]. The latency in the design is very high which makes it inefficient. Shahidinejad, Waje and Chabi proposed three designs of XOR gate [17, 18], 19]. These designs require 41, 36 and 30 cells with cell areas of 0.0132µm², 0.0116µm², 0.0097µm² respectively. All these designs are efficient in term of cell count and latency. Dallaki proposed an efficient design of an XOR gate [20] which requires 30 cells for implementation with cell area of 0.0097µm² and total area of 0.0233µm² and a latency of 3 clock cycles. Santra and Rani proposed a half adder circuit that requires only 63 cells for its realization with cell area of 0.0294µm² and total area of 0.0176µm². Their other design [27] uses 62 cells with cell occupation area of 0.0200µm² and total area of 0.0383µm². In both designs the latency is 4 clock phases. Wang, Walus and Jullien have designed a full adder circuit using 5 majority gates and 3 inverters [21]. This design is implemented on single layer and requires 145 cells with latency of 10 clock phases. Kyosum and Safavi implemented different full adder circuits using the 220, 180 and 150 cells [22, 24, 25]. All these designs employ large number of cells and the delay between the input and output is considerably high. Ahmad and Ebrahimi in their full adder circuit designs succeeded to reduce total number of cells, however, the delay considerably increased in these designs [26, 28]. Besides these, a number of full adder circuits have also been reported and the impetus in each design is on the optimized adder circuit designs.

Subtractor is another important arithmetic circuit, which performs the subtraction of inputs and provides the two outputs referred as difference and borrow. Several researchers have proposed subtractor circuit designs. Dallaki and Lakshmi proposed half subtractor circuits employing 55 and 77 cells in [20, 29]. The area occupancy and the delays in the two designed circuits remained 0.0178µm², and 0.0249µm² with a latency of 3 clock phases. The authors have also implemented a full adder circuits that consumes 136, 178 cells. The areas covered under the cells for the designs are 0.0440µm², and 0.0576µm². The latency in these reported designs remained 7 and 8 clock phases. Santanu proposed an optimized design of full subtractor that require only 108 cells for implementation with cell area of 0.0349µm² and total area of 0.0926µm² [25]. The latency of the reported design is 8 clock phases. In almost all the reported design of adder and subtractor circuits, the latency of sum, carry in case of adder circuits and difference, borrow in case of subtractor circuits are different thereby will produce the incorrect results when used in parallel adder and subtractor circuits.

4 Proposed work

This section discusses efficient designs of proposed adder and subtractor circuits. The proposed designs have been compared with the previously designed structures for validating their efficiency.

4.1 Design of XOR gate

The key block that is required for designing adder and subtractor circuits is an XOR gate which performs modulo-2 addition. In order to realize adder and subtractor circuits, an XOR function is required to be implemented. In adder circuit an XOR gate is used to obtain both SUM, CARRY outputs. An XOR gate can also be employed to produce the DIFFERENCE and BORROW outputs in case of subtractor circuit. Thus an efficient design of XOR gate will permit design of optimized arithmetic circuits. Several XOR gate designs are available in the literature. In this work the design proposed by Chabi, et.al, [19] has been optimized in terms of number of QCA cells and required area. The proposed optimized design has reduced the complexity by utilizing single layer design rather than multi-layer design. The proposed optimized XOR layout is shown in Figure 1.

![Figure 1: Schematic of the XOR gate [19]](image)

In this arrangement the five input majority gate and the three input majority gate is used in a feedback mode so as to provide the functionality of an XOR gate. The required XOR function can be obtained as:

\[ M5(X, M3(\overline{A}, 0, B), M3(\overline{A}, 0, B), \overline{B}, 1) \] \quad \quad (1)

\[ M5(A, \overline{A}B, \overline{A}B, \overline{B}, 1) = \overline{A}B + A\overline{B} \] \quad \quad (2)

The QCA implementation of the proposed XOR gate is shown in Figure 2 and Figure 3 respectively.

![Figure 2: QCA implementation of XOR gate](image)
A comparative statement of the already proposed XOR gates is mentioned in Table 1.

<table>
<thead>
<tr>
<th>XOR Designs</th>
<th>Cell Count</th>
<th>Cell Area (µm²)</th>
<th>Total Area (µm²)</th>
<th>Latency (in Clock Phases)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Shah et al [10]</td>
<td>35</td>
<td>0.011</td>
<td>0.06</td>
<td>4</td>
</tr>
<tr>
<td>Atips [11]</td>
<td>88</td>
<td>0.028</td>
<td>0.09</td>
<td>4</td>
</tr>
<tr>
<td>Jagarlamudi. et al [12]</td>
<td>58</td>
<td>0.018</td>
<td>0.10</td>
<td>3</td>
</tr>
<tr>
<td>Lakshmi et al [13]</td>
<td>64</td>
<td>0.020</td>
<td>0.07</td>
<td>4</td>
</tr>
<tr>
<td>Mustafa et al [14]</td>
<td>41</td>
<td>0.013</td>
<td>0.07</td>
<td>4</td>
</tr>
<tr>
<td>Roy et al [15]</td>
<td>95</td>
<td>0.030</td>
<td>0.1415</td>
<td>9</td>
</tr>
<tr>
<td>Teja et al [16]</td>
<td>121</td>
<td>0.039</td>
<td>0.1574</td>
<td>4</td>
</tr>
<tr>
<td>Shahidinejad et al [17]</td>
<td>41</td>
<td>0.0132</td>
<td>0.0262</td>
<td>3</td>
</tr>
<tr>
<td>Waje et al [18]</td>
<td>36</td>
<td>0.0116</td>
<td>0.0291</td>
<td>3</td>
</tr>
<tr>
<td>Chabi et al [19]</td>
<td>30</td>
<td>0.0097</td>
<td>0.0204</td>
<td>3</td>
</tr>
<tr>
<td>Dallaki et al [20]</td>
<td>30</td>
<td>0.0097</td>
<td>0.0233</td>
<td>3</td>
</tr>
<tr>
<td><strong>Proposed XOR Design</strong></td>
<td><strong>25</strong></td>
<td><strong>0.0081</strong></td>
<td><strong>0.0155</strong></td>
<td><strong>3</strong></td>
</tr>
</tbody>
</table>

Table 1: Comparison of various XOR gates

From table 1, it is evident that the proposed XOR design is efficient in terms of cell count, cell area and the total area. Therefore, by employing the proposed XOR gate for designing the various combinational circuits will definitely reduce the overall cell count and the area required.

### 4.2 Design of Adders

A half adder is combinational circuit which performs binary addition of the two inputs and produces two outputs labelled as SUM and CARRY. Considering A and B as the two inputs of the half adder and SUM and CARRY as the outputs, the relationship between the inputs and the outputs is given as:

\[
SUM = A \oplus B + A \bar{B} = A \oplus B \quad \ldots (3)
\]

\[
CARRY = AB \quad \ldots (4)
\]

In order to obtain the SUM function at the output, an XOR gate is employed and the CARRY is obtained by using an AND gate. The QCA outline and the simulation result of the half adder circuit are shown in Figure 4 and Figure 5 respectively.

A full adder has three inputs in which the third input is considered as the previous carry. Considering a full adder with three inputs labelled as A, B, and C and the two outputs labelled as SUM and CARRY, the relationship between the inputs and the outputs is given as:

\[
SUM = A \oplus B \oplus C \quad \ldots (5)
\]

\[
CARRY = AB + BC + AC \quad \ldots (6)
\]

The resulting SUM terminal will only require two XOR gates, while as the CARRY operation can be carried out using a single majority gate. The QCA arrangement and the simulation result of the full adder circuit are shown in Figure 6 and Figure 7 respectively.
4.3 Design of Subtractors

A subtractor is a combinational circuit which performs the subtraction of the inputs. In half subtractor there are two inputs called as minuend and subtrahend and two outputs referred as difference and borrow. In case of half subtractor there is no previous borrow and the input is restricted to only two. Considering a half subtractor with two inputs labeled as A, B and two outputs labelled as DIFF & BORROW, the relationship between the inputs and the outputs is given as:

\[
\text{DIFF} = A \oplus B \quad \ldots (7)
\]
\[
\text{BORROW} = \overline{A}B \quad \ldots (8)
\]

For a half subtractor one XOR gate and an AND gate is required for its implementation. The QCA implementation and the simulation result of the half subtractor are shown in Figure 8 and Figure 9 respectively.

Unlike a half subtractor, a full subtractor takes account of previous borrow as well. Consider a full subtractor with three inputs labelled as A, B, and C and two outputs labelled as

\[
\text{DIFF} = A \oplus B \oplus C \quad \ldots (9)
\]
\[
\text{BORROW} = \overline{A}B + (A \oplus B)C \quad \ldots (10)
\]

The QCA design and the simulation result of the proposed full subtractor are shown in Figure 10 and Figure 11 respectively.

4.4 Comparison

The designs of adder and subtractor circuits proposed in this section have been compared with the previously reported designs in term of cell count, cell area and latency. This comparison is mentioned in Table 2.

<table>
<thead>
<tr>
<th>Logic Structure</th>
<th>Cell Count</th>
<th>Cell Area (µm²)</th>
<th>Total Area (µm²)</th>
<th>Latency (in Clock Phases)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Santanu et al [25]</td>
<td>63</td>
<td>0.0204</td>
<td>0.01760</td>
<td>4</td>
</tr>
<tr>
<td>Rani et al [27]</td>
<td>62</td>
<td>0.0200</td>
<td>0.05832</td>
<td>4</td>
</tr>
<tr>
<td><strong>Proposed Half Adder</strong></td>
<td><strong>46</strong></td>
<td><strong>0.0149</strong></td>
<td><strong>0.0356</strong></td>
<td><strong>3</strong></td>
</tr>
<tr>
<td>Wang et al [21]</td>
<td>145</td>
<td>0.0469</td>
<td>0.1377</td>
<td>10</td>
</tr>
</tbody>
</table>
Table 2: Comparison of various adder subtractor circuits

The proposed designs proved to be efficient in term of parameters like cell count, cell area, total area and latency and the complexity. All designs in the Table 2 except one proposed by Dallaki in [20] have the different delays in the SUM/CARRY and DIFF/BORROW terminals, which can provide the wrong results if used in the advanced circuits like parallel adders. Practically when an input combination is applied at the adder/subtractor circuits they should provide the outputs simultaneously with equal delays. The proposed designs of adder and subtractor circuits have taken this thing into consideration and hence both the outputs of adder and subtractor circuits provide the same amount of delay. If the proposed circuits would have been designed like the previous ones, the cell count would have been further reduced. The QCA Designer 2.0.3 were used to carry out the simulations with cell size of 18x18 nm, cell spacing of 2nm, Dot diameter of 5nm, 12800 number of samples, and radius effect from 65.000000 nm.

Conclusion and future scope

This paper investigates the QCA technology as an alternative to conventional CMOS technology for implementing the nano-scale adder subtractor circuits. The proposed design of adder and subtractor circuits proved to be efficient in term of cell count, area, delay from input to output and complexity. The improvement in the performance parameters of the proposed circuits will help in less power consumption. Furthermore, the proposed circuits find an extensive use in designing the complex arithmetic systems as well.

Acknowledgements

This research work has been supported by University Grants Commission (UGC) of India under its SAP programme.

References


