DESIGN of HIGH SPEED, LOW AREA, CARRY FLOW BCD ADDER in QCA

D. Ajitha 1
Assistant Professor, Dept of ECE, S.I.T.A.M.S., Chittoor, A.P. India.
ajithavijay1@gmail.com

K. Venkata Ramanaiah2
Professor, Dept of ECE, N.B.K.R.I.S.T. Vidyanagar, Nellore, A.P. India.
ramanaiahkota@gmail.com

V. Sumalatha3
Associate Professor, Dept of ECE, J.N.T.U.C.E.A, Ananthapur, A.P. India.
sumaatp@yahoo.com

Abstract: We can overcome the CMOS scaling problems with emerging Nanotechnology. In Nanotechnology the basic building block of digital design is QCA. The problem in design of decimal adders on Quantum dot cellular automata with reduction of QCA primitives is very limited. In this paper we present a BCD adder with less number of QCA primitives and it is compared with existing BCD adder designs. Our proposed 4-bit QCA-BCD adder results yield increase in speed 66% and reduction in area approximately and 93% respectively compared to the best existing design [1].

Index terms: CMOS limitations, BCD/Decimal adder, Nano electronics, QCA and Majority gate.

Introduction: Traditional digital circuit technologies are mostly based on the CMOS technology, Where as these are reaching their practical and theoretical limits, due to the continuous down scaling of CMOS because of this the current silicon transistor technology faces the problems such as Quantum effects [4] etc, this motivates the researchers with alternative digital logic styles at nano scale.

QCA is one of the promising future solution and it was first introduced in 1993 by lent [5]. Recent papers showed that the QCA can achieve high density, Fast switching speed and room temperature operation [6], and a solution for CMOS scaling. QCA (Quantum dot cellular automata) is a new computation paradigm which encodes binary information by charge configuration with in a cell. The binary value propagates from input to output by couloumbic interactions between the cells. This revolutionary change leads to provide a solution for transistor less computation at the nano scale.

In design of digital arithmetic circuits binary adder is basic operation. Because by using binary adder we can compute most of the operations like addition, multiplication, division, etc. For human understanding we need to display binary data in decimal format. So we need to convert binary data to decimal form or Binary Coded Decimal (BCD) format for displaying.

Instead of converting binary data into BCD form we can perform all operations in decimal format, so that it can reduce hardware complexity. Since addition is a basic operation so we would like to design Decimal Addition or BCD addition in Nanotechnology especially in QCA. In QCA operations performed using polarization instead of electrons flow in CMOS.

The design of QCA based 4-bit BCD adder reported in [1-3]. The QCA based BCD adder design in [1] gives low area and delay compare to [3]. But the results in [1] are not optimal because it didn’t use optimal QCA full adder since full adder is basic unit for BCD adder. In this paper proposed QCA BCD adder for high speed and low in area is Carry Flow Adder (CFA) [2]. CFA provides high speed and small in area among all existing QCA full adders in literature. Our proposed 4-bit QCA-BCD adder results yield a reduction in delay and area approximately 66% and 93% respectively compared to the best existing design [1],and the CFA based method in [10]. We also presented detailed simulation results of QCA BCD adder. In next sections we explain QCA basics and subsequent sections we present design of QCA BCD adder. Final section gives comparisons with [1].

QCA basics: QCA is based on electrons confining in dots and each cell has four quantum Dots [2]. The four dots are located in the corners of squares structure as shown in fig 1. The electrons tunnel through neighboring dots to the proper location during the clock transition.

Fig 1: Schematic of a QCA cell: (a): Logic ‘1’ (b): Logic ‘0’.
**QCA Clock Zones:** QCA cells are clocked using a four-phase clocking scheme. A clock cycle in QCA logic can be divided into four phases, namely: 1) Switch, 2) Hold, 3) Release, 4) Relax [7]. Fig 2 shows QCA Clock Zones with 4 phase clocking scheme. One clock zone to next clock zone 90 degree phase shift is present. The cells begin computing during the switch phase of clock zone 0 (high to low), electrons in QCA cell polarize as per the input polarization and it holds (clock 1) value during the low state. The cell is released when the clock (2 & 3 one) is in the low to high state & inactive during the high state (relax).

A (logic) wire is nothing but series of QCA cells. Two types of crossovers are used to build various circuits (they are 1). Coplanar crossover, 2). Multilayer crossover. Based on comparison Multilayer crossovers [9] are used to construct all designs in this paper.

**Majority Gate & Basic gates as basic elements:**

Fig 3(a), (b) shows the majority gate and its layout, fig 3 (c), (d) shows NOT gate and its layout. By using these two gates various Boolean expression are realized.

![Majority gate layout](image)

The logic equation for a Majority gate is $M(A,B,C) = AB + BC + CA$. By using this gate expression all the basic gates & any defined function can be derived. Basic gates AND & OR gates were constructed. Fig 3(e),(f) is the AND gate and its layout, Fig 3(g),(h) is the OR gate and its layout.

The AND gate expression is derived as $A.B = M(A,B,0)$

![AND gate layout](image)
It is constructed by regular expression & modifying the Majority gate in the required expression.

The OR gate expression is derived as \( A + B = M(A, B, 1) \).

**Related work:**

**BCD Adder:**

Numbers are represented in digital computer either in binary or in decimal through a binary code. When specifying data, the user like to give the data in decimal form the input. Decimal numbers are stored internally in the computer by means of a decimal code.

In BCD number representation each decimal digit requires at least 4 binary storage elements. Consider the decimal numbers are constructed to binary when arithmetic operations are done internally with numbers represented in binary. Consider the arithmetic addition of two decimal digits in BCD together with a possible carry form a previous stage, each input digit does not exceed 9; the output sum cannot be greater than 9 + 9 + 1 = 19 & 1 in the sum being Input carry.

“A BCD Adder is a Circuit that adds two BCD digits in parallel & produces a sum also in BCD”. In order to get the final sum or result in the exact BCD format, for that internal correction logic is needed is implemented by majority gate. In this design the two decimal digits together with the input carry are fast added in the top 4-bit binary ripple carry adder to produce the binary sum.

\[
C_{out} = k + Z_0(Z_2 + Z_4).
\]

If Output carry=0 nothing added and when it is equal to one or when product of 2\(^{nd}\) bit & 4\(^{th}\) bit equal to ‘1’ or 3\(^{rd}\) bit & 4\(^{th}\) bit product is ‘1’ means 0110 is added i.e., (6) to the binary sum through the bottom 4-bit binary adder in the conventional BCD adder as shown in figure (4) In this design the bottom circuit is minimized to get low area & low delay with less number of QCA.

**Construction of BCD adder by using 4-bit full adder using majority gate:**

**Full Adder:**

One bit full adder can be constructed by using the following structure of majority gate.
\[ S_i = \sum (1,2,4,7) \]

\[ S_i = \overline{a_i}b_i\overline{c}_i + a_i\overline{b}_i\overline{c}_i + a_ib_i\overline{c}_i + a_ib_ic_i \]

\[ C_{i+1} = \sum (3,5,6,7) \]

\[ C_{i+1} = a_i b_i + b_i c_i + a_i c_i \]

By using the above equation full adder was constructed and then 4-bit Ripple carry adder.

**Layout diagram and Simulation Results of 4-Bit BCD adder:**

The final implementation of proposed BCD adder is shown in fig 6 (a) QCA layout and in fig 6(b) Simulation results of 4-bit BCD adder by using a QCA designer tool. In the proposed design
Characteristics Reported Design[1] Reported Design[10] Proposed design

Number of Cells 1903 932 750

Estimated area by QCA Designer tool(µm²) 5.50 1.36 0.89

Computation time 9 4.75 3

Conclusion:

In this paper we have presented an efficient QCA design for decimal adder. The results shows that in minimization of QCA primitives by using CFA adder and it is compared with the prior work as listed in the above table. Simulation results were obtained by QCAD designer with Coherence vector engine.

References:


