

# EFFICIENT ARCHITECTURE OF BCD SUBTRACTOR BY MAJORITY GATES IN VLSI

Saima Parveen, Saqlain Mirza  
Student, Assistant Professor,  
Alfaluh University.

## ABSTRACT

The modern world is getting advanced and enhanced in implementing the complex and huge structures in a minute design area, as per the moore's transistor integration law the integration of transistors are processed. As far the difficulty has been risen in identifying the individual transistor, power supply, power dissipation of each transistor ,for fixing this problems many studies and practices are performing . In such nano technologies evolved QCA is termed as the efficient technology.

This technology is employed as the transistor less structures which is termed as the replacement of cmos circuits. The qca technology is evolving day by day and are employed in arithmetic circuits. This paper flaunts QCA-based BCD subtractor.In this project BCD subtractor is developed using majority gates.

**KEYWORD-** Majority gates ,Quantum dot cellular automata,carry look ahead.

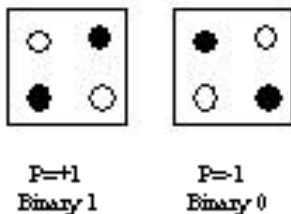
## 1.INTRODUCTION

As per the moore's integration of transistors law, the integration of the transistors for every two decdes will get doubled. This is proved true by the evaluation of the VLSI from LSI and LSI from MSI and MSI from SSI ,so by this the area will be minimizing by having a plum in the integrating transistors,where lot of challenges occur ,to solve that challenges lots of new technologies have took place.In that the nanotechnology is set as one of the efficient technologies.

The QCA is one among the nanotechnology where the replacement of the transistor is done with a qca cell.

### QCA Cells

The essential feature of a QCA cell [2] is that it possesses an electric quadrupole which has two stable orientations as seen in Fig. 1. These two orientations are used to represent the two binary digits, "1" and "0".A QCA cell is a structure comprise of four quantum-dots arranged in a square pattern.QCA information processing is based on the Columbic interactions between many identical QCA cells.



**Fig1: Basic QCA cell and Two Possible Polarizations**

The qca cell is the charge packet having four identical spots where they exhibit the diagonal similarity. These identical spots are tend to occupy by electrons or holes. These is decided by providing the selction process called polarization[P] .

As there only two possible chances of occupation so the two types of polarizations occur when polarization is '1' then the cell is occupied with charge identified as binary-1. If the polarization is '-1'then the cell will have the charge of binary-0.If the electrons are seen in 1<sup>st</sup> and 3<sup>rd</sup> spots in qca cell counting in clockwise then that cell can be termed as binary '0' if other then it is said to be binary '1' cell.

The flow of charge from one cell to another is done by the columbic reaction other than wire connections.when the three cells are placed in serial side by side the charge of the third cell will reflect the charge of the first cell.

As per the digital circuits relays or constructed with the basic blocks named logic gates. In the QCA circuits the basic blocks are majority gate and an invertors. The invertors are same as the not gate in digital literature.the equation of the majority gate is shown here.

$$M(a, b, c) = a \cdot b + a \cdot c + b \cdot c$$

By using this technology the construction of the BCD subtractor is constructed. The BCD subtractor is obtained by performing complement type of addition i.e first construction of the BCD adder is constructed .

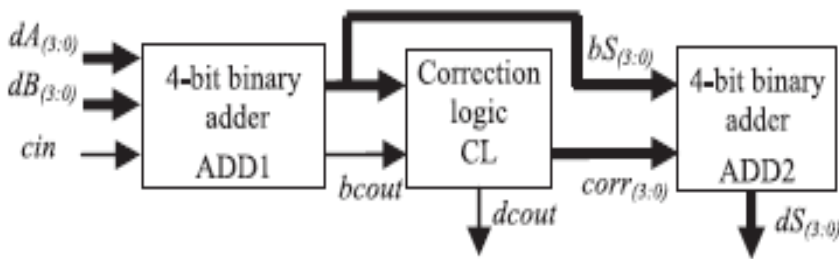


Fig 2:BCD adder block diagram with sub modules connections

The first block denotes the 4-bit adder where to advance over delay usage, CARRY LOOK AHEAD adder is employed instead of ripple carry adder. The second block is the correlation logic, which is used to convert the binary sum output to the BCD carry output.

And the final block is the adder block similar to the first block which produces the BCD sum output .

The SUBTRACTOR can be designed by making the complement of the second input and sending it to the bcd adder and again complementing the obtained output gives the SUBTRACTOR output.

**2.LITERATURE AND BACK GROUND**

The evolving technologies are making the solution path for previous unsolvable challenges . The most efficient technologies are the nano technologies as the nano is most related to scaling the design with the area in nanometers. The QCA is one the most efficient technology in such technologies. The logic gates will follows the physics while the QCA will follow the quantum physics. Every single thing in this wold is related to the quantum physics ,this is much faster ans efficient than normal physics .Reserches are going and construction of quantum computers has raised. The construction of the gates by using the QCA gates is done by making some changes in providing inputs and from that designing of the adders is done [3].by taking this[3] reference the evaluation of the BCD adder is done with making the adder into parallel adder[8] in [1]. From [2] the idea is taken to relate the structure mentioned in[2] to the QCA technology.

**3.IMPLEMENTATION OF BCD ADDER**

The BCD adder is implemented with the help of the majority gates and the relation of logic with majority gates has shown here

The 4-bit adder ADD1 is selected CLA so that the enhancement in new architechture to previous ripple carry adder. The calculation of carry and sum is done by using same circuit and this is related to the majority gate as shown below

$$C1 = M(dA0, dB0, cin) \dots\dots(1)$$

$$C2 = G1 + P1 \cdot G0 + P1 \cdot P0 \cdot Cin$$

$$= M (Cin, M(dA1, dB1, dA0), M(dA1, dB1, dB0)) \dots\dots(2)$$

$$C3 = M(dA2, dB2, C2) \dots\dots(3)$$

$$C4 = bcout = G3 + P3 \cdot G2 + P3 \cdot P2 \cdot C2$$

$$= M (C2, M(dA3, dB3, dA2), M(dA3, dB3, dB2)) \dots\dots\dots (4)$$

After carry generation sum calculation is done by the following equations

$$bSi = M(Ci, Ci + 1, M(dAi, dBi, Ci+1) ) \dots\dots\dots (5)$$

first block is constructed with carry generation and sum calculation and the next required blocks are correction logic and ADD2.the ADD2 is for obtaining the calculated value in decimal digit while CL logic makes the carry out which is of decimal carry out .

The process of obtaining the BCD output is shown with majority gates by the following theorem 1.

**Theorem 1:** If bcout and bS(3:0) are the binary carry-out and the binary sum computed by adding the BCD digits dA(3:0) and dB(3:0), then the BCD carry-out dcout is given by (6) and the decimal sum dS(3:0) is obtained by (7)

$$dcout = M (bcout, M(1, bcout, bS3), M(bS3, bS2, bS1)) (6)$$

$$dS0 = bS0 \dots\dots\dots (7a)$$

$$dS1 = M(\overline{M(0, bS1, dcout)}, M(1, bS1, dcout), 0) \dots (7b)$$

$$dS2 = M(\overline{M(bS2, dcout, M(0, bS1, dcout))},$$

$$M(\overline{M(bS2, dcout, M(0, bS1, dcout))}, bS2, dcout), M(0, bS1, dcout)) \quad (7c)$$

$$dS3 = M1, M(\overline{dcout}, bS3, 0), M, M(dcout, bS3, 0), bS1, 0$$

Proof: As demonstrated in

$$dcout = bcout + bS3 \cdot bS2 + bS3 \cdot bS1$$

that can be rewritten as

$$dcout = bcout \cdot (bcout + bS3) + bcout$$

$$\cdot (bS3 \cdot bS2 + bS3 \cdot bS1 + bS2 \cdot bS1) + (bcout + bS3)$$

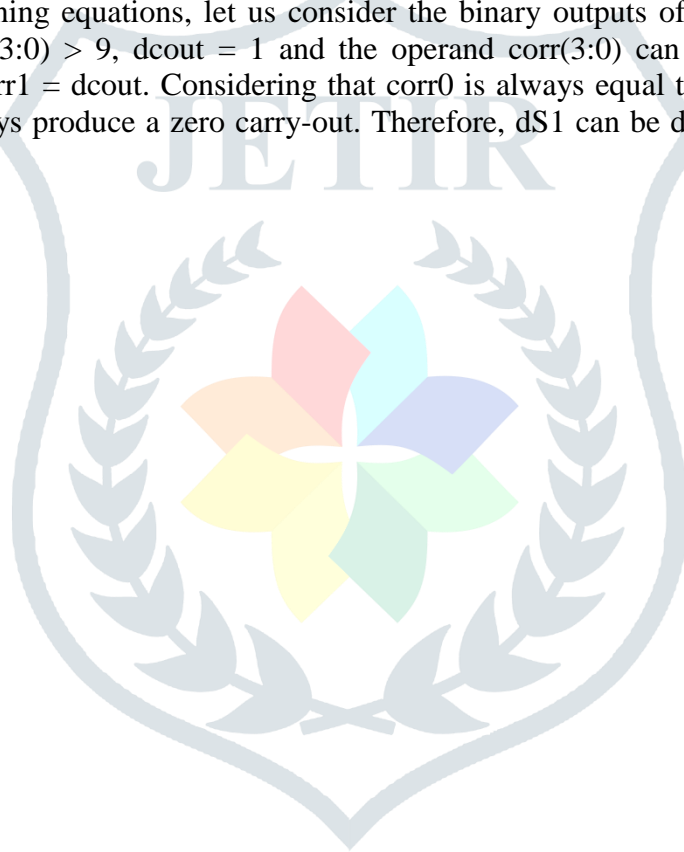
$$\cdot (bS3 \cdot bS2 + bS3 \cdot bS1 + bS2 \cdot bS1)$$

$$= M(bcout, bcout + bS3, bS3 \cdot bS2 + bS3 \cdot bS1 + bS2 \cdot bS1)$$

$$= M(bcout, M(1, bcout + bS3), bS3 \cdot bS2 + bS3$$

$$\cdot bS1 + bS2 \cdot bS1) \dots (8)$$

The decimal sum  $dS(3:0)$  is computed by adding  $bS(3:0)$  to the correction operand  $corr(3:0)$  that is equal to 6 when  $bS(3:0) > 9$ ; otherwise, it is set to 0. Due to this,  $dS0 = bS0 + corr0 = bS0$ , which trivially proves the correctness of (7a). To demonstrate the remaining equations, let us consider the binary outputs of ADD1 and their corresponding BCD representation. When  $bS(3:0) > 9$ ,  $dcout = 1$  and the operand  $corr(3:0)$  can be trivially obtained by setting  $corr3 = corr0 = 0$  and  $corr2 = corr1 = dcout$ . Considering that  $corr0$  is always equal to 0, it is easy to understand that adding  $bS0$  and  $corr0$  will always produce a zero carry-out. Therefore,  $dS1$  can be defined as the XOR between  $bS1$  and  $dcout$



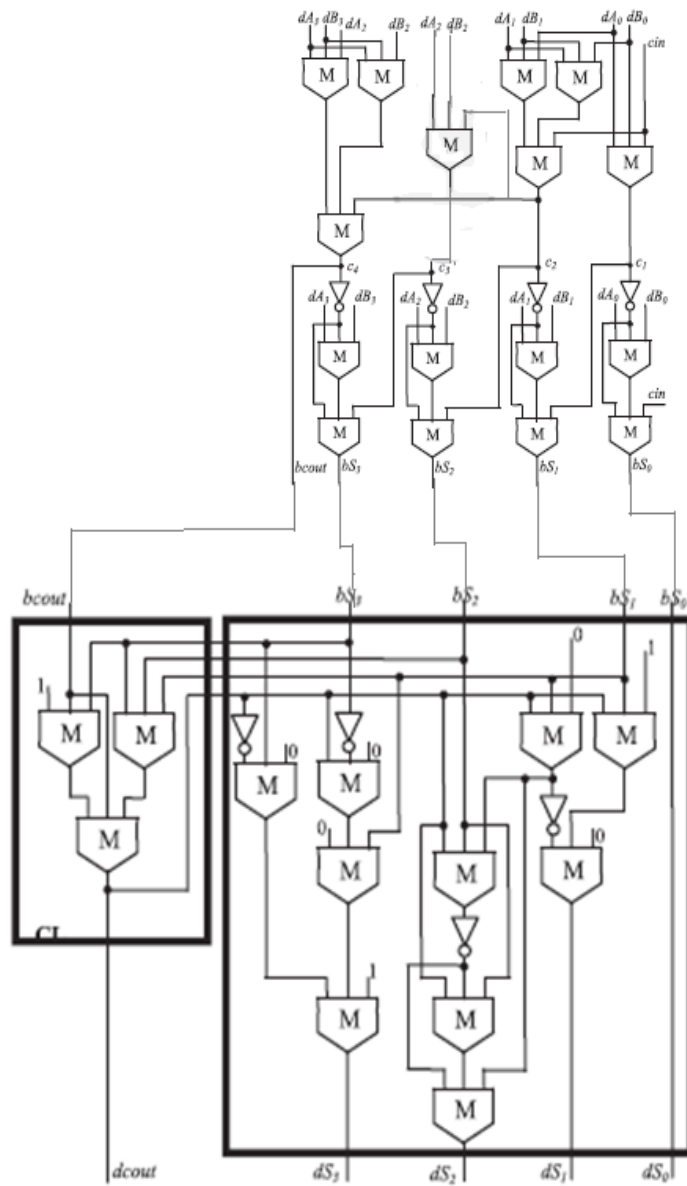


fig 3:bcd adder block diagram

**4.PROPOSED WORK**

**BCD SUBTRACTOR**

BCD subtractor is the type of complement addition in which complementation block and adder blocks are present. The 9's complement is made by making the 1's complement to the binary input and it is added with 10 which gives the nine complement.

The actual subtraction with complement addition is illustrated with an example

**Example** Subtract  $(1000)_2$  from  $(1001)_2$  using the 1's complement method.

**Solution.**

$$\begin{array}{r}
 1000 (+) \\
 1's\ complement\ 0110 \\
 \hline
 1110 \\
 1's\ complement\ 0001 \\
 \hline
 \text{final result} - 0001
 \end{array}$$

This is the normal binary subtraction so 1's complement is taken while in BCD subtraction the 9's complement will be taken and sent into the BCD adder and the result from that BCD adder is again sent into the complementation block to obtain sign and magnitude as outputs and the BCD subtractor is illustrated in the following diagram.

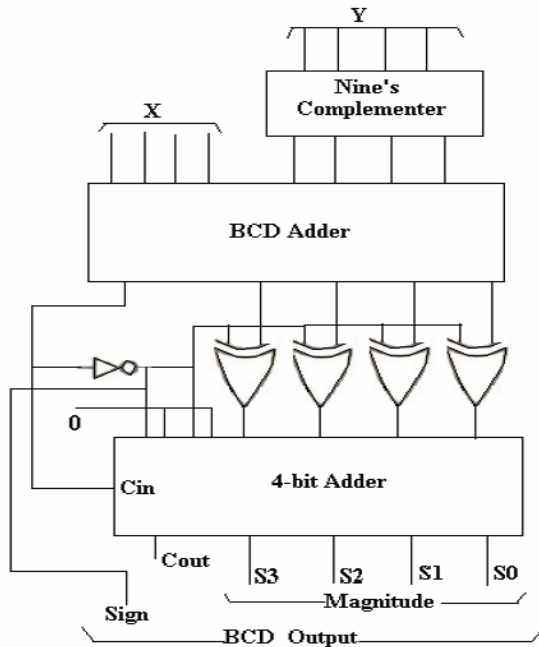


fig :4 bcd subtractor block diagram

The first block is the nines complement block here instead of the xor gates not gates are employed to obtain the same logic. In the subtraction arithmetic the subtraction will follow one kind of addition where if considered binary subtraction the subtrahend is added to another by performing the ones complement ,like wise in BCD the nines complement is performed .The implementation is shown below.

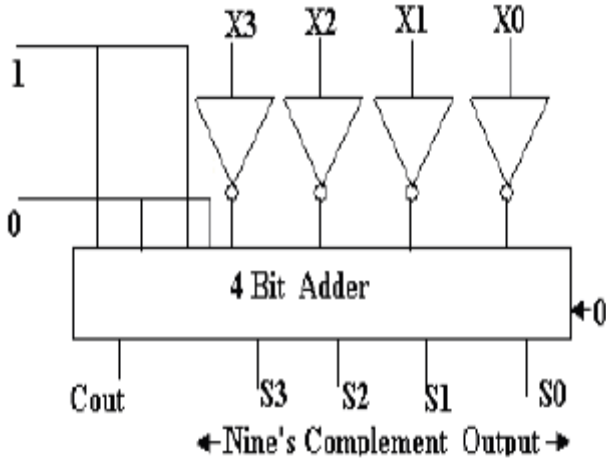


fig 5: nines complement block

The result obtained from the bcd adder is again sent into the process of nines complement block which has four two input xor gate architecture to obtain the actual difference with the sign output . construction of one xor gate by majority gates is flaunted below.

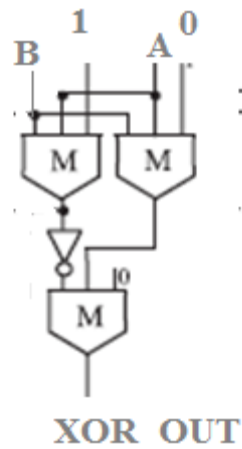


Fig 6:XOR GATE USING MAJORITY GATES

After the xor network there is a normal adder which is taken as previous adder which is used in construction of the BCD adder .The sign bit is obtained by inverting the BCD adder cout which shows the output sign i.e positive or negative .if sign bit =1 then the output is negative else it is positive.

**RESULTS**

RESULTS FOR BCD\_SUB[Subtrcator]

**Implementation**

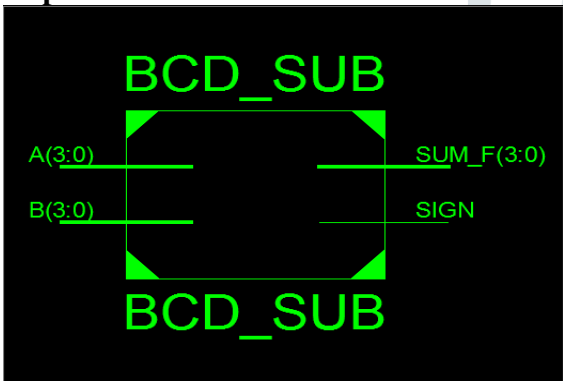


Fig 7:RTL Schematic view

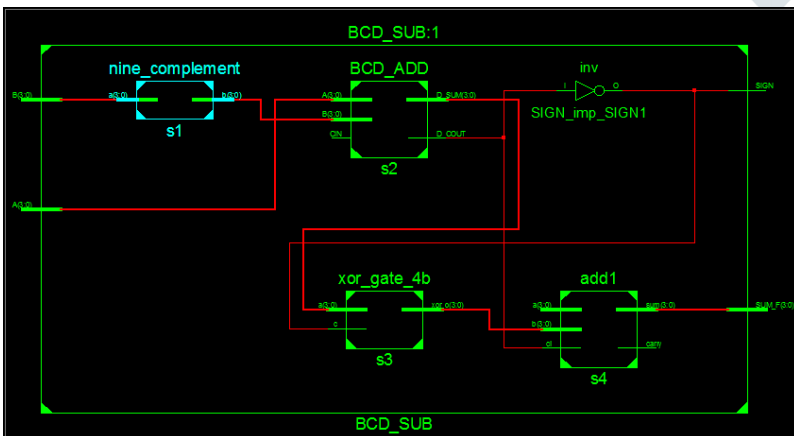


Fig 8:RTL Schematic view

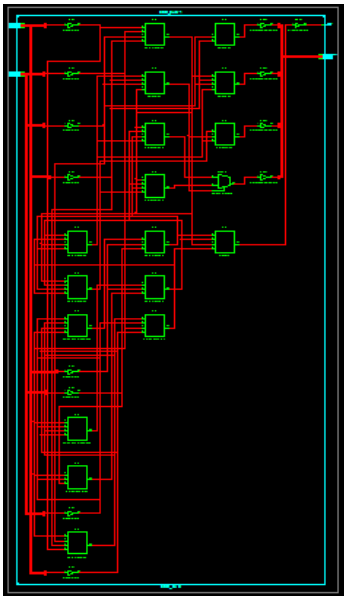


Fig 9:View technology Schematic Simulation

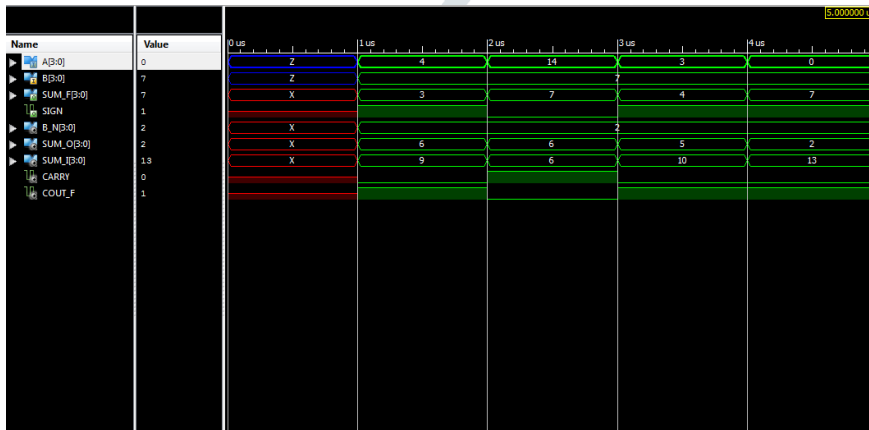


Fig 10: Simulation results for QCA

## CONCLUSION

Based upon the earlier studies the construction of an architecture is explored. The construction of the BCD-subtractor has done by using the majority gates analyzing with QUANTUM DOT CELLULAR AUTOMATA, which is termed as a nano architecture. The above study has the interlink to QCA and VLSI. where the evolving technologies are mostly preferring the QCA while the running technology is VLSI. This paper implementation is done in VLSI architecture making reference with QCA cells. The architecture is constructed using XILINX ISE 14.5 with preferred verilog HDL language.

## References:

- [1] Design of Efficient BCD Adders in Quantum-Dot Cellular Automata G. Cocorullo, P. Corsonello, F. Frustaci, and S. Perri
- [2] Efficient Reversible Logic Design of BCD Subtractors Himanshu Thapliyal<sup>1</sup>, Hamid R. Arabnia<sup>2</sup>, and M.B. Srinivas<sup>3</sup>
- [3] V. Pudi and K. Sridharan, "Low complexity design of ripple carry and Brent–Kung adders in QCA," IEEE Trans. Nanotechnol., vol. 11, no. 1, pp. 105–119, Jan. 2012.
- [4] V. Pudi and K. Sridharan, "Efficient design of a hybrid adder in quantumdot cellular automata," IEEE Trans. Very Large Scale Integr. (VLSI) Syst., vol. 19, no. 9, pp. 1535–1548, Sep. 2011.
- [5] S. Perri and P. Corsonello, "New methodology for the design of efficient binary addition in QCA," IEEE Trans. Nanotechnol., vol. 11, no. 6, pp. 1192–1200, Nov. 2012.



- [6] V. Pudi and K. Sridharan, “New decomposition theorems on majority logic for low-delay adder designs in quantum dot cellular automata,” *IEEE Trans. Circuits Syst. II, Exp. Briefs*, vol. 59, no. 10, pp. 678–682, Oct. 2012.
- [7] K. Walus and G. A. Jullien, “Design tools for an emerging SoC technology: Quantum-dot cellular automata,” *Proc. IEEE*, vol. 94, no. 6, pp. 1225–1244, Jun. 2006.
- [8] S. Bhanja, M. Ottavi, S. Pontarelli, and F. Lombardi, “QCA circuits for robust coplanar crossing,” *J. Electron. Testing, Theory Appl.*, vol. 23, no. 2, pp. 193–210, Jun. 2007.

