IOSR Journal of VLSI and Signal Processing (IOSR-JVSP)
Volume 5, Issue 6, Ver. II (Nov -Dec. 2015), PP 59-66
e-ISSN: 2319 – 4200, p-ISSN No. : 2319 – 4197
www.iosrjournals.org

Multiplexer-Based Design of Adders/Subtractors and Logic Gates for Low Power VLSI Applications

[1]Ramesh Boda*, [2]M.Charitha, [3]B.Yakub, [4]R.Sayanna
[1, 4](Department of physics, Osmania University, Hyderabad, Telangana, India)
[2](ECE, NRI College of engineering, JNTU Hyderabad, Telangana, India)
[3](ECE, HITAM engineering college, JNTU Hyderabad, Telangana, India)

Abstract: *The emphasis in VLSI design has shifted from high speed to low power due to the rapid increase in number of portable electronic systems. Power consumption is very critical for portable video applications such as portable videophone and digital camcorder Motion estimation (ME) in the video encoder requires a huge amount of computation, and hence consumes the largest portion of power. Many of the techniques have already been used in low power design with additional techniques emerging continuously at all levels. The aim of this work is to provide a comprehensive study of low-power circuit and design techniques using CMOS (complementary metal-oxide-semiconductor) technology. This consist of circuit design; transistor size, number of transistors used in design, layout technique, cell topology, and circuit design for low power operation while paying particularly attention on the methodology of logic style. This paper presents, high-speed and high-performance multiplexer based 1 bit adders/subtractors for low-power applications such as ASIC (Application Specific Integrated Circuits), ALU (Arithmetic logic Unit). Simulations were performed by EWB (Electronic Work Bench, unisim, Xilinx) for analysis of various feature sizes. The simulation results demonstrate clearly the improvement of the proposed design in terms of lower power dissipation, less propagation delay, less occupying area compared to other widely used existing adder/ subtractor circuits on the basis of multiplexer.*

Key words: *VLSI design, half adder, full adder, half subtractor, full subtractor, CMOS.*

I. INTRODUCTION

With the rapid growth in laptops, portable personal communication systems and the evolution of shrinking technology, the research effort in low-power microelectronics has been intensified and low-power Very Large Scale Integration (VLSI) systems have emerged to be high in demand [1, 2]. There are an increasing number of portable applications requiring small-area, low-power and high-speed circuitry in recent trends. Therefore, circuits with low power consumption have become the major identities for design of microprocessors and system components [3-5]. However, the battery technology does not advance at the same rate as the microelectronics technology, and there is a limited amount of power available for the mobile systems. The goal of extending the battery life span of portable electronics is to reduce the energy consumed per arithmetic operation. To execute an arithmetic operation, a circuit can consume very low power by clocking at an extremely low frequency but it may take a very long time to complete the operation. Therefore, designers have been facing many constraints in terms of high speed, low power, and small layout area [6]. Addition is one of the fundamental arithmetic operations and is used mostly in many VLSI systems. The main objective of addition is adding two binary numbers; it is the base of many other useful operations such as subtraction, multiplication, division, and ALU circuits [3, 7]. The adder plays an important role in most of these systems, to estimate the critical path, which determines the overall performance of the system, and the full adder is the fundamental element of complex arithmetic circuits. The main aim of this work is to enhance the performance of 1-bit full-adder cell [8]. Designed with certain logic styles [9], the proposed adder/subtractor and other six types of logic gates are mainly designed and simulated by CMOS technology. Reducing the supply voltage and increasing the transistor count are the major constraints for the design of the full adder cell. The proposed full adder circuit has a very short critical two level paths and reduced transition activity [10, 11]. We know that the universal gates are NAND and NOR, they are called as the universal gates because they can create any of the logic gates, any other digital circuit. There is a concept called "Universal Logic", Universal logic can also be used to create any of the logic gates and digital circuits. MUX and Decoders are called "Universal Logic" In this paper, we presented how a 2:1 MUX can be used to create different logic gates, half adder and half subtractor and how a 4:1 MUX can be used to create full adder and full subtractor and all other circuits design also.

II. EXPERIMENTAL DETAILS

The designed circuits are implemented, simulated and tested the results with ISE software from Xilinx, and Electronic work bench (Unisim). ISE software is chosen as the standard to synthesize the designs and compare the results.

III. THEORY

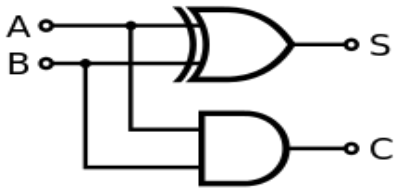
Logic gates:

The reason for which the computers are capable of performing complex operation is due to the interconnection of these logic gates. Logic gates are implemented by using transistors, diodes, relays, optics and molecules or even by several mechanical elements. Due to this reason logic gates can also be considered as electronic circuits. The logic gates can be build up in a wide variety forms such as large-scale integrated circuits (LSI), very large-scale integrated circuits (VLSI) and also in small-scale integrated circuits (SSI). Here the inputs and output of all the gates of integrated devices can be accessible and also the external connections are made available to them just like discreet logic gates.

Inputs and outputs of logic gates are in two levels termed as HIGH and LOW, or TRUE and FALSE, or ON and OFF, or simply 1 and 0. A table which list out the combination of input variables and the corresponding output variables is termed as "TRUTH TABLE". It explains how the logic circuit output responds to various combinations of logic levels at the inputs. Here we are following level logic, in which the voltage levels are represented as logic 1 and logic 0. Level logic is of two types such as positive logic or negative logic. In the positive logic system, higher of the two voltage levels are represented as 1 and lower of the two voltage levels are represented as 0. But in the negative logic system, higher of the two voltage levels are represented as 0 and lower of the two voltage levels are represented as 1. While considering the transistor-transistor logic (TTL), the lower state is assumed to be zero volts (0V) and the higher state is considered as five volts positive (+5V).

Half adder:

The half adder adds two single binary digits *A* and *B*. It has two outputs, sum (*S*) and carry (*C*). The carry signal represents an overflow into the next digit of a multi-digit addition. The simplest half-adder design, pictured on the right, incorporates an XOR gate for *S* and an AND gate for *C*. With the addition of an OR gate to combine their carry outputs, two half adders can be combined to make a full adder. The half adder adds two input bits and generates a carry and sum, which are the two outputs of a half adder. The input variables of a half adder are called the augends and addend bits. The output variables are the sum and carry. The truth table for the half adder is given in the next section



The value of the sum is $2C + S$.

The dataflow Boolean logic for half adder is given by

Sum(S) = A XOR B.

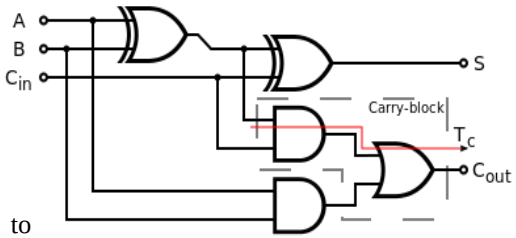
Carry(C) = A AND B.

Where A and B are the 1-bit binary inputs to the half adder.

Full adder:

A full adder adds binary numbers and accounts for values carried in as well as out. A one-bit full adder adds three one-bit binary numbers, often written as A, B, and C_{in}; A and B are the operands, and C_{in} is a bit carried in from the previous less significant stage. The full adder is usually a component in a cascade of adders, which add 8, 16, 32, etc. bit binary numbers. The circuit produces a two-bit output, output carry and sum typically represented by the signals C_{out} and Sum, The one-bit full adder's truth table is given in the next section

The value of sum is given by ***Sum = 2 x Cout + S***.



The dataflow Boolean logic for full adder is given by

Sum(S) = A XOR B XOR C_{in}.

Carry(C) = (A AND B) OR (C_{in} AND (A XOR B)).

Where A and B are the 1-bit binary inputs to the full adder.

Half subtractor:

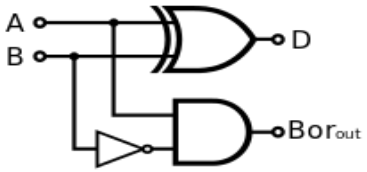
The half-subtractor is a combinational circuit which is used to perform subtraction of two bits. It has two inputs, A (minuend) and B (subtrahend) and two outputs D (difference) and B(borrow). An important point difference should be noticed is that the half substractor diagram aside implements (b-a) and not (a-b) as borrow is calculated from equation.

The dataflow Boolean logic for half subtractor is given by

D (difference) = A XOR B.

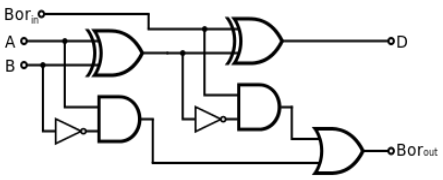
B (borrow) = A AND (NOT B).

Where A and B are the 1-bit binary inputs to the half subtractor.



Full subtractor:

The full-subtractor is a combinational circuit which is used to perform subtraction of three bits. It has three inputs, A (minuend) and B (subtrahend) and Bi (borrow in) and two outputs D (difference) and Bo (borrow out).



The dataflow Boolean logic for full subtractor is given by

D (difference) = A XOR B XOR B_{in}.

Bo (borrow out) = NOT A AND (B XOR B_{in}) OR (B AND B_{in}).

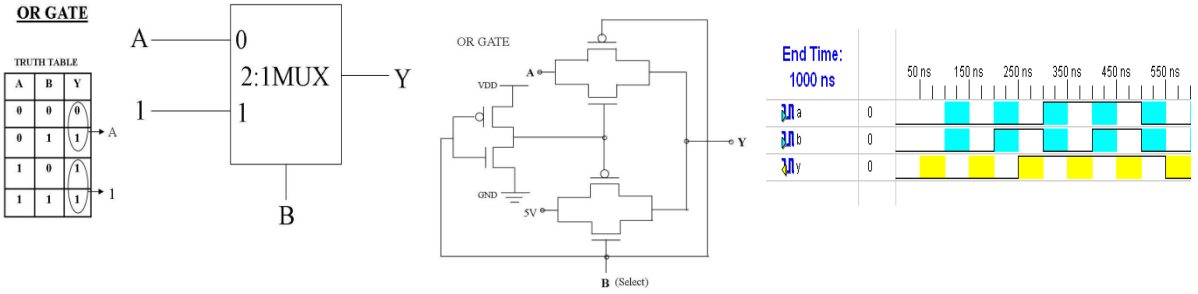
Where A and B are the 1-bit binary inputs to the full subtractor.

IV. RESULTS AND DISCUSSION

1. Designing an OR Gate using 2:1 MUX

To design an OR using 2:1 mux, we need to tie the "First" input to "Logic 1" and the "Zeroth" input to the one of the input of the OR Gate. The other input of OR gate would be connected with the select line of the MUX. Now, the output of the MUX would be "A" when any of the two inputs on B would be "0" otherwise it would be "1" for all conditions. The truth table is solved and it is simplified that the two inputs of the MUX are 'A' and "1", in CMOS technology the logic "1" can fixed as 5V (volts) which indicates logic high. In this design logic "1" is assigned to 5Volts. And the simulated wave form is given in the figure below.

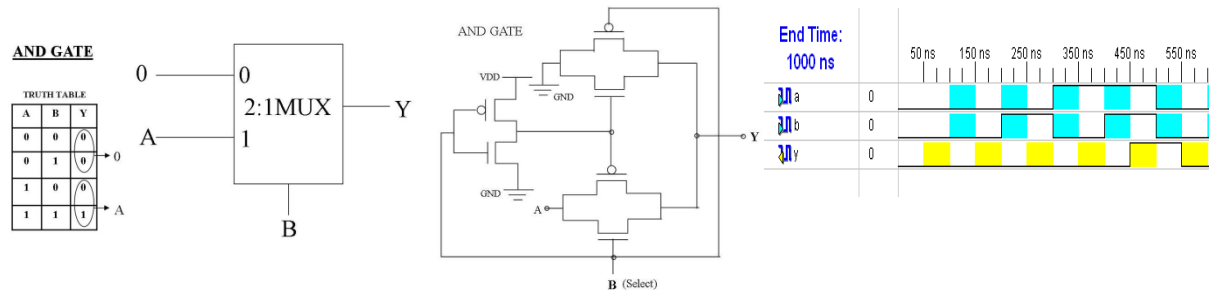
Fig 1: MUX to OR Gate:



2. Designing an AND Gate using 2:1 MUX

To design an AND using 2:1 mux, we need to tie the "zeroth" input to "Logic 0" and the "First" input to the one of the input (A) of the AND Gate. The other input of AND gate would be connected with the select line of the MUX. Now, the output of the MUX would be "0" on low on the select line otherwise it would be "A" for all conditions. The truth table is solved and it is simplified that the two inputs of the MUX are "0" and "A", in CMOS technology the logic "0" can fixed as GROUND which indicates logic low. In this design logic "0" is assigned to GROUND. And the simulated wave form is given in the figure below for AND gate.

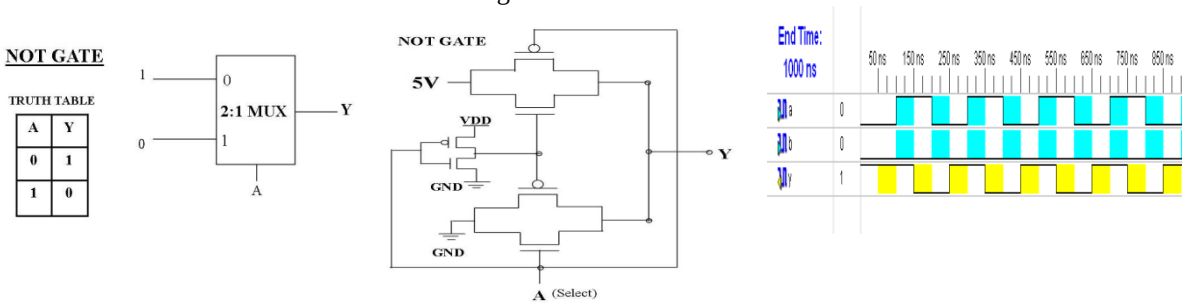
Fig 2: MUX to AND Gate:



3. Designing an Inverter using 2:1 MUX

To design an inverter using 2:1 mux, we have to use the input as the select line of the MUX and the "zeroth" select line would be tied with "Logic 1 " and "First" select line would be tired with "Logic 0", Now when the select line (Input) goes to "1" the output will be "0" (inverted). Now, the output of the MUX would be "0" on high on the select line otherwise it would be "1" condition. The truth table is solved and it is simplified that the two inputs of the MUX are "0" and "1", in CMOS technology the logic "0" can fixed as GROUND which indicates logic low, the logic "1" can be fixed to 5V which indicates logic high. In this design logic "0" is assigned to GROUND, logic "1" is assigned to 5V. And the simulated wave form is given in the figure below for NOT gate.

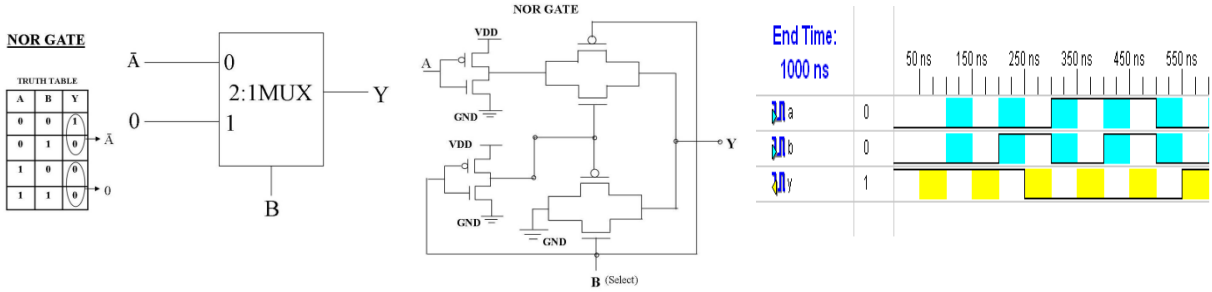
Fig 3: MUX to inverter



4. Designing an NOR Gate using 2:1 MUX

To design the NOR using 2:1 mux, we need to tie the "Zeroth" input of mux to one of the input of NOR and another input of MUX is tied to "0". Another input of the NOR gate would be applied to the select line of the MUX. Now, the output of the MUX would be A'B' = (A+B)', which is same as the output of the 'NOR' Gate. Now, the output of the MUX would be "0" on high on the select line otherwise it would be "not A" for all conditions. The truth table is solved and it is simplified that the two inputs of the MUX are "0" and "not A", in CMOS technology the logic "0" can be fixed as GROUND which indicates logic low. In this design logic "0" is assigned to GROUND, and the "not A" input is provided through CMOS inverter. And the simulated wave form is given in the figure below for AND.

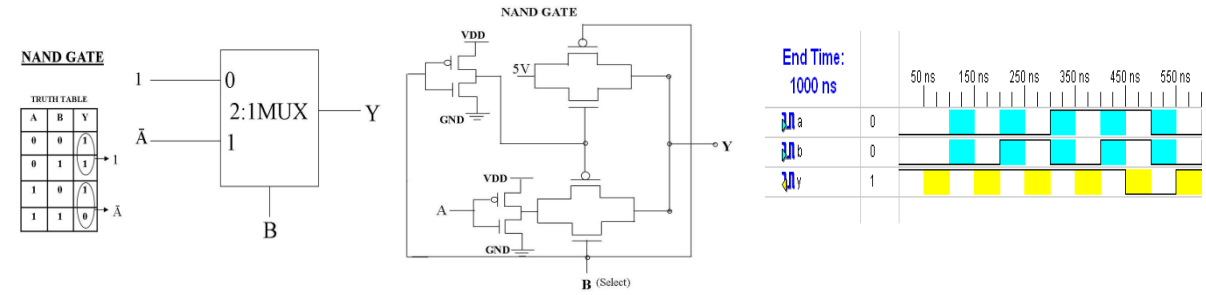
Fig 4: MUX to NOR Gate:



5. Designing an NAND Gate using 2:1 MUX.

To design the NAND using 2:1 mux, we need to combine the AND Gate and inverter implementation. To design an NAND using 2:1 mux, we need to tie the "zeroth" input to "Logic 1" and the "First" input to the one of the input of the NAND Gate. The other input of NAND gate would be connected with the select line of the MUX. Now, the output of the MUX would be "1" if there is low on the select line otherwise it would be "not A" for all conditions. Truth table is solved and it is simplified that the two inputs of the MUX are "0" and "not A", in CMOS technology the logic "1" can be fixed as 5V which indicates logic high. In this design logic "1" is assigned to 5V, and the not A input is provided through CMOS inverter. And the simulated wave form is given in the figure below for NAND.

Fig 5: MUX to AND Gate:

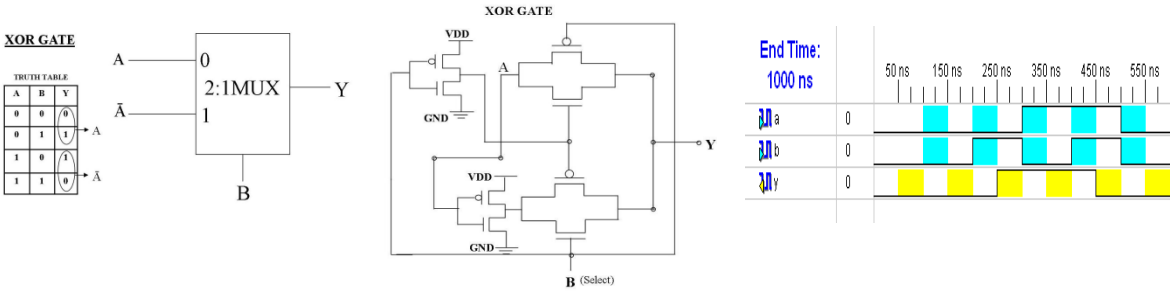


6. Designing an XOR Gate using 2:1 MUX

To design the XOR using 2:1 mux, we need to tie the "Zeroth" input of mux to one of the input of XOR and another input of MUX to the inverted of first input. Another input of XOR gate would be applied to the select line of the MUX. Now, the output of the MUX would be AB' + A'B which is same as the output of

XOR Gate. Now, the output of the MUX would be "A" if there is low on the select line otherwise it would be "not A" for all conditions. Truth table is solved and it is simplified that the two inputs of the MUX are "A" and "not A", in CMOS technology the "not A" input is provided through CMOS inverter. And the simulated wave form is given in the figure below for XOR gate.

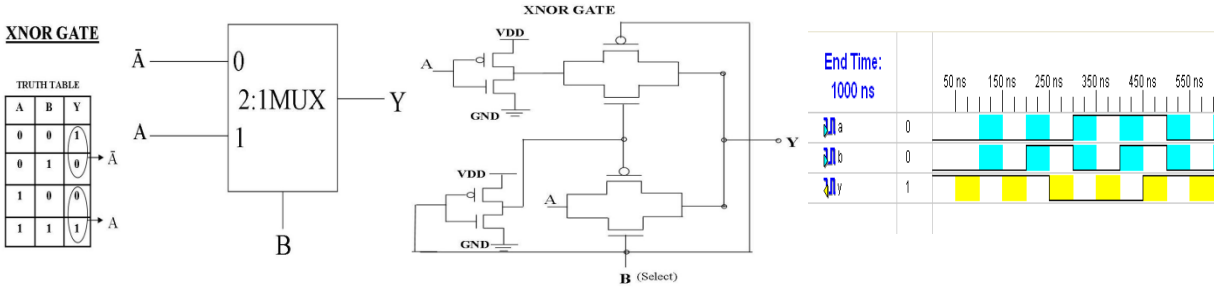
Fig 6: MUX to XOR Gate:



7. Designing an XNOR Gate using 2:1 MUX

To design the XNOR using 2:1 mux, we need to tie the "First" input of mux to one of the input of XNOR and another input of MUX to the inverted of first input. The other input of XOR gate would be applied to the select line of the MUX. Now, the output of the MUX would be A'B' + AB which is same as the output of XNOR Gate. Now, the output of the MUX would be "notA" if there is low on the select line otherwise it would be "A" for all conditions. Truth table is solved and it is simplified that the two inputs of the MUX are "not A" and "A", in CMOS technology, the not A input is provided through CMOS inverter. And the simulated wave form is given in the figure below for XNOR gate.

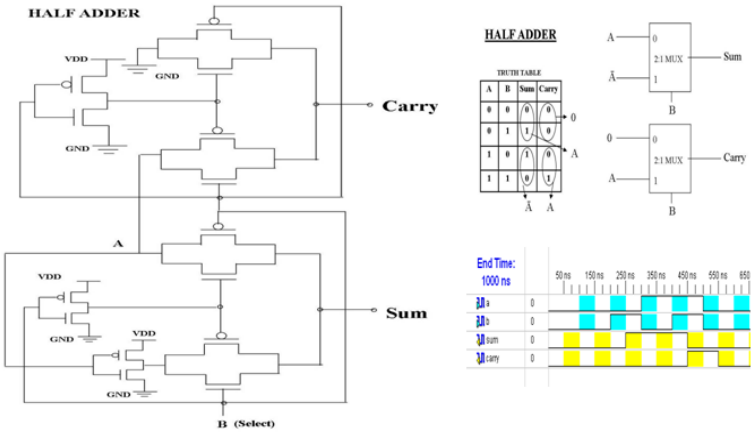
Fig 7: MUX to XNOR Gate:



8. Designing a HALF ADDER using two 2:1 MUX

To design the half adder using two 2:1 MUX, we need to tie the "First" input of "First"mux to one of the input of SUM block and another input of MUX to the inverted of first input. The other input of half adder would be applied to the select line of both MUX of SUM and CARRY block. Now, the output of the "First" MUX would be A'B + A B' which is same as the SUM output of half adder. Now, the output of the "first" MUX would be "A" if there is low on the select line otherwise it would be "not A" for all conditions and the output of the "second" MUX would be "0" if there is low on the select line otherwise it would be "not A" for all conditions. Truth table is solved and it is simplified that the two inputs of the half adder are "not A" and "A", in CMOS technology, the not A input is provided through CMOS inverter and logic "0" is fixed to 0V i.e. to GROUND. And the simulated wave form is given in the figure below for half adder.

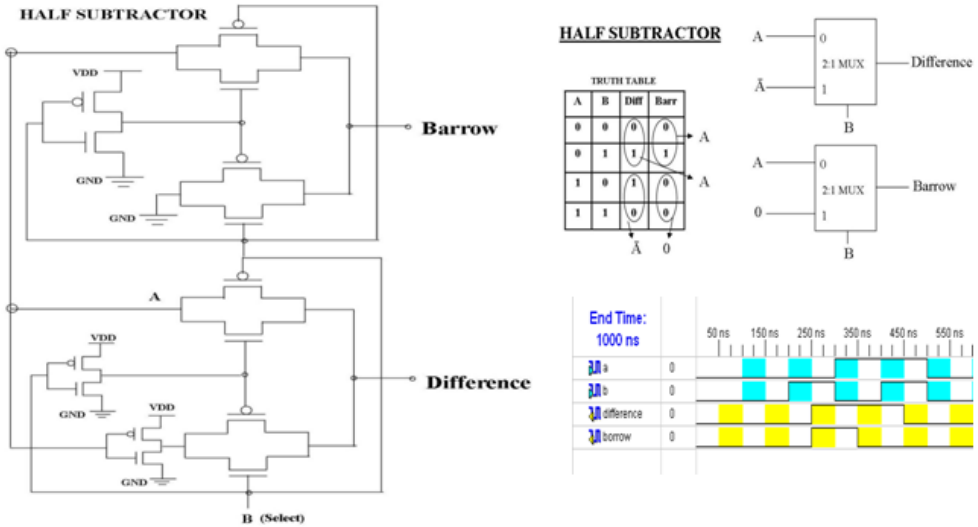
Fig 8: MUX to Half adder:



9. Designing a HALF SUBTRACTOR using two 2:1 MUX

To design the half subtractor using two 2:1 MUX, we need to tie the "First" input of "First"mux to one of the input of DIFFERENCE block and another input of MUX to the inverted of first input. The other input of half subtractor would be applied to the select line of both MUX of DIFFERENCE and BARROW block. Now, the output of the "First" MUX would be A'B + A B' which is same as the DIFFERENCE output of half subtractor. Now, the output of the "first" MUX would be "A" if there is low on the select line otherwise it would be "not A" for all conditions and the output of the "second" MUX would be "A" if there is low on the select line otherwise it would be "0" for all conditions. Truth table is solved and it is simplified that the two inputs of the half adder are "not A" and "A", in CMOS technology, the not A input is provided through CMOS inverter and logic "0" is fixed to 0V i.e. to GROUND. And the simulated wave form is given in the figure below for half subtractor.

Fig 9: MUX to Half subtractor:

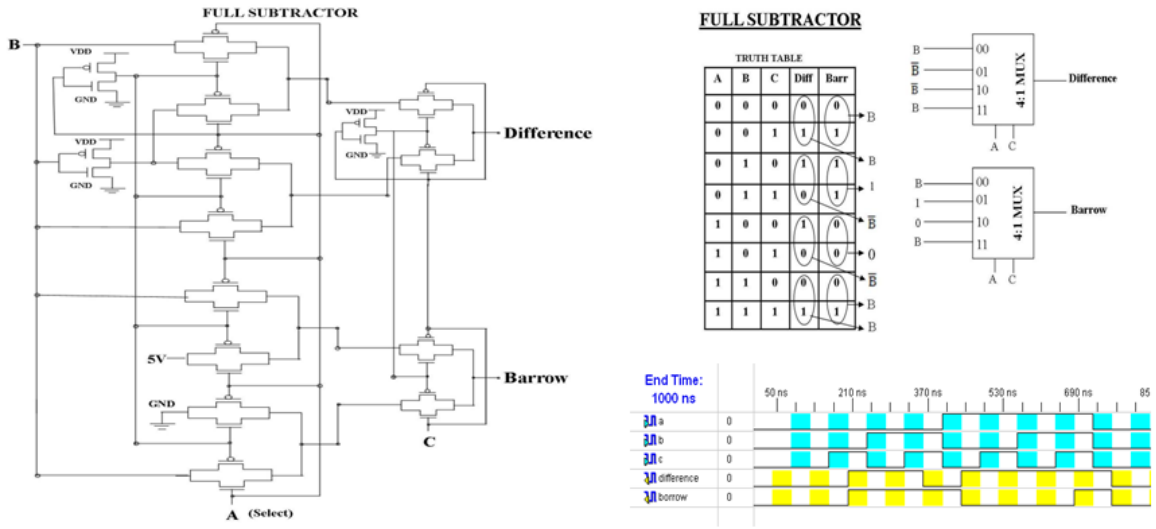


10. Designing 1-bit full adder using 4:1 MUX

The 1-bit full adder is one of the most critical components of a processor that determines its throughput, as it is used in the ALU, the floating-point unit, and address generation for cache or memory accesses [12]. It is therefore inherent that full-adders have great impact on the systems that use these adders. A variety of full adders using static and dynamic logic styles have been reported in literature [12]-[19]. Including the most well known static complementary CMOS adders using 28 transistors and 40 transistors [12] and [15]. In the present work a 1-bit full adder that utilizes a novel multiplexer-based architecture, built upon two 4:1 identical multiplexers are used, which requires a total of 28 transistors to realize the function of a full adder (Fig. 10). This new adder has low response time, low power utilization and reduced transition activity than previously

proposed low-power adders, because it utilizes only two levels of circuit [16]. There are three major sources of power dissipation in a digital CMOS circuit: logic transition, short-circuit current and leakage current [17], [18]. The following, is the low power consumption circuit adder due to reduction in short circuit current and switching activity. The short-circuit current is defined to be the direct current passing through the supply and the ground, when both the NOMS and PMOS transistors are simultaneously active [13][17], in the two level adder, both the NOMS and PMOS transistors does not takes place simultaneously hence low transition time can be observed. All adders suffer the same short-circuit current problem as they have some internal nodes driven by signals with slow raise and/or fall times. This leads to significant (20%) short circuit power dissipation for loaded inverters [17]. Such problem was partially solved in this design.

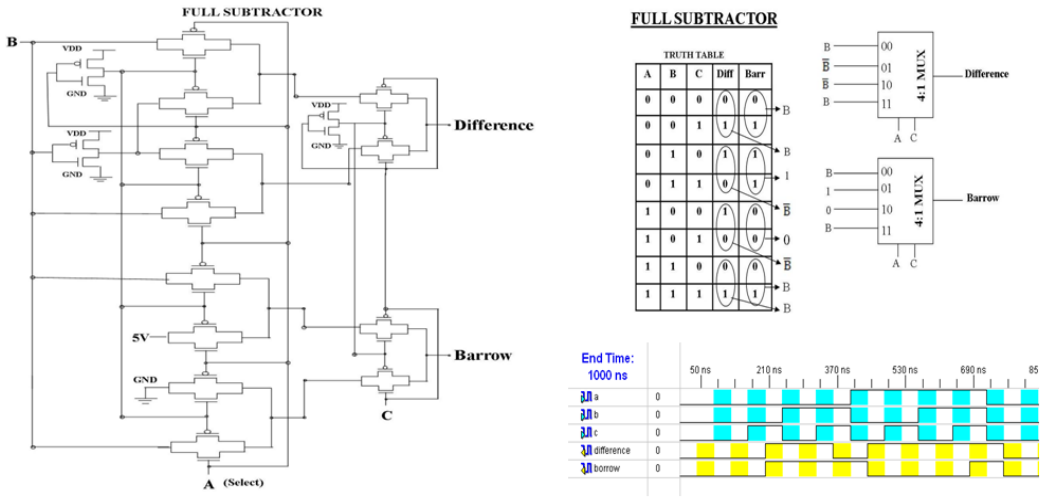
Fig 10: MUX to full adder:



11. Designing 1-bit full subtractor using 4:1 MUX

Subtraction is executed by two's components, so we need to use one XOR gate to invert one bit and add one into carry. The DIFFERENCE output is same as that of SUM output of full adder circuit but the BARROW output is not same as that of carry output of full adder but it is inverted and complimented, i.e. ***A-B = A+ (-B) = A+ 2's compliment of B.***

Fig 11: MUX to full subtractor:



V. CONCLUSIONS

This paper aims to propose the new architecture of ADDER/SUBTRACTOR platform with two optimization goals. The first one is to optimize the ADDER/SUBTRACTOR in order to decrease power consumption and increase the speed. The second one is to develop new circuit for ADDER/SUBTRACTOR to enable a production with fewer gates. As a result, the proposed design is based on the multiplexer, and it has only two kinds of components: NOT gate and multiplexer and important noticeable thing is it is designed as only two level circuit which takes less delay and fast performance, which makes the design easily implemented on FPGA/CPLD chip. A design and simulation is carried on FPGA/CPLD chip.

ACKNOWLEDGMENT

This work is supported by "University College of Science, Osmania University, and Hyderabad "the Fundamental Research Funds for my work is provided by UGC New Delhi under RGNF scheme and deep gratitude to my supervisor Prof.R.Sayanna, for his valuable assistance over the period of carrying out this work in VHDL laboratory.

REFERENCES

[1] G. Ramana murthy, C. Senthilpari, P. Velrajkumar, Lim tien sze, a novel design of multiplexer based full-adder cell for power and propagation delay optimizations journal of engineering science and technology vol. 8, no. 6 (2013) 764 – 777 © school of engineering, Taylor's university.
[2] Rabaey, J.M.; and Pedram, M. (1995). Low power design methodologies. Kluwer Academic Publishers, Boston.
[3] Alito, M.; and Palumbo, G. (2002). Analysis and comparison of the full adder block. IEEE Transactions on VLSI, 10(6), 806-823.
[4] Shalem, R.; John, E.; and John, L.K. (1999). A novel low power energy recovery full adder cell. Proceedings of the Great Lakes Symposium on VLSI, 380-383.
[5] Pedram, M.; and Rabaey, J.M. (2002). Power aware design methodologies. Kluwer Academic Publishers, Boston.
[6] Soudris, D.; Piguet, C.; and Goutis, C. (2002). Designing CMOS circuits for low power European low-power initiative for electronic system design. Kluwer Academic Publishers, Boston/Dordrecht/London.
[7] Song, P.J.; and Micheli, G.D. (1991). Circuit and architecture trade-offs for high speed multiplication. IEEE Journal of Solid-State Circuits, 26(9), 1119-1184.
[8] Chandrakasan, A.P.; and Brodersen, R.W. (1995). Low power digital CMOS design. Kluwer Academic Publishers, Norwell, MA.
[9] Kuroda, T.; and Hamada, M. (2000). Low-power CMOS digital design with dual embedded adaptive power supplies. IEEE Journal of Solid-State Circuits, 35(4), 652-655.
[10] Senthilpari, C.; Ajay, K.S.; and Arokiasamy, A. (2006). Statistical analysis of power delay estimation in adder circuit using non-clocked pass gate families. IEEE 4th International Conference on Electrical and Computer Engineering, Dhaka, Bangladesh, 509-513.
[11] Senthilpari, C.; Ajay K.S.; and Diwakar, K. (2008). Design of a low power, high performance, 8×8 bit multiplier using a Shannon based adder cell. Microelectronics Journal, 39(5), 812-821.
[12] R. Shalem R., E. John E., and L. K. John L. K." A Novel Low Power Energy Recovery Full Adder Cell". Proc.of the Great Lakes Symposium of VLSI, Feb. 1999, pp. 380-383.
[13] A. P. Chandrakasan, S. Sheng and R. W. Broderson, "Low-Power CMOS Digital Design, " IEEE Journal of Solid-State Circuits, vol. 27, no. 4, 1992, pp.473-483.
[14] R. Pedram R. and M. Pedram M. Low Power Design Methodologies. Kluwer Academic, US, 1996.
[15] H. T. Bui, A. K. Al-Sheraidah, and Y. Wang ,"Design and Analysis of 10-Transistor Full Adders Using Novel XORXNOR Gates," International Conference on Signal Processing 2000, Wold Computer Congress, August 2000, Beijing, China
[16] J. Wang, S. Fang and W. Feng, "New Efficient Designs for XOR and XNOR Functions on the Transistor Level," IEEE Journal of Solid -State Circuits, vol.29, no. 7, July 1994, pp.780-786.
[17] N. Weste N. and K. Eshraghian. Principles of CMOS VLSI Design: A System Perspective. MA: Addison- Wesley, 1993.
[18] R. Zimmermannn and W. Fichtner, "Low-Power Logic Styles: CMOS versus Pass-Transistor Logic," IEEE Journal of Solid-State Circuits, vol.32, no.7, July 1997, pp. 1079-1090.
[19] Y. Jiang, Y. Wang and J. Wu, "Comprehensive Power Evaluation of Full Adders," Tech. Report, Florida Atlantic University, 2000
[20] TranBich Thuan Pham, Yi Wang, Renfa Li Designing one-bit Full-Adder/Subtract or based on Multiplexer and LUT's architecture on FPGA.