ISSN (Online) 2321 – 2004
ISSN (Print) 2321 – 5526

INTERNATIONAL JOURNAL OF INNOVATIVE RESEARCH IN ELECTRICAL, ELECTRONICS, INSTRUMENTATION AND CONTROL ENGINEERING
Vol. 2, Issue 3, March 2014

Design and Implementation of Area Optimized ALU using GDI Technique

Akshay Dhenge[1], Abhilash Kapse[2], Sandeep Kakde[3]

Student, Electronics Department, Y. C. College of Engineering, Nagpur, India[1,2]

Professor, Electronics Department, Y.C. College of Engineering, Nagpur, India[2]

Abstract: Area and power are the two vital issues in analog circuit design and synthesis of ULSI and VLSI circuits which depends on various critical design parameters. The purpose of this paper is the design and implementation of an Arithmetic Logic Unit (ALU) using area optimizing techniques such as complementary & Gate-Diffusion-input (GDI). The main sub-blocks of ALU are Adder, Subtractor, shifter and Logical Block. This work evaluates and compares the performance and optimized area of ALU with Static CMOS technique and GDI technique in 250nm CMOS (1P5M-1 Poly 5 Metal) process technology. Simulations are performed by using Tanner EDA 13.2 tools using model file 250nm CMOS technology. At first, using Tanner 13.2 EDA S-Edit Tool, the circuits are implemented with Static CMOS technology and then with GDI techniques. Simulations results validate the proposed concept and verify that GDI technique decreases the area used by ALU and increase the speed of ALU.

Keywords: ALU, VLSI, GDI, CMOS, Low Power, Power Dissipation, Optimized ALU.

I. INTRODUCTION

A processor is a main part of any digital system. And an ALU is one of the main components of a micro-processor. To give a simple analogy, CPU works as a brain to any system & and ALU works as a brain to CPU. So it's a brain of computer's brain. They consists of fast dynamic logic circuits and have carefully optimized structures. Of total power consumption in any processor, CPU accounts a significant portion of it. ALU also contribute to one of the highest power-density locations on the processor, as it is clocked at the highest speed and is busy mostly all the time which results in thermal hotspots and sharp temperature gradients within the execution core. Therefore , this motivate us strongly for a energy-efficient ALU designs that satisfy the high-performance requirements, while reducing peak and average power dissipation.[1,2] Basically ALU is a combinational circuit that performs arithmetic and logical operations on a pair of n bit operands e.g. A [0:7] & B [0:7] for 8 bits. The typical internal structure of a 8 bit ALU is shown in Fig.1.

The architecture can be modified similarly for lower bits. Our work is divided into following sections: Section (II) give description of various units and operations of ALU, Section (III) briefly presents the designing of various ALU components such as Shifter, adder, subtractor etc. using conventional complementary design. Section (IV) present the GDI technique. Section (V) present the simulation results and performance analysis of various ALU blocks with complementary and GDI technique and comparison of their performance & optimized area. Finally the work is concluded in Section (VI)

II. ARITHMETIC LOGIC UNIT

2.1 Arithmetic Unit

Employing fast and efficient adders in arithmetic logic unit will aid in the design of low power high performance system. Other operations such as subtraction and multiplication also employ addition in their operations, and their internal hardware is almost similar though not identical to addition hardware. Various adder families have been proposed in the past to trade-off power, area and speed for possible use in ALUs. The performance criticality of the ALU demands a dynamic adder implementation. Dynamic logic family of adders are the most efficient in terms of transistor-count, speed and power dissipation. This work covers the design of 3 bit adder using Complementary logic. This same Adder unit is used for the implementation of subtractor unit. This reuses the current hardware we made for adder and saves area.[5]

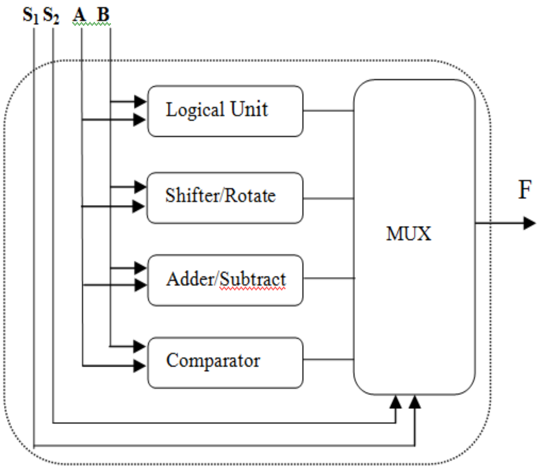


Fig.1 Internal Architecture of ALU

2.2 Logic Unit

ALU can perform various logic operations like NOT, AND, OR, NAND, NOR, XOR, XNOR etc. For these operations a special unit is made called as Logical Unit. This Logic Unit performs all logic operations asked to perform. A MUX operated by select lines, for which particular logic operation to perform, is used inside this logic block.[5]

2.3 *Comparator & Shifter unit*

A 1-bit comparator is made with the help of Complementary logic. It compare the two I/Ps and give three states of O/P for three different conditions. Also a 2:1 MUX is made which is used as a cell in the barrel shifter. Used for shifting and rotating operation.

2.4 *Low Power ALU using 4:1 Multiplexer*

There is a substantial increase in the standby mode leakage power, When technology is scaled from say 250nm to 180nm. Reducing the power consumption of the ALU of high-end processors is important not only because they consume a considerable percentage of processor energy, but also because they are one of the busiest component of the processor [5]. As a result of that they dissipate a lot of dynamic energy. This is aggravated by the exponential dependence of leakage on the temperature, & ALU also become a site of high leakage. The total leakage of the ALU can be given as

IS,T=N.IS,i

Where, N= number of transistors in the ALU IS,i= sub threshold leakage of gate i which is a function of gate length L.[2]

Similarly, the dynamic power of the ALU is given as
ID=α.Ceff.Vdd2.f

where α is the switching factor
Ceff. is the total effective capacitance
Vdd is the supply voltage
& f is the frequency of operation.

III. COMPONENTS DESIGN WITH COMPLEMENTARY LOGIC

The various components are designed using complementary logic(CMOS logic). Figure shows the schematic design of various components such as ADDER, SUBTRACTOR, COMPARATOR, SHIFTER, LOGICAL GATES (AND,OR,NOT).

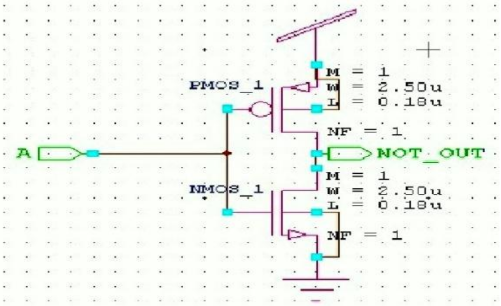


Fig. 2 Inverter Using CMOS Logic

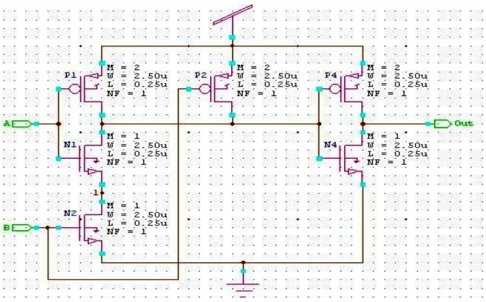


Fig.3 2-input AND gate using CMOS logic

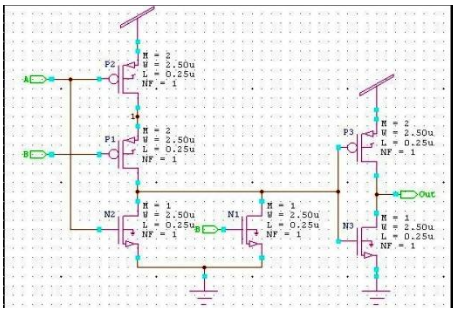
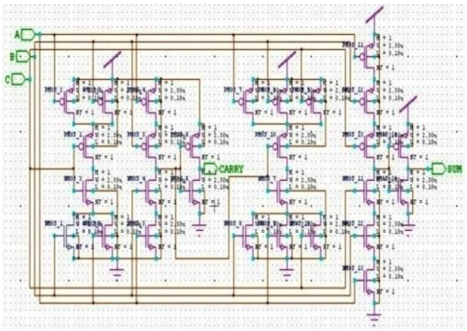


Fig..4 2-input OR gate using CMOS logic

Mostly all the other circuits are implemented with the above three gate circuit.



.

Fig.5 1-bit adder circuit using CMOS logic

Similarly subtractor is implemented. Also with this logic MUX can be implemented. Using which a barrel shifter will be made. The schematic with complementary logic of 2:1 MUX and Barrel Shifter using that MUX cell is given below

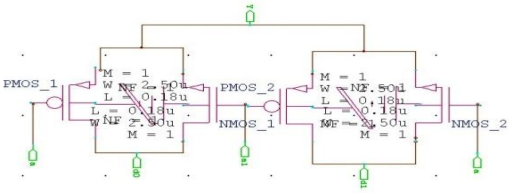


Fig. 6 2:1 MUX with CMOS technology

Here the a and a1 are complementary I/P used as select line.

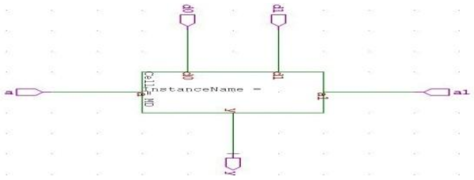


Fig. 7 Cell representation of MUX

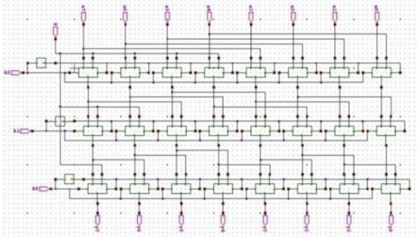


Fig. 8 Barrel shifter using MUX cell

Using the CMOS logic MUX the above barrel shifter is made and so are the other components of ALU. But the problem associated with this logic is that we require too much number of transistors, Which results in more power consumption and more area for chip. So in order to get a more optimized and low power consuming circuit we've to use a technique where the no. of transistor used will be less. That's why we will use one such technique known as Gate-Diffusion-Input (GDI) method.

IV. GDI METHOD AND COMPONENTS DESIGN

The Gate-Diffusion-Input (GDI) method is based on the use of a simple cell as shown in Fig. 9. One may be think of the CMOS inverter in the first look of this circuit, but there are some major differences in the two: (1) The GDI cell contains three inputs—*G* (common gate input of NMOS and PMOS), *P* (input to the source/drain of PMOS), and *N* (input to the source/drain of NMOS). (2) Bulks of both NMOS and PMOS are connected to *N* or *P* (respectively), so it can be arbitrarily biased in contrast to CMOS inverter.[4] The basic GDI cell is shown in Fig. 9

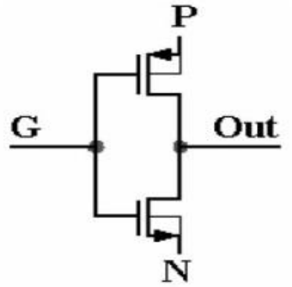


Fig. 9 Basic Gate-Diffusion-Input cell[4]

The GDI cell with four ports can be recognized as a new Multi-functional device, which can achieve six functions with just different combinations of inputs *G*, *P* and *N*. TABLE 1. shows that simple configuration changes in the inputs *G*, *P*, and *N* of the basic GDI cell can lead to very different Boolean functions at the output *Out*. Most of these functions are complex (usually consume 6-12 transistors) in CMOS, while very simple (only 2 transistors per function) in the GDI design methodology. Meanwhile, multiple-input gates can be implemented by combining several GDI cells.

Sr. No.	Input			Output	Function
	P	G	N		
1	B	A	0	$\overline{A}.B$	F1
2	1	A	B	$\overline{A} + B$	F2
3	B	A	1	$A + B$	OR
4	0	A	B	$A.B$	AND
5	B	A	C	$\overline{A}.B + A.C$	MUX
6	1	A	0	$\overline{A}$	NOT

TABLE I- Functions of the basic GDI cell[4]

The XOR gate made with the help of GDI cell is a application of the GDI technique. As it can be seen in Fig. 10, the XOR made using GDI technique requires only four transistors. Obviously, the proposed GDI XOR gate use less transistors compared with the conventional CMOS XOR.

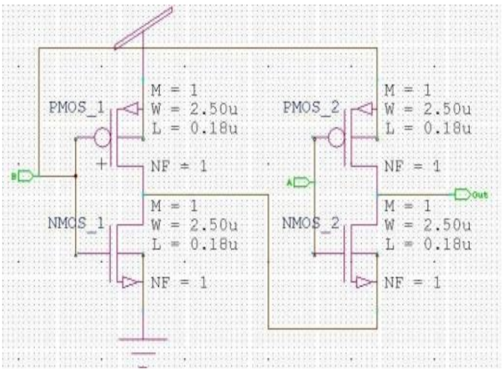


Fig. 10 XOR using GDI cell

Now using this same XOR circuit many other more complex circuit can be implemented such as a Full Adder which will require very less transistor than its counterpart. A such 1-bit full adder is shown in fig. 11.

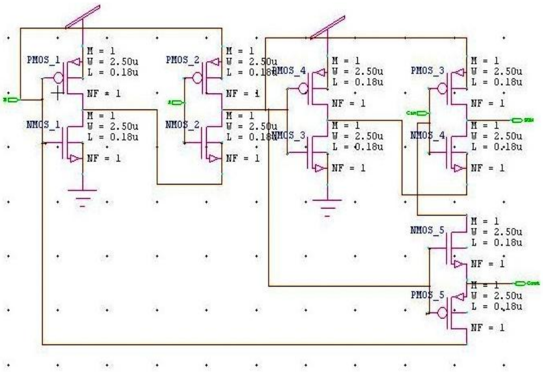


Fig. 11. Full adder using XOR-GDI gate

Simulation result of some of the operation using GDI technique are given below using Tanner EDA W-Edit tool.

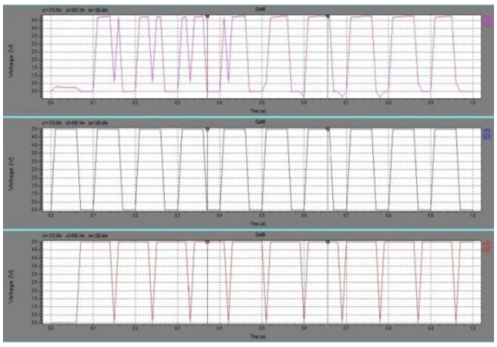


Fig. 12. W-Edit Simulation of AND gate with GDI technique

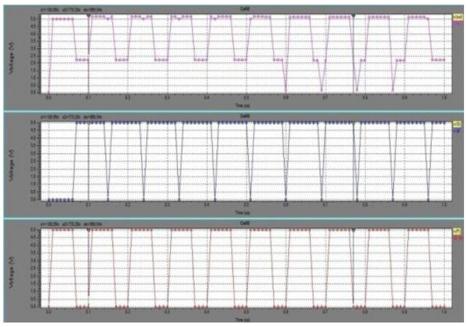


Fig. 13. W-Edit Simulation of OR gate with GDI technique

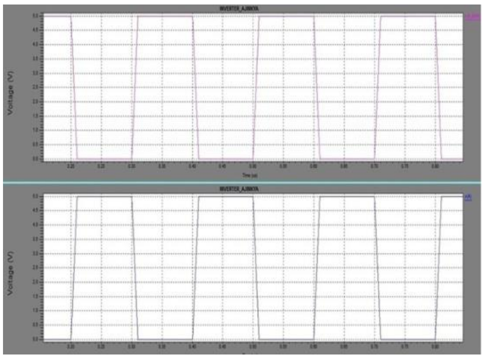


Fig. 14. W-Edit Simulation of NOT gate with GDI technique

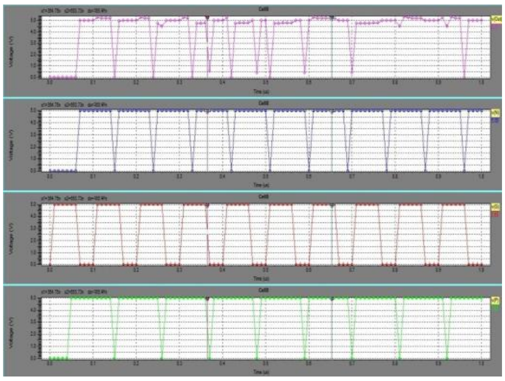


Fig. 15. W-Edit Simulation of MUX 2:1 with GDI technique

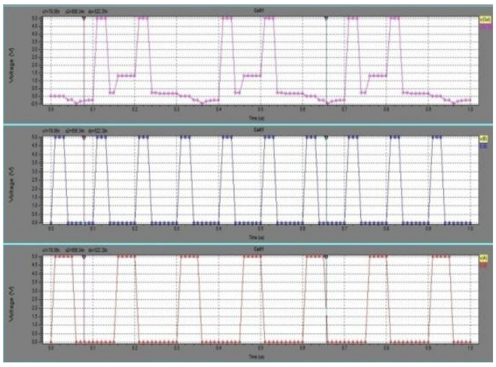


Fig. 16. W-Edit Simulation of XOR gate with GDI technique

V. RESULT

As the number of transistor is reduced in the GDI technique ALU it is obvious that its area is optimized. Apart from this optimized area of ALU the other evident advantage we get is speed. Apparently as the number of transistor used is reduced the operating time is also reduced and operation are done in less time. So our new ALU is also fast in operation as compare to its counterpart. Also due to some attractive features which allow improvements in design complexity, transistor counts, static power dissipation and logic level swing, research on GDI is becoming feverish in VLSI area. However, the GDI scheme suffers the defect of special CMOS process, specifically, it requires twin-well CMOS or silicon on insulator (SOI) process, which are more expensive than the standard p-well CMOS process[4][9].

Sr. No.	Sub Components	No. of transistor required	
		CMOS	GDI
1	Adder	28	10
2	Subtractor	26	12
3	AND2	6	2
4	OR2	6	2
5	XNOR	16	4
6	XOR	16	4
7	NOT	2	2
8	NAND2	4	4
9	NOR2	4	4
10	Shifter 8-bit	96	48
11	2:1 MUX	4	2

TABLE II- Comparison of Transistor used

Similarly with certain modification the subtractor circuit can be made which again will use less transistor than its CMOS counterpart. Same is the case with each gate's and other component's circuit. Every design made with this GDI technique have considerably reduced the no. of transistor. The following table compares the no. of transistor used in Complementary Logic Design and The GDI design technique.

VI. CONCLUSION

In this paper, the traditional CMOS technique and GDI Technique for the designing of ALU is discussed first. An area optimizing technique is introduced and the components with GDI technique are implemented. Later the comparison between the number of transistor used in old and GDI design of ALU is done. And in the demonstration it is shows that this GDI design evidently reduces the number of transistor and hence optimize the area of ALU as well increase its working speed.

REFERENCES

[1] Shrivastava, A.; Kannan, D.; Bhardwaj, S. Vrudhula, Reducing functional unit power consumption and its variation using leakage sensors, IEEE transactions on very large scale integration (VLSI) systems, vol. 18, no. 6, pp.988-997, June 2010

[2] Sanu K. Mathew, Mark A. Anders, and Ram K. Krishnamurthy, High-Performance Energy-Efficient Dual- Supply ALU Design, High-Performance Energy Efficient Microprocessor Design, Springer, pp.171-187, 2006.

[3] Beom Seon Ryu, Jung Sok Yi, Kie Young Lee and Tae Won Cho, "A DESIGN OF LOW POWER 16-B ALU",IEEE TENCON, 0-7803-5739-6/99/$1O.OO ©1999 IEEE.

[4] Dan Wang, Maofeng Yang, Wu Cheng, Xuguang Guan, Zhangming Zhu, Yintang Yang, "Novel Low Power Full Adder Cells in 180nm CMOS Technology", ICIEA 2009, 978-1-4244-2800-7/09/$25.00 ©2009 IEEE

[5] MEETU MEHRISHI, S. K. LENKA, "VLSI Design of Low Power ALU Using Optimized Barrel Shifter", International Journal of VLSI and Embedded Systems-IJVES, ISSN: 2249 – 6556

[6] M.Kamaraju, K.Lal Kishore, A.V.N.Tilak, "Power Optimized ALU for Efficient Datapath", International Journal of Computer Applications (0975 – 8887), Volume 11– No.11, December 2010

[7] Vaibhav Neema, Pratibha Gupta, "Design Strategy for Barrel Shifter Using Mux at 180nm Technology Node ", International Journal of Science and Modern Engineering (IJISME), ISSN: 2319-6386, Volume-1, Issue-8, July 2013

[8] Arun Prakash Singh, Rohit Kumar, "Implementation of 1-bit Full Adder using Gate Diffusion Input (GDI) cell ", International Journal of Electronics and Computer Science Engineering, ISSN-2277-1956/V1N2-333-342

[9] Arkadiy Morgenshtein, Alexander Fish, and Israel A. Wagner, "Gate-Diffusion Input (GDI): A Power-Efficient Method for Digital Combinatorial Circuits", IEEE Transactions On Very Large Scale Integration (VLSI) Systems, Vol. 10, No. 5, October 2002

[10] Hamid Reza Naghizadeh, Mohammad Sarvghad Moghadam, Saber Izadpanah Tous and Abbas Golmakani, "Design of Two High Performance 1-Bit CMOS Full Adder Cells", International Journal of Computing and Digital Systems 2, No. 1, 47-52 (2013)