

Low Power-Delay Design of 4-Bit ALU Using GDI Technique and Its Comparison

Sukesh B Pednekar¹, Sheela Kore²

¹IV Semester M.Tech. Student, Dept. of Electronics and Communication Engineering
K.L.E. Dr. M.S.Sheshgiri College of Engineering and Technology, Belagum-590008, Karnataka, India-590018

²Faculty Dept. of Electronics and Communication Engineering
K.L.E. Dr. M.S.Sheshgiri College of Engineering and Technology, Belagum-590008, Karnataka, India-590018

Abstract: This paper propose the new technology called Gate Diffusion Input [GDI] method for low power-high speed implementation of 4-bit ALU. The ALU is the most important block (brain) of central processing unit and is essential in the applications such as DSP, microprocessors and embedded systems. In this implementation the ALU block needs full adder, 2-bit multiplexer, 4-bit multiplexer and some basic gates, which are implemented to perform Logic Operations, Arithmetic Operations, Increment And Decrement. The sub-components of the ALU block are designed using GDI cell in order to reduce total power of the circuitry. The simulation is done using 180nm technology on cadence virtuoso platform.

Keywords: Arithmetic Logic Unit, Full Adder, Gate Diffusion Input(GDI), CMOS logic, Power, Delay .

1. Introduction

As a VLSI design engineers the main issues which we are facing today are Power dissipation and delay in the circuit. In Present era, As the technology is increasing tremendously and when we speak in terms of IC design, more number of transistors getting invoked into an IC, which in any physical design increases the power dissipation and total area of device. So, today, it is very important for scaling design in any device. With the evolution of the nanotechnology and due to the tremendous progress of modern electronic system, the low- power & high speed microelectronic devices has been enlightened. There are various technologies and design styles on which we can work to design our circuits but at the same time on every design technology there are some limitations which cannot be denied completely.

1 bit full adders are mostly used in the complicated circuits where the computational complexity is increased, So in order to increase the overall performance of the circuit, it is necessary to apply the low power techniques on the basic units followed by performance analysis. The most important block of any intelligent system is microprocessor, where microprocessor works as the heart of intelligent systems. When we speak in terms of microprocessor, it consist of 4 essential blocks namely a ALU unit, Program counter, a Register unit, a Control unit as shown in Figure 1.

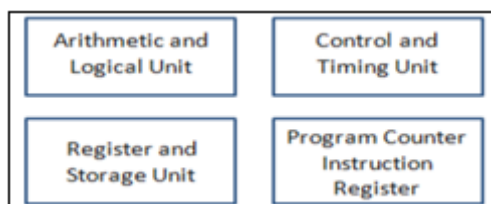


Figure 1: Basic block diagram of Microprocessor

Among the 4 essential blocks of microprocessor, Arithmetic logic unit is the most important block in microprocessor. As

mentioned, microprocessor is the heart of CPU, likewise ALU also functions as the heart of the microprocessor. As the ALU performs very important and complex computational operations of the microprocessor, the need of high speed microprocessor with low power dissipation can be met with the considerations, that the low power techniques should be applied on the ALU as it is the main block of microprocessor.

2. Gate Diffusion Input Technique

Gate diffusion input technique is a new low power technique, that over comes the existing logic style in many ways. The gate diffusion input(GDI) cell consist of only two transistor, where one is NMOS and the other is PMOS, it assembles similar to that of CMOS inverter. In CMOS inverter, source of PMOS is connected to VDD and source of the NMOS is connected to the ground. Unlikely in GDI both the terminals are not connected to VDD and ground, instead we use both the terminals as the control inputs [3].

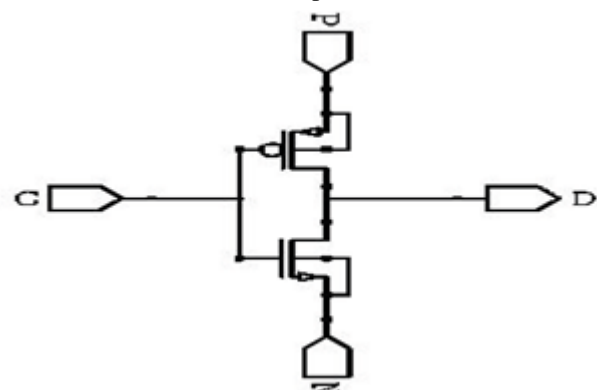


Figure 2: GDI basic cell

The Figure 2 shows the basic GDI cell where it consists of: 3 input terminals:

- G (shorted gate input of NMOS and PMOS)
- P (source terminal of PMOS)

- N (source terminal of NMOS)

1 output terminal:

- D (shorted drain terminals of NMOS and PMOS)

In case of GDI cell both the bulk and the corresponding source terminals are not connected to supply, instead both are shorted in order to provide the random bias to the transistors.

Table 1: Different functions of GDI cell

N	P	G	OUTPUT	FUNCTION	TRANSISTOR COUNT
0	1	A	A'	Inverter	2
0	B	A	A'B	F1	2
B	1	A	A'+B	F2	2
1	B	A	A+B	OR	2
B	0	A	AB	AND	2
C	B	A	A'B+AC	MUX	2
B'	B	A	A'B+AB'	EXOR	4
B	B'	A	AB+A'B'	EXNOR	4

We can realize many Boolean functions using only 2 transistors GDI cell. Table 1 shows the realization of different Boolean operations just by changing the control inputs of the cell [5]. When the 'N' terminal is at logic 0, 'P' terminal at logic 1 then the input at terminal 'G' will make the cell to function as an inverter. Similarly when terminal 'N' is applied with input B, 'G' applied with input A and 'P' terminal at VDD, then the cell functions as a OR gate. In the same way different basic gates can be realized just by changing the control inputs of the GDI cell.

3. Methodology

This chapter gives the detailed description and implementation of the GDI Block and its sub components.

3.1 Block Diagram of ALU

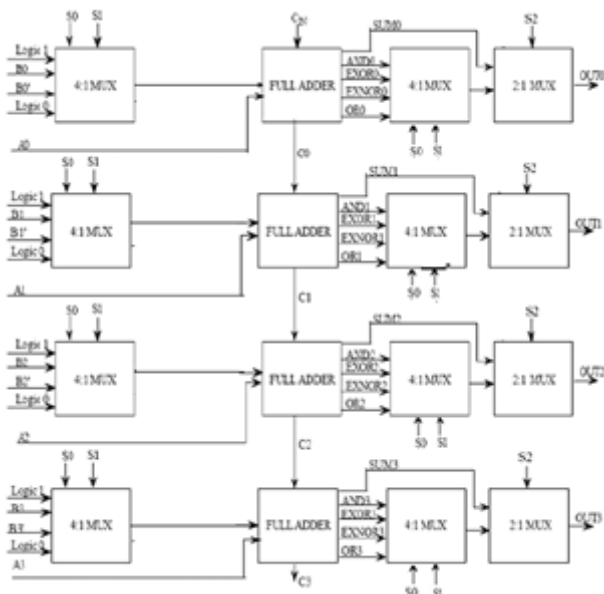


Figure 3.1: General block diagram of 4-bit ALU

The Figure 3.1 shows the generalized block diagram of 4-bit ALU which consists of '8' 4:1 Multiplexers, '4' 2:1

Multiplexers and 4 Full-adders and some of the basic gates which are required to perform the logic operations. The block diagram consists of 3 select lines those are S0, S1 and S2 where for each combination of the select lines the ALU block performs unique operation.

Table 2: Truth table for 4-bit ALU

SELECTION LINES			OPERATION
S2	S1	S0	
0	0	0	AND
0	0	1	EXOR
0	1	0	EXNOR
0	1	1	OR
1	0	0	DECREMENT
1	0	1	ADDITION
1	1	0	SUBTRACTION
1	1	1	INCREMENT

The 4-bit ALU consists of 2 input combinations each of 4-bit a0-a3 and b0-b3 with output lines from Q0-Q3. The Table.2 shows the truth table for 4-bit ALU where the control inputs are the select lines from the multiplexers. ALU block computes a specific operation for every combination of select lines from 000-111. In the case when select lines S0..S2=000 the ALU performs AND operation, and that for 001 combination it performs the EXOR operation and similarly it performs different operations for all the remaining combinations.

When the INCREMENT and DECREMENT Operations are need to be performed, the Logic '1' and Logic '0' are applied as the inputs. An INCREMENT operation can be analyzed by adding '1' to the addend and DECREMENT operation can be seen as subtraction operation [2].

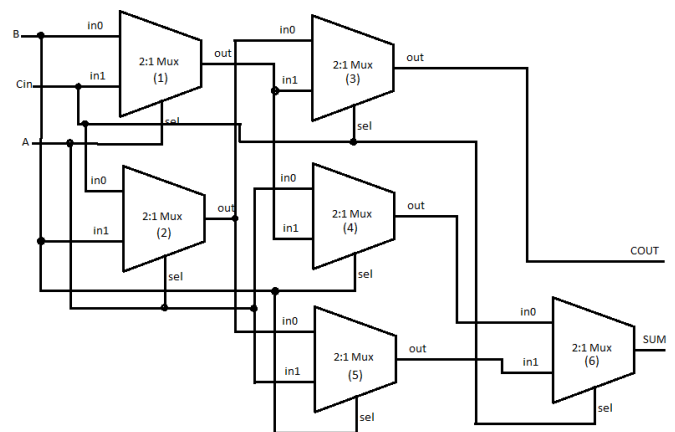


Figure 3.2: 10 transistors Full Adder using GDI technique

As the technology is increasing day by day, there is significant development in the structure of design in last few decades. Adders that are considered to be basic building blocks of complex application like ALU, DSP and microprocessors has been always under perusal now or then [6].

The Figure 3.2 shows the Full Adder block where it has 3 input lines and 2 output lines, input lines are from the 4:1

Multiplexer. The input from 4:1 Multiplexer will be either logic 0-1, input B or its inversion B', to perform Arithmetic operations like Addition, Subtraction, Increment or Decrement.

4. Implementation

This chapter shows the implementation of ALU block and all its sub components carried on the Cadence Virtuoso tool, which consist of schematics and layouts of the design.

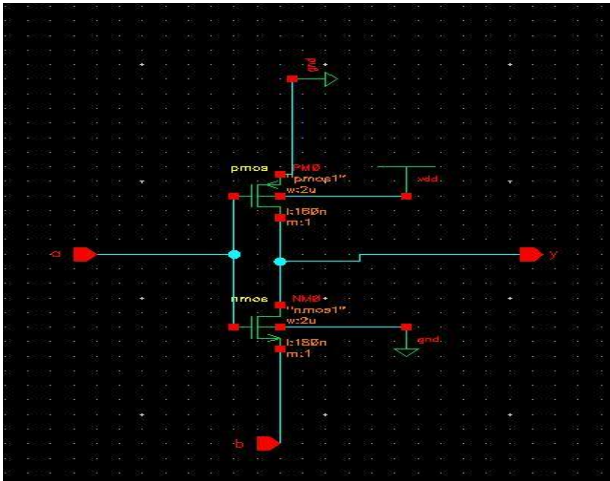


Figure 4.1: GDI AND gate

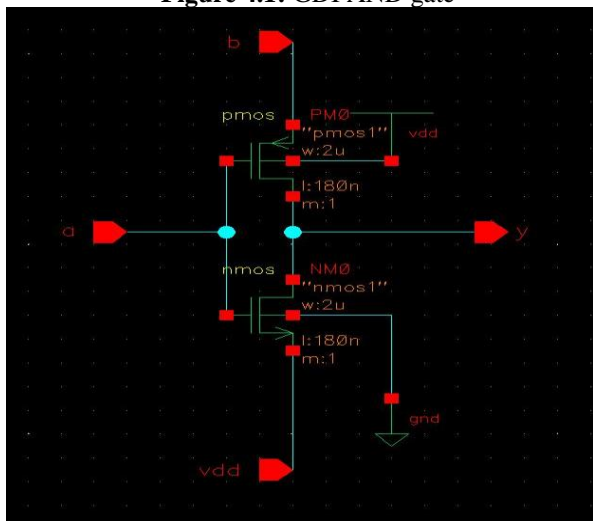


Figure 4.2: CMOS OR gate

Figure 4.1-4.4 shows the implementation of the basic gates using GDI technique where it is shown that AND, OR, EXOR realization can be done using fewer transistors than that of the CMOS technique which results in reducing the power consumption. Figure 5.4.2 shows the 2 transistor 2:1 multiplexer that is further used to for applicable in implementing an efficient Full Adder.

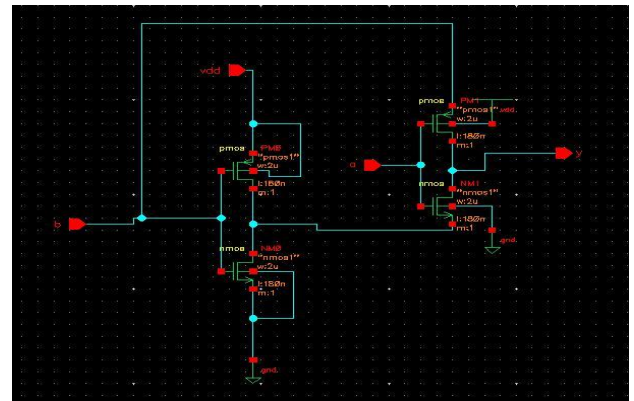


Figure 4.3: GDI EXOR gate

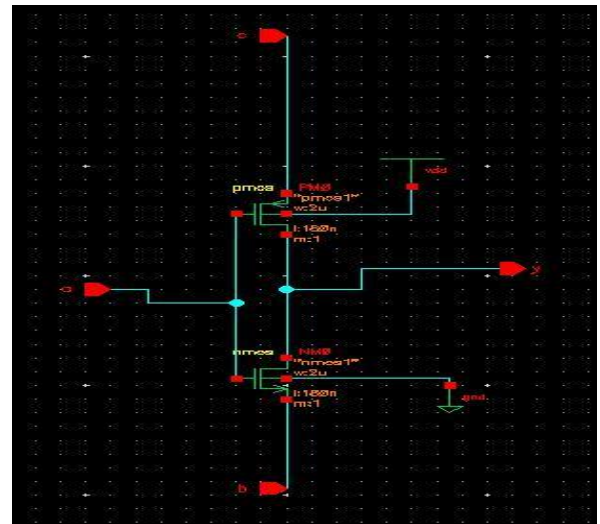


Figure 4.4: GDI 2:1 Multiplexer

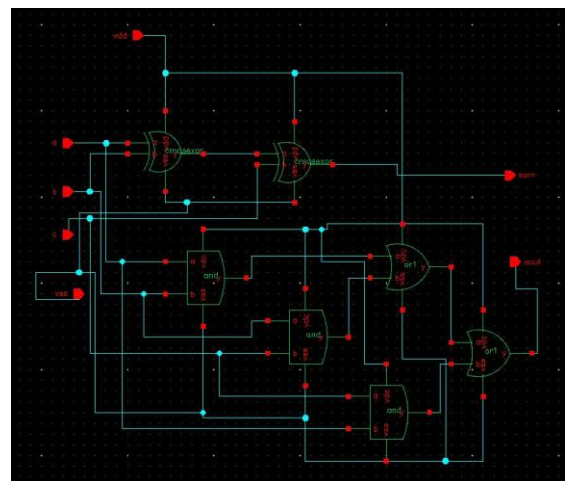


Figure 4.5: 1-bit full adder CMOS implementation

The Figure 4.5 and 4.6 shows the implementation of 1-bit full adder using CMOS and GDI technique. We can see that the complexity is more in the currently being used CMOS full adder as it requires of about 54 transistors due to which it exhibits a large power dissipation and is time consuming. As we know that full adder is the very important component in ALU, and with the aim of reducing power and delay of the circuit, we can use the new technique called GDI as it uses only 12 transistors, hence resulting in an efficient architecture. The proposed full adder is designed by cascading 2:1 multiplexers.

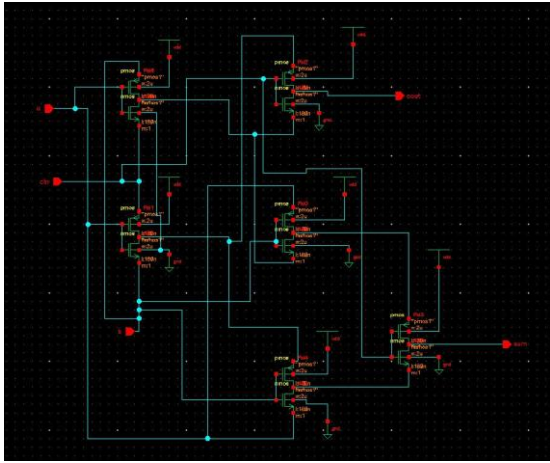


Figure 4.6: 1-bit full adder GDI implementation

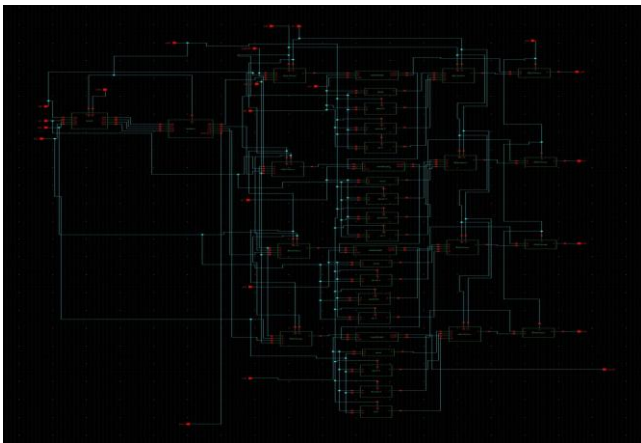


Figure 4.7: Schematic implementation of 4-bit ALU

The schematic implementation of 4-bit ALU is shown in Figure 4.7. Where it requires 4:1 Multiplexer, 2:1 Multiplexer, Full adders and some basic gates to perform Logical operations, which is then extended by paralleling all the interconnecting blocks to implement the 4-bit ALU. As seen from the Figure 5.6.1 the design consists of 3 select lines where each combination of the select line selects unique set of blocks to perform specific operation. Therefore, as we have 3 select lines there will be total 8 combinations which corresponds to 8 different operations. The 4:1 Multiplexer is responsible for selecting logic 0, logic 1, b and b' (any one at the specific unit of time), where b is selected whenever the Addition operation is need to be performed and b' is selected whenever the Subtraction operation has to be performed. The logic 0 and logic 1 lines are selected whenever the Increment or Decrement operations are to be performed, during Increment the input data needs to be added by 1 and that for Decrement the input data is decremented by 1. Then the Full Adder perform Addition on the values that arrive at its input lines along with the carry input, and propagating the carryout through the concurrent stages producing the carryout of the ALU from the final stage of the design. 2:1 multiplexer at the output stage is very important block because it is responsible for selecting the Arithmetic or Logic operations to be performed depending on its select line(S2). When S2=0, it selects the blocks to perform Logical operation, and when S2=1, it selects the blocks to perform the Arithmetic operation.

The subtraction is done by taking the 2's complement of the input which need to be subtracted by the other input value, like in our design we have been subtracting input b from input a. To obtain the 2's complement of input b, first we have to invert all the bits of input b and then adding 1 to its LSB and then to perform Subtraction, the 2's complement of b is added to the input a.

The Figure 4.8 shows the Layout of 4-bit ALU which is efficiently designed and then verified by performing all the required checks like, DRC check(Design Rule Check), ERC check(Extracted Resistance and Capacitance) and LVS check (Layout versus Schematic) and found NO ERRORS, further the layout helps us in calculating the total area required by the design.

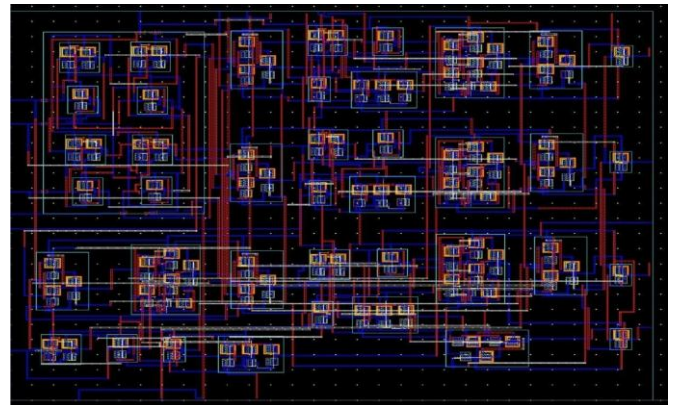


Figure 4.8: Layout of 4-bit ALU

5. Results and Analysis

This section shows the Transient Simulation results of 4-bit ALU designed using GDI technique.

5.1 Simulation Results

As seen from the Figure 5.1.1, we have the first 3 control signals which corresponds to select line S0-S2, then we have 2 inputs signals a and b, each of 4-bit i.e. a0-a3 and b0-b3, the 4 output lines y0-y3 and then finally the Carryout and the Borrow signals.

The simulation is performed in Virtuoso Analog Design Environment (ADE) using Spectre tool with the stop time of 100ms. For simplification, the input signals are of constant value throughout the period i.e., a=1111 and b=0111. When we analyse the simulation, the results are as expected i.e., it performs different operations with the alteration in its control bits. The only thing that the GDI lags in, is that the output voltage level is bit degraded as compared to that in CMOS.

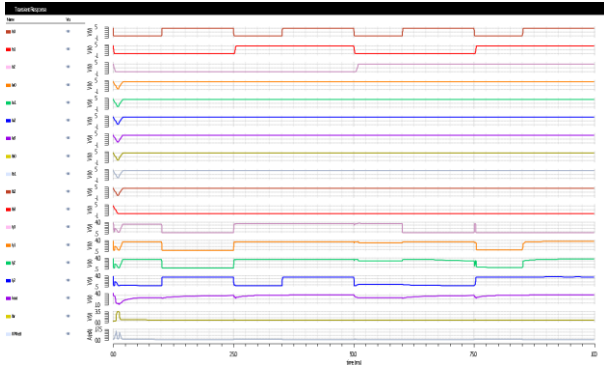


Figure 5.1.1: Simulation results for GDI based 4-bit ALU.

5.2 Power Analysis

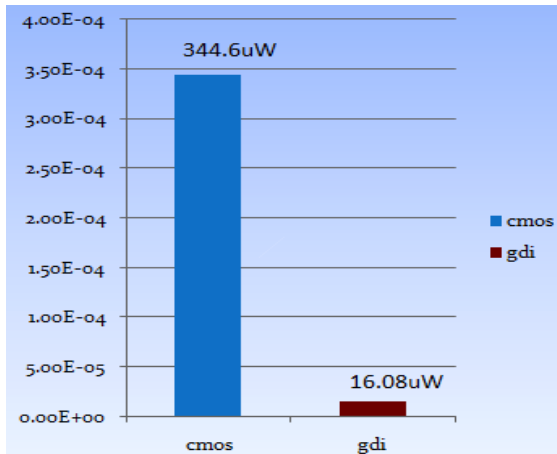


Figure 5.2.1: Power Comparison (OR)

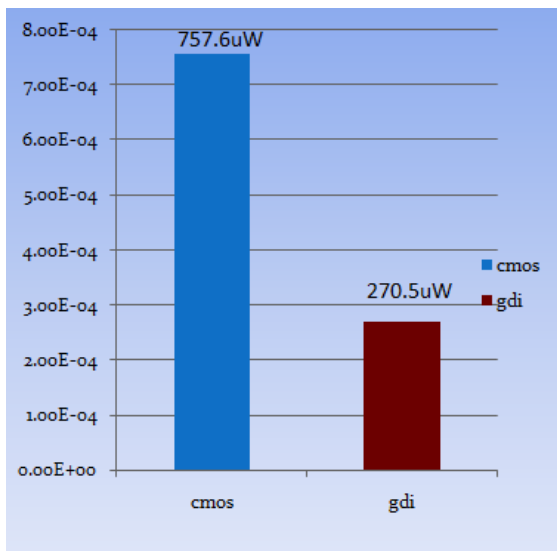


Figure 5.2.2: Power Comparison (AND)

The Figure 5.2.1-5.2.7 shows the power comparison between CMOS and GDI for ALU block, full adder and all the subcomponents of ALU

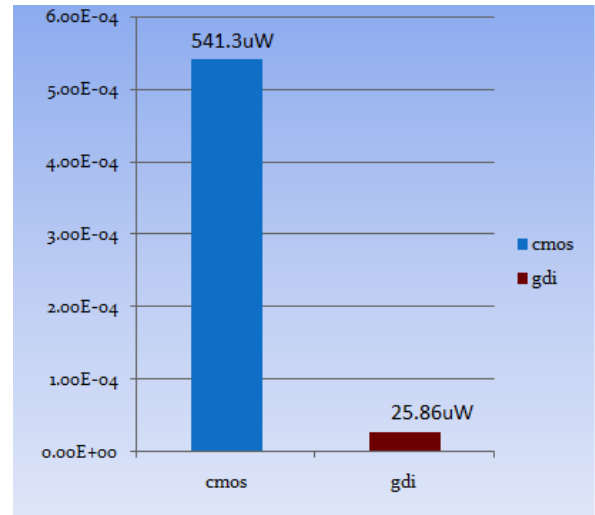


Figure 5.2.3: Power Comparison (EXOR)

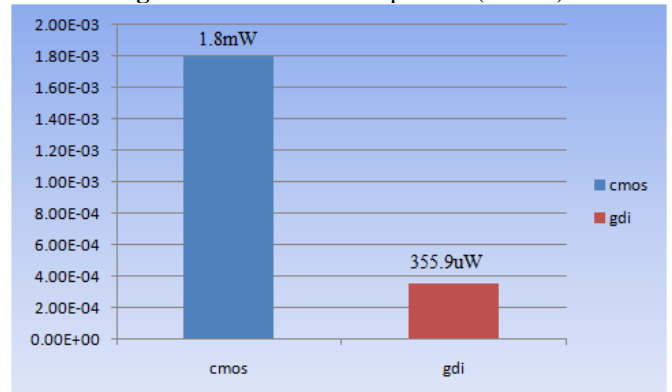


Figure 5.2.4: Power Comparison (2:1 Mux)

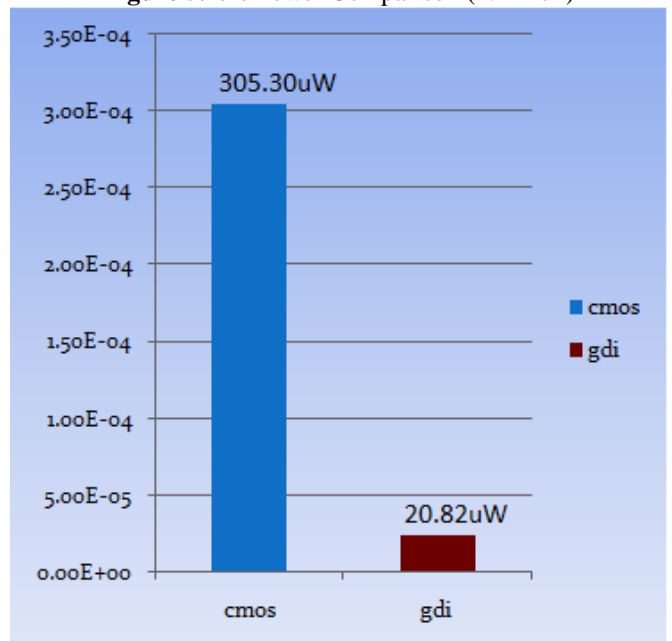


Figure 5.2.5: Power Comparison (1-bit FA)

The Figure 5.2.6 shows the Power comparison of 4-bit ALU block for CMOS and GDI implementation. The ALU designed using presently being used CMOS method exhibits of about 32.80mW for specific time interval as it is 10ns in our design, likewise the same ALU block can be successfully realized using new low power technique GDI, which reduces about half of the power consumption as compared to CMOS i.e., 10.610mW.

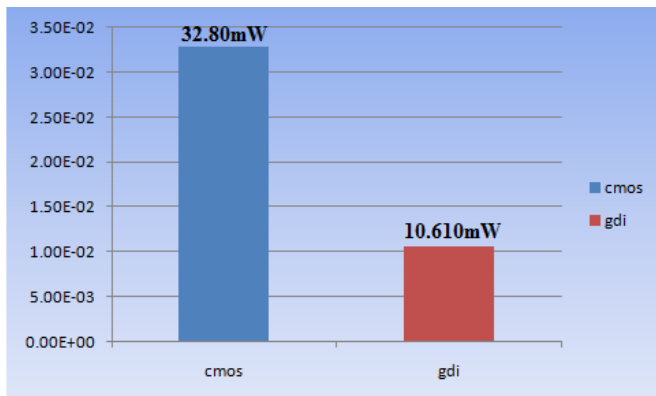


Figure 5.2.6: Power Analysis of ALU

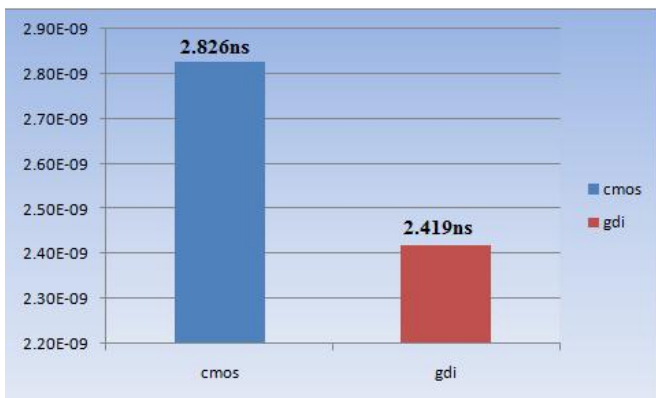


Figure 5.2.7: Delay Analysis of ALU

Similarly the Figure 5.2.7 shows that the delay of the ALU block is also reduced from 2.82ns to 2.419ns.

The Table 3 shows the comparison of the ALU block and its subcomponents using CMOS and the GDI method. As seen from Table, the number of transistors required in realization of OR gate using CMOS technique needs about 6 transistors and that using GDI requires just 2 transistors, which leads to the reduction in power dissipation of about 95.3% as compared to CMOS technique, Similarly the amount of power dissipation is reduced for all the basic gates using GDI method. Then for Full Adder using CMOS we require about 54 transistors which dissipates 1.8mW amount of power and using GDI it requires only 12 transistors which dissipates about 355.9uW of power i.e., 80.2% of power reduction can be achieved.

The low power techniques applied on all the subcomponents of ALU helps in reducing the power consumption of the entire Architecture. When a ALU is designed using CMOS method with 448 transistors, it results in high power dissipation of 32.08mW. Which can be reduced to greater extent using new low power GDI technique which requires only 104 transistors(one fourth number of transistors as compared to CMOS) causing the power dissipation of 10.610mW. Therefore using GDI technique power dissipation of ALU can be reduced by 67.6%, Delay can be reduced by 407ps and Area can be reduced from 210nm to 27.4nm.

Table 3: CMOS V/S GDI Comparison Table Of ALU And Its Sub Components

CMOS					GDI				
Functional Block	No. of Transistors	Power Dissipation	Delay	Area	No. of Transistors	Power Dissipation	Delay	Area	Reduction in Power Dissipation (%)
OR	6	344.6uW	1.04ns	-	2	16.08uW	992.4ps	-	95.3%
AND	6	305.30uW	1.011ns	-	2	20.82uW	989.6ps	-	93.18%
EXOR	6	757.6uW	509.2ps	-	2	270.5uW	506.3ps	-	64.2%
2:1 Mux	14	541.3uW	533.2ps	-	2	25.86uW	499.7ps	-	95.2%
4:1 Mux	26	1.567mW	1.636ns	-	6	122.8uW	1.004ns	-	92.16%
Full adder	54	1.8mW	1.077ns	-	12	355.9uW	1.02ns	-	80.2%
ALU	448	32.80mW	2826ps	210nm	104	10.610mW	2419ps	27.4nm	67.6%

6. Conclusion

This proposed GDI based implementation of the circuits reduces the dynamic power of the circuit. A 4-bit low power ALU is designed and implemented using this technique in 180nm technology. The low power technique is applied on multiplexers, full adders as these are the main components to build ALU. So the total power, area and delay of the ALU is reduced. The Simulation shows that the ALU is implemented with less power that is 505.8mW using GDI technique and this is very less as compared to 1.11W (CMOS technology)

i.e. 67.6% of power reduction is achieved. While reducing the Area from 210nm to 27.4nm and delay of 407ps.

References

- [1] T. Sakurai, "Closed-form Expressions for Interconnection Delay, Coupling, and Crosstalk in VLSI's", IEEE Transactions on Electron Devices, vol. 40, no. 1, pp. 118-124, January 1993.
- [2] F. Najm. A Survey of Power Estimation Techniques in VLSI Circuits (Invited Paper). IEEE Transactions on VLSI Systems, 2(4):446-455, December 1994.

- [3] A.Morgenshtein, A. Fish, and Israel A. Wagner, "Gate-Diffusion Input (GDI): a novel power efficient method for digital circuits: a design methodology", *Proceedings of 14th Annual IEEE International ASIC/SOC Conference*, pp. 39-43, 2001.
- [4] N. H. E. Weste, D. Harris, and A. Banerjee, *CMOS VLSI Design: A Circuits and Systems Perspective*, 3rd ed. Delhi, India: Pearson Education, 2006.
- [5] Mohan Shoba, Rangaswamy Nakkeeran, "GDI based full adders for energy efficient arithmetic applications" *Engineering Science and Technology, an International Journal* 19 (2016) 485–496.
- [6] Kshitij Shant, Rita Mahajan, "1-Bit Hybrid Full Adder by GDI and PTL Technique" *International Journal of Innovative Research in Computer and Communication Engineering*, Vol. 4, Issue 5, May 2016.