A Novel and Efficient Low Power 8-Bit Arithmetic Logic Unit Using FS-GDI Technique

K.Srinivasa Rao[1],Tadiboina Pavan Kumar[2],Pendem Sudheer[3],Gudimetla Lakshmi Vara Prasad[4]

[1]Assistant Professor,Department of ECE,Andhra Loyola Institute of Engineering and Technology,Vijayawada,A.P.,India.
[2,3,4]B.Tech,Student,Department of ECE,Andhra Loyola Institute of Engineering and Technology,Vijayawada,A.P.,India
(E-mail: jntuksr@gmail.com[1],pavankumar6163@gmail.com[2],sudheerpaani232@gmail.com[3],prasadgudimetla434@gmail.com[4])

***Abstract*— Power dissipation and area of the circuit are the main issues in the electronics industry.This paper provides a design of 8-Bit Arithmetic Logic Unit (ALU) using Full-Swing GDI and CMOS techniques.The proposed ALU design consists of 2x1 Multiplexer, 4x1 Multiplexer and low power Full Adder circuits to realize the arithmetic and logic operations. The simulation carried out using Tanner EDA Tools V13.0. The results show that the GDI design consume less power using less number of transistors, while achieving full swing operation compared to CMOS design.**

***Keywords*— Arithmetic Logic Unit (ALU); Gate Diffusion Input (GDI); Full-Swing GDI,CMOS.**

I. INTRODUCTION

We use a lot of portable electronic devices in our daily life,; which are basically low power high speed VLSI circuits. One of these circuits is the Arithmetic logic unit (ALU) which considered as an essential component in many applications such as Microprocessor, digital signal processing, image processing, etc.

ALU is used to perform both the logical and arithmetic operations.The power and area are the major problems in ALU implementation,so that we preferred low power logic design style i.e..,GDI.But in the GDI technology,output voltage swing degradation,fabrication complexity & power consumption are the disadvantages.Due to this reason Modified GDI was proposed.The Modified gate diffusion input (MOD-GDI) technology is more efficient in terms of power consumption,fabrication complexity and low output voltage swing degradation when compared to GDI technology.

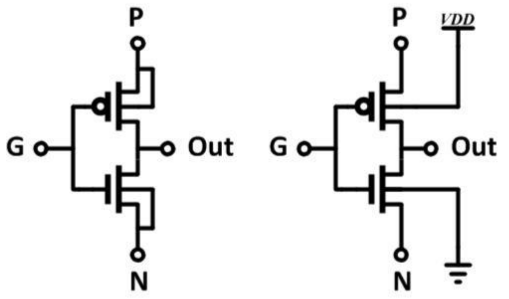
In this paper, an 8-bit alu is designed using an adder,logic block and multiplexers which are realized by the Full-Swing GDI technique and compared to CMOS work in terms of power dissipation and transistor count.

Simulations are carried out using S-EDIT,T-SPICE and W-EDIT in the Tanner EDA Tools V13.0.

This paper is organized as follows: Section II overviews the GDI methodology and presents its benefits and limitations. The circuits needed to design Arithmetic logic unit are discussed in Section III. The design of the arithmetic logic unit is discussed in Section IV.Section V presents simulation results and comparison. Section VI concludes the paper.

II. GATE DIFUSSION INPUT (GDI) TECHNIQUE

GDI Technique was first proposed by Arkadiy Morgenshtein, IdanShwartz and Alexander Fish [1] this technique allows implementation of various complex logic functions using only two transistors as listed in Table I. The original GDI was based on using a simple cell, as shown in Fig. 1(a)



(a) *(b)*

GDI cell; (a) originally proposed, (b) standard CMOS compatible

However, it was proposed for fabrication in twin-well CMOS or silicon on insulator (SOI) processes, it allowed improvement in power consumption, delay and area of digital circuits compared to CMOS and PTL techniques. The drawback in GDI cell was it suffered from reduced voltage swing due to threshold drops, which leads to performance degradation and increasing static power dissipation.

To improve the output of the GDI cells Swing restoration circuits utilized. Morgenshtein et al. [2] Proposed the Modified -GDI approach, shown in Fig. 1 (b) where the substrate terminals of NMOS and PMOS transistors connected permanently to GND and VDD, respectively. This modification enables fabrication of GDI cell in standard CMOS processes which is cost efficient compared to twin-well and (SOI) processes.

TABLE I. DIFFERENT LOGIC FUNCTIONS REALIZATION USING GDI CELL

N	P	G	OUT	Function
0	B	A	$\bar{A}B$	F1
B	1	A	$\bar{A}+B$	F2
1	B	A	A+B	OR
B	0	A	AB	AND
C	B	A	$\bar{A}B+AC$	MUX
0	1	A	$\bar{A}$	NOT

III.CIRCUITS IN ARITHMETIC LOGIC UNIT

In this section, the Full-Swing GDI technique is used to realize the circuits required to design the ALU as follows:

A. 2x1 Multiplexer

A multiplexer is a digital switch which chooses the output from several inputs based on a select signal[4]as shown in Fig. 2 .A 2x1 multiplexer consists 2 inputs and 1 selection line and 1 output.It is realized using GDI technique with 6 transistors.

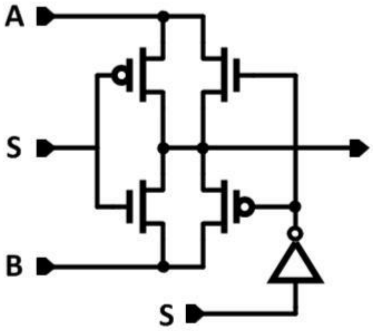


Fig. 2. Full-Swing GDI 2x1 Multiplexer

B. 4x1 Multiplexer

Using the previously discussed 2x1 multiplexer a 4x1 multiplexer realized as shown in Fig. 3.It consists only of 16 transistors.A count of 3 2x1 multiplexers are required to realize a 4x1 multiplexer.

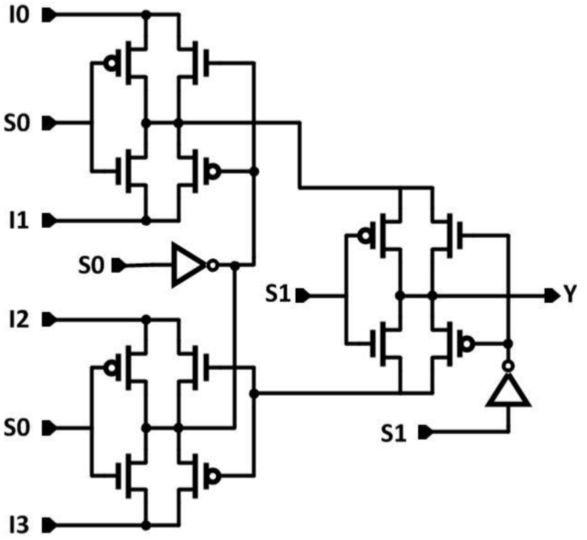


Fig. 3. Full-Swing GDI 4x1 Multiplexer

C. Full Adder

A full adder is a combinational circuit that performs the arithmetic sum of three input bits. It consists of three inputs and two outputs.

Adders are widely used in digtal VLSI systems.By improving the performance of adder cell,we can increase the performance of the ALU.Here Adder is the key component in the ALU to perform arithmetic operations.By realizing the full adder in less transistors reduces the area of the ALU.Full adder is realized using 18 transistors.

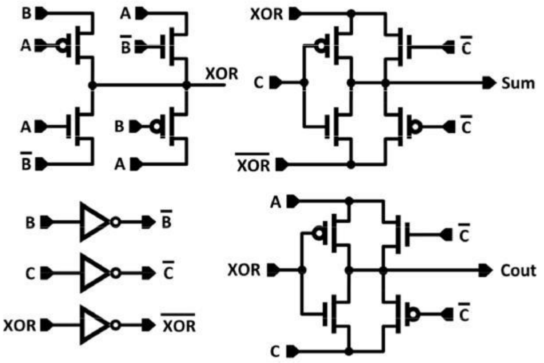


Fig. 4. Full-Swing Full Adder cell

E.Logic Block

Logic block is used to perform logical operations in the ALU.We can take these operations in the full adder cell also but,the circuit becomes complex and delay becomes high.So,we take a separate block in order to reduce the delay time .The Logic block can mainly perform 4 operations i.e..,AND,OR,XOR and XNOR.

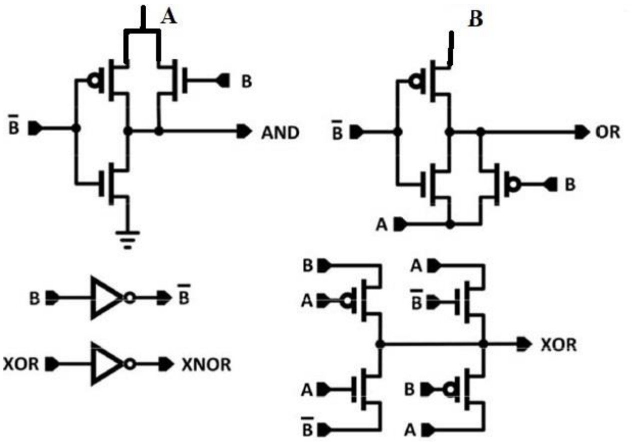


Fig. 5.Logic block

IV. Arithmetic Logic Unit

An ALU is a key component in the Central Processing Unit (CPU) of a computer.It is a combinational circuit which performs arithmetic operations such as addition, subtraction, increment, decrement and logic operations such as AND, OR, XOR and XNOR .

The proposed design of the 8-Bit ALU consists of 2 stages, each stage is an 4-Bit ALU tand 4-bit consists 4 stages,each stage is an 1-Bit ALU realized using the previously discussed circuits as follows:

Each 1-Bit ALU stage consists of two 2x1 multiplexers, two 4x1 multiplexers and one full adder cell,one logic block and one invertor as shown in fig 6..It has a number of selection lines(S0-S2) used to determine the operation to be performed.S2 selects between Arithmetic and Logic operaton.If S2=0,Arithmetic operations are realized and If

S2=1, Logical operations are realized.S0 and S1 specify the operation.

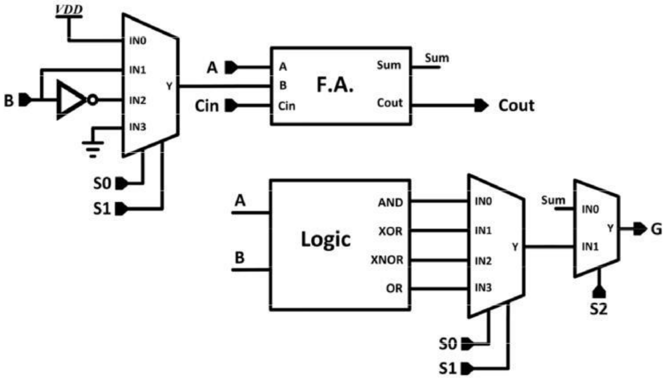


Fig. 6. Schematic of 1-Bit ALU Stage

The first 4x1 multiplexer is responsible for the B operand based on the values of S0 and S1 selection lines it selects from logic 1, B, B and logic 0 to generate the decrement,addition,subtraction and the increment operations respectively.

The output of first multiplexer is given to the operand B input of the full adder alongside with operand A and C output of the arithmetic operation is calculated form the next equation.

$$G = A + B + CIN$$

For decrement operation operand A is summed with logic 1 which represents -1 in 2's complement with C = 0, this gives G = A – 1. In addition operation operand A is summed with operand B with C = 0, this gives G = A + B. Subtraction is achieved using 2's complement operand A is summed with the complement of operand B with C = 1, this gives G = A +B+1 which is equivalent to A – B. And for increment operation,operand A is summed with logic 0 with C = 1, this gives G = A + 1.

The second 4x1 multiplexer used for selection of logic operation according to S0 and S1, while the second 2x1 multiplexer used to selects between arithmetic and Logic operation. Table II summarizes the truth table of the proposed 8-Bit ALU.

The 4-bit ALU design consists of four stages of 1-bit ALU and one 2x1 Multiplexer as shown in Fig. 7.

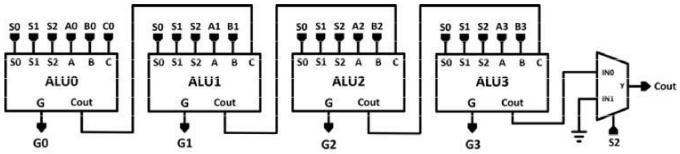


Fig. 7. Schematic of 4-Bit ALU Stage

The proposed 8-Bit ALU consists of 2 stages of 4-Bit ALU.It can perform a total of 12 operations as shown in TABLE II. Out of those,8 are Arithmetic and 4 are Logical. In Arithmetic operations,CIN plays an important role but in in logical operations,it does not needed because logical operations only depend upon both A and B inputs only.

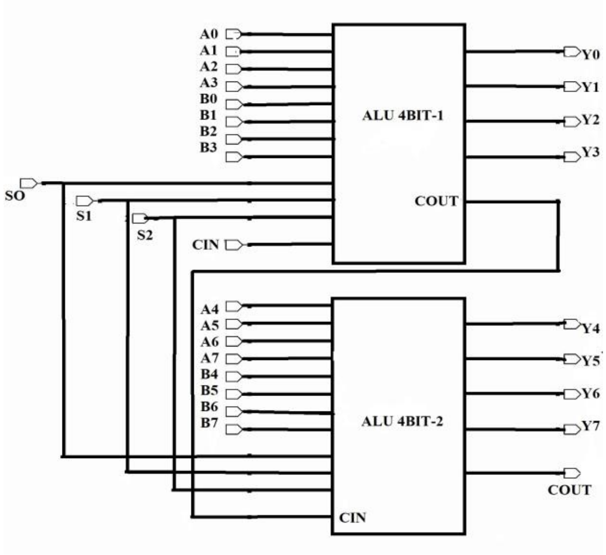


Fig. 8. Proposed 8-Bit ALU block diagram

TABLE II. TRUTH TABLE OF THE PROPOSED 8-BIT ALU

S2	S1	S0	CIN	OPERATION
0	0	0	0	DECREMENT OF A
0	0	0	1	TRANSFER OF A
0	0	1	0	A+B
0	0	1	1	A+B+CIN
0	1	0	0	A-B-CIN BAR
0	1	0	1	A-B
0	1	1	0	TRANSFER OF A
0	1	1	1	INCREMENT OF A
1	0	0	0	LOGICAL AND
1	0	1	0	LOGICAL XOR
1	1	0	0	LOGICAL XNOR
1	1	1	0	LOGICAL OR

IV. SIMULATION RESULTS

The proposed 8-Bit ALU designed using 250nm CMOS process.The simulations were done using the S-EDIT,T-SPICE and W-EDIT in Tanner EDA Tools V13.0 Using A=01111111, B=01010101 as test inputs. The proposed design compared to CMOS design in terms of power consumption and transistor count. Simulation results for the proposed 8-bit ALU are shown in Table III.

TABLE III .SIMULATION RESULTS OF THE 8-BIT ALU DESIGN

Design	Technology	No. of Transistors	Power(µW)
CMOS design	250nm	1194	4800.05.µW
GDI design	250nm	464	2100.40 µW

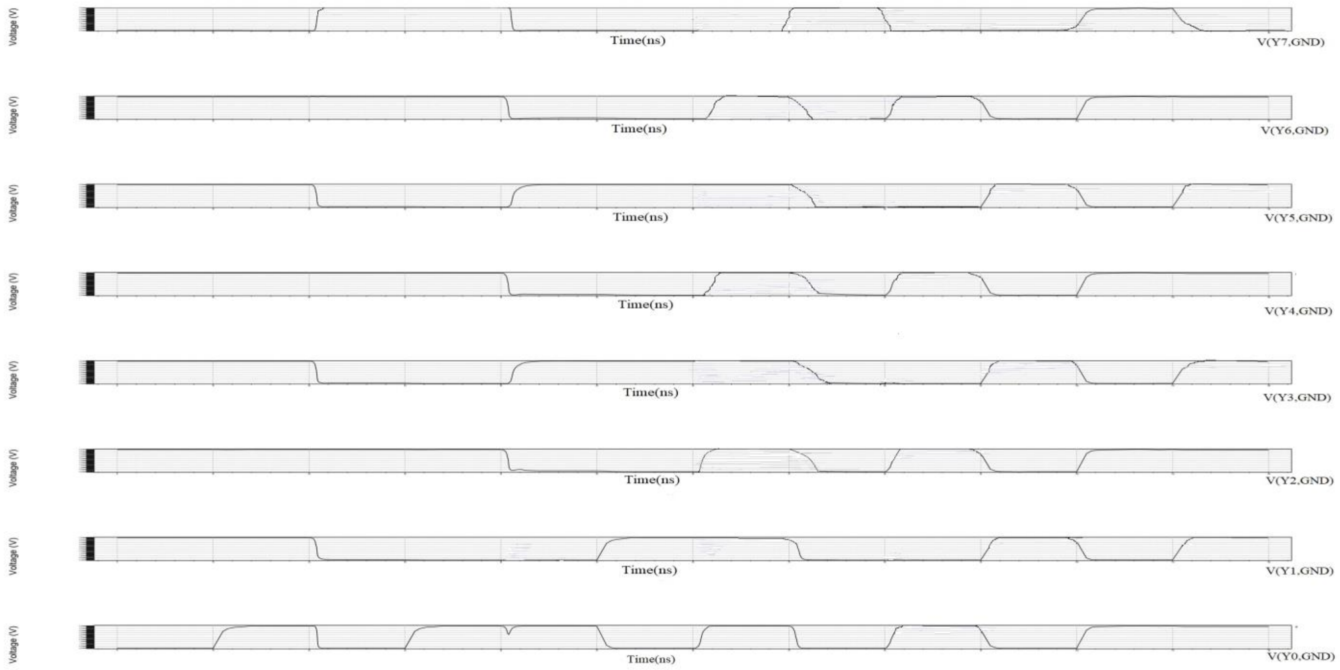


Fig. 9.The waveform of the Proposed 8-Bit ALU

The analysis and simulation indicate that the proposed ALU design technique optimized and reduced the area by 61% compared to the CMOS design, while maintaining full-swing operation and the power consumption of the 8-bit ALU is reduced by 52%.

CONCLUSION

In this paper a 8-Bit ALU is designed using the full-swing GDI technique optimized and reduced the area by 61% compared to the CMOS design, while maintaining full-swing operation. Hence the power consumption of the 8-bit ALU reduced by 52%.The proposed design consists of 464 transistors compared to CMOS 1184 transistors.Based on the results, it can be concluded that the proposed 8-bit ALU in full-swing GDI technique is suitable for low energy high - speed VLSI applications. Further study in this work would be using the 8-bit ALU as a building block to implement 16-bit and 32-bit ALU.

REFERENCES

[1] A. Morgenshtein, A. Fish, and I. A. Wagner, “Gate-diffusion input (GDI): A power-efficient method for digital combinatorial circuits,” *IEEE Trans. Very Large Scale Integr. Syst.*, vol. 10, no. 5, pp. 566– 581, 2002.

[2] A. Morgenshtein, I. Shwartz, and A. Fish, “Gate Diffusion Input (GDI) logic in standard CMOS nanoscale process,” *2010 IEEE 26th Conv. Electr. Electron. Eng. Isr. IEEE 2010*, pp. 776–780, 2010.

[3] A. Morgenshtein, V. Yuzhaninov, A. Kovshilovsky, and A. Fish, “Full-swing gate diffusion input logic - Case-study of low-power CLA adder design,” *Integr. VLSI J.*, vol. 47, no. 1, pp. 62–70, 2014.

[4] M. M. Mano and C. R. Kime, *Logic and Computer Design Fundamentals*. 2015.

[5] M. A. Ahmed and M. A. Abdelghany, “Low power 4-Bit Arithmetic Logic Unit using Full-Swing GDI technique,” in *Proceedings of 2018 International Conference on Innovative Trends in Computer Engineering, ITCE 2018*, 2018, vol. 2018–March, no. Itce, pp. 193– 196.

[6] M. Shoba and R. Nakkeeran, “GDI based full adders for energy efficient arithmetic applications,” *Eng. Sci. Technol. an Int. J.*, vol. 19, no. 1, pp. 485–496, 2016.

[7] M. A . Ahmed, M.A.Mohamed El-Bendary,Fathy Z.Amer,Said M.Singy,” Delay Optimization of 4-Bit ALU Designed in FS-GDI Technique” 2019 international conference on Innovative trends in computer engineering,ITCE’2019,vol.2019-February,pp 534-537,2019