

Generic 64-bit ALU Design for Low Power Processors Using GDI Technology

¹N. Solmon Udai Kiran,²Dr. M. Madhusudhan Reddy

¹PG Student, ²Assistant Professor

^{1,2} Dept of Electronics and Communication Engineering

¹G. Pullareddy Engineering College (Autonomous), Kurnool, India

Abstract—For building Arithmetic Logic Units (ALUs), numerous methods have been developed based on conventional CMOS technology due to its robustness and scalability. However, CMOS implementations often suffer from larger transistor counts, increased area, and higher power consumption. In this work, the Gate Diffusion Input (GDI) technique is proposed for designing a 64-bit Arithmetic Logic Unit (ALU) with enhanced efficiency. The GDI approach significantly reduces the number of transistors required compared to CMOS, which in turn lowers power consumption and minimizes silicon area—two of the most critical parameters in modern digital VLSI design. The proposed 64-bit ALU has been implemented and simulated using GPDK 180nm technology in Cadence Virtuoso at an operating voltage of 1.8V. Transistor sizing is carefully chosen, with PMOS devices sized at 1.2 μm and NMOS devices at 0.4 μm , to achieve optimized performance and maintain proper switching behavior. Simulation results validate that the GDI-based ALU achieves substantial power and area savings while maintaining reliable logic functionality, making it a promising alternative for low-power, high-performance VLSI applications.

Indexed Terms—FS-GDI m-GDI, ALU, low power.

I. INTRODUCTION

Arithmetic Logic Units (ALUs) are fundamental components of digital systems, as they perform essential arithmetic and logical operations in processors and signal processing units. With the increasing demand for portable electronics and high-performance computing, low power consumption and reduced chip area have become critical objectives in VLSI design. Conventional designs rely on CMOS technology, which is robust and widely adopted but requires a relatively large number of transistors to implement logic functions. This results in increased silicon area, dynamic power dissipation, and propagation delay, which limit efficiency in large-scale designs. As device scaling continues, these drawbacks make it challenging to meet modern requirements for power and area optimization.

To overcome these issues, this work proposes the use of the Gate Diffusion Input (GDI) technique for ALU design. GDI enables complex logic functions to be realized with fewer transistors, leading to significant reductions in power consumption and chip area compared to CMOS. In this paper, a 64-bit ALU is implemented using GDI in Cadence Virtuoso with GPDK 180 nm technology at 1.8 V supply voltage. Transistor sizing is chosen as PMOS = 1200 nm and NMOS = 400 nm to achieve optimized switching. The results highlight that GDI-based designs can offer substantial improvements over CMOS for power-efficient and compact VLSI systems.

II. METHODOLOGY

CMOS TECHNIQUE

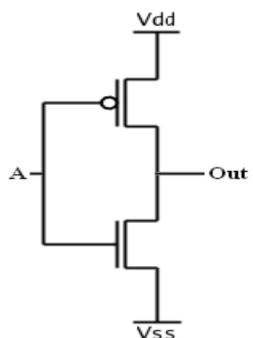


Fig 1: Basic CMOS cell

CMOS (Complementary Metal Oxide Semiconductor) is the most widely used design technique in VLSI due to its high noise immunity, scalability, and robustness. It relies on complementary pairs of pMOS and nMOS transistors to implement logic functions. CMOS logic is power-efficient in static conditions, as it only dissipates significant power during switching. However, it often requires a larger number of transistors to implement even simple logic gates, leading to increased area, capacitance, and delay. The Figure below depicts a CMOS basic cell.

B. GATE DIFFUSION TECHNIQUE

GDI (Gate Diffusion Input), on the other hand, is a relatively newer technique that modifies the basic cell structure of CMOS by allowing the gate, source, and drain terminals of transistors to be biased in different ways. This enables the realization of complex logic functions with fewer transistors compared to CMOS. As a result, GDI circuits exhibit reduced power consumption, lower propagation delay, and smaller silicon area. However, GDI suffers from certain drawbacks, such as voltage swing degradation, reduced noise margin, and design complexity when scaled for large systems, which limit its straightforward replacement of CMOS in standard processors without additional corrective measures like buffers or restoration circuits.

TABLE I: REALIZATION OF VARIOUS LOGICAL FUNCTIONS WITH GDI

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

In m-GDI, the bulk of PMOS and NMOS are connected to VDD and ground, respectively, as shown in Fig. 2(b). m-GDI can be fabricated in regular CMOS processes, which reduces the rate compared to silicon-on-insulator and twin-well processes.

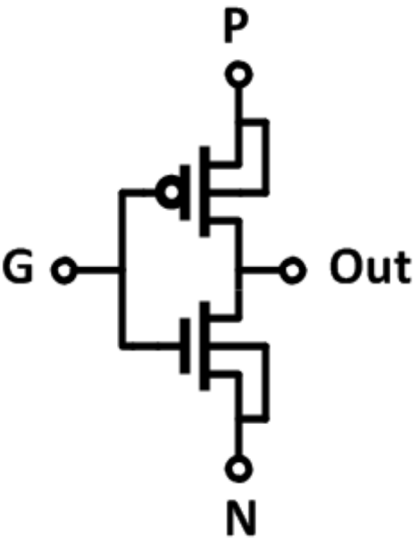


Fig. 2 (a) Basic Cell GDI

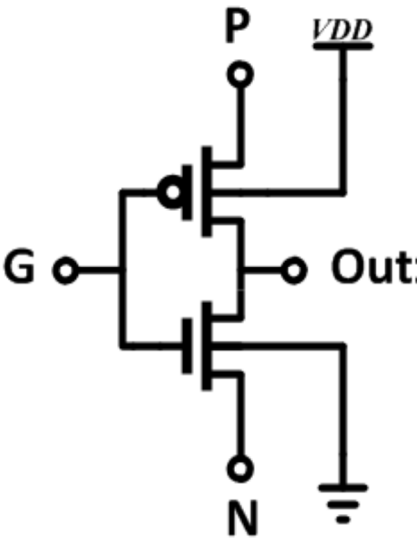


Fig. 2(b) CMOS Compatible

FS-GDI cells are proposed as an alternative to swing restoration buffers in this method, which uses swing restoration transistors to augment the output swing of F1 and F2 gates (universal gates). Using more transistors to achieve whole

swing operation is possible with this technology; however, fewer transistors are required than with CMOS realization, resulting in a smaller circuit size, improved power efficiency, and reduced latency.

IV. ARITHMETIC AND LOGIC UNIT

This study employs the FS-GDI technique to implement the circuits necessary for designing the ALU as follows.

INVETER

An inverter is designed using the Gate Diffusion Input (GDI) technique, as shown in Fig. 3, where the PMOS transistor has a width of 1.2 μm and the NMOS transistor has a width of 0.4 μm . This configuration ensures proper switching characteristics while reducing transistor count compared to conventional CMOS[4]. The inverter serves as the basic building block, from which the design is extended to include multiplexers, logic gates, and the full adder.

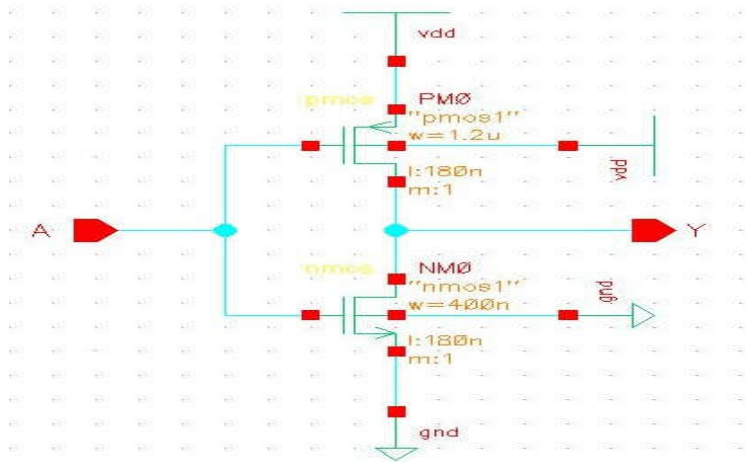


Fig 3. Inverter Using FS-GDI

AND GATE

The Gate, utilizing FS-GDI technology, is designed as shown in Fig. 4, and a symbol is created for use in further designs. The AND gate is realized using the GDI approach, which minimizes the number of transistors needed. This helps achieve lower power dissipation and area savings over CMOS implementation.

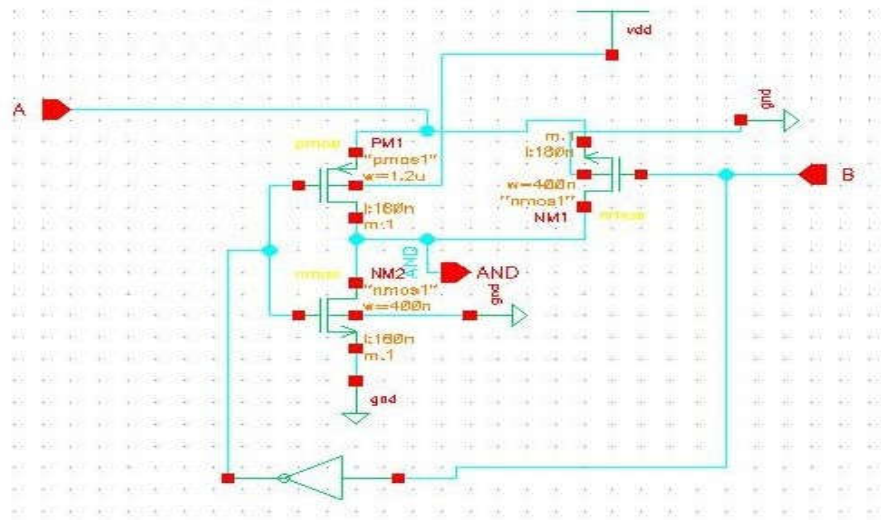


Fig.4 AND using FS-GD

OR GATE

Using the GDI method, the OR gate is designed with fewer transistors as shown in Fig. 5, leading to efficient logic operation. The reduced transistor count directly contributes to low power and area usage. The gate has been designed using FS GDI technology, and a symbol has been created for use in future designs.

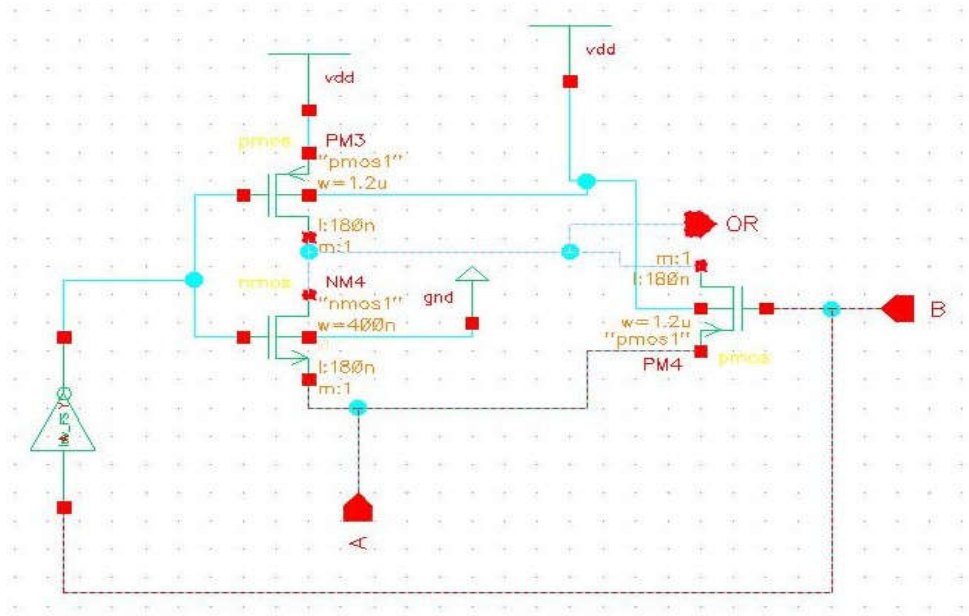


Fig 5. OR using FS-GDI

XOR GATE

The XOR gate is designed using GDI technology, as shown in Fig. 6, to perform exclusive logic with high efficiency. Compared to CMOS, it achieves better performance in terms of delay and power consumption. XOR has been designed using FS GDI technology; a symbol is created for use in future designs.

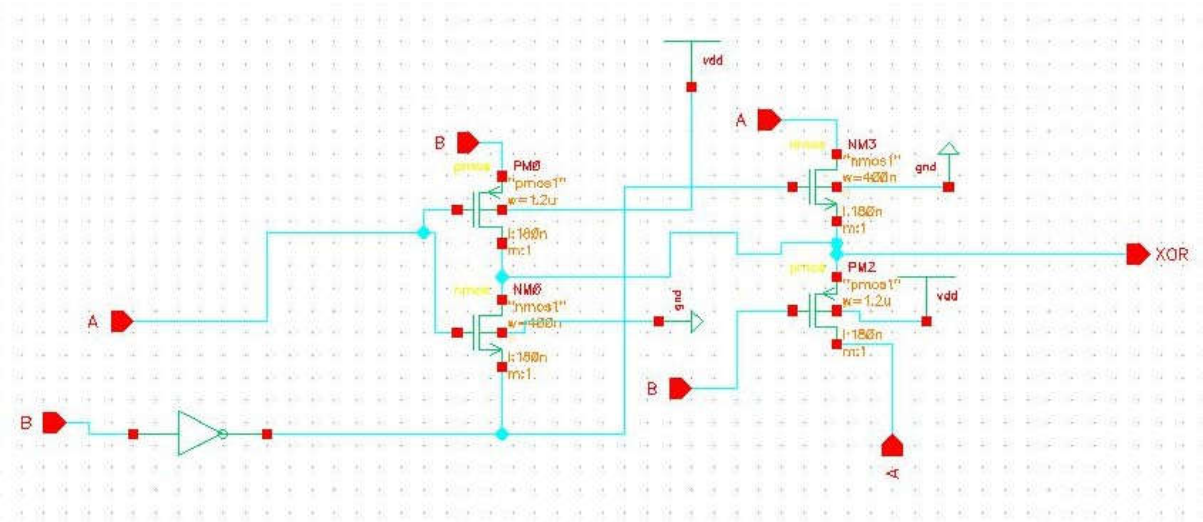


Fig.6 XOR using FS-GDI

XNOR

The GDI-based XNOR gate is implemented as shown in Fig. 7 to provide a compact realization of equivalence logic. This circuit consumes less power and achieves better speed efficiency. The XNOR Gate has been designed. A symbol is created for use in future designs.

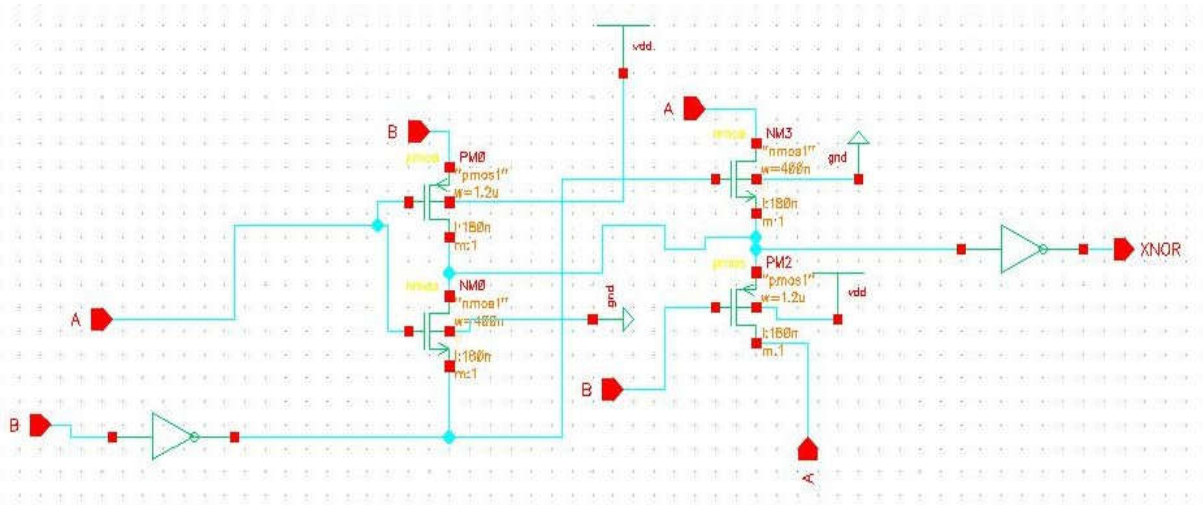


Fig.7 XNOR using FS-GDI

2×1 MULTIPLEXER

A 2×1 MUX is implemented using the GDI technique as shown in Fig. 8, which efficiently selects one of the two input signals based on the control input. This design reduces transistor count and power consumption. A 2X1 Multiplexer has been designed using FS-GDI technology[2], and a symbol has been created for use in future designs.

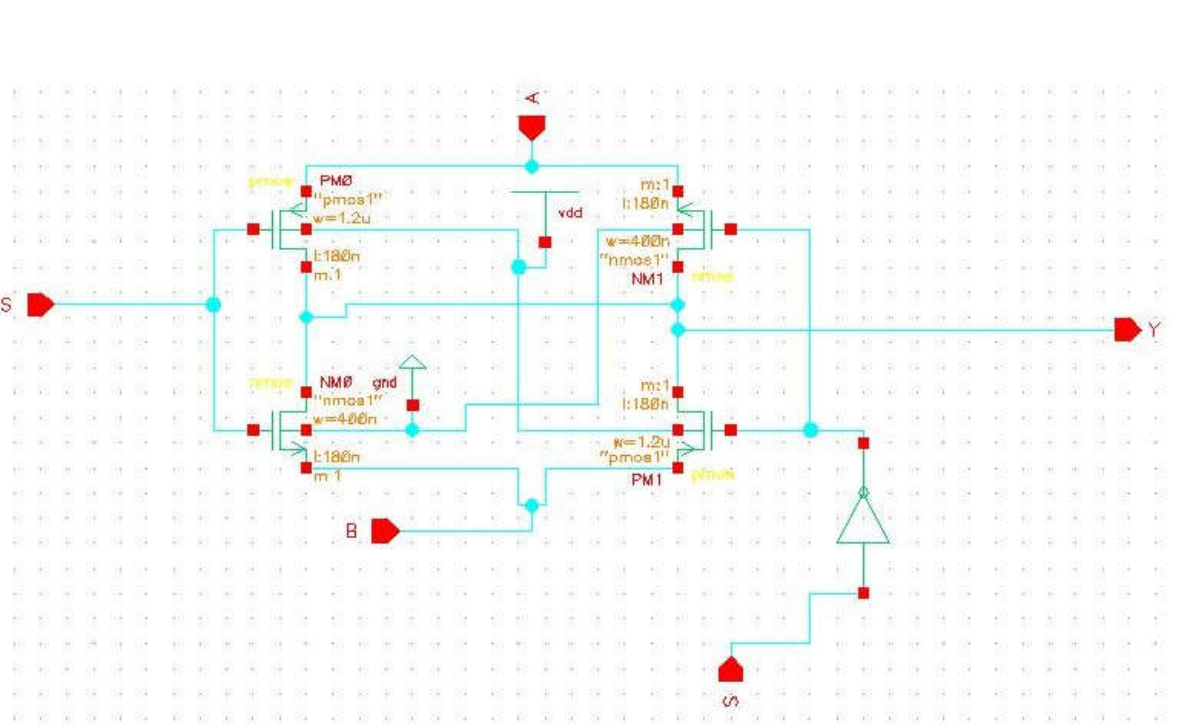


Fig.8 2X1Multiplexer using FS-GDI

4×1 MULTIPLEXER

The 4:1 MUX using GDI is constructed as shown in Fig. 9 by cascading smaller multiplexers, allowing for the selection among four input signals. The design ensures compact structure and reduced switching power. The multiplexer has been designed using FS GDI technology, and a symbol has been created for use in future designs.

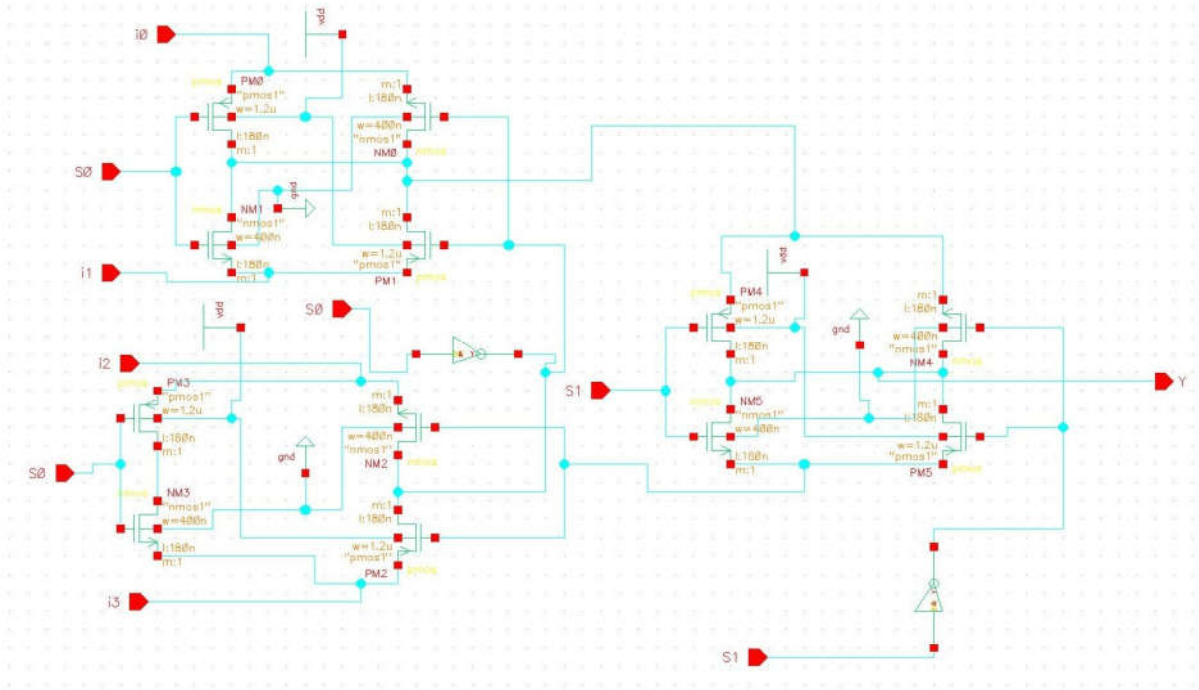


Fig.9 4X1 Multiplexer using FS-GDI

H. FULL ADDER

A complete adder circuit is constructed using GDI-based logic blocks such as XOR, AND, and OR. The design achieves reduced complexity, lower power, and improved speed, making it suitable for arithmetic operations in ALU design. A full Adder has been designed using FS GDI technology, as shown in Fig. 10, and a symbol has been created for use in future designs.

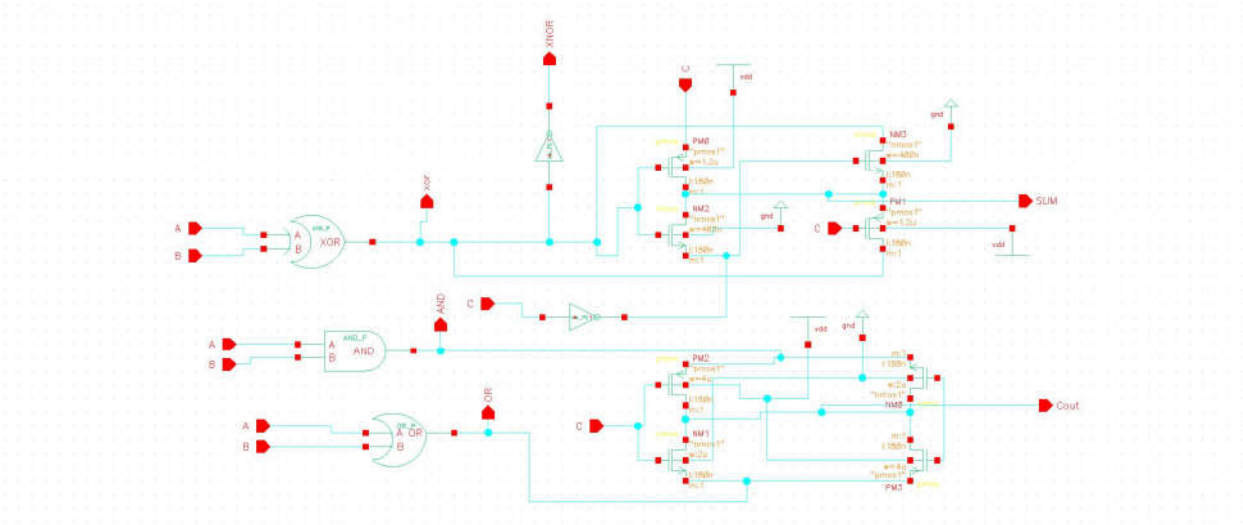


Fig.10 Full adder using FS-GD

1-Bit ALU

A one bit ALU as shown in Fig. 11 has been designed using previously made symbols Inverter, two 2×1 Multiplexer one 4X1 Multiplexer, and a Full adder.

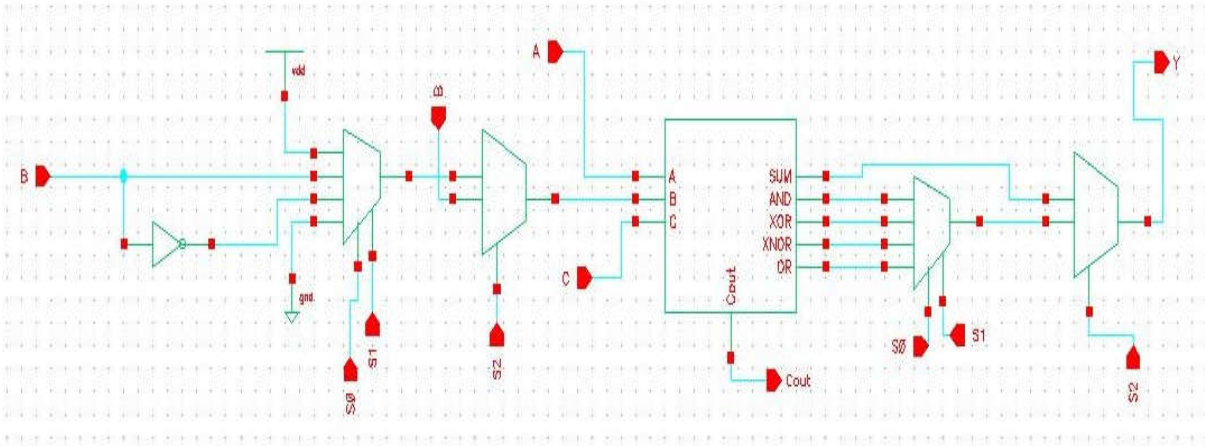


Fig 11. 1-Bit ALU

The sub-blocks of the 1-bit ALU are realized using FS-GDI: AND, OR, XOR, XNOR gates, 2×1 and 4×1 multiplexers, and a full adder. Transistors are resized to balance power and delay. In the proposed Design, the size of PMOS and NMOS is sized at 1200nm and 400nm, respectively, maintaining a 3:1 ratio to optimize switching characteristics. The design and simulation were carried out in Cadence Design Suite using the 180nm GPDK process technology and a supply voltage of 1.8V.

TABLE II Truth Table of the Proposed 64-Bit ALU

S2	S1	S0	OPERATIONS
0	0	0	DECREMENT
0	0	1	ADDITION
0	1	0	SUBSTRACTION
0	1	1	INCREMENT
1	0	0	AND
1	0	1	XOR
1	1	0	XNOR
1	1	1	OR

4 Bit ALU

The schematic in Fig. 12 below shows a 4 Bit ALU, which has been designed using four blocks of a one-bit ALU.

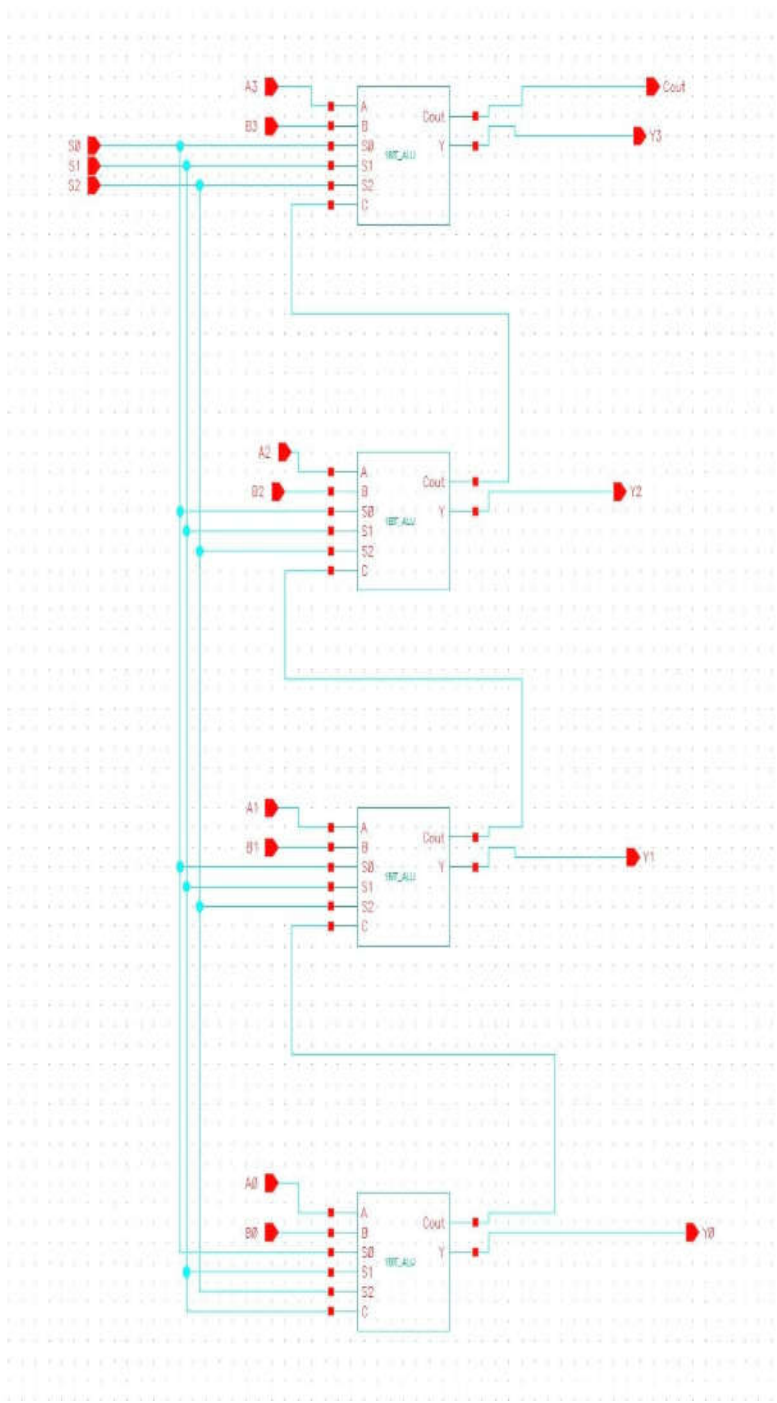


Fig 12. 4-Bit ALU using GDI Technology

16 Bit ALU

A 16 Bit ALU has been designed using two 8-bit ALUs. Figure 14 shows the implemented schematic of the 16 Bit ALU.

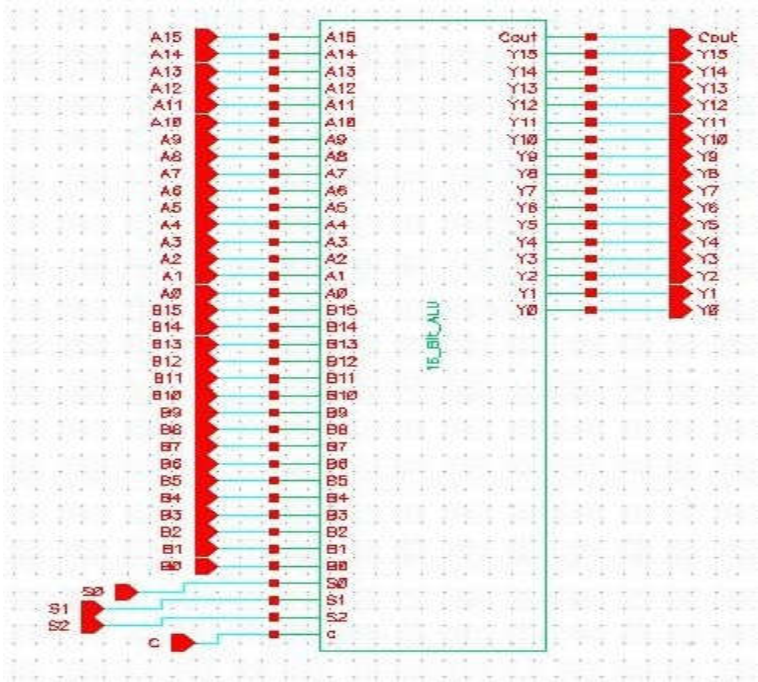


Fig 14. 16-Bit ALU Using GDI Technology

V. Proposed 64-Bit ALU Design

The designed 64-bit ALU performs both arithmetic and logic operations, including addition, subtraction, increment, decrement, AND, OR, XOR, and XNOR. Each 1-bit stage, as shown in Fig. 11, consists of two 2×1 MUXs, two 4×1 MUXs, and one full adder cell. As shown in Fig. 16, two 32-bit stages are cascaded to form a 64-bit ALU. The type of operation performed by the ALU will be determined by the selection line inputs (S0, S1, S2). The Truth Table 2 provides a summary of operations performed by the ALU. Based on the values of the select lines S0 and S1, the 4x1 multiplexer connected to the B input selects one of four inputs logic 1, B, $\bar{B}$, or logic 0 to perform the operations of decrease by one, sum, difference, and increase by one, respectively. The control signal S2 decides whether the circuit performs an arithmetic or a logic operation.

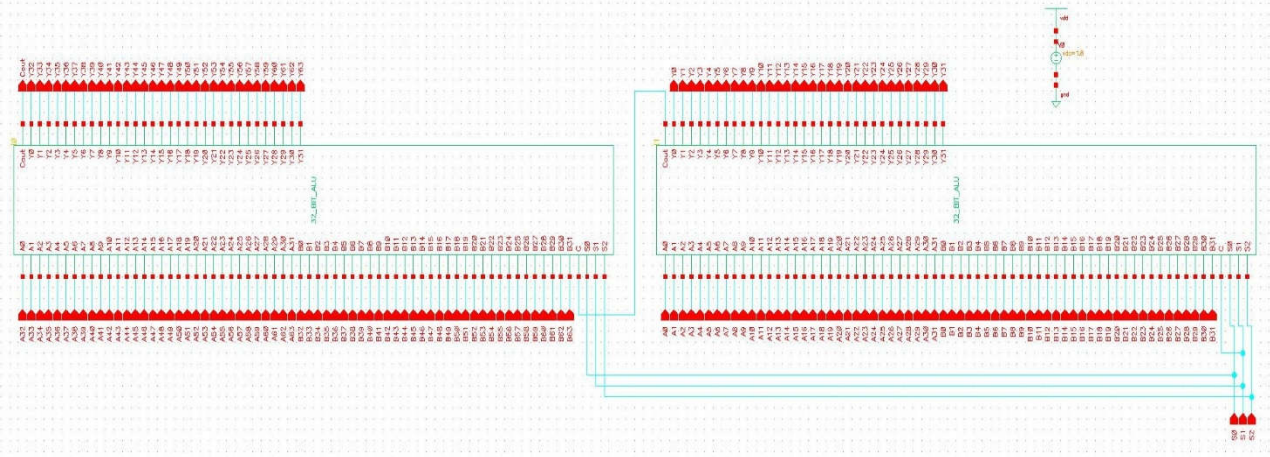


Fig 13. 64 Bit ALU

Two phases of 32-bit were employed to implement the 64-bit ALU, as seen in Fig. 4. While the selection line S1 is linked to the carry input of ALU0 to achieve logic 1, which is required for the outcomes of the logic operations to be unaffected by the other values, but subtraction and increment operations do not.

VII. SIMULATION AND RESULTS

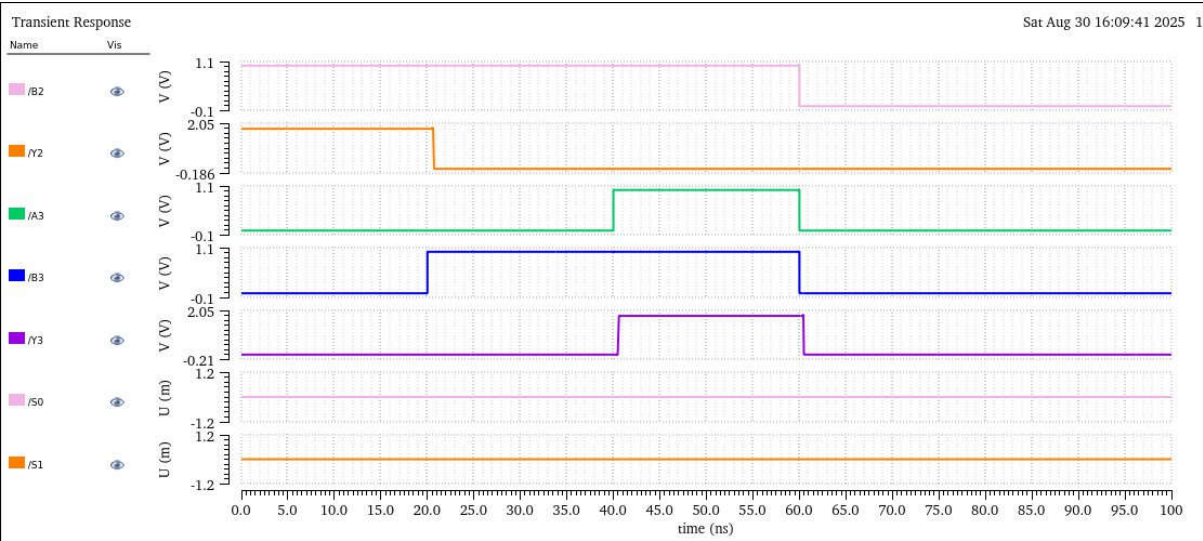


Fig. 14 output waveform of the proposed 4-bit ALU S0=0,S1=0, S2=1

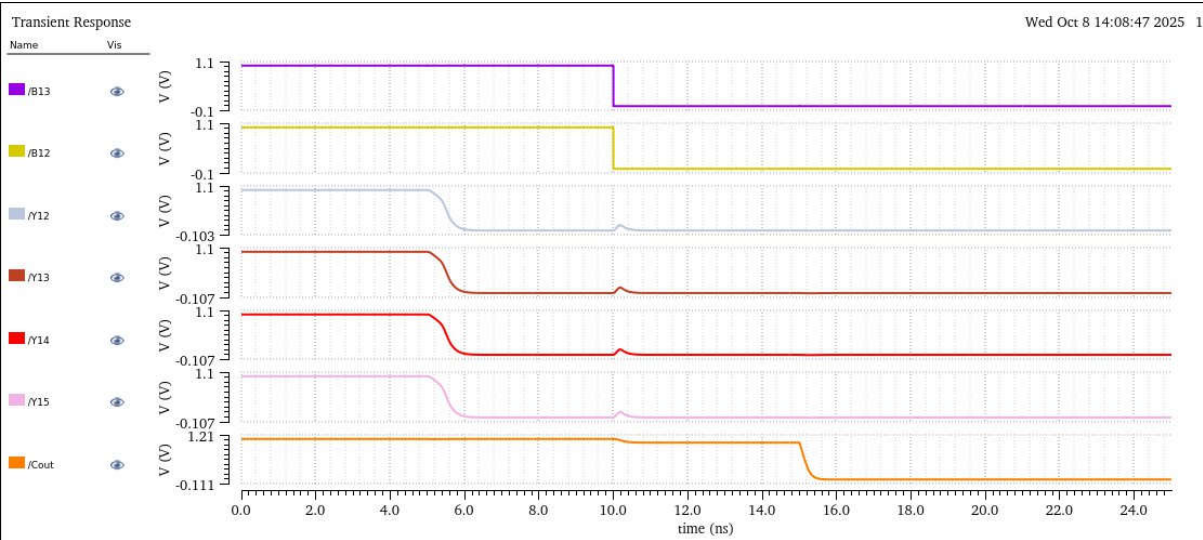


Fig. 15 output waveform of the proposed 16-bit ALU S0=0, S1=0,S2=1

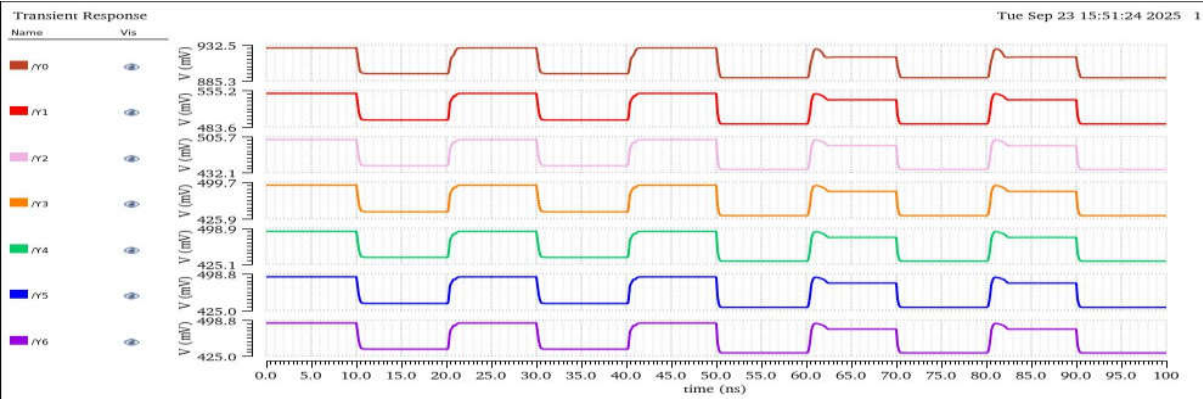


Fig. 16 output waveform of proposed 64-bit ALU S2=1 S1=1 S0=1.

Depending on the selected signals S0, S1, and S2, Operations will be performed between A and B. The simulation of the proposed ALU is performed in Cadence Virtuoso, and the inputs for A and B are specified as A = 111110000 and B = 1010101010, with a supply voltage of 1.8V.

Table III Average Power Consumption of ALU

Type of ALU	CMOS (in mWatts)	Proposed GDI (inµ watts)
4 Bit ALU	0.75[6]	98.95
16 Bit ALU	2.73[6]	969.6
64 Bit ALU	10.01[6]	4.985

VI CONCLUSION

The proposed work presents a 64-bit ALU designed in Cadence Virtuoso Tool using 180nm technology with the full-swing GDI technique. According to the simulation results, the proposed ALU exhibits a significant reduction in power consumption compared to the conventional CMOS-based design. In addition to low power, the GDI technique significantly reduces the transistor count, thereby minimizing silicon area and improving circuit compactness. The results also highlight that GDI-based circuits effectively overcome power-delay limitations of CMOS, making the architecture more efficient for large-scale integration. These findings establish that the proposed 64-bit ALU is a promising candidate for modern VLSI design, particularly in low-power, high-speed applications such as embedded processors, portable systems, and next-generation computing devices.

REFERENCES

- [1] M. Hasan, H. U. Zaman, M. Hossain, P. Biswas, and S. Islam, "Gate Diffusion Input technique based full swing and scalable 1-bit hybrid Full Adder for high performance applications," *Engineering Science and Technology, an International Journal*, vol. 23, pp. 1364–1373, 2020.
- [2] M. Hasan, H. U. Zaman, M. Hossain, P. Biswas, and S. Islam, "Gate Diffusion Input technique based full swing and scalable 1-bit hybrid Full Adder for high performance applications," *Engineering Science and Technology, an International Journal*, vol. 23, pp. 1364–1373, 2020.
- [3] A. Morgenshtein, A. Fish, and I. Wagner, "Gate-diffusion input (GDI): a power-efficient method for digital combinatorial circuits," *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, IEEE Trans. VLSI Syst., vol. 10, no. 5, pp. 566–581, 2002.
- [4] A. Morgenshtein, I. Shwartz, and A. Fish, "Gate Diffusion Input (GDI) logic in standard CMOS Nanoscale process," 2010 IEEE 26th Convention of Electrical and Electronics Engineers in Israel, 2010.
- [5] A. Morgenshtein, V. Yuzhaninov, A. Kovshilovsky, and A. Fish, "Fullswing gate diffusion input logic—Case-study of low-power CLA adder design," *Integration, the VLSI Journal*, vol. 47, no. 1, pp. 62–70, Jan. 2014
- [6] Rajesh pidugu, P. Mahaesh Kannan design of 64 bit low power alu for DSP applications. In international journal of advanced research in electrical, electronics and instrumentation Engineering vol 2, issue 4, April 2013.
- [7] M. Mukhedkar and B. P. Wagh, "A 180 nm Efficient Low Power and Optimized Area ALU design using Gate Diffusion Input technique," in *Proceedings of the 2017 International Conference on Data Management, Analytics and Innovation (ICDMAI)*, Pune, India, 2017.
- [8] M. A. Ahmed and M. A. Abdelghany, "Low Power 4-Bit Arithmetic Logic Unit Using Full-Swing GDI Technique," in *Proceedings of the 2018 International Conference on Innovative Trends in Computer Engineering (ITCE 2018)*, Aswan University, Egypt, 2018.
- [9] N. Jatav and A. Marwah, "Implementation of Arithmetic Logic Unit using GDI Technique," in *Proceedings of the 2016 Symposium on Colossal Data Analysis and Networking (CDAN)*, 2016.
- [10] M. Shoba and R. Nakkeeran, "GDI-based full adders for energy-efficient arithmetic applications,"

- Engineering Science and Technology, an International Journal, vol. 19, pp. 485–496, 2016. A. Sharma and R. Tiwari, "Low Power 8-bit ALU Design Using Full Adder and Multiplexer," in Proceedings of IEEE WiSPNET 2016 Conference, 2016.
- [11] G. K. Reddy, "Low Power-Area Pass Transistor Logic Based ALU Design Using Low Power Full Adder Design," in IEEE Sponsored 9th International Conference on Intelligent Systems and Control (ISCO) 2015, 2015.
- [12] V. Shekhawat, T. Sharma, and K. G. Sharma, "2-Bit Magnitude Comparator using GDI Technique," in IEEE International Conference on Recent Advances and Innovations in Engineering (ICRAIE-2014), Jaipur, India, May 2014.
- [13] V. Dubey and R. Sairam, "An Arithmetic and Logic Unit Optimized for Area and Power," in 2014 Fourth International Conference on Advanced Computing & Communication Technologies.