ABHIYANTRIKI
An International Journal of Engineering & Technology
(A Peer Reviewed & Indexed Journal)
Vol. 3, No. 5 (May, 2016) http://www.aijet.in/ eISSN: 2394-627X

Enhancement of Design Quality for an 8-bit ALU

Shreelakshmi*
M Tech Student
NCET, Bangalore, India

Sendamarai P
Assistant Professor
Deptt. of EC, NCET, Bangalore, India

Abstract

Paper presents a design of an 8-bit arithmetic logic unit (ALU) by using the hybrid of two techniques i.e. gate diffusion input (GDI) technique and substrate biasing technique (SBT). ALU is the most crucial and core component of central processing unit as well as of numbers of embedded system and microprocessors. In this work, ALU consists of 4x1 multiplexer, 2x1 multiplexer and full adder designed to implements logic operations such as AND, OR, XOR, XNOR; arithmetic operations such as INCREMENT, DECREMENT, ADD and SUBTRACT. GDI cells are used in the design of multiplexers and substrate biasing technique and GDI is used to design full adder AND, OR, XOR, XNOR gates which are then associated to realize ALU. The simulation is carried out using cadence virtuoso with gpdk 180nm technology and compared with previous designs realized with CMOS logic. The simulation shows that the design is more efficient with less power consumption, less surface area and is faster as compared to CMOS techniques.

Keywords*: GDI technique, ALU, Substrate biasing, Tool: cadence VIRTUOSO gpdk 180.*

****Author for correspondence*** shreelakshmi005@gmail.com

1. Introduction

In VLSI, the three major constraints are area power and time (delay). Here the ALU is designed with less complexity, low power and les delay. ALU performs arithmetic and logical operations, where these arithmetic operations are key and basic functions for all high speed and low power devices which uses the technology related to image processing, microprocessors etc. Addition is most important part of the arithmetic. In this paper it uses the idea of gate diffusion technique and substrate biasing. Sub blocks are as follows 2x1 mux, 4x1 mux and full adder. Full adder is the main block in ALU so this paper utilizes both technique to implement, and for mux it utilizes GDI technique.

2. Previous Works

Static power consumption is a challenge for the designer so to reduce this, author utilized the concept of 11T full adder using CMOS VLSI circuits [1]. Full adder is the basic building block of ALU so to reduce the power and area consumption. Paper proposed TG full adder using 6T XOR and carry logic was implemented by 2x1 mux [2]. Average power consumption is very

high in 28T full adder this paper used TG technology hence 18T transistors are used hence transistor count decreased [3]. PTL method 6 transistor full adder and PTL based multiplexer are used to reduce area by using less number of active devices in PTL power consumption is very less when compared with the CMOS logic [4]. Later 4-bit ALU is implemented using the concept of GDI and simulation is carried using mentor graphics 130nm technology and compared with pass transistor logic and CMOS logic realization [5]. Gate diffusion input technique is enhanced to provide some measure of enhanced hazards tolerance and more suitable for low voltage operation [6]. Then the design of 2T XOR gate using pass transistor logic was proposed it has least number of transistors and when compared with the earlier design it uses less silicon area consumption and power delay [7]. Optimization of devices with respect to speed and power is a significant issues in low voltage low power applications this which can be overcome by GDI technique [8]. In ALU to increase the switching speed in this paper, work has utilized two technique i.e. GDI and substrate biasing, the input and output part consists of 2x1 mux and 4x1 mux and ALU is implemented by using full adder.

3. Gate Diffusion Input Technique

Morgenstern has proposed basic GDI cell shown in Fig. 1 [8]. This is a new approach for designing low power digital combinational circuit. GDI technique is basically two transistor implementation of complex logic functions which provides in-cell swing restoration under certain operating condition. This approach leads to reduction in power consumption, propagation delay and area of digital circuits is obtained while having low complexity of logic design. An important feature of GDI cell is that the source of the PMOS in a GDI cell is not connected to VDD and the source of the NMOS is not connected to GND. Therefore, GDI cells gives two input pins for use which makes the GDI design more flexible than CMOS logic. Therefore, three inputs in a GDI cell-G (common gate input of NMOS and PMOS). P (input to the source /drain of PMOS) and N (input to the source and drain of NMOS). Bulks of both NMOS and PMOS are connected to N and P respectively. Following table shows different logic functions which is implemented by GDI logic, based on different input values so, various logic functions can be implemented with low power and high speed with GDI technique as compared with the CMOS logic.

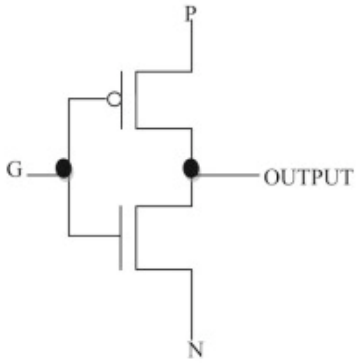

Fig. 1: Basic GDI Cell

Table 1: Logic Functions for Basic GDI Cell

SN	N input	P input	G input	Output	Function
1	0	1	A	A’	Inverter
2	A	0	B	AB	AND
3	1	A	B	A+B	OR
4	A’	A	B	A’B+AB’	XOR
5	A	A’	B	AB+A’B’	XNOR
6	0	B	A	A’B	F1
7	B	1	A	A’+B	F2
8	C	B	A	A’B+AC	MUX

4. Substrate Biasing Technique

Substrate biasing in PMOS biases the body of the transistor to a voltage higher than V_{dd}; in NMOS, to a voltage lower than V_{ss}. Since leakage currents are a function of device V_{th}, substrate biasing also known as back biasing can reduce leakage power. With this advanced technique, the substrate or the appropriate well is biased to raise the transistor thresholds, thereby reducing leakage. In PMOS, the body of transistor is biased to a voltage higher than V_{dd}. In NMOS, the body of transistor is biased to a voltage lower than V_{ss} as shown in figure 2.

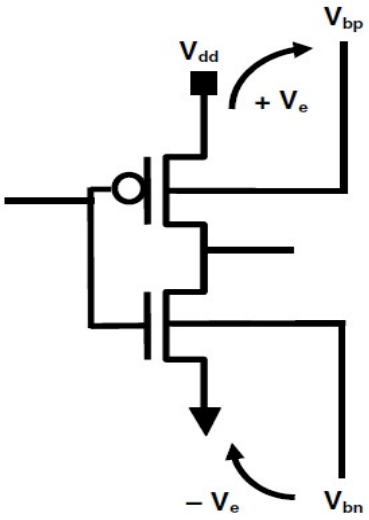

Fig. 2: Basic Substrate Biasing Cell

5. Sub Blocks

A. Multiplexer

Multiplexer is a digital switch. A multiplexer has numbers of input data lines and one output line. The selection of a particular input line is controlled by a set of selection line. There are '2n' input lines and 'n' selection lines whose bit combinations determine which input is selected. Fig. 3 shows implementation of basic 2x1 multiplexer using GDI cell. The 4x1 multiplexer has four inputs, two selection lines and one output. Depending on the two selection lines, one output is selected at a time among the four input lines. Fig. 4 shows implementation of 4x1 multiplexer using GDI cell.

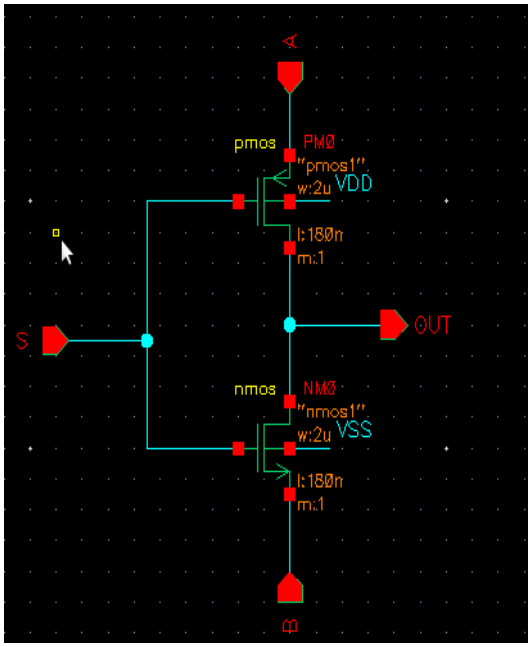

Fig. 3: 2X1 MUX Schematic

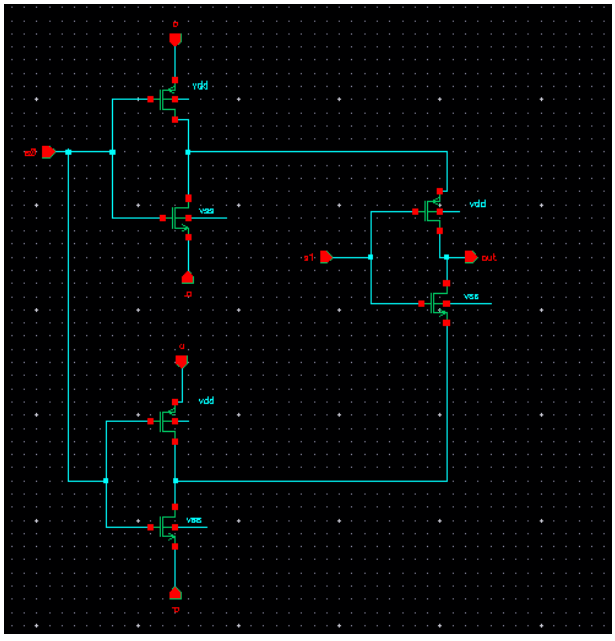

Fig. 4: 4x1 MUX Schematic

B. XOR Gate

The main building block of full adder circuit is XOR gate which gives sum output. So the overall performance of full adder circuit can be improved by optimizing XOR gate. Fig. 5 shows the implementation of XOR gate using GDI technique as well as substrate. It uses less number of

transistors as compared to conventional design of XOR gate using CMOS logic Units. The number of transistor in GDI is 4 and in CMOS are 12.

C. Full Adder

The Full Adder circuit adds three one-bit binary numbers (A, B & C) and outputs two one-bit binary numbers, a sum (S) and a carry (Cout). The full adder is usually a component in cascade of adders, which add 4, 8, 16 etc. binary numbers. Implementation of full adder circuit using GDI technique which is a basic building block of arithmetic and logic unit has been shown in figure 6.

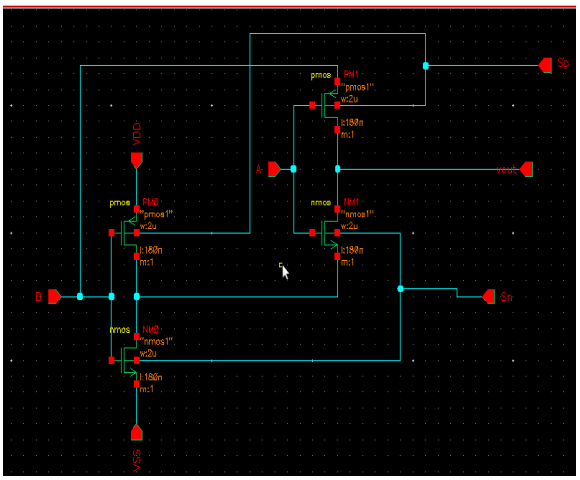

Fig. 5: Schematic of XOR Gate using BIAS and GDI

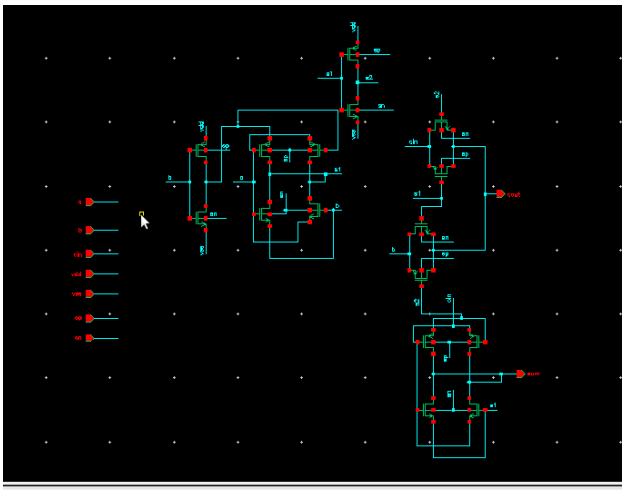

Fig. 6: Schematic of 1 Bit Full Adder

6. Design of Arithmetic and Logic Unit (ALU)

An arithmetic logic unit (ALU) is a fundamental building block of the Central Processing Unit (CPU) of a computer, and even the simplest microprocessors contain one. It is responsible for performing arithmetic and logic operations such as addition, subtraction, increment, decrement, logical AND, logical OR, logical XOR and logical XNOR. ALU consists of sixteen 4x1 multiplexers, eight 2x1 multiplexers and eight full adders. The 8-bit ALU is designed in 180nm. When logic '1' and logic '0' are applied as an input INCREMENT and DECREMENT operations takes place respectively. An INCREMENT operation is analyzed as adding '1' to the addend and DECREMENT is seen as a subtraction operation [6]. Two's complement method is used for SUBTRACTION in which complement of B is used. The outputs obtained from the full adder are SUM, EXOR, EXNOR, AND & OR. Figure 7 shows the block diagram of 8-bit ALU where first stage to fourth stage is cascaded with the CARRY bit. Symbolic representation of 8-bit ALU has been visualized in figure 7. Some of the operations performed by the ALU based on the status of the select signal are shown in table 2.

Selection Lines			Operations
S2	S1	S0	
0	0	0	INCREMENT
0	0	1	DECREMENT
0	1	0	ADDITION
0	1	1	SUBTRACTION
1	0	0	AND
1	0	1	OR
1	1	0	EXOR
1	1	1	EXNOR

Table 2: Operations of ALU

The schematic of ALU is designed using cadence virtuoso gpdk 180nm. It shows connectivity between the components and describes aspect ratios of the transistor that can be modified along

with the design. Figure 7B represents the complete schematic view of ALU. The 8-bit ALU consists of two 8-bit inputs, three selecting lines, and one carry input, one carry output and four output bits.

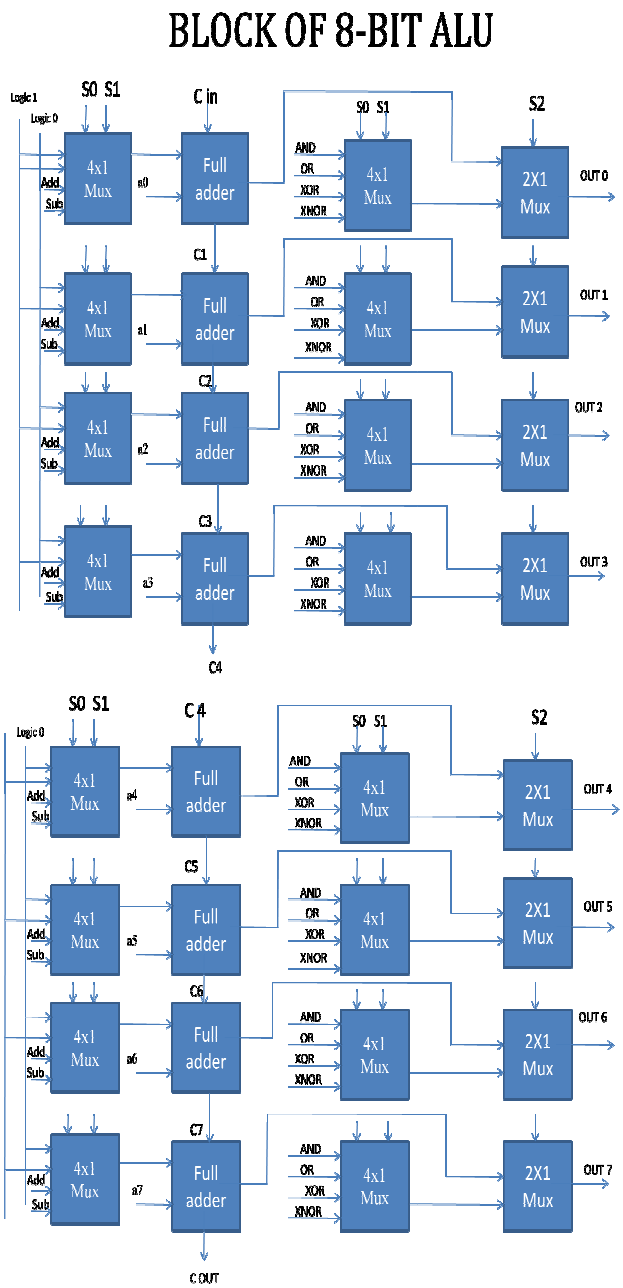

Fig.7A: 8-Bit ALU

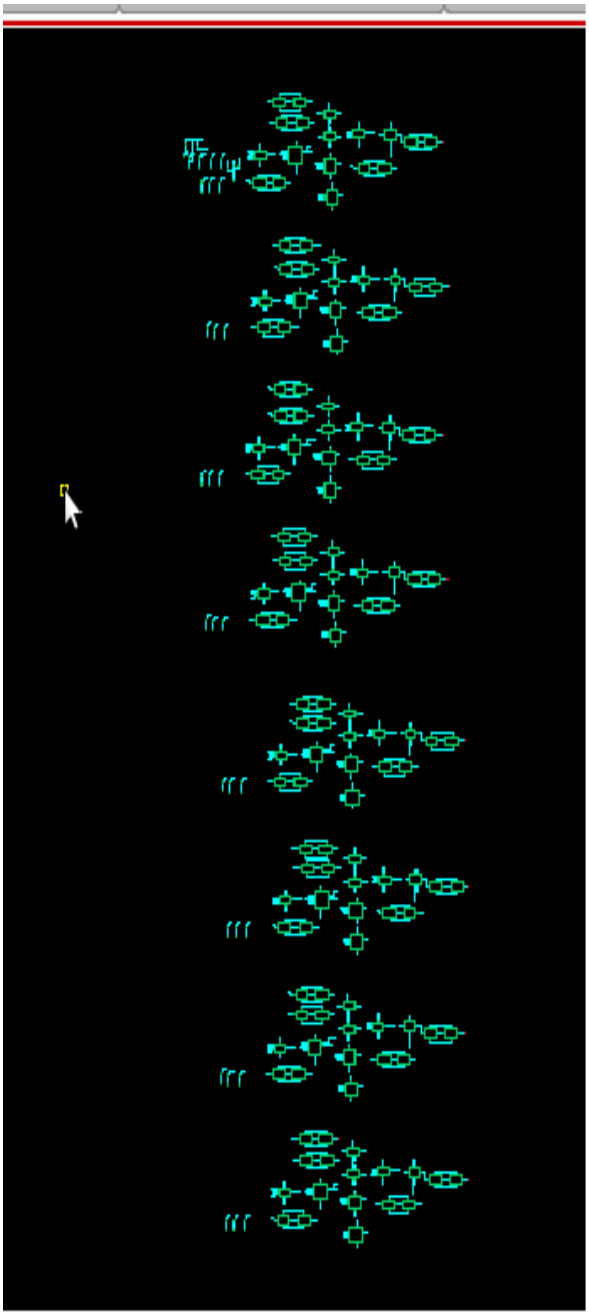

Fig. 7B: Schematic of 8-Bit ALU

The multiplexer stage selects the appropriate inputs based on the condition of the select signals, and gives it to the full adder which then computes the results. The multiplexer at the output stage selects the appropriate output and route it to output port. Table 2 shows the truth table for the operations performed by the ALU based on the status of the select signal. The operation being performed and the inputs and outputs being selected are determined by set of three select signals incorporated in the design. The multiplexer stage selects the appropriate inputs based on the

condition of the select signals, and gives it to the full adder which then computes the results. The multiplexer at the output stage selects the appropriate output and route it to output port.

The schematic of ALU is designed using cadence virtuoso gpdk 180nm. It shows connectivity between the components and describes aspect ratios of the transistor that can be modified along with the design. Figure 7 represents the complete schematic view of ALU. The 8-bit ALU consists of two 8-bit inputs, three select lines, and one carry input, one carry output and four output bits. Paper presents a new approach using concept of GDI technique and substrate to design an arithmetic and logic unit. In an ALU, for appropriate selection of input to perform particular operation and for obtaining output accordingly multiplexer is the most applicable device. In earlier designs of ALU, the multiplexer unit is either implemented by conventional CMOS logic which proven to have high power consumption. The approach gives better result than previous designs in terms of power consumption, propagation delay as well as area.

Table 3: Power Analysis

	2x1 mux	4x1 mux	AND	OR	XOR	XNOR
CMOS	430.1 e^-6	15.84 e^-6	2.6e^-6	2.45e^-6	8.8e^-6	10.1e^-6
GDI	-22.03 e^-6	15.12e^-9	782.2e^-9	37.7e^-3	75.55e^-9	871.5e^-9
BIAS	328.2 e^-6	3.134e^-6	767.5e^-9	13.6e^-6	38.62e^-6	752.4e^-9

Table 4: Number of Transistors (Area)

	2X1 MUX	4X1 MUX	AND	OR	XOR	XNOR
CMOS	12	30	6	6	12	14
GDI	2	6	2	2	4	6
BIAS	2	6	2	2	4	6

7. Final Simulated Output

Simulation is carried out using cadence virtuoso gpdk 180 nm technology and the final simulated results of 2x1 mux, 4x1 mux, XOR, full adder and 8-bit ALU is shown below.

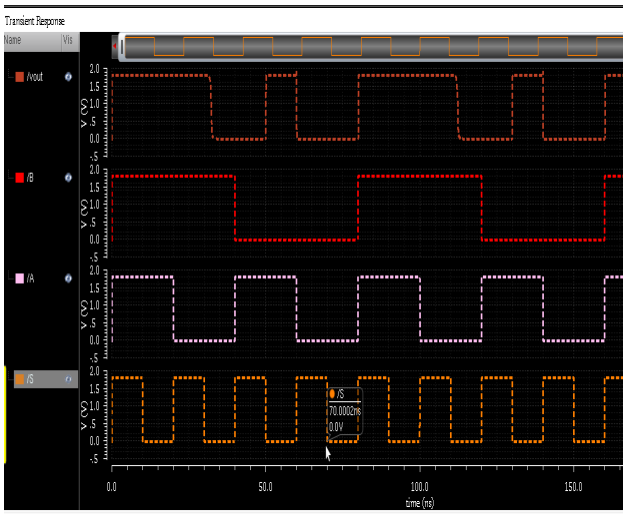

Fig. 8: Simulated result of 2x1 mux

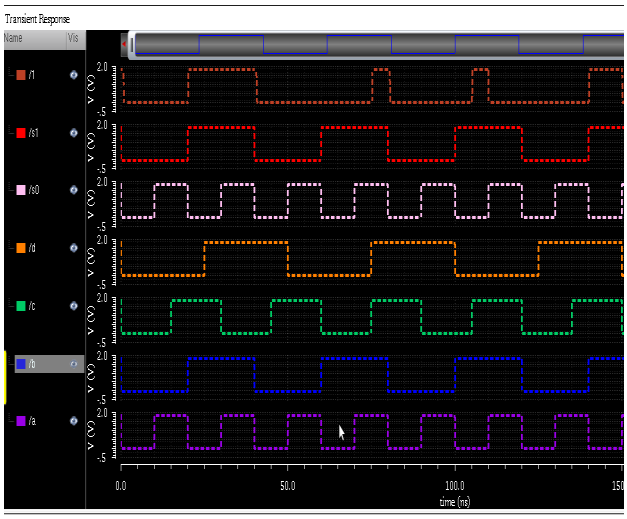

Fig. 9: Simulated result of 4x1 mux

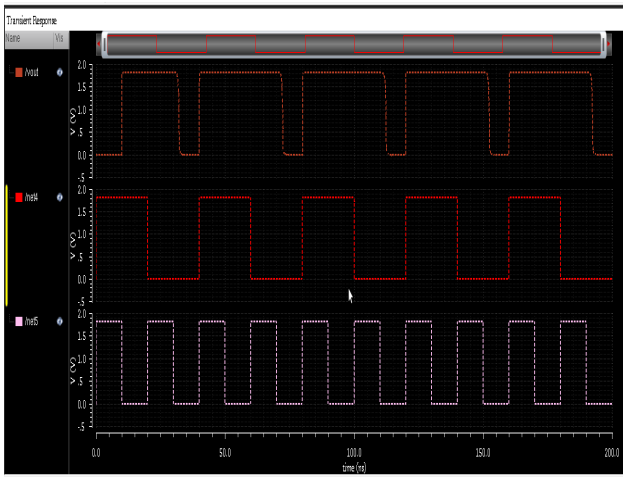

Fig. 10: Simulated Result of XOR

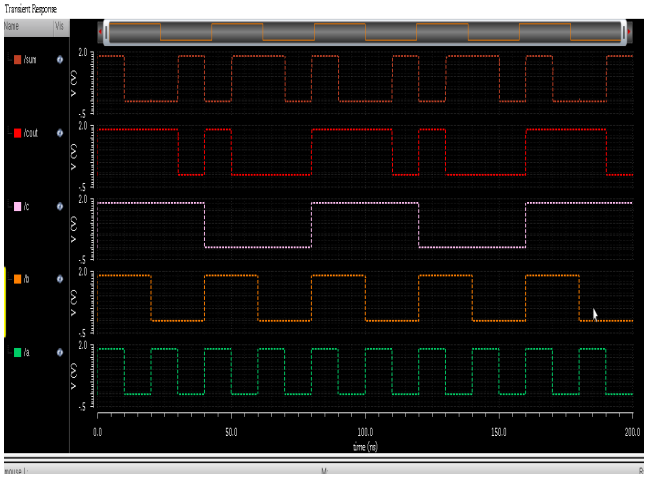

Fig. 11: Simulation Result of 1 Bit Full Adder

8. Conclusion

In the current project 8-bit ALU is successfully implemented using GDI and substrate biasing technique and power and area is compared with CMOS logic to show GDI and substrate biasing technique is more efficient than CMOS.

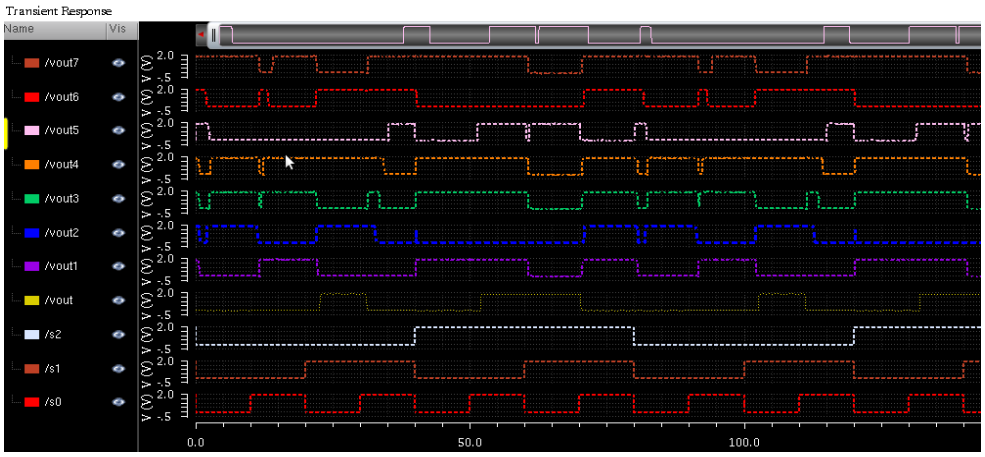

Fig. 12: Simulated result of 8-bit ALU

References

[1] R Shalem, E John, & LK John. A novel low-rao, area optimized low power energy recovery full adder cell. Proc. Great Lakes Symp. VLSI, Feb. 1999, pp.380–383.

[2] A Sharma, R Singh, & R Mehra. Low power TG full adder design using CMOS nanotechnology.

[3] S Panda, N Mohan Kumar, & CK Sarkar. Transistor count optimization of conventional CMOS full adder & optimization of power and delay of new implementation of 18 Transistor full adder by dual threshold node design with submicron channel length. 4th Inter. Conf. on Computers and Devices for Communication, 2009. pp. 1-4.

[4] T Esther Rani, M Asha Rani, & Rameshwar. Power Arithmetic and Logic Unit. IEEE, 2011.

[5] M Kumar, MA Hussain, & LLK Singh. Design of a low power high speed ALU in 45nm using GDI technique and its performance comparison. CNC 2011, CCIS 142, pp. 458–463.

[6] Morgenshtein, A, Fish, A, & Wagner, IA. Gate-Diffusion Input (GDI) – a technique for low power design of digital circuits: analysis and characterization. ISCAS 2002, USA, 2002.

[7] K-H Cheng, & C-S Huang. The novel efficient design of XORlXNOR functions for adder applications. Proc. 6th IEEE Inter. Conf. on Electronic Circuits and System, Vo1. 1, pp.29-32, 1999.

[8] R Uma, & P Dhavachelvan. Modified Gate Diffusion Input Technique: a new technique for enhancing performance in full adder circuits. 2nd Inter. Conf. on Communication, Computing & Security, 2012.
[9] R Uma, V Vijayan, M Mohanapriya, & S Paul. area, delay and power comparison of adder topologies.