Design of 4-bit ALU using 45nm Technology in Cadence Virtuoso

***Abstract*—This paper presents the design, implementation and verification of a 4-bit Arithmetic Logic Unit (ALU) using Cadence Virtuoso and Spectre simulation tools using 45nm Technology with PMOS width adjusted to 2.5 times the NMOS width. The ALU designed integrates individual operational circuits—Adder, Subtractor, Multiplier, Divider and various logical gates—into a single unit using a 16:1 multiplexer (MUX). The selection of operations is determined by the MUX select lines. Each operational sub-unit was designed to accept two 4-bit binary inputs and generate accurate outputs. To facilitate seamless data handling and verification, an 8-bit parallel-to-serial converter was employed between the outputs of operational blocks and the MUX. The design and simulation process includes the creation of schematic diagrams in Cadence Virtuoso and verification using input pulses. Simulations were conducted in Cadence Spectre, and results were verified against truth tables for all possible input combinations. The integration of sub-units and the overall performance of the ALU were validated using specific test cases, demonstrating the successful implementation of the multi-operational ALU.**

***Index Terms*—Arithmetic Logic Unit, Cadence Virtuoso, Schematic Circuits, Spectre simulation, CMOS**

I. Introduction

In this paper, we have meticulously developed and integrated a 4-bit Arithmetic Logic Unit (ALU) by employing a sophisticated design flow that combines individual operational circuits into a cohesive unit. This integration process leverages the versatility of a 16:1 multiplexer (MUX) to selectively determine the operation performed by the ALU, based on the configuration of its select lines.

The design and implementation of the ALU were carried out within the Cadence Virtuoso environment [6], a robust platform renowned for its precision in analog and digital circuit design. The integration of the operational circuits into a single 4-bit ALU was achieved through a structured approach that ensured seamless functionality and optimal performance. The use of the 16:1 MUX was pivotal in facilitating the selection of different arithmetic and logic operations [4], which are fundamental to the ALU's capabilities.

To validate the functionality of the integrated ALU, the design was translated into a symbolic representation. This was followed by extensive simulation and verification using Cadence Spectre, a powerful tool for circuit simulation and analysis. The testbench setup involved applying various input pulses to the ALU and scrutinizing the output results to confirm the accuracy and reliability of the design.

The block diagram of our circuit design flow encapsulates the entire process, from the initial circuit integration to the final verification phase. This detailed workflow not only underscores the technical proficiency employed in the design but also highlights the systematic approach taken to ensure the ALU's correct operation. This endeavor reflects our commitment to precision in digital circuit design, paving the way for future advancements in integrated circuit technology.

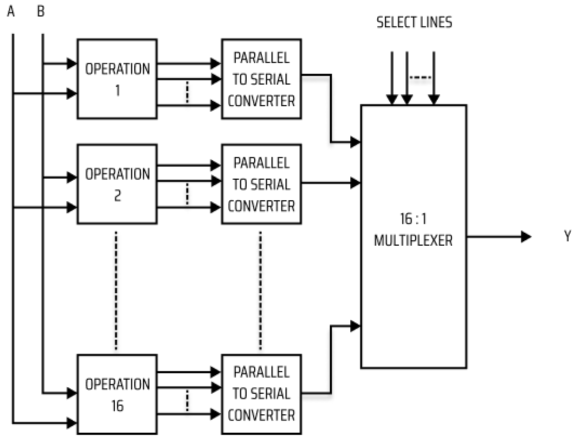
A
B
OPERATION 1
OPERATION 2
OPERATION 16
PARALLEL TO SERIAL CONVERTER
PARALLEL TO SERIAL CONVERTER
PARALLEL TO SERIAL CONVERTER
SELECT LINES
16 : 1 MULTIPLEXER
Y

Fig. 1. Block diagram of 4-bit ALU

II. Literature Survey

In [1], a Serial Divider circuit was implemented by the author using a mux, 4-bit ripple carry adder, D-flip flop, and synchronous up counter. In [2], the author designed and implemented 4-bit ALU performing various arithmetic and logical operations using simulation tools with the Verilog HDL and analyzed the FPGA-based ALU. In [3], the author analyzed Full Adder Circuit using simulation with input pulses for truth table verification. In [4] array multipliers using full adder and gates for multiplying 4-bit inputs,is implemented by the author. The authors in [5] have given MUX implementation. In [6], a full 8-bit ALU, including DRC and pre-layout simulation in Cadence, was given. In [7], 16-bit power-optimized ALU with power and input gating in Cadence was implemented. Reference [8] is a Survey Paper that highlights an ALU with low power dissipation. In [9], 8-bit serial in parallel out (SIPO) shift register using True Single-Phase Clock is implemented. D-FF implementation for making parallel to serial converters (PISO) is given in [10].

III. Methodology

We have followed a top-down approach for deciding the block diagram of the ALU and bottom-up approach for designing the circuits. The ALU accepts two 4 bit values as input and returns an 8 bit output.

A. Arithmetic Operations

- 4-bit Full Adder: Takes two 4-bit binary numbers as input along with a 1-bit input Carry in (Cin) and produces a 4-bit sum along with a 1-bit output Carry out (Cout).
- 4-bit Subtractor: Takes two 4-bit binary numbers as input along with a 1-bit input Borrow in (Bin) and produces a 4-bit difference along with a 1-bit output Borrow out (Bout).
- 4-bit Multiplier: Takes two 4-bit binary numbers as input and produces an 8-bit product.
- 4-bit Divider: Takes two 4-bit binary numbers as input and produces two separate 4-bit outputs: a Quotient and a Remainder.

B. Logical Operations

There are 7 basic gates, which can be used to implement any Boolean expression. These act as building blocks for all circuits used in this Arithmetic Logic unit.
The NAND gate was implemented with CMOS transistors: PMOS and NMOS. The AND, OR, NOT, NOR, EX-OR, EX-NOR were then implemented using NAND gates.

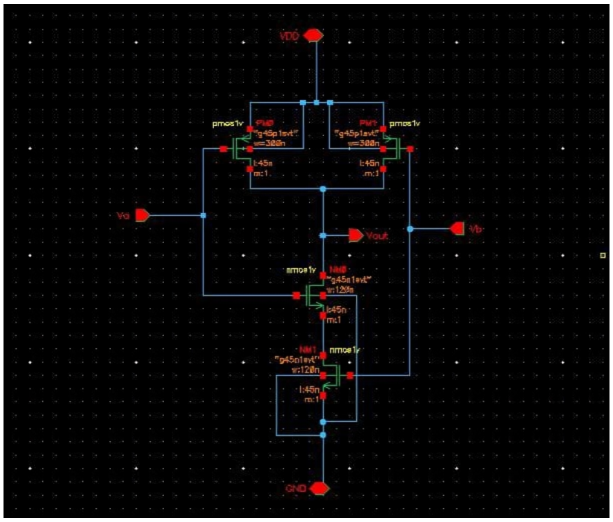

Fig. 2. CMOS Implementation of NAND

C. Parallel-to-Serial Converter (PISO)

The parallel output of each sub-unit in the ALU is converted into a string of serial 8-bit Binary using a PISO convertor. This helped us verify the simulation results for each operation performed.

D. Integration using 16:1 Mux

A 16:1 Multiplexer is designed using 2:1 MUX and all the sub-units given as input to a PISO. Then, the MUX selects the input according to the select lines. Based on the input from the user, select lines activate the operation on the particular input line of the mux. An 8-bit PISO is used between output of each sub-unit and input of MUX to facilitate efficient data handling. The converter optimizes data flow and enhances the overall system efficiency.

IV. Design

All the schematics have been created in Cadence Virtuoso using CMOS transistors.

A. Arithmetic Operations

The operations mentioned in Methodology are performed and the respective Schematic circuit is shown below to perform these operations. The building blocks of a 4-bit Full Adder Fig. 3 are four 1- bit Full Adders cascaded together. For 4-bit Full Subtractor Fig. 4 four 1-bit Full Subtractors were cascaded. A 4-bit Array Multiplier Fig. 5 comprises 4-bit full adders and AND gates. The 4-bit Serial Divider Fig. 6 is made of an SB block, which is a 1-bit Full Subtractor and 2:1 Mux cascaded together, AND, NOT and OR gates.

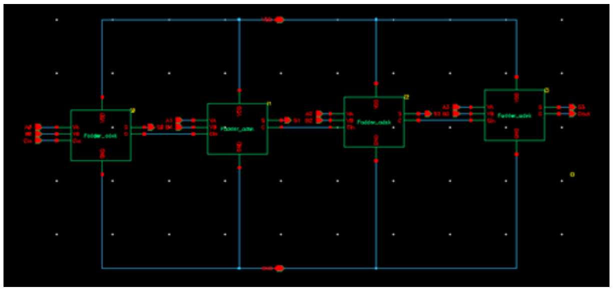

Fig. 3. Schematic of 4-bit Adder

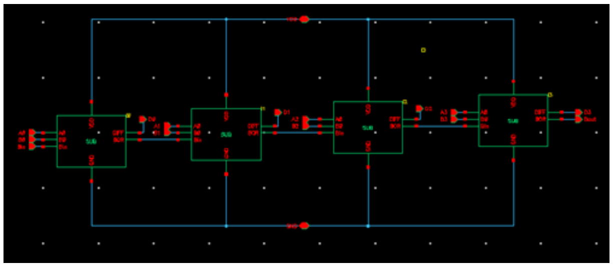

Fig. 4. Schematic of 4-bit Subtractor

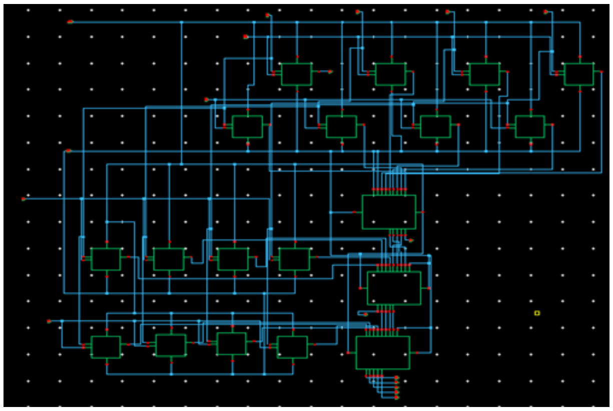

Fig. 5. Schematic of 4-bit Multiplier

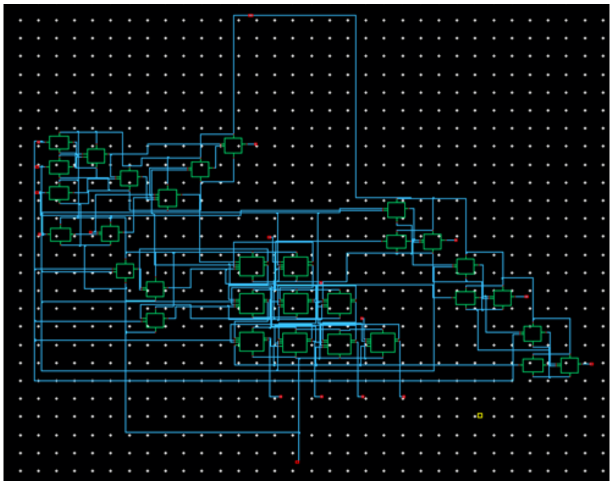

Fig. 6. Schematic of 4-bit Divider

B. Logical Operations

The logical operations mentioned above are performed and the Schematic of one of them is given below. The 4-input NAND gate shown in Fig. 7 is designed by cascading four 2-input NAND gates in parallel. All the other logic gates have been implemented similarly.

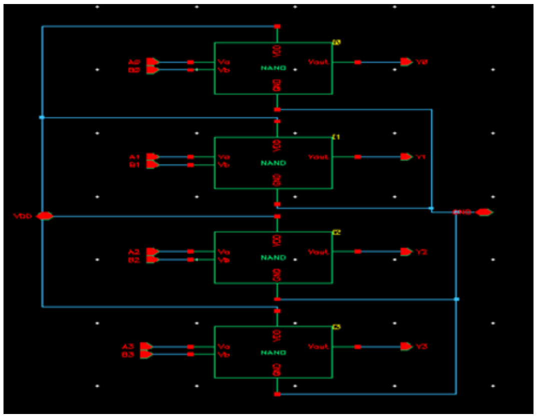

Fig. 7. Schematic of 4-bit NAND

C. Total Integration

The output of each of these operations is given to a PISO convertor and then to the MUX. The following test case is considered: A = 1011 and B = 0101. All the sub-units are integrated together by connecting each of them to a PISO converter as shown in Fig. 8. Outputs of these sub-units are further connected to a 16:1 MUX. Fig. 9 is the testbench in which the select lines, controlled by input pulses, determine the operation to be performed over 8 clock cycles.

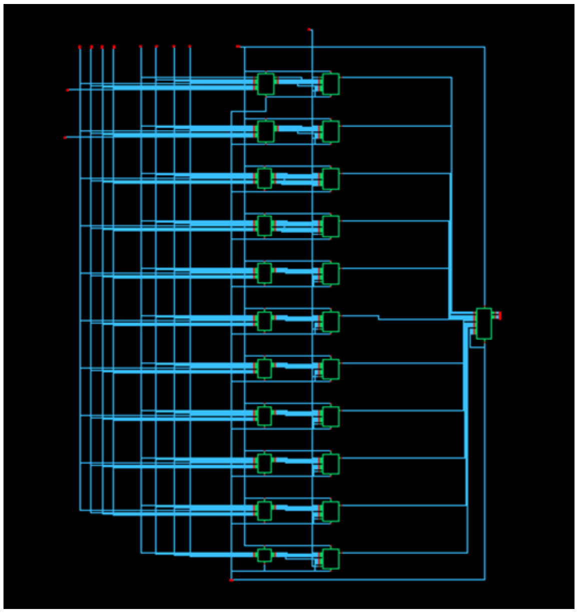

Fig. 8. Total Integration of 4-bit ALU

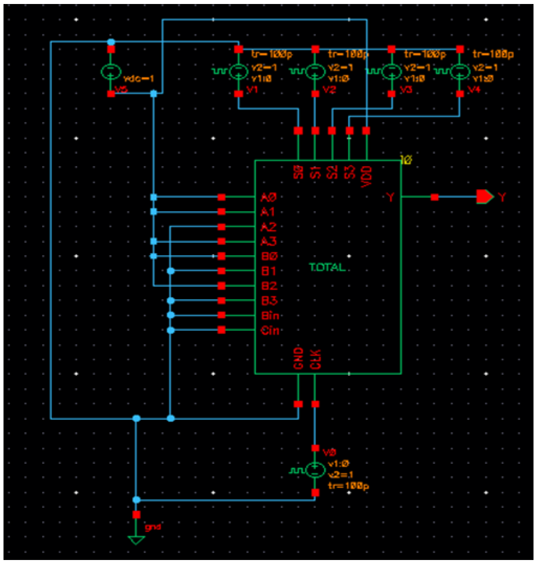

Fig. 9. 4-bit ALU Symbol

TABLE I
ORDER OF OPERATIONS IN THE ALU I.E OPERATIONAL CODES

Input Select Line (S3 S2 S1 S0)	**Operation activated on Mux**	**Sample Results**
0000	Adder	0001 0000
0001	Subtractor	0000 0110
0010	Multiplier	0011 0111
0011	Divider	R0001 Q0010
0100	AND	0000 0001
0101	OR	0000 1111
0110	NAND	0000 1110
0111	NOR	0000 0000
1000	EX-NOR	0000 0001
1001	EX-OR	0000 1110
1010	NOT	0000 0100

V. Simulation and Results

The simulation results for a test case using a= 1011 and b=0101 are shown in Fig. 10 to Fig. 14. The first waveform is the common clock for the entire circuit.The second waveform is SHIFT/LOAD of the PISO. The third waveform is the final output. In the Output Figures the last four waveforms represent the Input pulses of the Select Lines of the 16:1 MUX, with the last waveform being MSB.

The select lines determine the Opcode associated with every operation performed by the ALU as listed in Table I. The output of every sub-unit is 8- bit wide, therefore the select lines change every 8 clock pulses to accommodate all the significant output bits in the waveform.

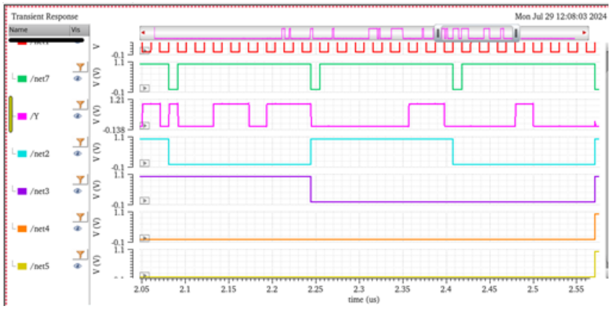

Fig. 10. Opcode: 0000 to 0011

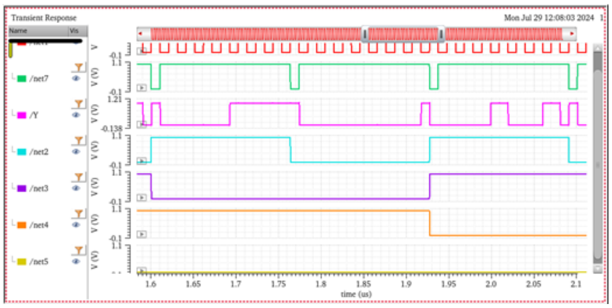

Fig. 11. Opcode: 0011 to 0110

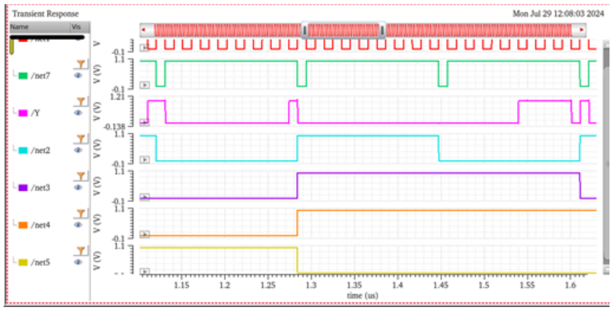

Fig. 12. Opcode: 0110 to 1000

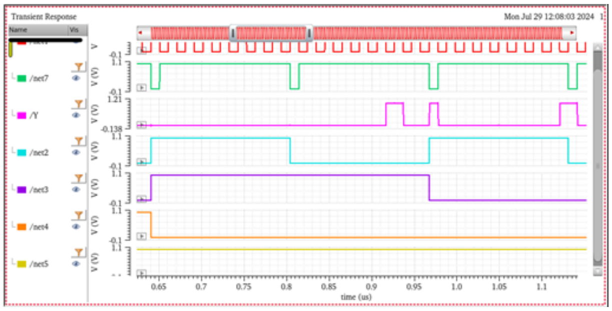

Fig. 13. Opcode: 1001 to 1011

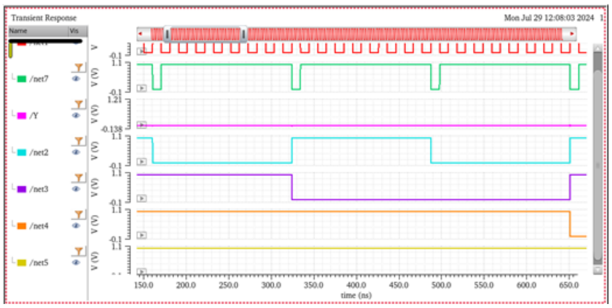

Fig. 14. Opcode: 1100 to 1111

VI. Conclusions

The simulation and verification of the ALU circuits using Cadence Spectre, with input voltage pulses in a test-bench setup, confirm the proper functioning of each sub-unit. By thoroughly testing all possible 4-bit input combinations against the truth tables, it has been ensured that the ALU performs its arithmetic and logic functions accurately. This rigorous approach validates that each sub-unit and the overall ALU are functioning as intended, providing reliable performance in practical applications. The successful completion of these simulations demonstrate the robustness and accuracy of the designed ALU.

VII. Future Scope

The future scope lies in the integration of code converters for operations like binary to hexadecimal and decimal number systems. Progressing to the layout process will enhance the design. Incorporating memory and time management blocks will further optimize efficiency, improving both performance and resource utilization.

Acknowledgement

This work was supported by the Chips to Startup (C2S) grant from the ministry of Electronics and Information Technology (MeitY). We thank MeitY for their financial support and encouragement,which was instrumental in our project's success and allowed us to explore and implement an ALU.

References

[1] Y. Hiremath, A. L. Kulkarni, and J. S. Baligar, "Design and implementation of serial divider using 180nm process technology," *Int. J. Eng. Sci. Res. Technol.*, vol. 4, no. 2, pp. 1–5, Feb. 20XX.

[2] C. Patel, A. Saxena, A. Rawat, and O. Nautiyal, "Implementation of LVCMOS based 4 bit FPGA based ALU on SP 701 board for new digital age technologies," *Int. J. Recent Technol. Eng.*, vol. 11, no. 6, pp. 1–8, Mar. 2023.

[3] M. K. Manjunath, A. L. Haroon, A. Pagi, and J. Ulaganathan, "Analysis of various full-adder circuits in Cadence," presented at the *Int. Conf. Emerging Res. Electron. Comput. Sci. Technol. (ICERECT)*, Mysuru, India, Dec. 2015.

[4] B. V. Krishna and K. Dhanunjaya, "Analysis and modeling of low power array multipliers using Cadence Virtuoso simulator in 45 nm technology," *Int. J. Electron. Commun. Instrum. Eng. Res. Dev.*, vol. 3, no. 4, pp. 49–64, Oct. 2013.

[5] K. I. Shohail, W. Awsaf, S. U. Sayel, M. K. Nitu, and M. H. Bhuyan, "Design steps, simulation, and analysis of a 1-bit ALU in Cadence at 90 nm CMOS node," *IOSR J. VLSI Signal Process.*, vol. 13, no. 6, pp. 1–14, Nov.-Dec. 2023.

[6] N. Zhang, T. M. Mamo, and W. H. Mahmoud, "VLSI design, verification and fabrication of an arithmetic logic unit (ALU) using the Cadence Virtuoso: A case study," *Proc. ASEE*, Paper ID 36234, 2022.

[7] K. Bikshalu and P. Soma, "Design and simulation of 16 bit arithmetic unit using gating techniques in Cadence 45nm technology," *Int. J. Eng. Adv. Technol.*, vol. 6, no. 3, pp. 1–8, Feb. 2017.

[8] S. Sumanth and Y. G. S, "A survey on arithmetic logic unit using Cadence," *Stud. Indian Place Names - J.*, vol. 40, no. 3, pp. 1–5, Feb. 2020.

[9] S. Kakde, S. R. Thakare, S. Kamble, U. Mandawkar, and S. Mali, "Design and implementation of efficient 8-bit SIPO shift register," *J. Univ. Shanghai Sci. Technol.*, Dec. 2020.

[10] A. Saxena, M. Kaur, H. Pahuja, V. Chhabra, and A. Kumar Saxena, *Design and Performance Analysis of CMOS Based D Flip-Flop Using Low Power Techniques*. June 8, 2023.