Design of efficient 4x4 Enhanced Pipeline multiplier Based with Various Optimization Techniques Department of Electrical and Computer Engineering, Birzeit University

Mahmoud Hamdan[1], *Osaid Hamza*[2], *Yazeed Hamdan*[3]
1201134@student.birzeit.edu[1],1200875@student.birzeit.edu[2],1201133@student.birzeit.edu[3]

Abstract— This paper discusses the creation of a 4x4 Enhanced Pipeline Multiplier optimized through various straightforward techniques to improve its efficiency and speed. Our design built using electric tool, and it focuses on refining the standard multiplication process, commonly used in computing, to meet the demands of modern digital applications. The proposed design aims to address the challenges associated with conventional multipliers and provide a solution that demonstrates enhanced throughput and minimized processing time. The effectiveness of the design is evaluated through the output's simulations. This work helps to advancing the field of digital circuit design by introducing a simple approach to pipeline multiplication process with good efficiency.

I. Introduction

Multipliers is one of the basic building blocks in all digital processors, they are fundamental blocks of arithmetic units in the CPU, nowadays, digital signal processing systems are essential to improve the quality of digital signals and one of the major components used in DSP systems are multipliers.[1]

multipliers contributed for the multiplication process. Since it contains a large amount of computation, therefore high speed and greater efficiency are highly required. There are several approaches such as decrease the delay time, lessen the number of partial products, reduce the processing time of accumulation of the partial products, decrease the number of stages to enhance the overall speed of the computation, which directly improves the efficiency of the device along with to advance the multiplication performance, it is also essential to take care off some important factors such as compactness, consumption of power speed, area, regularity of layout. [2]

Multiplication operations, being fundamental to many digital applications such as signal processing, image processing, and various computing tasks, so its need to be as fast and efficient as possible. The focus of this paper is to present a 4x4 Enhanced Pipeline Multiplier that has been optimized to improve both efficiency and speed, catering to the modern demands of digital applications. Through the use of an innovative electric tool, we have refined the standard multiplication process to foster higher throughput and reduced processing time. By taking a straightforward approach to pipeline multiplication, we aspire to address the associated challenges prevalent in conventional multipliers and carve a path towards more progressive digital circuit designs.

The figure below depicts a general design for 4x4 multiplier, this design needs to include a flip flop between the stages to consider as pipeline system, the pipelined architecture is most commonly used for high-speed designs, pipelining is a design technique for increasing the throughput of a digital circuit or system without introducing pipelining registers between adjacent combinational logic blocks in the circuit/system. However, this requires balancing of the delays along all the paths from the input to the output which comes the way of its implementation.[3]

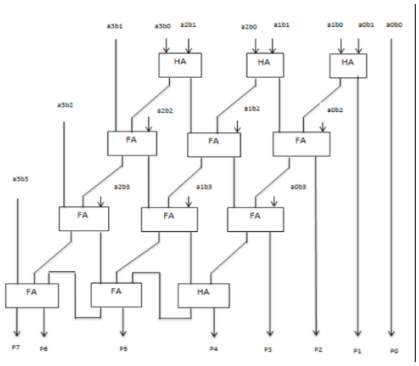

Figure 1: general design of 4x4 multiplier [4]

The using of D flip-flops (data or delay flip-flops) in pipeline systems is a common technique to manage data flow and

increase the throughput of a digital system. In a pipeline, data moves through several stages of processing, with each stage doing a part of the work. The use of D flip-flops helps to synchronize the data movement between these stages.[5]

II. DESIGN AND IMPLEMENTATION

In this project, the design was implemented using a 300nm CMOS library to ensure accurate timing and mitigate signal skew. Our primary focus was on achieving balanced rise and fall times for both NMOS and PMOS devices. This required careful consideration of factors such as the intrinsic delay and output resistance of each transistor. Through extensive simulation and testing, our objective was to optimize the performance and functionality of the multiplier design while adhering to the constraints imposed by the 300nm CMOS process technology.

We determined that a ratio of width approximately 2:1 between PMOS and NMOS devices, this ration gives an optimal result in terms of achieving equal rise and fall times within the 300nm process.

The components used in our design are as the following:

A. Invertor

The inverter is universally accepted as the most basic logic gate doing a Boolean operation on a single input variable. Fig.1 depicts the schematic representation of a CMOS inverter. As shown, the simple structure consists of a combination of an PMOS transistor at the top and a NMOS transistor at the bottom.[6]

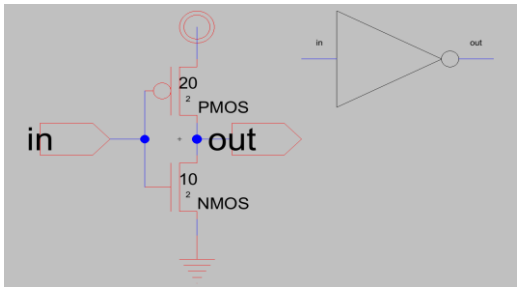

Figure 2: Schematics of the Invertor

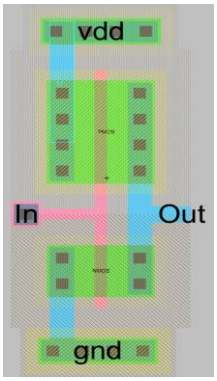

Figure 3: Layout of the Invertor

B. 2-input NAND

The NAND gate design done by using two PMOS with width value equals 20 and two NMOS transistors in series each one takes width equals 20. NAND implementation will be used when build the full adder and half adder blocks. The following two figures shows both schematics and layout representations for the NAND gate.

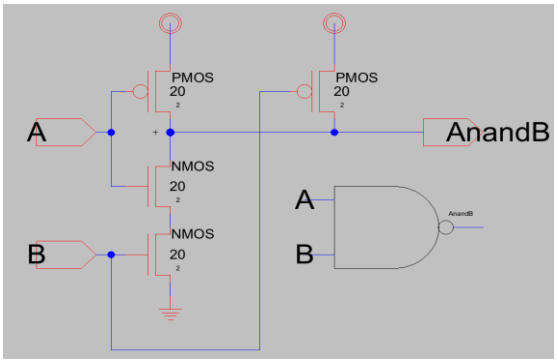

Figure 4: Schematics of NAND gate

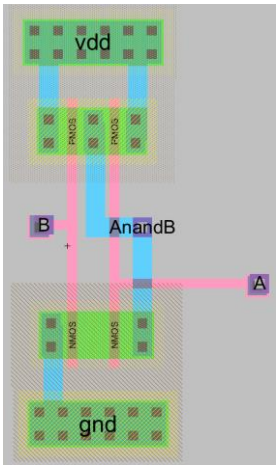

Figure 5: Layout of NAND gate

Table I summarizes the operation of 2-input NAND gate

Input		Output
A	B	Y
0	0	1
0	1	1
1	0	1
1	1	0

Table I: NAND truth table

C. 2-input AND

The AND gate is more complex than NAND gate because it needs 6 transistors to build, it is configured from NAND gate and invertor as shown at Figure 6 which represents the schematics building of the gate, AND will be used at the final multiplication system, between the inputs before enter the stages which include the full and half adder blocks.
Sizing of the PMOS and NMOS transistors shown at the figure also, The following figures shows the schematics and layout design of the gate.

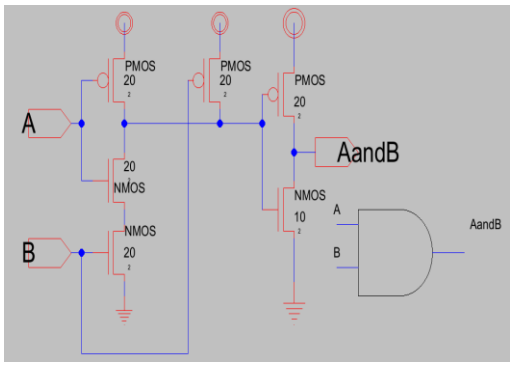

Figure 6: Schematics of AND gate

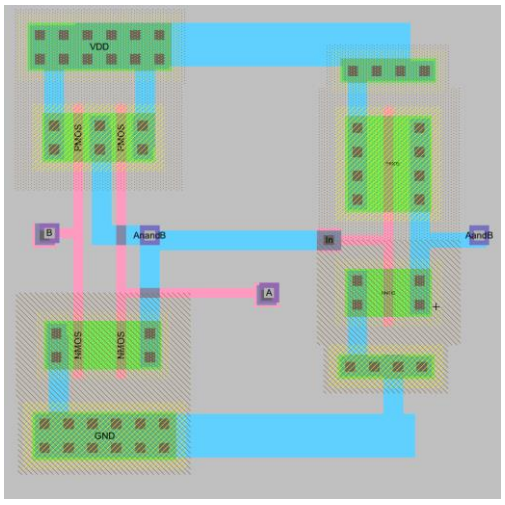

Figure 7: Layout of AND gate

Table II summarizes the operation of 2-input AND gate

Input		Output
A	B	Y
0	0	0
0	1	0
1	0	0
1	1	1

Table II: AND truth table

D. 2-input XOR

XOR gate design done by using six PMOS transistors and six NMOS transistors. XOR implementation will be used when build the full adder and half adder blocks, these blocks will be considering as main components in our multiplication system. The following two figures shows both schematics and layout representations for the XOR gate.

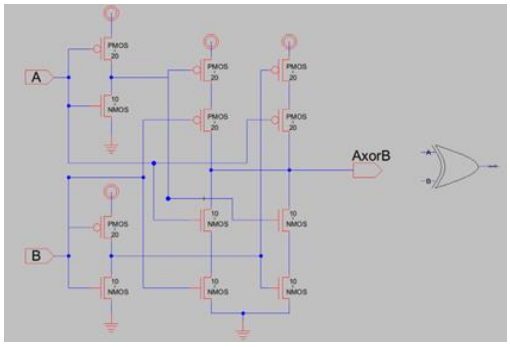

Figure 8: Schematics of XOR gate

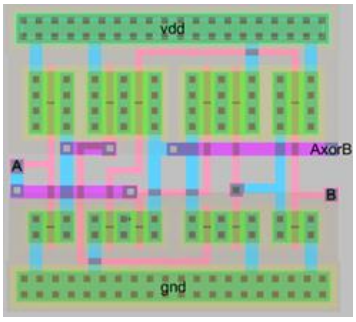

Figure 9: Layout of XOR gate

Table III shows the truth table of 2-input XOR gate.

Input		Output
A	B	Y
0	0	0
0	1	1
1	0	1
1	1	0

Table III: XOR truth table

E. Half adder

Half adder is the simplest of all adder circuits, it implemented from XOR with AND gates, with total 18 transistors.
The SUM output is the least significant bit of the result, which is the XOR of the two inputs. The Carry output is the most significant bit of the result. The Carry output is the AND of the two inputs.[7] Half adder used as main component in our final

design, the following two figures shows both schematics and layout representations for the half adder.

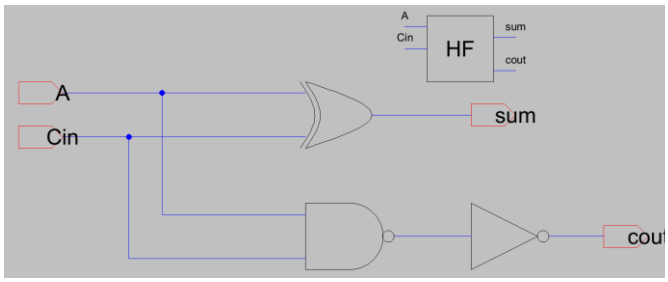

Figure 10: Schematics of half adder

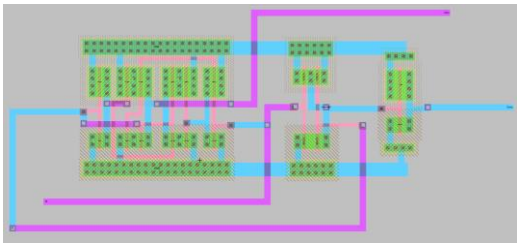

Figure 11: Layout of half adder

Table IV shows the truth table of the Half Adder

Input		Output	
A	B	Sum	Carry
0	0	0	0
0	1	1	0
1	0	1	0
1	1	0	1

Table IV: Half adder truth table

F. Full adder

The full adder is used to add three 1-bit binary numbers A, B, and carry in. Full adder has three input states and two outputs sum and carry. They are utilized in math circuits to add twofold numbers. At the point when different full adders are associated in a chain, they can add multi-bit paired numbers.[8]

Full adder considering as main component in our final multiplication design, the following two figures shows both schematics and layout representations for the full adder.

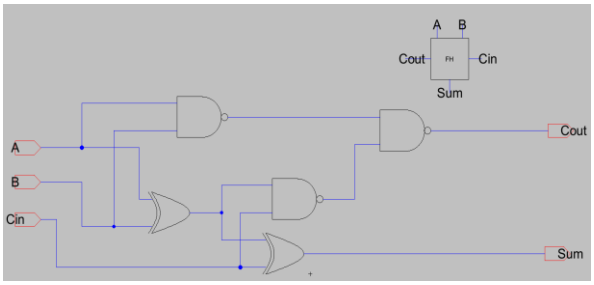

Figure 12: Schematics of full adder

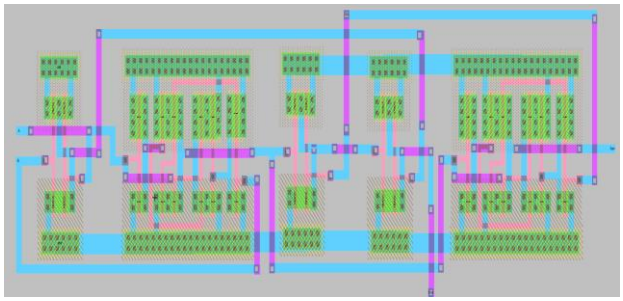

Figure 13: Layout of full adder

Table V represents the truth table of the full adder

Input			Output	
A	B	Cin	Sum	Cout
0	0	0	0	0
0	0	1	1	0
0	1	0	1	0
0	1	1	0	1
1	0	0	1	0
1	0	1	0	1
1	1	0	0	1
1	1	1	1	1

Table V: Full adder truth table

G. N-Latch

In general, latch it has two stable states. It is used to store one bit of data. A D latch is one of the simplest types of latches which operates with a data input ('D') and a clock input.

N-latch built from 4 NMOS transistors and 4 PMOS transistors and invertor, width sizing shown at schematics design, it has one output and two inputs, Clock and D.

The latch will be transparent when the clock is high and latched when the clock is low. The following two figures shows the schematics and layout designs for N-Latch.

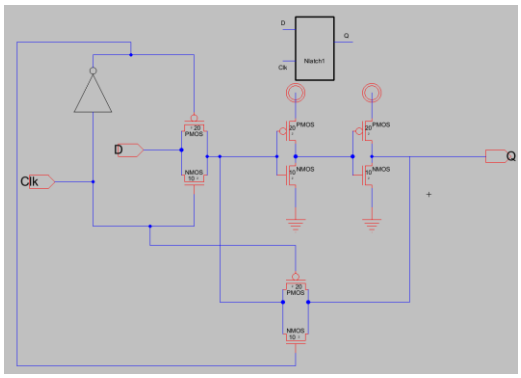

Figure 14: Schematics of N-latch

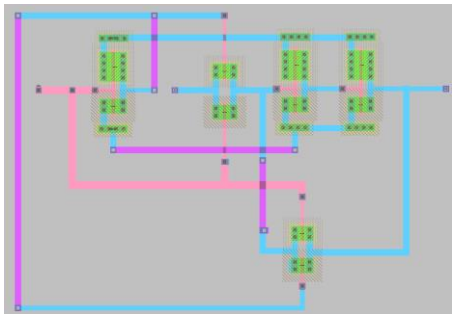

Figure 15: Layout of N-latch

H. P-Latch

P-latch built from 4 PMOS transistors and 4 NMOS transistors and invertor, width sizing shown at schematics design, it has one output and two inputs, Clock and D.

The latch will be transparent when the clock is high and latched when the clock is low. The following two figures shows the schematics and layout designs for P-Latch.

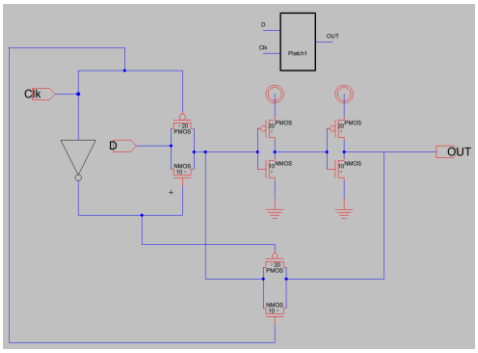

Figure 16: Schematics of P-Latch

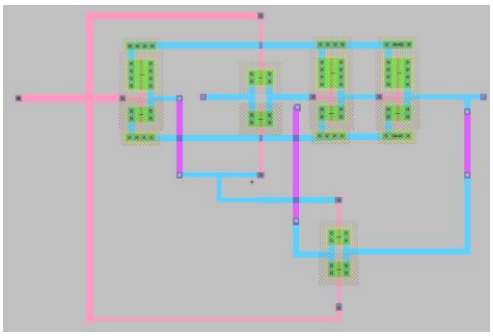

Figure 17: Layout of P-Latch

I. D-Flip flop

D-flip flop considering as storge element like the registers, it used to delay the change of state of its output signal until the next rising edge of a clock timing input signal occurs. Our design for this circuit implemented by use N-latch and P-Latch. In this circuit, when the clock signal is low, the flip flop holds its current state and ignores the D input. On the other hand, When the clock signal is high, the flip flop samples and stores D input. D flip-flop used in our multiplier design between the stages to store the data, the using of d-flip flop considering our design as pipeline system.

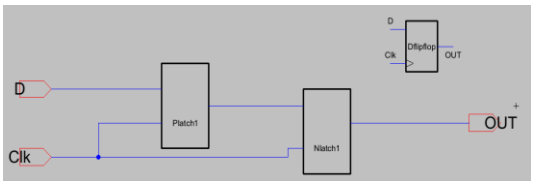

Figure 18: Schematics of D-flip flop

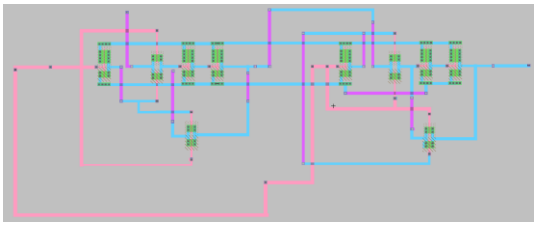

Figure 19: Layout of D-flip flop

J. 4x4 Enhanced Pipeline multiplier

Our multiplication system design was done in both schematics and layout implementations as depicted in the accompanying figures, in multiplier design, the pipelined architecture is usually employed to improve the data throughput.

Furthermore, an important computing block of the multiplier is the full adder. The performance of a multiplier is greatly influenced by the full adder and half adder design.[9] AND gate between the two 4-bit inputs was used 16 time in the complete design.

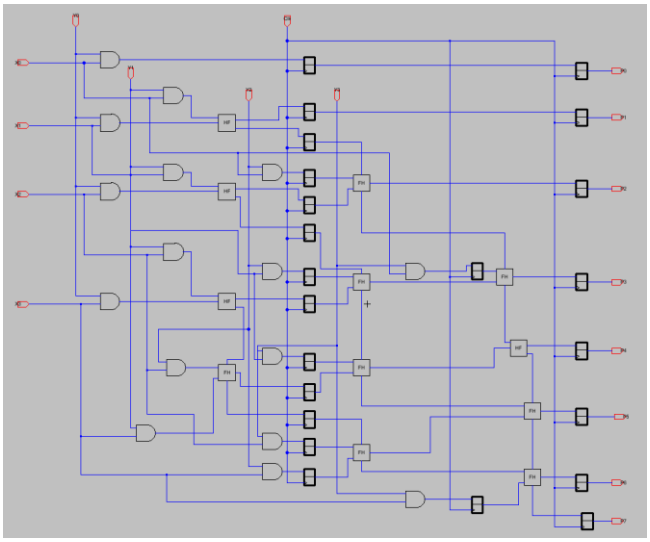

Figure 20: Schematics of 4x4 enhanced pipeline multiplier

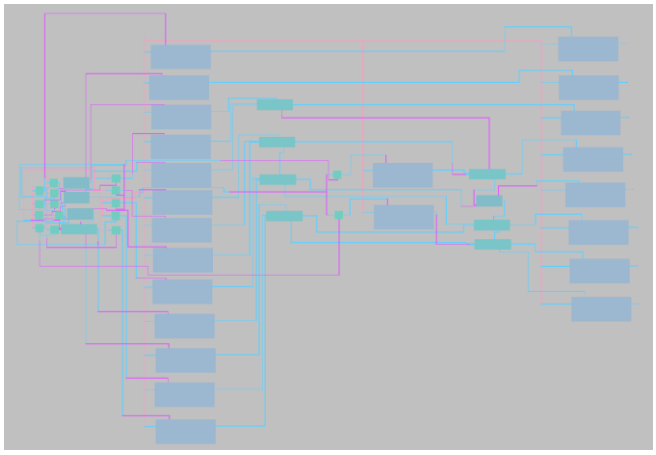

Figure 21: Layout of 4x4 enhanced pipeline multiplier

III. RESULTS AND SIMULATIONS

A. Invertor

The simulation of the inverter circuit, demonstrated the expected behavior and functionality. The output of the inverter was observed to be the logical complement of the input signal.

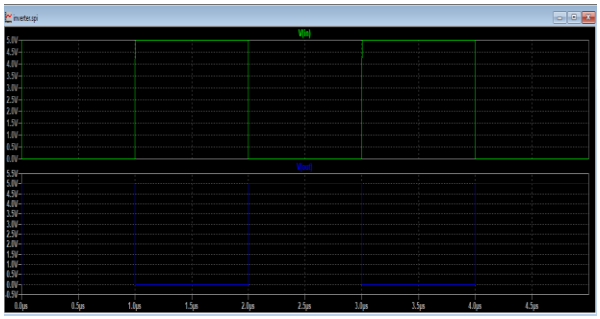

Figure 22: The simulation of the invertor

B. 2-input NAND

The output of the NAND gate was observed successfully as expected, by following the truth table of NAND gate at Table I, it can be noticed that the output simulation gives true values.

Figure 23: The simulation of 2-input NAND

C. 2-input AND

The output of the AND gate was observed with the expected result, by following the truth table of AND gate at Table II, it can be noticed that the output simulation gives true values.

Figure 24: The simulation of 2-input AND

D. 2-input XOR

The output of the XOR gate was observed with the expected result, by following the truth table of XOR gate at Table III, it can be noticed that the output simulation gives true values.

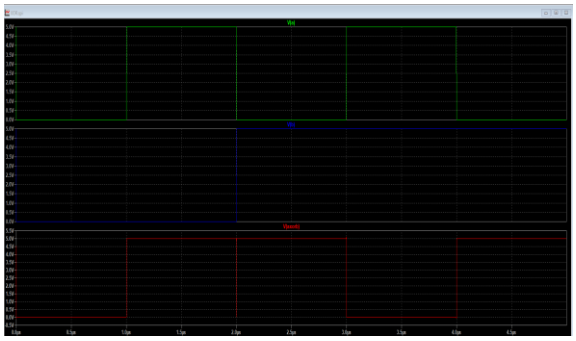

Figure 25: The simulation of 2-input XOR

E. Half adder

The simulation of half adder circuit was observed as expected, by following the truth table of Half adder circuit at Table IV, it can be noticed that the output simulations for the sum and the carry out give true values in all states.

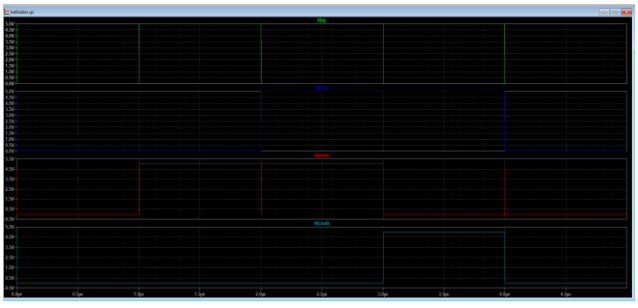

Figure 26: The simulation of Half adder

F. Full adder

The simulation of full adder circuit was observed as expected, by following the truth table of full adder circuit at Table V, it can be noticed that the output simulations for the sum and the carry out give true values in all states.

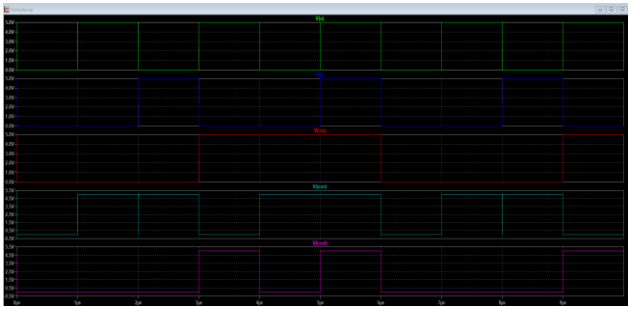

Figure 27: The simulation of Full adder

G. N-latch

The simulation of N-Latch circuit was observed and it can be noticed that it is working as successfully expected as depicted in the accompanying figure.

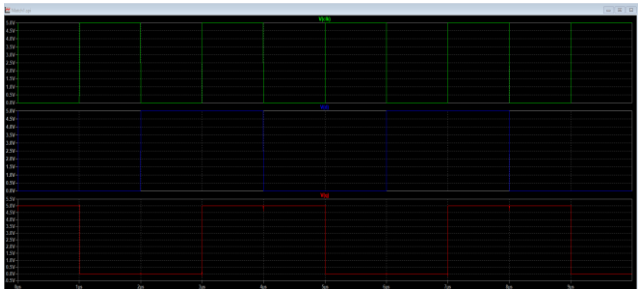

Figure 28: The simulation of N-latch

H. P-latch

The simulation of P-Latch circuit was observed and it can be noticed that it is working as successfully expected as depicted in the accompanying figure.

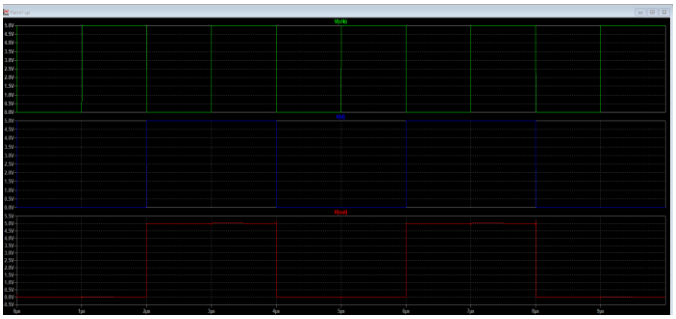

Figure 29: The simulation of P-latch

I. D-Flip Flop

The output of D-Flip flop circuit was observed and it can be noticed that it is working successfully as expected, at the positive edge of the clock the output takes the input (D) value, the result depicted in the accompanying figure.

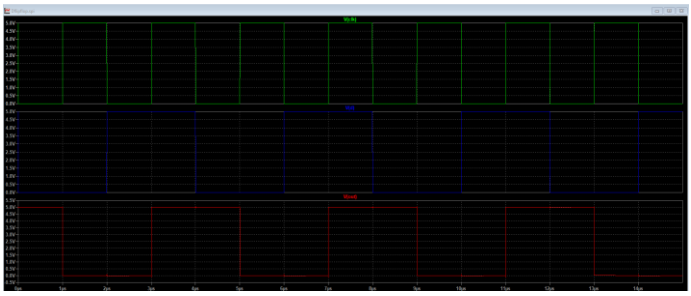

Figure 30: The simulation of D-flip flop

J. 4x4 Enhanced Pipeline multiplier

The simulation of the final system 4x4 Enhanced Pipeline multiplier circuit was observed as expected, by enter two 4-bit input values as a test case, input X consider as 2 (0010), input Y consider as 7 (0111), it can be noticed that the output simulation for the multiplication process equals 14 (1110), the result depicted in the accompanying figure.

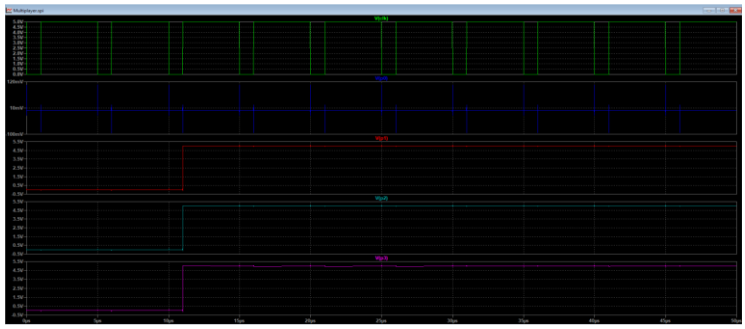

Figure 31: The simulation of 4x4 Enhanced Pipeline multiplier

As a second test case, by enter two 4-bit input values, input X consider as 3 (0011), input Y consider as 6 (0110), it can be noticed that the output simulation for the multiplication process equals 18 (10010), the result depicted in the accompanying figure.

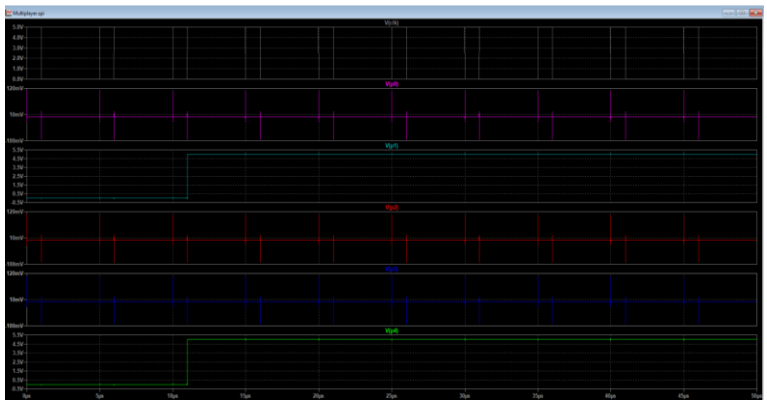

Figure 32: Second test case

As a third test case, by enter two 4-bit input values, input X consider as 4 (0100), input Y consider as 5 (0101), it can be noticed that the output simulation for the multiplication

process equals 20 (10100), the result depicted in the accompanying figure.

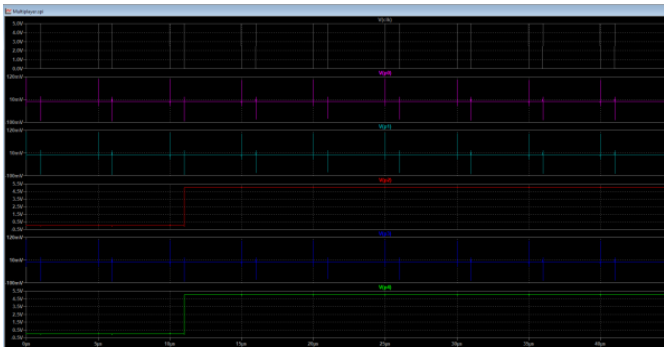

Figure 33: Third test case

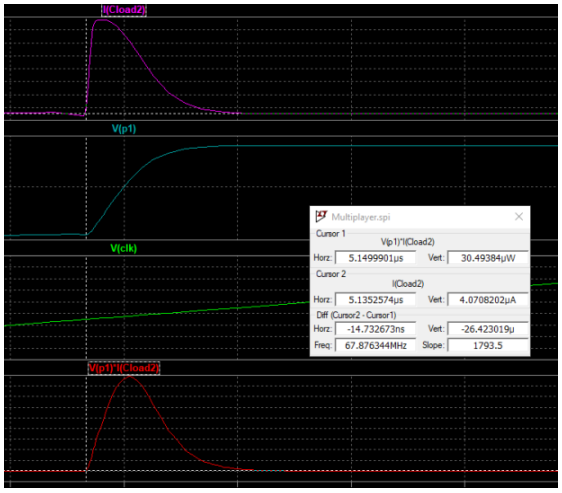

Figure 35: Min Power

To evaluate the rise time delay, two cursors were used to measure the difference at the rising time in the curve, the result of this difference is the value of rise time delay, ash shown in the figure the value is 1.2048495ns.

IV. POWER, AND DELAY OPTIMIZATION

We have taken steps to make the circuit more efficient and cost-effective without affecting its output quality or performance. The sizing of the NMOS and PMOS transistors were considered at a ratio of 2:1 yielded optimal results in terms of achieving equal rise and fall times within the 300nm process.

After multiply the current by the output voltage we get power curve, the max power was measured by using a cursor at the curve of V*I, the result 3.4574496 mW was observed as shown in the figure below.

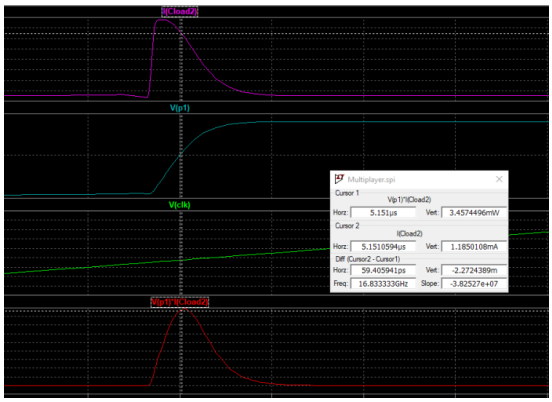

Figure 34: Max power

The min power was measured by using a cursor at the curve of V*I, the result 30.49384 uW was observed as shown in the figure below.

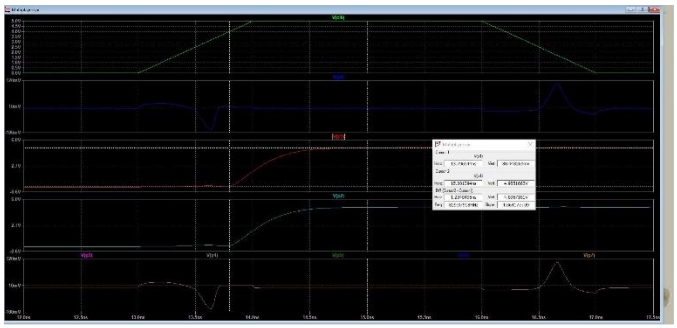

Figure 36: Rise time delay

To evaluate the fall time delay, two cursors were used to measure the difference at the falling time in the curve, the result of this difference is the value of fall time delay, as shown in the figure the value is 1.0147715ns.

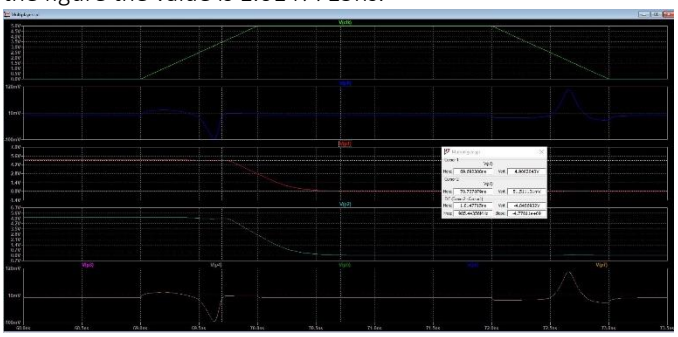

Figure 37: Fall time delay

The following table compares between three types of multipliers, Conventional 90-T 4x4 multiplier, Proposed 4x4 multiplier, and our system 4x4 Enhanced Pipeline multiplier. The factors to compare are the power and delay, it can be noticed that the most average power value consider for Conventional multiplier. Furthermore, it turns out that the fall and rise time delays are very close between the three multipliers.

	Conventional 90-T 4x4 multiplier	Proposed 4x4 multiplier	4x4 Enhanced Pipeline multiplier (our design)
Avg. Power	18.15621 mw	14.32673 mw	1.7439694mW
Max Power	0.2267146 w	0.2471635 w	3.4574496 mW
Min Power	0.6970550 mw	0.49741495 mw	30.49384 uW
Rise time delay	1.3875 ns	1.1020 ns	1.2048495ns
Fall time delay	1.2046 ns	0.9098 ns	1.0147715ns

Table VI: Comparison table [10]

V. CONCLUSION

In conclusion, the schematics and layout designs and implementation steps of 4x4 Enhanced Pipeline multiplier were completed successfully with observed of the expected results thoroughly examined and presented in a comprehensive way. The layout and schematic designs for each component utilized in the final circuit were clearly shown. Notably, simulation results of all circuits were carefully obtained, meticulously analyzed, and effectively communicated, the delay and power for the multiplier were measured and compered with another systems.

This project not only stands as a significant contribution to the field of Integrated circuits, but also opens avenues for future research and developments in this realm.

REFERENCES

[1] https://www.iioab.org/IIOABJ_9.S1_54-66.pdf [Accessed 3-Sep-2023].

[2] Parate PG, Patil PS, Subbaraman S. [2008] Asic implementation of 4 bit multipliers. In Emerging Trends in Engineering and Technology, 2008. ICETET'08. First International Conference on (pp. 408-413). IEEE [Accessed 3-Sep-2023].

[3] D. A. Henlin, M. T. Fertsch, M. Mazin, and E. T. Lewis, "A 16 bit x 16 bit pipelined multiplier macrocell," IEEE J. Solid-state Circuits, vol. 20, no. 2, pp. 542-547, Apr. 1985. [*Accessed* 4 – *Sep* – 2023].

[4] https://www.researchgate.net/?_tp=eyJjb250ZXh0Ijp7ImZpcnN0UGFnZSI6Il9kaXJlY3QiLCJwYWdlIjoiX2RpcmVjdCJ9fQ [Accessed 4-Sep-2023].

[5] https://web.iitd.ac.in/~shouri/eel201/tuts/pipeline_notes.pdf . [Accessed 4-Sep-2023].

[6] https://vlsi-iitg.vlabs.ac.in/CMOS_theory.html [Accessed 6-Sep-2023].

[7] https://www.geeksforgeeks.org/half-adder-in-digital-logic [Accessed 5-Sep-2023].

[8] https://www.geeksforgeeks.org/full-adder-in-digital-logic/ . [Accessed 5-Sep-2023].

[9] https://www.sciencedirect.com/science/article/abs/pii/S0026269210002053 [Accessed 6-Sep-2023]

[10] file:///C:/Users/hp/Downloads/Ref_paper_2ajast-23.pdf [Accessed 7-Sep-2023]