VLSI Project Reprot

Harshit Goyal-2023102054
Electronics & Communication Engineering
International Institute of Information Technology, Hyderabad
Hyderabad, India
harshit.goyal@students.iiit.ac.in

I. Introduction

This report includes SPICE simulation, MAGIC layout, FPGA and Verilog implementation of a 4-bit Carry Look Ahead adder(CLA Adder) circuit that can drive the inverter of size $W_n = 10 * LAMBDA$ and $W_p = 2 * W_n$, where $LAMBDA = 0.09\mu m$. Implemented circuit follows the arrangement given below.

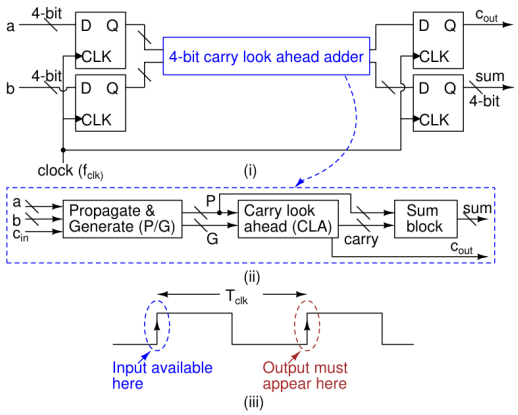

Fig. 1. Arrangement of the circuit

II. Design of the Circuit

To implement the 4-bit CLA adder the block for Carry propagation and generation is used.
The propogate and generate equations are as follows:

$$P_i = A_i \oplus B_i$$

$$G_i = A_i.B_i$$

The carry equations for a Carry-Lookahead Adder (CLA) are as follows:

$$C_{i+1} = G_i + P_i C_i$$

On Simplifying the Equations (substituting the Ci value in Ci+1 Equations),we get:

$$C_1 = G_0 + P_0.C_0$$

$$C_2 = G_1 + P_1.G_0 + P_1.P_0.C_0$$

$$C_3 = G_2 + P_2.G_1 + P_2.P_1.G_0 + P_2.P_1.P_0.C_0$$

$$C_4 = G_3 + P_3.G_2 + P_3.P_2.G_1 + P_3.P_2.P_1.G_0 + P_3.P_2.P_1.P_0.C_0$$

Static CMOS implementation is used for most of the gates, and for some gates (specifically XOR) pass transistor logic styling is used.
2-input gates are used in place of multiple input gates in order to minimize the delay.
Standard CMOS static sizing is used, so that the delay is minimized and the other unwanted effects don't bother.

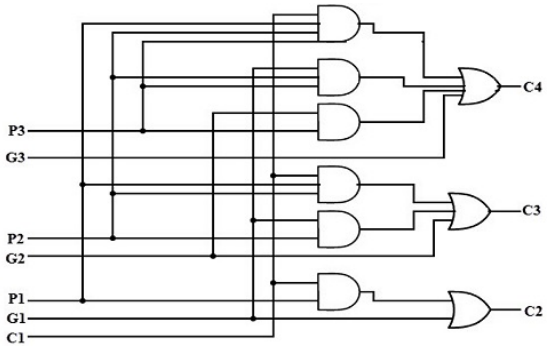

Fig. 2. The carry generator part of the circuit

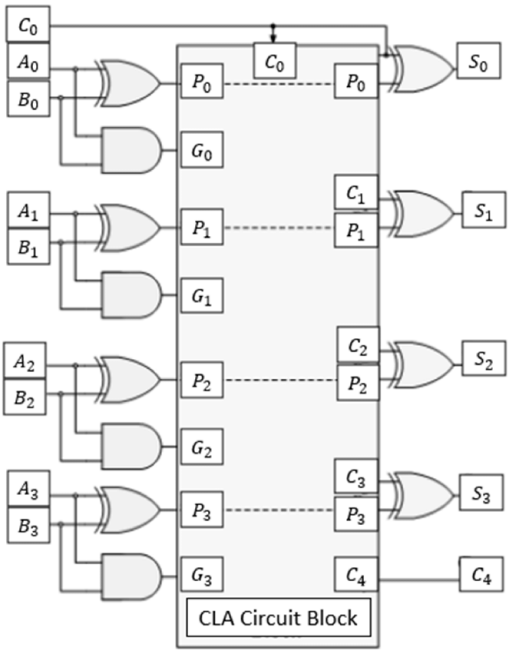

Fig. 3. General Arrangement of 4-bit CLA

III. Design topolgy

$W_n = 10 * LAMBDA$ and $W_p = 2 * W_n$, where $LAMBDA = 0.09\mu m$ are used as a standard size of static CMOS gates. The sizing of the following gates are now described in terms of W_n and W_p. The length of each MOSFET used is $2 * LAMBDA$ (min length).

A. Inverter

A CMOS inverter uses complementary PMOS and NMOS transistors to achieve low power dissipation, high noise margins, and efficient signal inversion. The static CMOS inverter used at all the place has the $\frac{W_p}{W_n}$ ratio=2 and W_n and W_p are the same as described above.

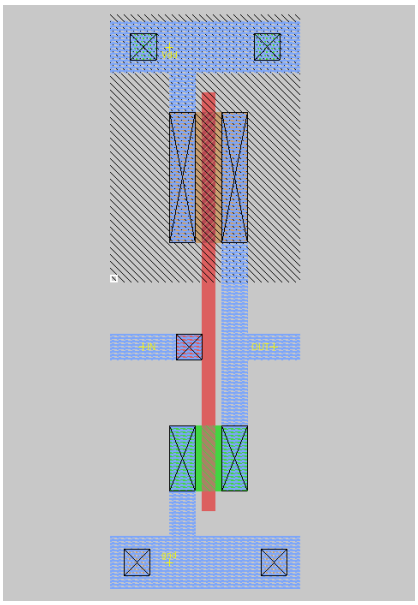

Fig. 4. Topology of INVERTER

B. XOR gate

The XOR gate is implemented using PTL style, with PMOS and NMOS sized identically to those in the CMOS inverter.

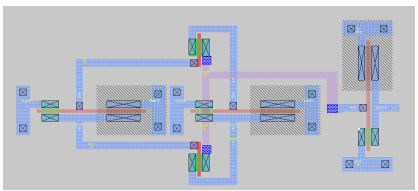

Fig. 5. Topology of XOR gate

C. AND gate

The AND gate is implemented by cascading a NAND gate and an inverter, using series NMOS and parallel PMOS transistors in the NAND stage.

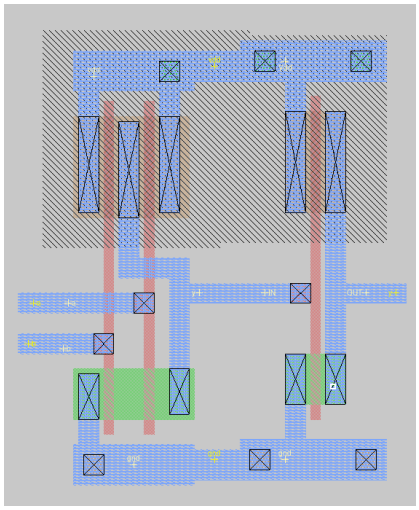

Fig. 6. Topology of AND gate

D. OR gate

The OR gate is implemented by cascading a NOR gate and an inverter, using parallel NMOS and series PMOS transistors in the NOR stage.

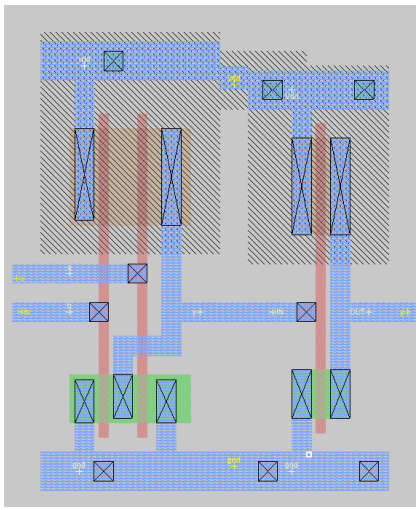

Fig. 7. Topology of OR gate

E. D flip flop

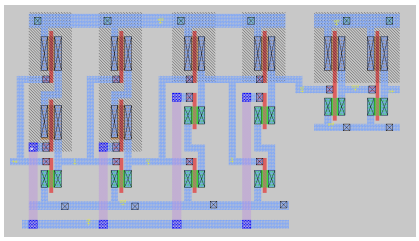

Fig. 8. Topology of D Flip flop in TSPC style

The D Flip-Flop is implemented using TSPC (True Single Phase Clock) logic for high-speed operation and reduced clock

skew, with a buffer added at the output to prevent signal degradation.

F. CLA Adder

The Carry Look-Ahead (CLA) adder topology utilizes generate and propagate signals to quickly compute carries in parallel. It reduces carry propagation delay by using multiple levels of carry generation, improving the speed of binary addition. This structure enhances performance while maintaining efficient area utilization in high-speed arithmetic operations

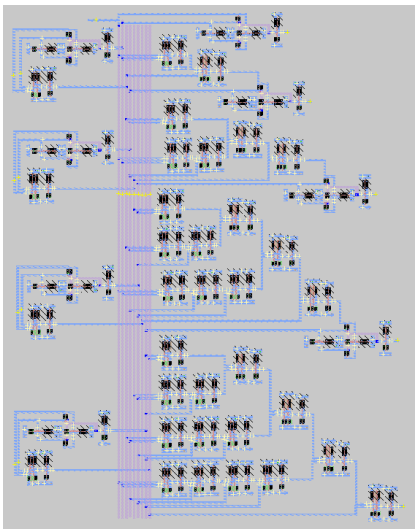

Fig. 9. Topology of CLA Adder

IV. Simulations

Here is the schematic and post-layout simulation results for each block and gate used, with inputs generally labeled as A,B, and IN, and outputs labeled as OUT and Y, demonstrating the functionality and performance of the design.

A. INVERTER

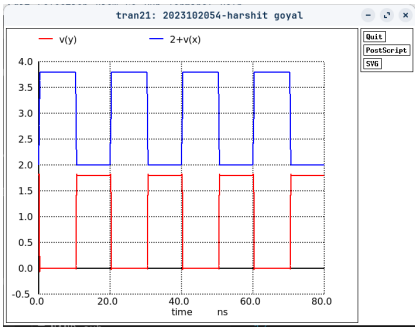

Fig. 10. Schematic Simulation

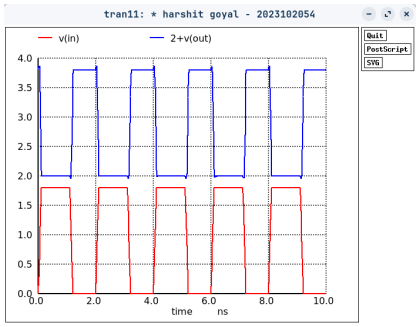

Fig. 11. Post Layout Simulation

B. And Gate

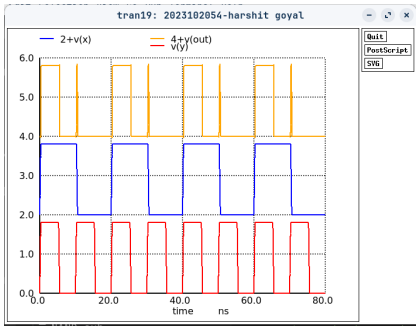

Fig. 12. Schematic Simulation

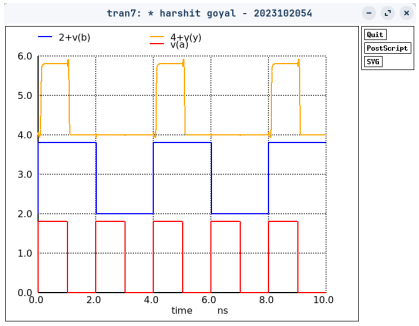

Fig. 13. Post Layout Simulation

C. OR Gate

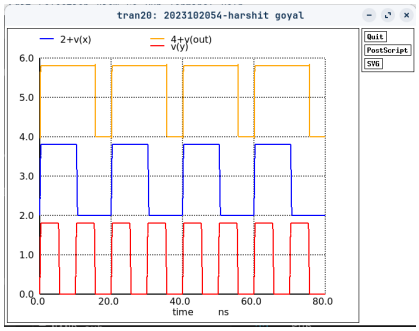

Fig. 14. Schematic Simulation

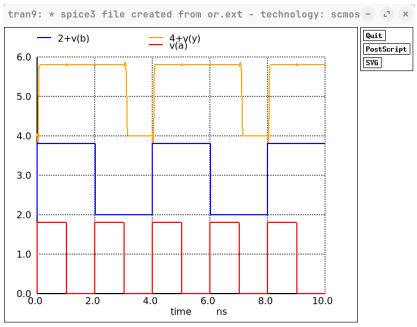

Fig. 15. Post Layout Simulation

D. XOR Gate

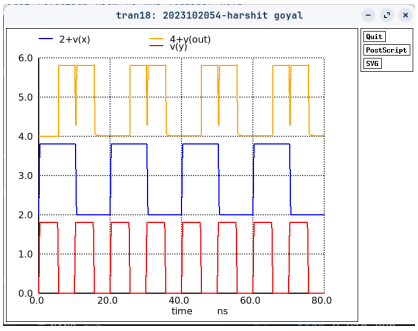

Fig. 16. Schematic Simulation

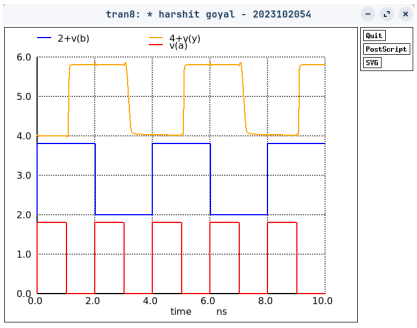

Fig. 17. Post Layout Simulation

E. D-flipflop

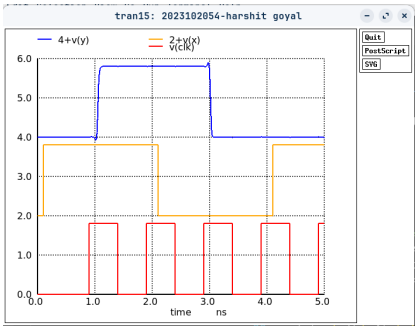

Fig. 18. Schematic Simulation

The schematic simulation plot shows that the clock to Q delay is measured to be 0.135 ns, representing the time it takes for the output to respond to a change in the clock signal.

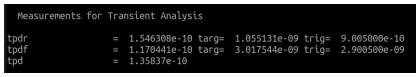

Fig. 19. Clock to Q Delay

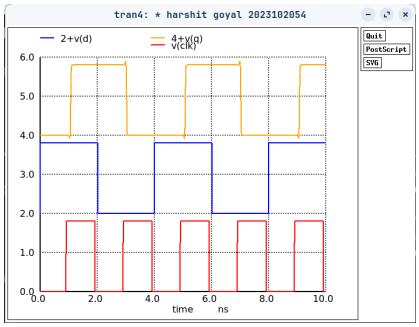

Fig. 20. Post Layout Simulation

The Post layout simulation plot shows that the clock to Q delay is measured to be 0.140 ns, representing the time it takes for the output to respond to a change in the clock signal.

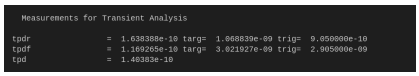

Fig. 21. Clock to Q Delay

F. Hold time

The plot illustrates that changing the input during the clock pulse does not affect the output, confirming that the hold time is zero, as the D flip-flop retains its output state even when the input changes at this point.

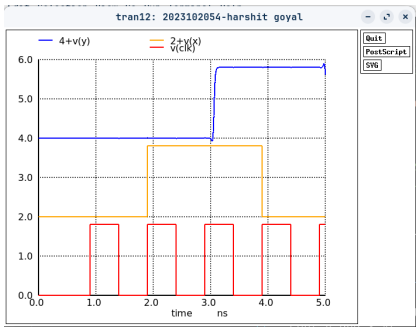

Fig. 22. Topology of or gate

G. Set-up time

The plot demonstrates that the clock pulse arrives at 1.9 ns, and when the input changes at 1.792 ns, the output is correct. However, changing the input at 1.793 ns results in an incorrect output, indicating a setup time of 107 ps for the D flip-flop.

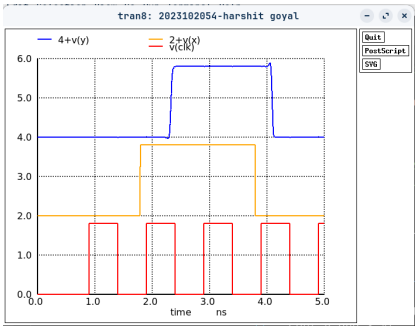

Fig. 23. Input changes at 1.792ns

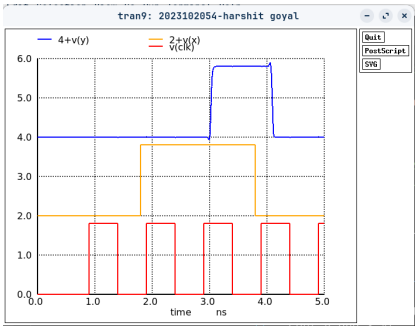

Fig. 24. Input changes at 1.793ns

H. CLA Adder

The plot below indicates the schematic simulation for the Carry Look-Ahead (CLA) adder, demonstrating the functionality and timing behavior of the adder's components.

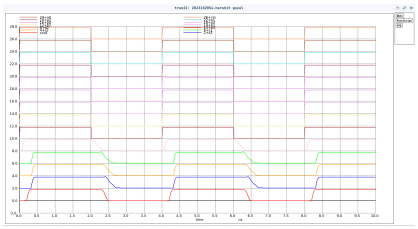

Fig. 25. Schematic Simulation

Below is the delay for the Carry Look-Ahead (CLA) adder from the respective simulation, with the worst delay 0.44ns occurs for the input sets 1111, 1111, and 1 from 0000 and 0000 and 0.

```
Measurements for Transient Analysis

tpdrs0           =  8.739337e-11 targ=  9.239337e-11 trig=  5.000000e-12
tpdfs0           =  1.762120e-10 targ=  2.191212e-09 trig=  2.015000e-09
tpds0            =  1.31803e-10
tpdrs1           =  3.472768e-10 targ=  3.522768e-10 trig=  5.000000e-12
tpdfs1           =  4.057833e-10 targ=  2.420783e-09 trig=  2.015000e-09
tpds1            =  3.76530e-10
tpdrs2           =  3.572159e-10 targ=  3.622159e-10 trig=  5.000000e-12
tpdfs2           =  4.765129e-10 targ=  2.491513e-09 trig=  2.015000e-09
tpds2            =  4.16864e-10
tpdrs3           =  3.442143e-10 targ=  3.492143e-10 trig=  5.000000e-12
tpdfs3           =  5.351191e-10 targ=  2.550119e-09 trig=  2.015000e-09
tpds3            =  4.39667e-10
tpdrcout         =  2.244167e-10 targ=  2.294167e-10 trig=  5.000000e-12
tpdfcout         =  3.895860e-10 targ=  2.404586e-09 trig=  2.015000e-09
tpdcout          =  3.07001e-10
```

Fig. 26. Delay of schematic

The plot below indicates the post layout simulation for the Carry Look-Ahead (CLA) adder, demonstrating the functionality and timing behavior of the adder's components.

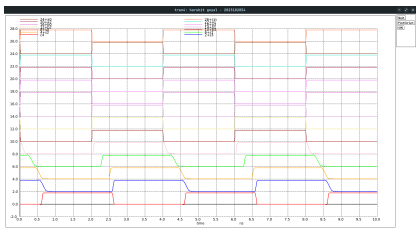

Fig. 27. Post layout Smulation

Below is the delay for the Carry Look-Ahead (CLA) adder from the respective simulation, with the worst delay of 0.62ns occurs for the input sets 1000,0111,1 from 0111,1000,0.

```
Measurements for Transient Analysis

tpdrs0           =  4.137175e-11 targ=  2.056372e-09 trig=  2.015000e-09
tpdfs0           =  8.191686e-11 targ=  4.086917e-09 trig=  4.005000e-09
tpds0            =  6.16443e-11
tpdrs1           =  2.707694e-10 targ=  2.285769e-09 trig=  2.015000e-09
tpdfs1           =  3.327400e-10 targ=  4.337740e-09 trig=  4.005000e-09
tpds1            =  3.01755e-10
tpdrs2           =  4.991994e-10 targ=  2.514199e-09 trig=  2.015000e-09
tpdfs2           =  5.377252e-10 targ=  4.542725e-09 trig=  4.005000e-09
tpds2            =  5.18462e-10
tpdrs3           =  5.912790e-10 targ=  2.606279e-09 trig=  2.015000e-09
tpdfs3           =  6.500281e-10 targ=  4.655028e-09 trig=  4.005000e-09
tpds3            =  6.20654e-10
tpdrcout         =  5.940222e-10 targ=  2.609022e-09 trig=  2.015000e-09
tpdfcout         =  6.131378e-10 targ=  4.618130e-09 trig=  4.005000e-09
tpdcout          =  6.03580e-10
```

Fig. 28. Delay of Post Layout

V. STICK DIAGRAM

Below is the implementation of the stick diagram for each gate used to implement this project, illustrating the physical layout of the gates and their interconnections.

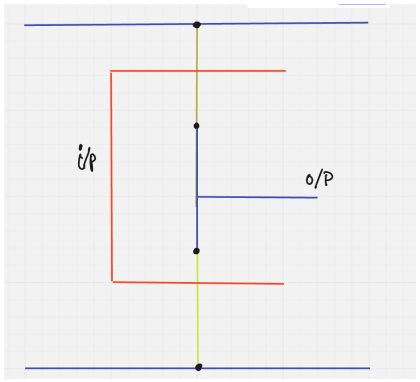

Fig. 29. INVERTER

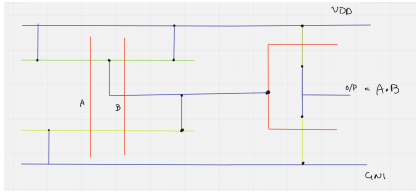

Fig. 30. AND GATE

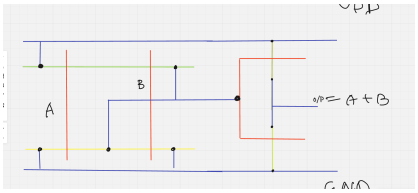

Fig. 31. OR GATE

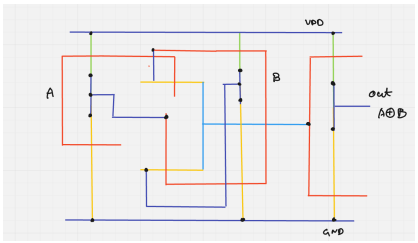

Fig. 32. XOR GATE

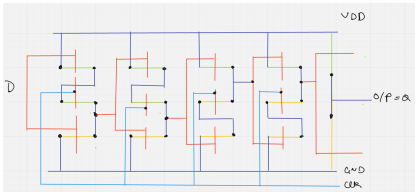

Fig. 33. D Flipflop

VI. Floor Plan of the Circuit

Below is the floor planning of the circuit, depicting the spatial arrangement of the components and gates used in the design. This layout optimizes the placement of each block to minimize routing delays and enhance performance.

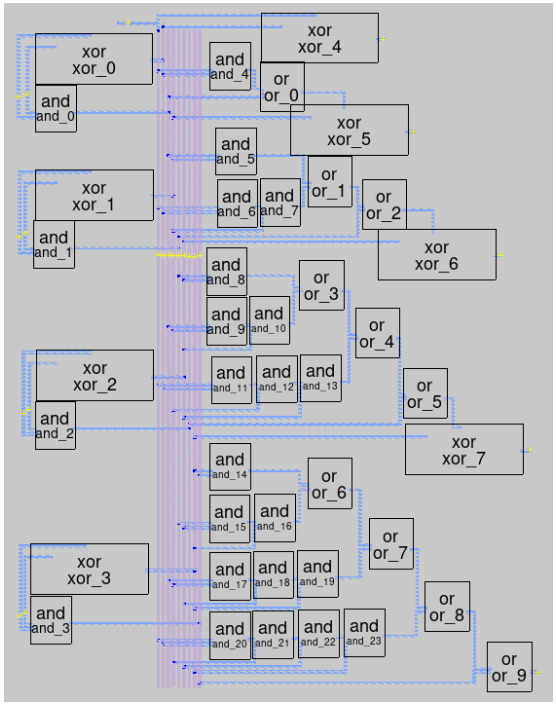

Fig. 34. CLA without D Flipflops

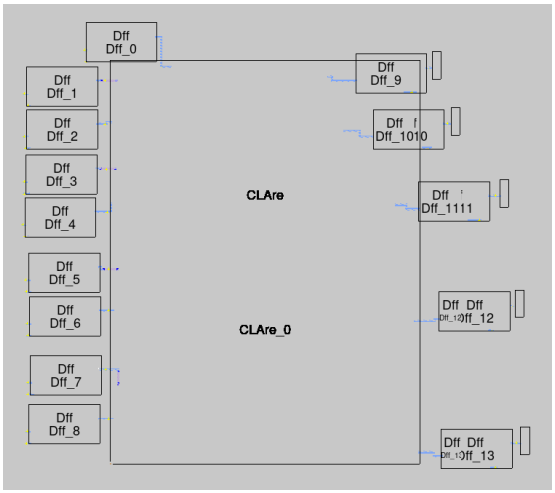

Fig. 35. CLA Block with D Flipflops

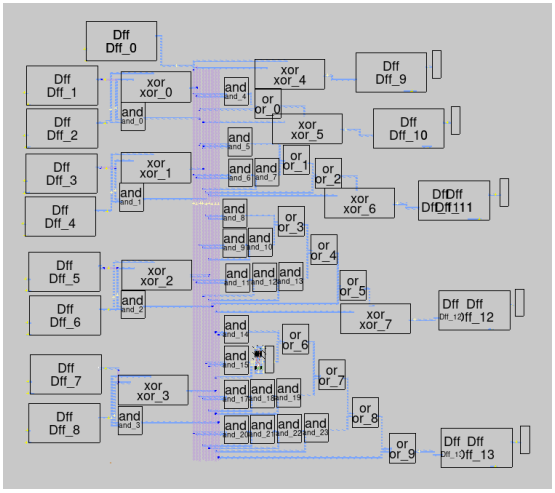

Fig. 36. Final Magic Layout

The final magic layout of the whole circuit.

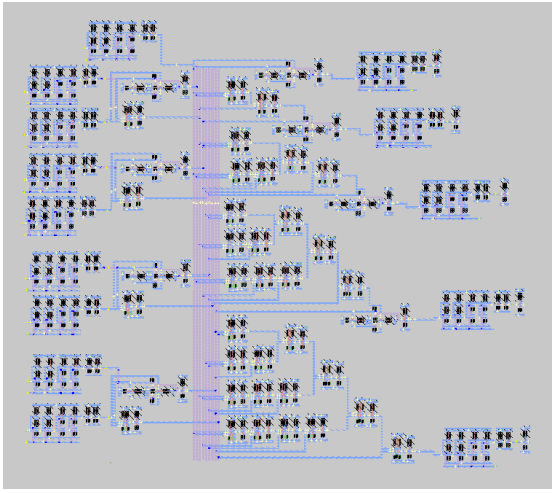

Fig. 37. Layout of full circuit

VII. Full Circuit Simulation

The schematic and Post-Layout simulation was conducted for full circuit, and the output was observed to be correct. However, there was an decrease in the delay, resulting in a smaller time period for the clock signal.
The inputs $A3, A2, A1, A0, B3, B2, B1, B0,$ CIN are fed to the D Flipflop. After one clock cycle, the updated values $A3D, A2D, A1D, A0D, B3D, B2D, B1D, B0D,$ CIND are generated and processed within the D Flipflop and fed to CLA Block. The final outputs are $S3, S2, S1, S0,$ COUT.

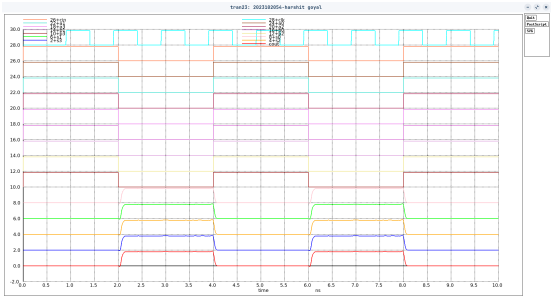

Fig. 38. Schematic Simulation of the circuit

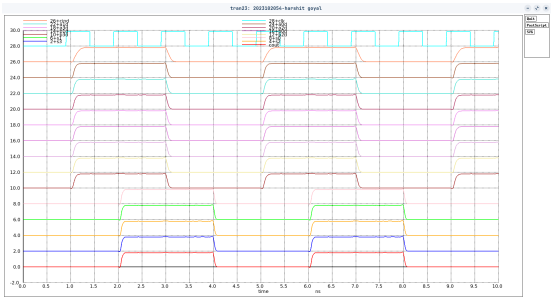

Fig. 39. Schematic Simulation of the circuit

The inputs $A3, A2, A1, A0, B3, B2, B1, B0,$ CIN are fed to the D Flipflop block. After one clock cycle, the updated values $A13, A12, A11, A10, B13, B12, B11, B10,$ CIN1 are generated and processed within the D Flipflop and fed to CLA Block. The final outputs are $S3, S2, S1, S0,$ COUT.

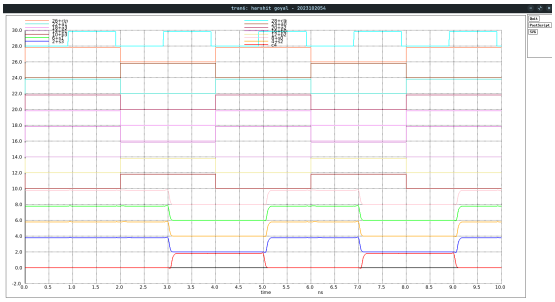

Fig. 40. Post layout Simulation of the circuit

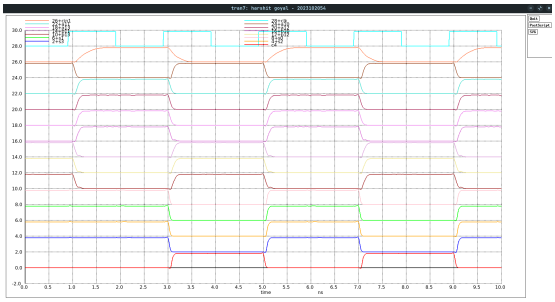

Fig. 41. Post layout Simulation of the circuit

VIII. Post layout delay

The post layout delay of the circuit has increase as compared to the prelayout delays. The increase in delay is mostly accounted due to the increase in parasitics.

```
 Measurements for Transient Analysis

tpdrs0            =  1.678189e-10 targ=  2.068319e-09 trig=  1.900500e-09
tpdfs0            =  1.215505e-10 targ=  4.022050e-09 trig=  3.900500e-09
tpds0             =  1.44685e-10
tpdrs1            =  1.678333e-10 targ=  2.068333e-09 trig=  1.900500e-09
tpdfs1            =  1.215455e-10 targ=  4.022046e-09 trig=  3.900500e-09
tpds1             =  1.44689e-10
tpdrs2            =  1.678334e-10 targ=  2.068333e-09 trig=  1.900500e-09
tpdfs2            =  1.215467e-10 targ=  4.022047e-09 trig=  3.900500e-09
tpds2             =  1.44690e-10
tpdrs3            =  1.678332e-10 targ=  2.068333e-09 trig=  1.900500e-09
tpdfs3            =  1.215512e-10 targ=  4.022051e-09 trig=  3.900500e-09
tpds3             =  1.44692e-10
tpdrcout          =  1.678192e-10 targ=  2.068319e-09 trig=  1.900500e-09
tpdfcout          =  1.215542e-10 targ=  4.022054e-09 trig=  3.900500e-09
tpdcout           =  1.44687e-10
```

Fig. 42. pre delay of the circuit

```
 Measurements for Transient Analysis

tpdrs0            =  1.836614e-10 targ=  5.084161e-09 trig=  4.900500e-09
tpdfs0            =  1.293770e-10 targ=  3.029877e-09 trig=  2.900500e-09
tpds0             =  1.56519e-10
tpdrs1            =  1.833914e-10 targ=  5.083891e-09 trig=  4.900500e-09
tpdfs1            =  1.291149e-10 targ=  3.029615e-09 trig=  2.900500e-09
tpds1             =  1.56253e-10
tpdrs2            =  1.836423e-10 targ=  5.084142e-09 trig=  4.900500e-09
tpdfs2            =  1.293511e-10 targ=  3.029851e-09 trig=  2.900500e-09
tpds2             =  1.56497e-10
tpdrs3            =  1.836905e-10 targ=  5.084190e-09 trig=  4.900500e-09
tpdfs3            =  1.293937e-10 targ=  3.029894e-09 trig=  2.900500e-09
tpds3             =  1.56542e-10
tpdrcout          =  1.837863e-10 targ=  3.084286e-09 trig=  2.900500e-09
tpdfcout          =  1.294568e-10 targ=  5.029957e-09 trig=  4.900500e-09
tpdc4             =  1.56622e-10
```

Fig. 43. Post layout delay of the circuit

TABLE I
Comparison of Pre-Layout and Post-Layout Results

Quantity	Pre-Layout	**Post-Layout**
Tpcq	0.144ns	0.156ns
Tcla(max)	0.44ns	0.62ns
Tclk(min)	0.9ns	1.2ns
freq(max)	1.1GHz	833MHz

The minimum time period of the clock must be greater than or equal to $tpcq + tcla + tsetup$. Therefore,the minimum time period of our circuit prelayout was found to be around 0.8ns and for post layout it was something around 0.9ns. The minimum time period in which our circuit works correctly comes out to be 0.9ns for schematic , and for post layout it was 1.2ns.

IX. Verilog implementaion

The following are the outputs from the structural implementation of the Verilog code for the D flip-flop (DFF)

Fig. 44. GTKwaves plot For Dff

The following are the outputs from the structural implementation of the Verilog code for the 4 bit D flip-flop (DFF)

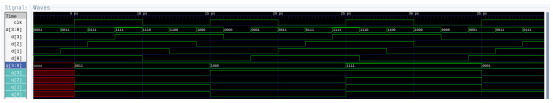

Fig. 45. GTKwaves plot for 4 input DFF

The following are the outputs from the structural implementation of the Verilog code for the 4 bit CLA Adder.

```
harshit@phosgene:~/Desktop/VLSI/2023102054_Project/Verilog$ vvp CLA
VCD info: dumpfile CarryLookAheadAdder_tb.vcd opened for output.
A=1111 B=1111 Cin=1 A1=1111 B1=1111 Cin1=1 S=xxxx Cout=x
A=1111 B=1111 Cin=1 A1=1111 B1=1111 Cin1=1 S=1111 Cout=1
A=1010 B=1100 Cin=1 A1=1010 B1=1100 Cin1=1 S=1111 Cout=1
A=1010 B=1100 Cin=1 A1=1010 B1=1100 Cin1=1 S=0111 Cout=1
A=1010 B=1000 Cin=0 A1=1010 B1=1000 Cin1=0 S=0111 Cout=1
A=1010 B=1000 Cin=0 A1=1010 B1=1000 Cin1=0 S=0010 Cout=1
```

Fig. 46. Terminal outputs

The gtkwave plots for the same set of inputs.

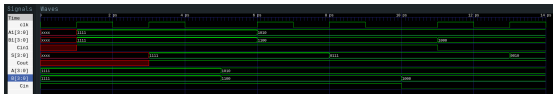

Fig. 47. GTKwaves plot for CLA

A (4-bit)	B (4-bit)	CIN	COUT	S (4-bit)
1111	1111	1	1	1111
1010	1100	1	1	0111
1010	1000	0	1	0010

It can be observed that the inputs are made available to the CLA after the first clock edge, and the output is obtained at the second clock edge.

X. FPGA implementaion

The same verilog code was synthesized and run on the FPGA. The following are the outputs observed for the inputs.

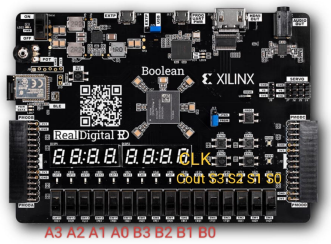

Fig. 48. FPGA with labelled inputs and outputs

A. Example Testcases

Given Inputs:
$A = 1010$ (Decimal 10)
$B = 1000$ (Decimal 8)
$C_{\text{in}} = 0$

Output:
$S = 1000$ (Sum)
$C_{\text{out}} = 1$

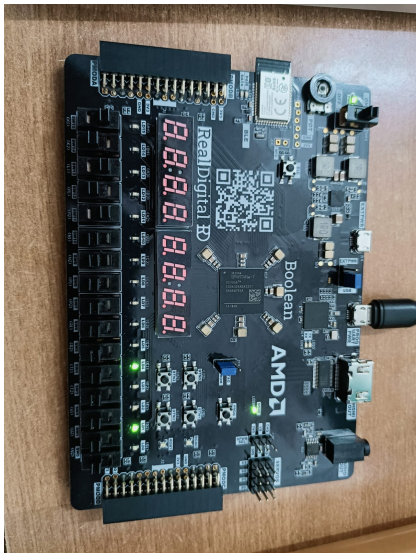

Fig. 49. Input-1010+1000 Output-10010

Given Inputs:
$A = 1010$ (Decimal 10)
$B = 1100$ (Decimal 12)
$C_{\text{in}} = 1$

Output:
$S = 0111$ (Sum)
$C_{\text{out}} = 1$

Fig. 50. Input-1010+1100+1 Output-10111

XI. FPGA IMPLEMENTAION ON OSCILLOSCOPE

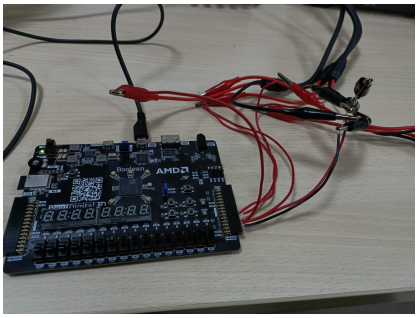

Fig. 51. FPGA with osscilloscope

The link for video of implementation: Click here
The given input was A=7(0111) and B=15(1111) and the output was observes as 22(10110) with cout=1 and s=0110

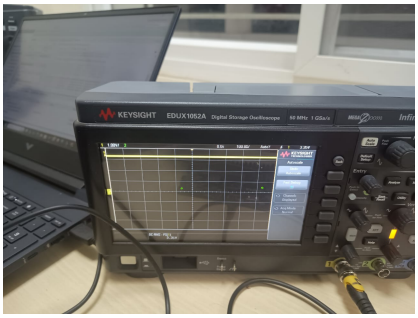

Fig. 52. cout

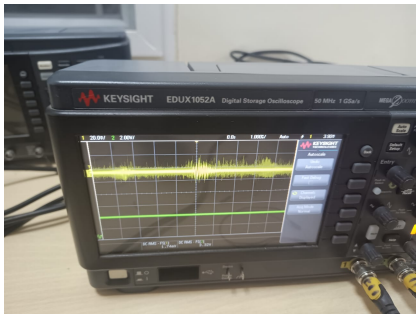

Fig. 53. S3 and S2 bits

Fig. 54. S1 and S0 bits

REFERENCES

[1] Digital Logic and Computer Design by Morris Mano.
[2] Jan M Rabaey, Digital Integrated Circuits
[3] Computer Architecture and Organization by John P. Hayes.
[4] CMOS VLSI Design (fourth edition) by Weste and Harris.
[5] Fundamentals of Digital Logic with Verilog Design by Brown and Vranesic
[6] S. Pradha, R. K. A, V. G., and J. Ajayan, "Comparison of Static and Dynamic Carry Look Ahead Adders," *International Journal of Creative Research Thoughts (IJCRT)*, vol. 6, no. 2, pp. 250–256, April 2018. [Online]. Available: https://www.ijcrt.org
[7] D. Rajesh, S. Sowmya, M. Subashini, P. Suryaprabha, G. Naveen Balaji, and A. Gautami, "Investigations of Carry Look Ahead Adder at VDD=1.0V at 65nm CMOS Technology for High Speed and Low Power Applications," *International Journal of Latest Technology in Engineering, Management and Applied Science (IJLTEMAS)*, vol. VII, no. IV, pp. 191–196, April 2018. [Online]. Available: https://www.researchgate.net/publication/325170876
[8] Class Notes