भारतीय सूचना प्रौद्योगिकी संस्थान, नागपुर
Indian Institute of Information Technology, Nagpur
An Institution of National Importance By An Act of Parliament

Design and Implementation of a CMOS-Based 2-Bit ALU Using 3:8 Decoder

AIM : To implement complete ALU using 3:8 decoder , also designing it in microwind 3.1 . Analysis of performance by writing Netlist and simulation in ngSpice .

NAME	GAGAN DIXIT
Roll Number	BT22ECE098
Year	3rd : 6th Semester
College Name	Indian Institute of Information Technology , Nagpur
Subject	CMOS
Guidance	Dr. Paritosh D. peshwe
Date	**15th April 2025**

About our Arithmatic-Logic-Unit (ALU)

1. SPECIFICATION
    a. 4-Control Lines :

These control lines include 3-Select Lines and 1-Enable decoder line . which are all together used to determine the action of our ALU

b. 5-Input Data Lines :

There are 5 input data lines used to provide ALU with input of : A1-A0 , B1-B0 , Cin

c. 3-Output Data Lines :

The 3 output lines are the outcome of any arithmetic operation performed in the ALU : Cout , D1 , D0

2. MICROWIND

Microwind 3.1 Software was used to carry out the designing part of the layout .

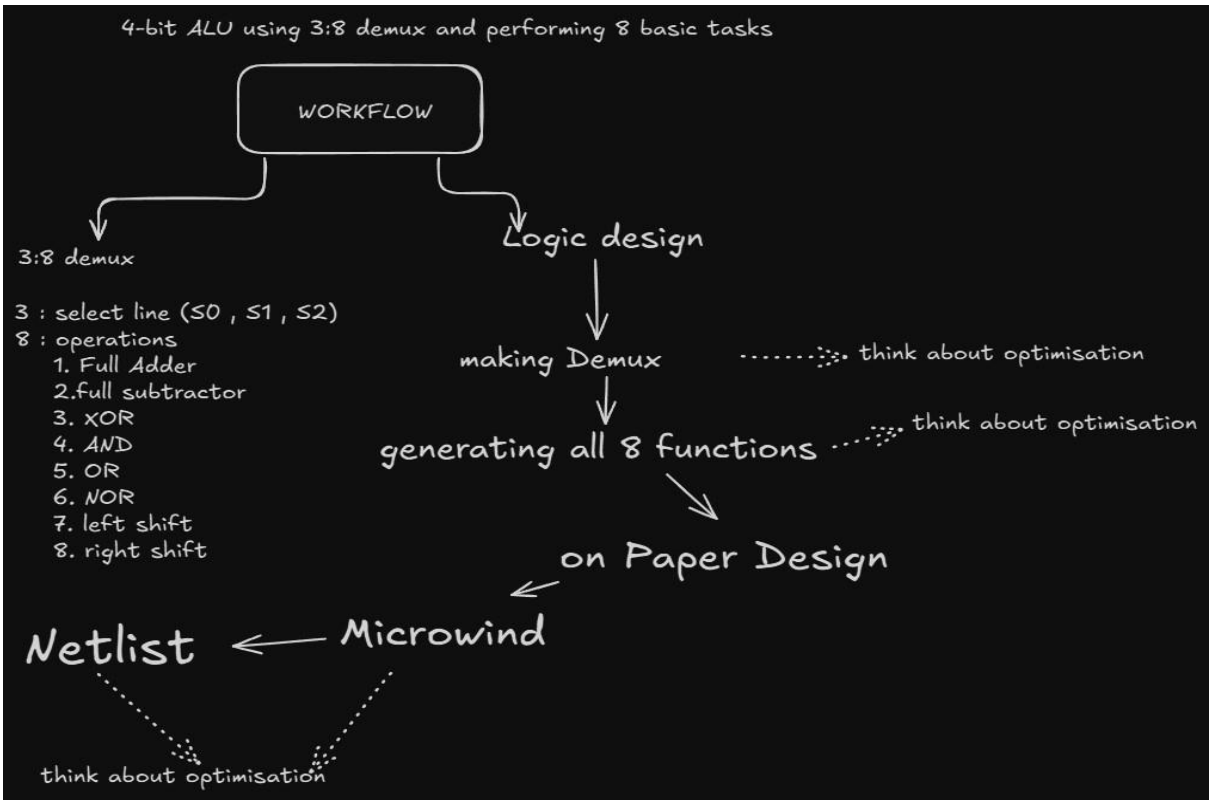
Rule File : 018 was used i.e **180 nm** Based technology

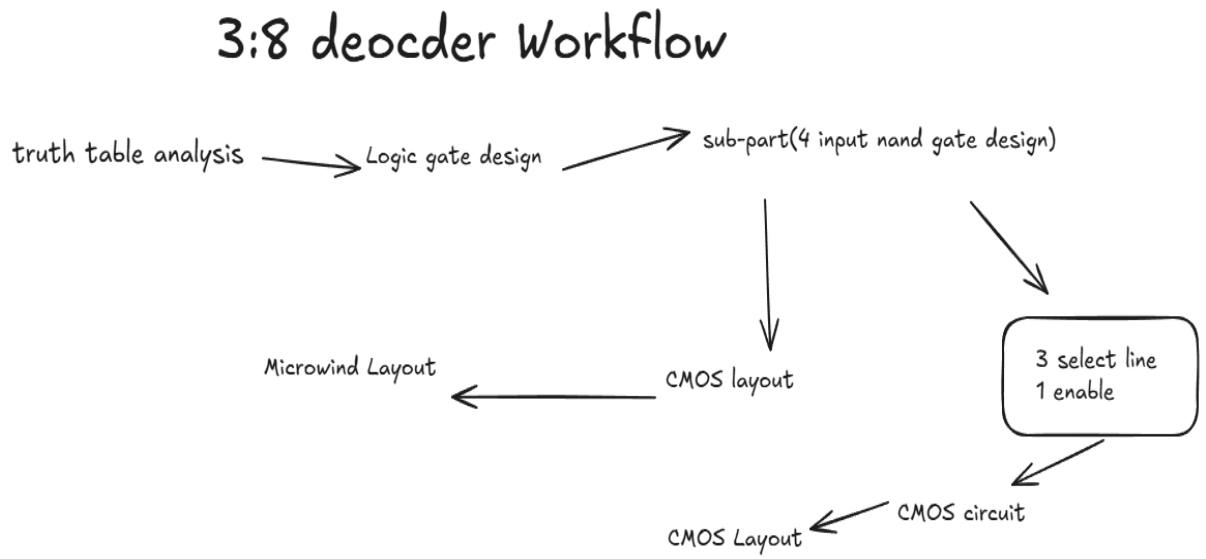
Clocks : ranging from 0.5 uSec to - 17uSec

3. NG-SPICE

Used Ng-Spice for simulation and performance analysis

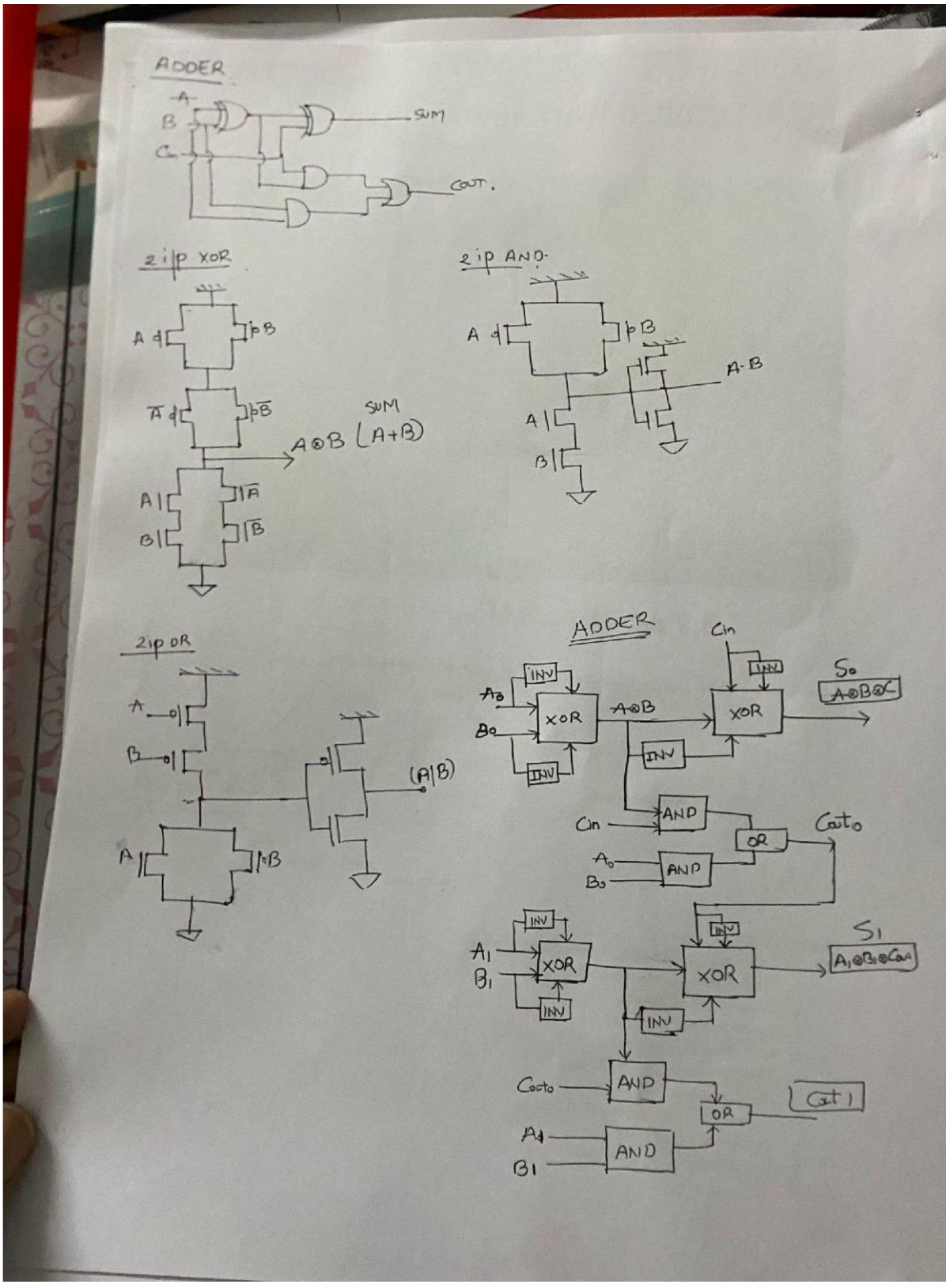
Detailed Workflow Analysis





WALKTHROUGH :

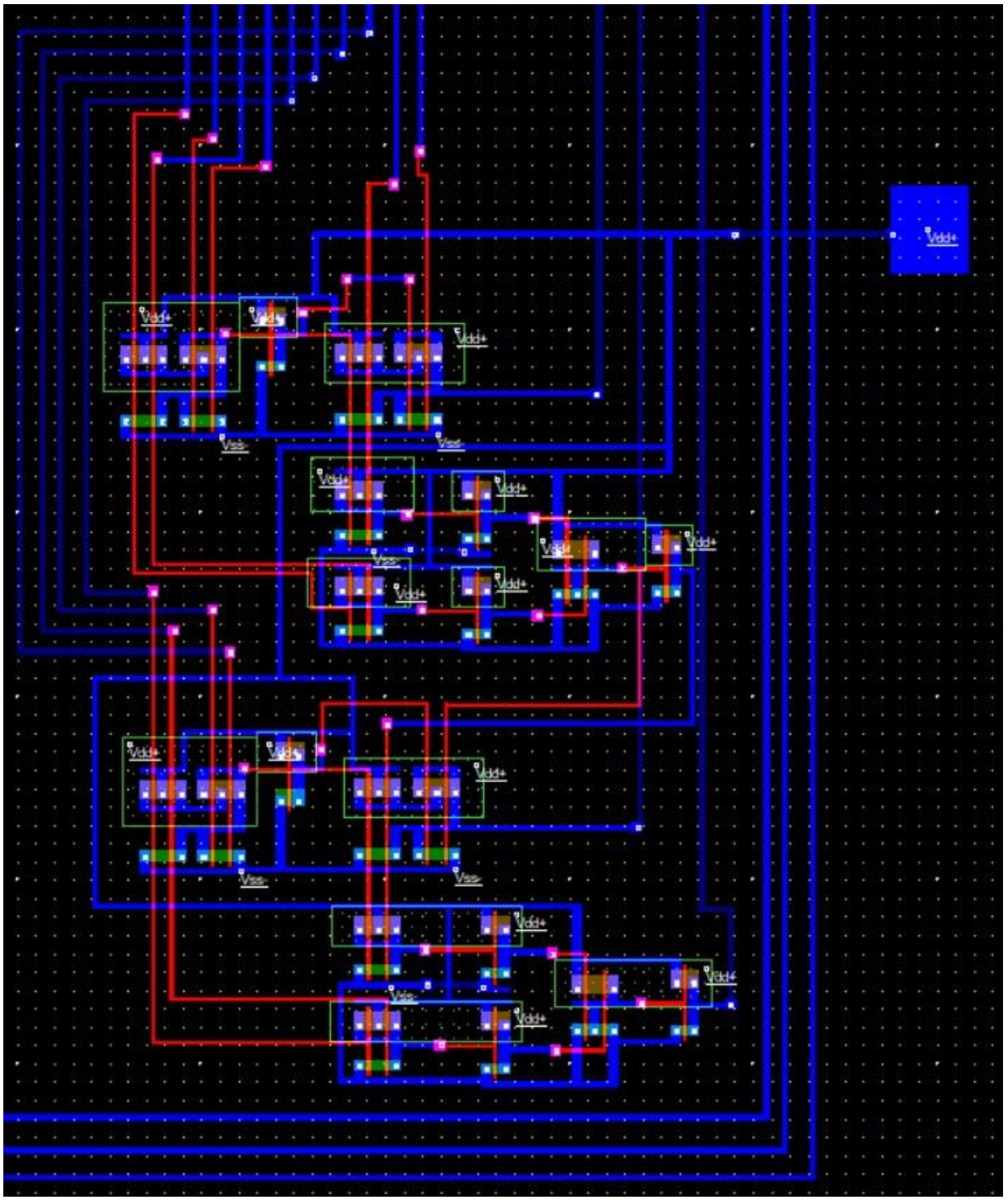
1. Implementing 3:8 Decoder using Nand-Nor logic
2. Having 3-select and 1-enable line
3. 8-line output of decoder
4. Implementing Full-Adder
5. Implement Full-Subtractor
6. Xor Operation
7. And Operation
8. Connection Arithmetic units to Decoder
9. Output on 3-DataLines
10. Optimisation using techniques



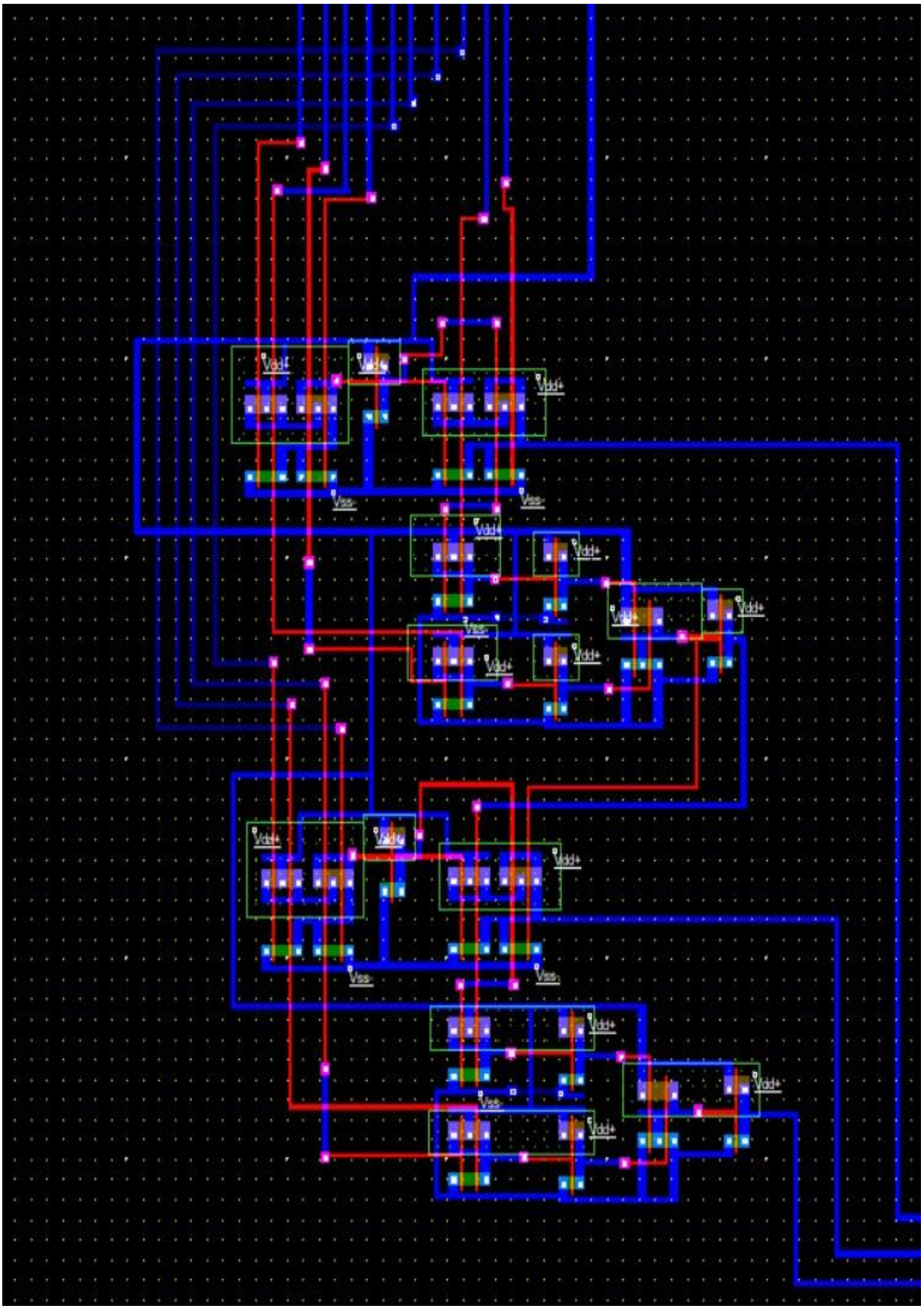
LAYOUT DESIGNING : MICROWIND

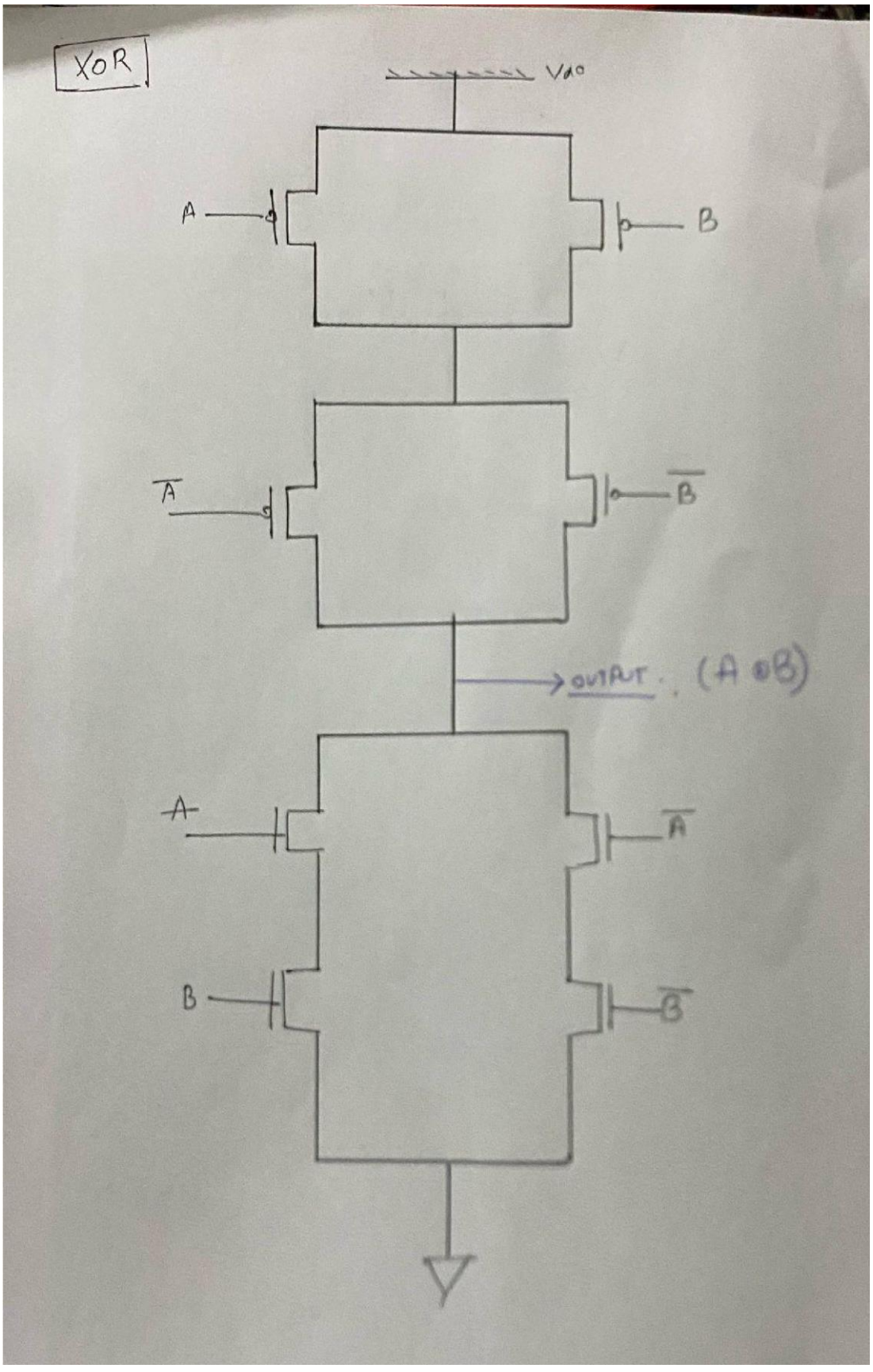
SUB-UNITS :

1. 2-Bit Full Adder

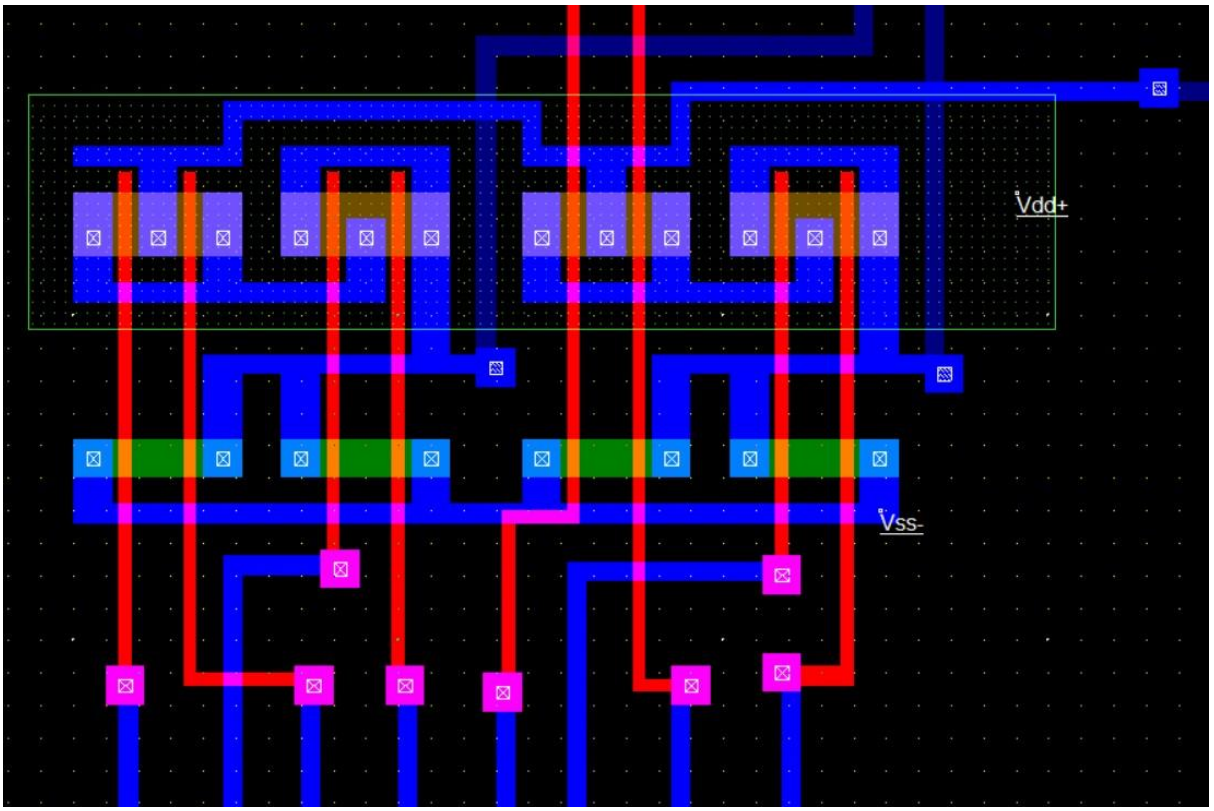


2. 2-Bit Full Subtractor

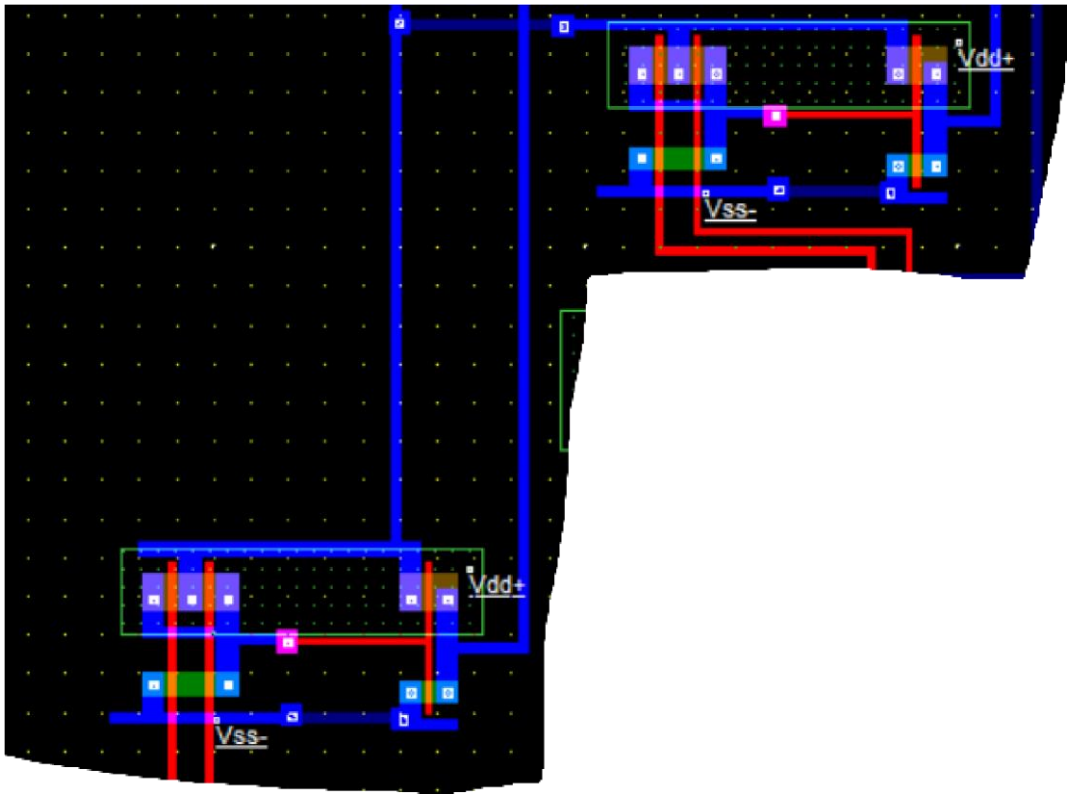

Vdd+
Vss-

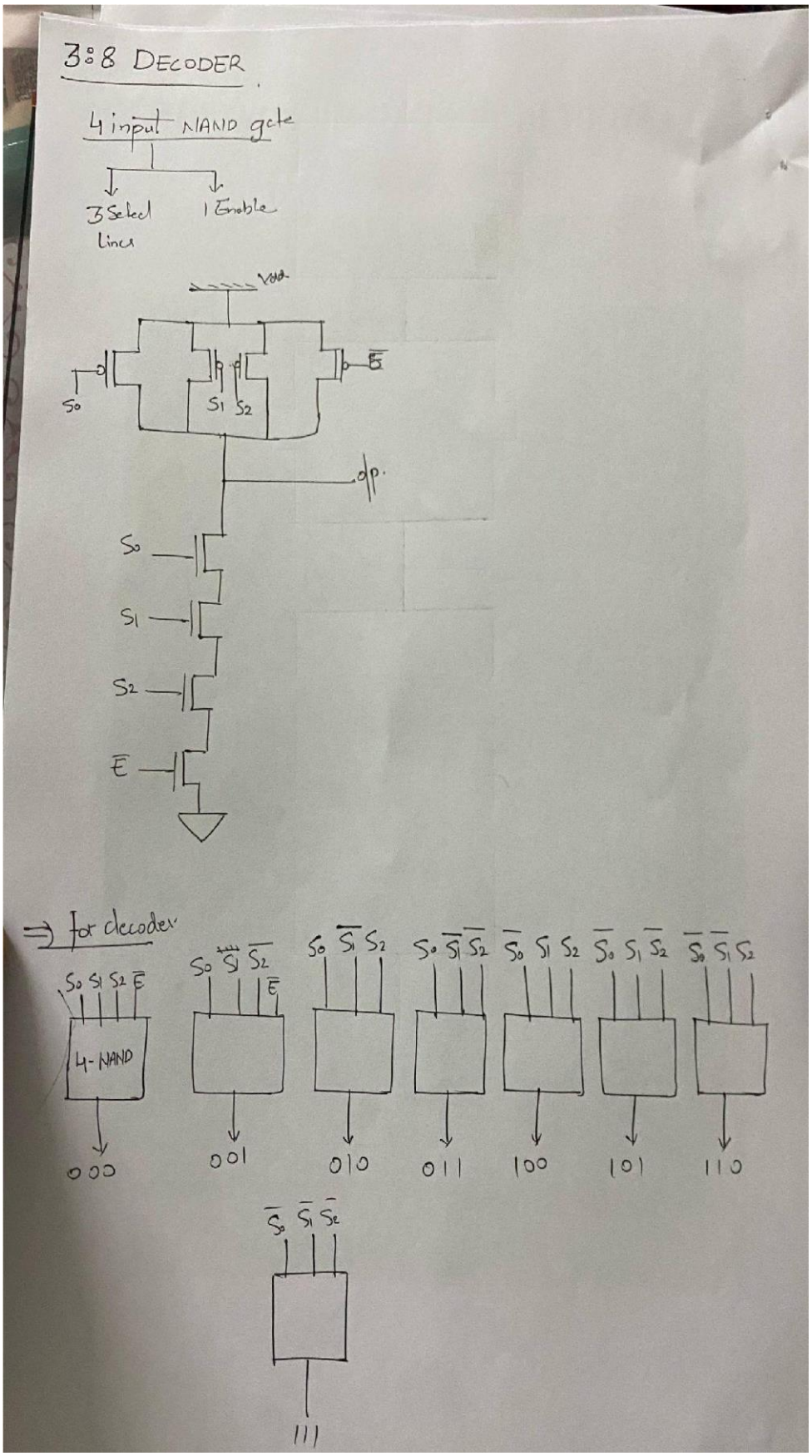

XOR
Vdd
A
B
$\overline{A}$
$\overline{B}$
OUTPUT : (A ⊕ B)
A
$\overline{A}$
B
$\overline{B}$

3. Xor Unit

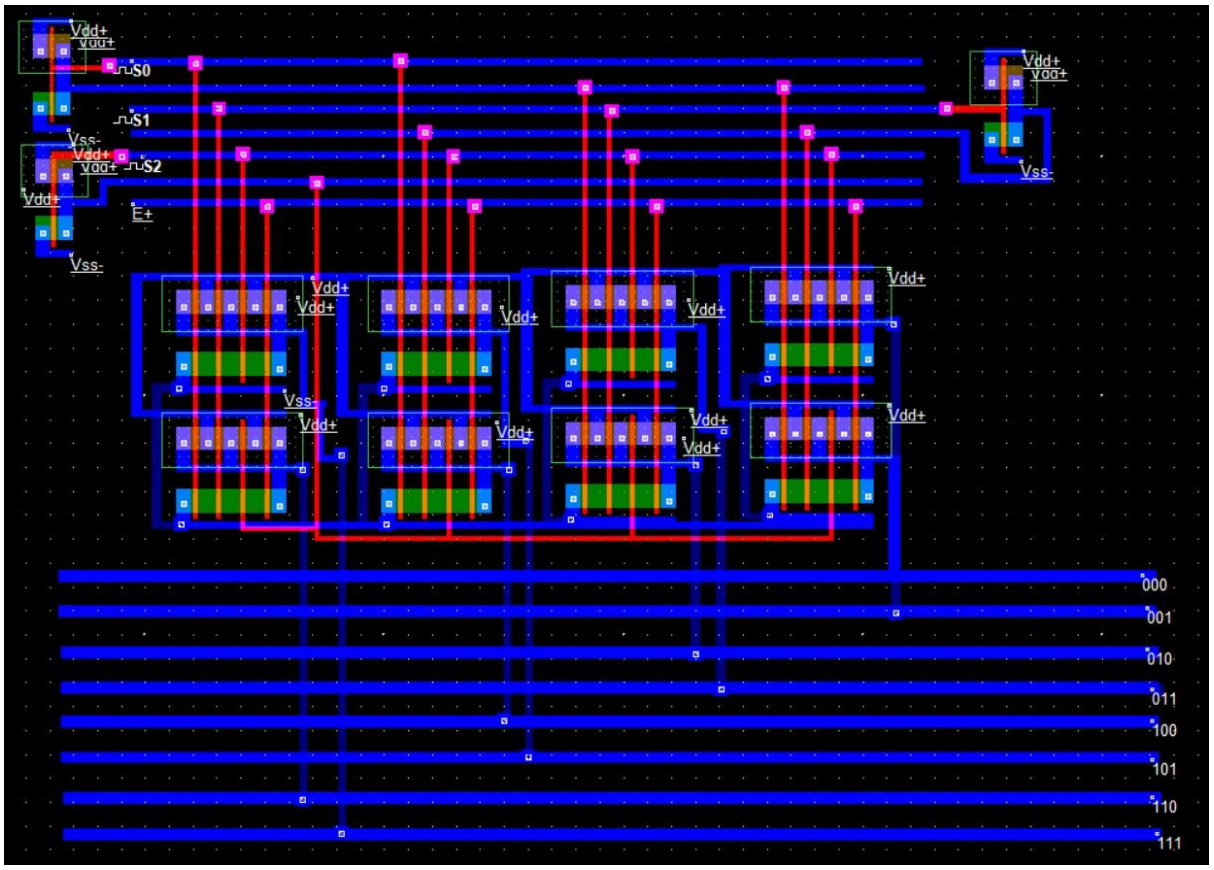


4. And Unit

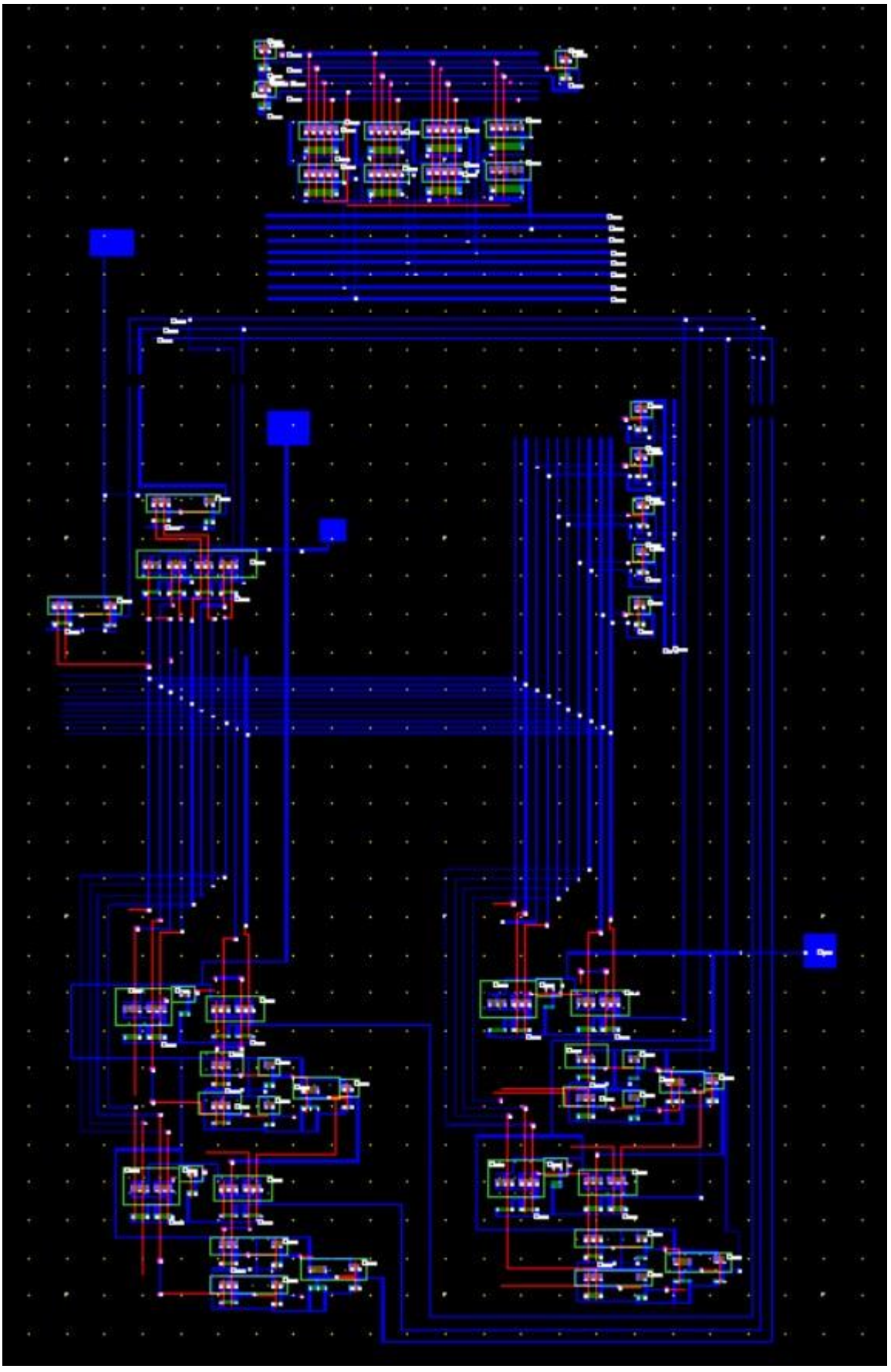




5. 3:8 Decoder Block

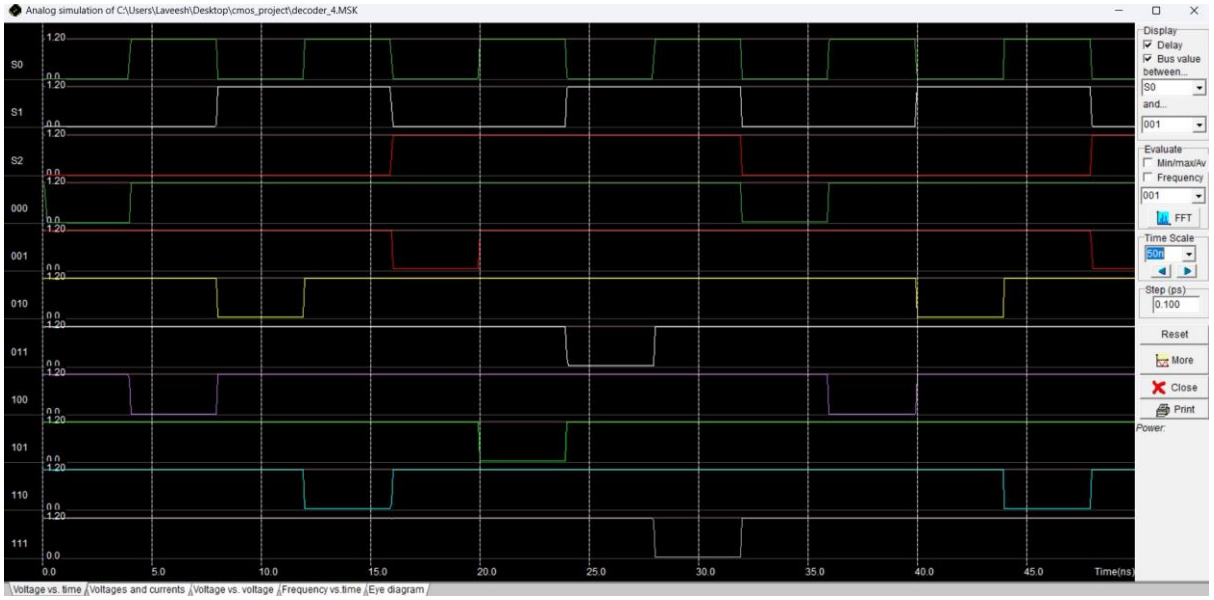


COMPLETE ALU

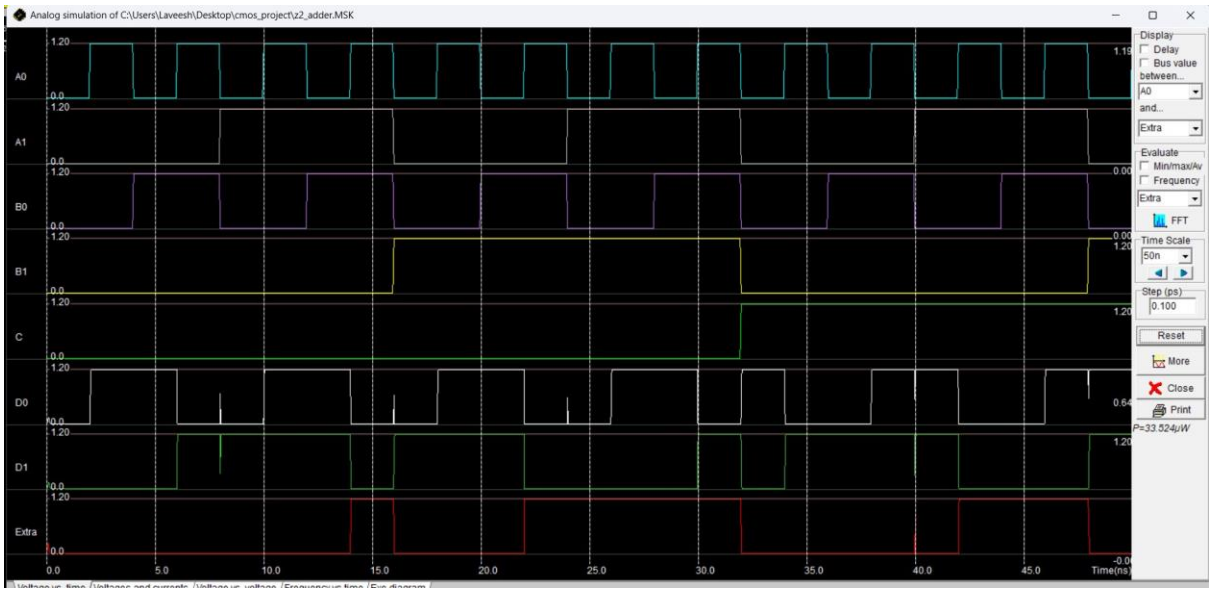


Waveforms : MICROWIND

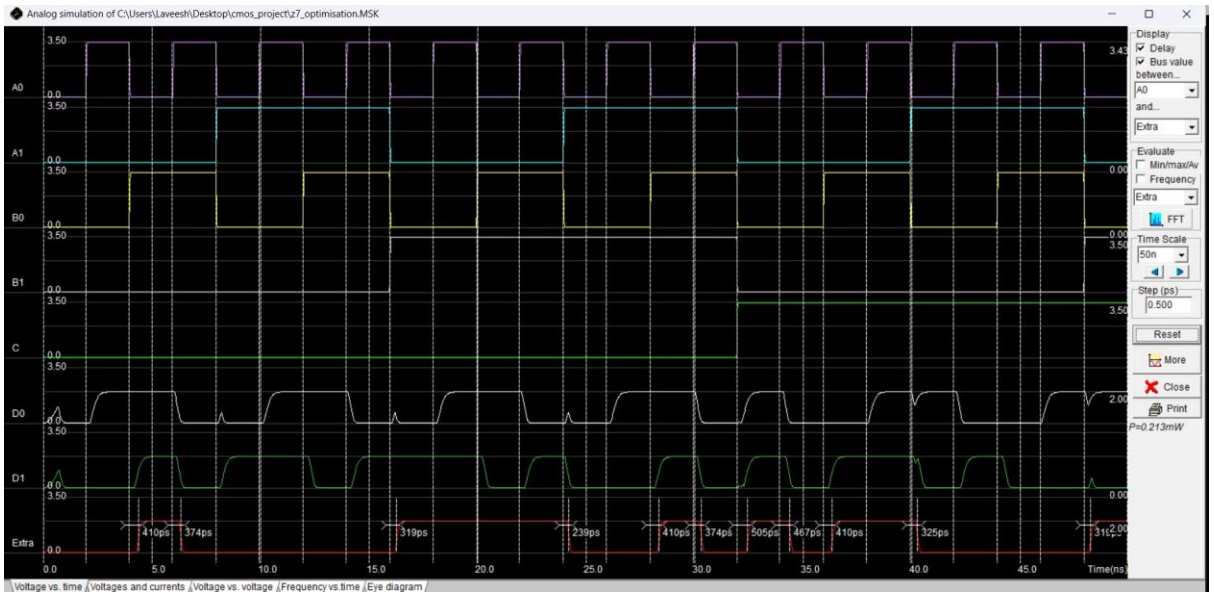
1. 3:8 DECODER :



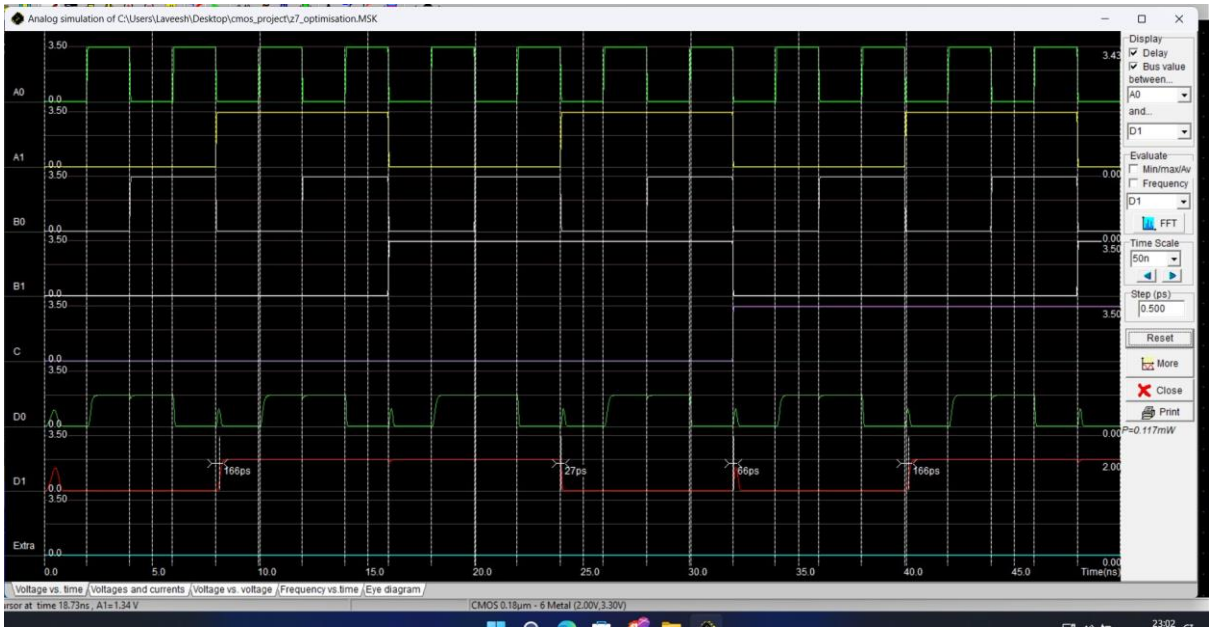
2. FULL-ADDER :



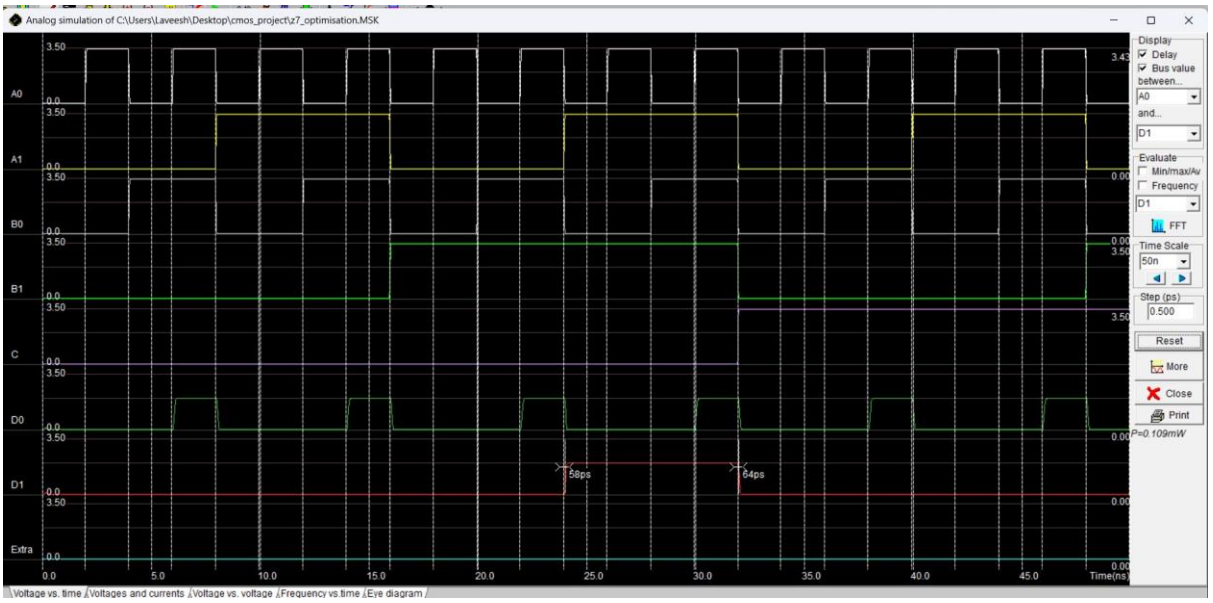
3. FULL-SUBTRACTOR


Analog simulation of C:\Users\Laveesh\Desktop\cmos_project\z7_optimisation.MSK
A0
A1
B0
B1
C
D0
D1
Extra
410ps
374ps
319ps
239ps
410ps
374ps
505ps
467ps
410ps
325ps
Time(ns)
Display
Delay
Bus value between...
and...
Evaluate
Min/max/Av
Frequency
FFT
Time Scale
Step (ps)
Reset
More
Close
Print
P=0.213mW
Voltage vs. time
Voltages and currents
Voltage vs. voltage
Frequency vs.time
Eye diagram

4. XOR


Analog simulation of C:\Users\Laveesh\Desktop\cmos_project\z7_optimisation.MSK
A0
A1
B0
B1
C
D0
D1
Extra
166ps
27ps
66ps
166ps
Time(ns)
Display
Delay
Bus value between...
and...
Evaluate
Min/max/Av
Frequency
FFT
Time Scale
Step (ps)
Reset
More
Close
Print
P=0.117mW
Voltage vs. time
Voltages and currents
Voltage vs. voltage
Frequency vs.time
Eye diagram
CMOS 0.18µm - 6 Metal (2.00V,3.30V)

5. AND



NETLIST : ng-SPICE

SUB-UNITS :

1. 2-Bit Full Adder

```
* 2-Bit Ripple Carry Ader using full Adders


.subckt inverter in out Vdd Gnd
Mp out in Vdd Vdd PMOS L=1u W=2u
Mn out in Gnd Gnd NMOS L=1u W=1u
.ends inverter

.subckt nand2 A B Y Vdd Gnd
Mp1 Y A Vdd Vdd PMOS L=1u W=2u
Mp2 Y B Vdd Vdd PMOS L=1u W=2u
Mn1 net A Gnd Gnd NMOS L=1u W=1u Mn2
Y B net Gnd NMOS L=1u W=1u
.ends nand2

.subckt xor2 A B Y Vdd Gnd
* Intermediate nodes
X1 A n1 Vdd Gnd inverter
X2 B n2 Vdd Gnd inverter
X3 A n2 n3 Vdd Gnd nand2
X4 n1 B n4 Vdd Gnd nand2
X5 n3 n4 Y Vdd Gnd nand2
.ends xor2

.subckt and2 A B Y Vdd Gnd Xnand
A B n1 Vdd Gnd nand2
Xinv n1 Y Vdd Gnd inverter
.ends and2

.subckt or2 A B Y Vdd Gnd
Xinv1 A nA Vdd Gnd inverter
Xinv2 B nB Vdd Gnd inverter
Xnand nA nB nY Vdd Gnd nand2
Xinv3 nY Y Vdd Gnd inverter
```

```
.ends or2

.subckt full_adder A B Cin Sum Cout Vdd Gnd
Xxor1 A B n1 Vdd Gnd xor2
Xxor2 n1 Cin Sum Vdd Gnd xor2
Xand1 A B n2 Vdd Gnd and2
Xand2 B Cin n3 Vdd Gnd and2
Xand3 A Cin n4 Vdd Gnd and2
Xor1 n2 n3 n5 Vdd Gnd or2
Xor2 n5 n4 Cout Vdd Gnd or2
.ends full_adder
****************** 2-Bit Ripple Carry Adder ********************
* Inputs: A0 A1 B0 B1
* Outputs: S0 S1 Cout

Vdd Vdd 0 DC 5
Vss Gnd 0 DC 0

Va0 A0 0 PULSE(0 5 0 1n 1n 10n 20n)
Va1 A1 0 PULSE(0 5 0 1n 1n 20n 40n)
Vb0 B0 0 PULSE(0 5 0 1n 1n 40n 80n)
Vb1 B1 0 PULSE(0 5 0 1n 1n 80n 160n)

* Ground
Vcin Cin 0 DC 0

* Full Adders
XFA0 A0 B0 Cin S0 C1 Vdd Gnd
XFA1 A1 B1 C1 S1 Cout Vdd Gnd

* simulation
C1out S0 0 1p
C2out S1 0 1p
C3out Cout 0 1p

.tran 1n 200n
.control run
plot S0 S1 Cout
.endc
.end
```

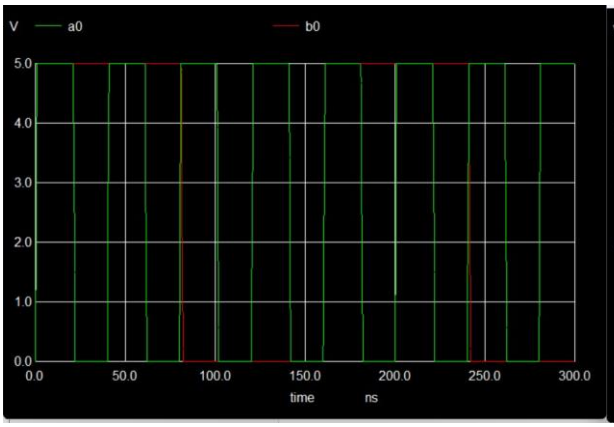
2. 2-Bit Xor - And

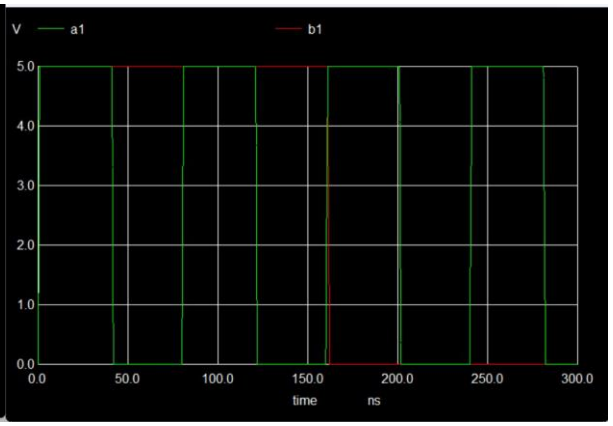
```
* 2bit XOR

* NAND Gate
.subckt nand2 A B Y Vdd Gnd
Mp1 Y A Vdd Vdd PMOS L=1u W=2u
Mp2 Y B Vdd Vdd PMOS L=1u W=2u
Mn1 n1 A Gnd Gnd NMOS L=1u W=1u Mn2
Y B n1 Gnd NMOS L=1u W=1u
.ends nand2

*xor using nand
.subckt xor2 A B Y Vdd Gnd
```

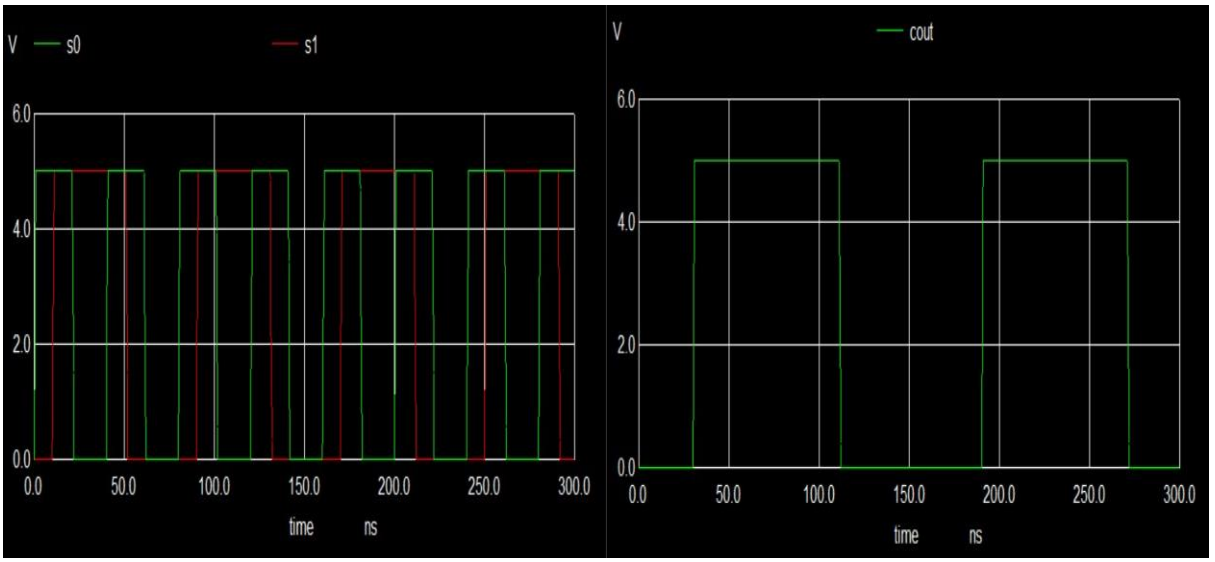
```
X1 A B n1 Vdd Gnd nand2
X2 A n1 n2 Vdd Gnd nand2
X3 B n1 n3 Vdd Gnd nand2
X4 n2 n3 Y Vdd Gnd nand2
.ends xor2

Vdd Vdd 0 DC 5
Vss Gnd 0 DC 0

* Input
Va0 A0 0 PULSE(0 5 0 1n 1n 10n 20n)
Va1 A1 0 PULSE(0 5 0 1n 1n 20n 40n)
Vb0 B0 0 PULSE(0 5 0 1n 1n 40n 80n)
Vb1 B1 0 PULSE(0 5 0 1n 1n 80n 160n)

* XOR
Xxor0 A0 B0 S0 Vdd Gnd xor2
Xxor1 A1 B1 S1 Vdd Gnd xor2

.tran 1n 200n
.control run
plot A0 B0 S0 A1 B1 S1
.endc
.end
```

GRAPHS : ng-SPICE

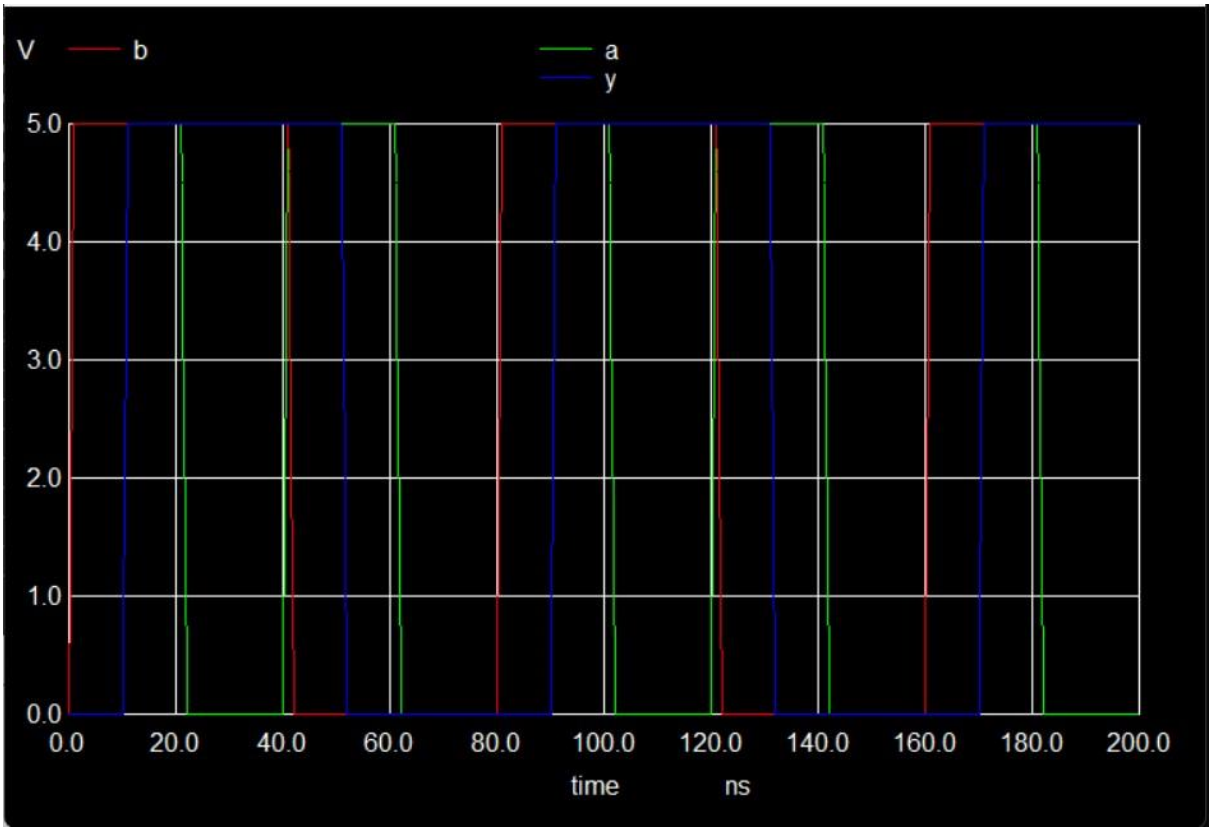
1. Adder :







2. Xor :



Optimisation (ALU)

1. Logic Delay , low voltage levels at output (high)

: After several simulations , I saw that the output waveform on output high logic was not satisfactory .
Realised the issue was with **Device sizing** .
Made all the **PMOS** transistors width **Twice** that of **NMOS**

2. Premature Discharge problem

: Observed that the charge was **decaying** even before the logic was going zero
Applied **Domino Logic** to avoid this issue , connected consecutive **NMOS** preserving the actual logic .

Bibliography

International Journal of Latest Technology in Engineering, Management & Applied Science (IJLTEMAS)

D. Garg and M. K. Rai / IJECCT 2012, Vol. 2 (4)

Layout Design of a 2-bit Binary Parallel Ripple Carry Adder Using CMOS NAND Gates with Microwind
Arif Ul Alam1 , Nishatul Majid1 and S. K. Aditya2