VLSI CIRCUIT DESIGN
COMPE-572

PROJECT-3

SAN DIEGO STATE UNIVERSITY

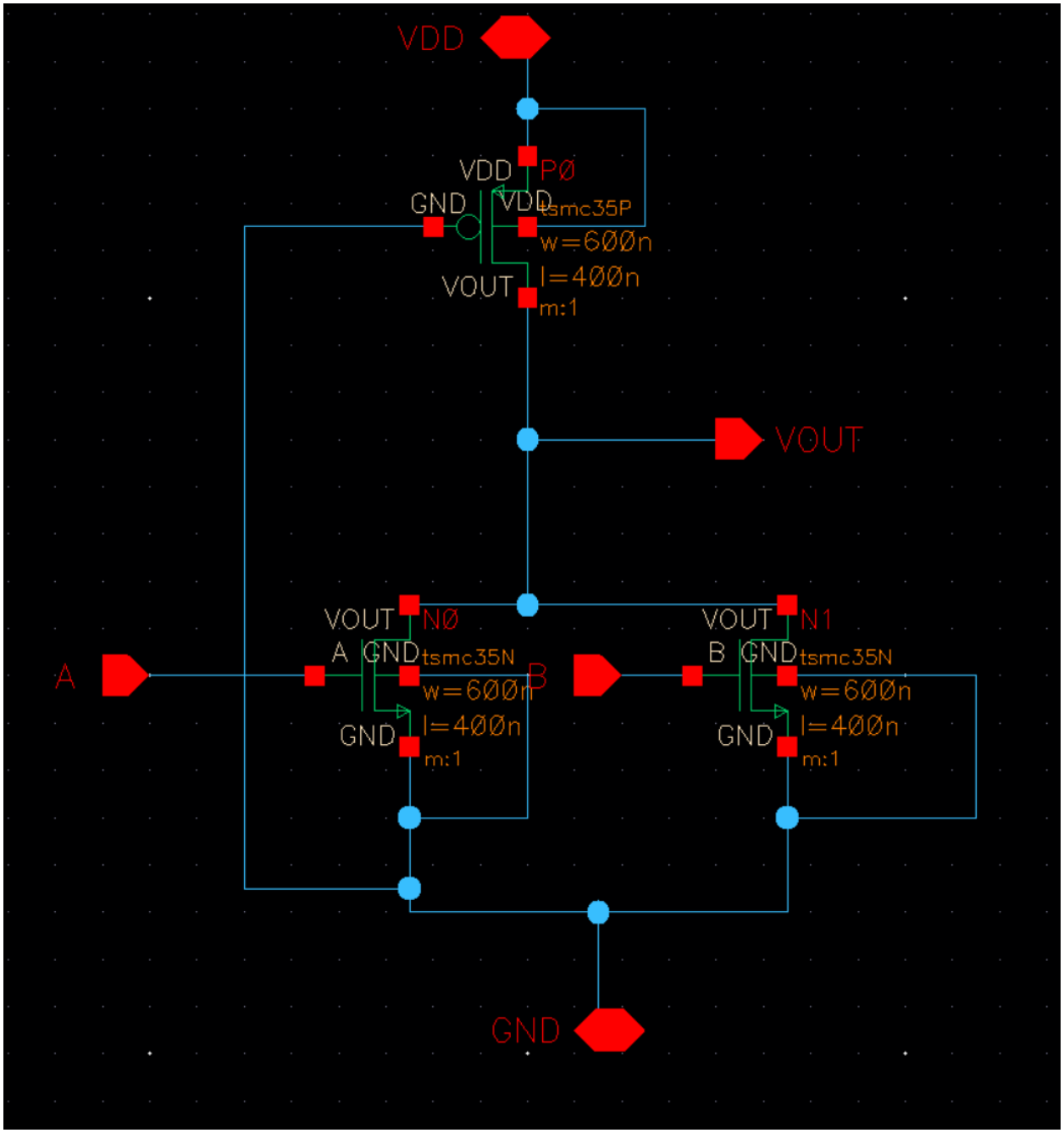
Preparations

In this project, we have chosen the TSMC04_4M_2P as the transistor with Lmin = 600nm technology. The standard supply voltage for this transistor is 3.3V and has 5 routing metal layers.

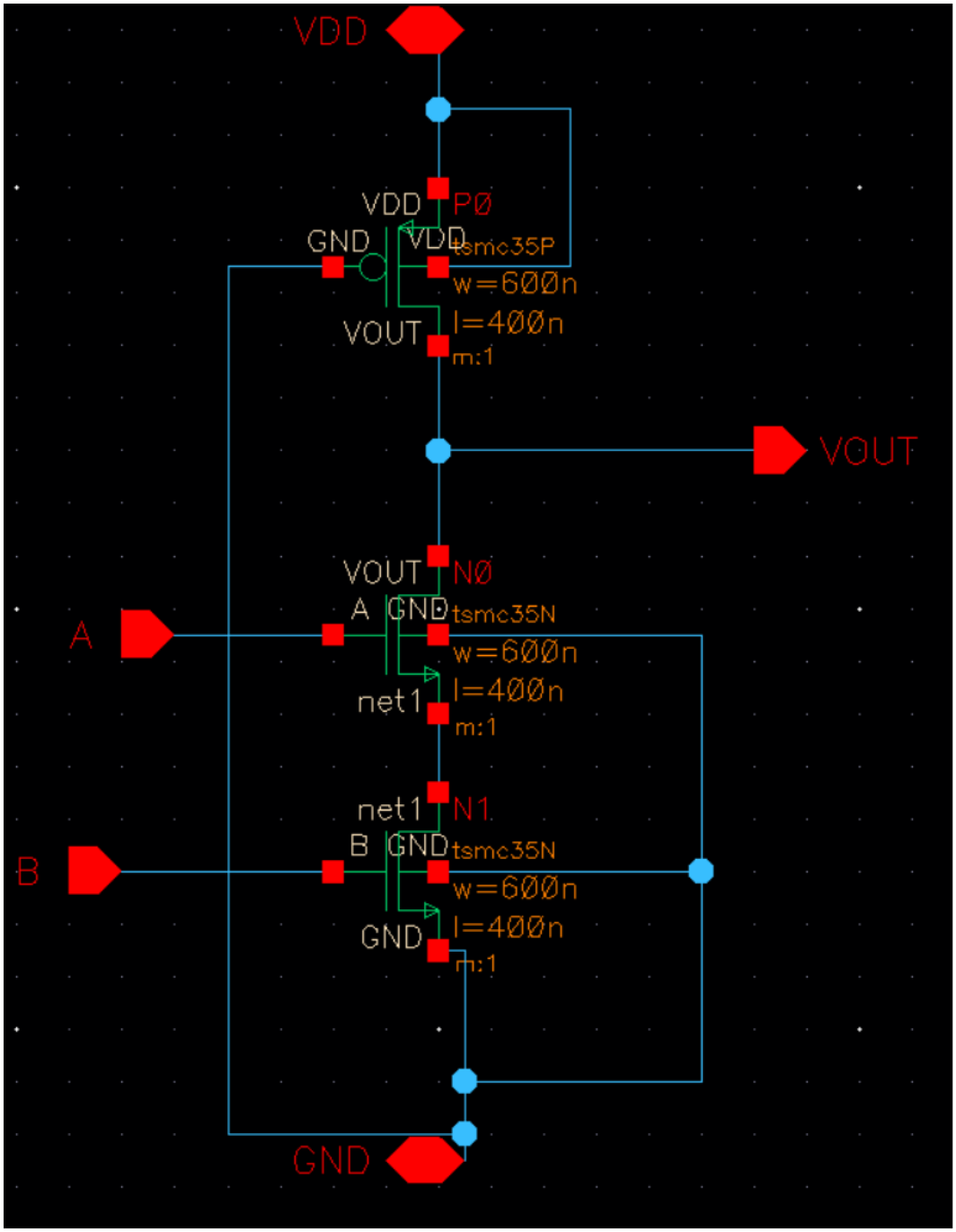
Step 1. 2-Input Nand and NOR Gate with Psuedo-Nmos Technique

In the first step, we implemented the Psuedo-Nmos technique for a NAND and NOR gate with 2 inputs. To implement the technique, we would reduce the PMOS network to 1 PMOS and connect its gate to ground. Then, we would implement the CMOS technique on only the NMOS network. Lastly, the testbenches for our gates will ALSO be used for the pass transistor logic simulations, so we omitted the test bench screenshots for redundancy.

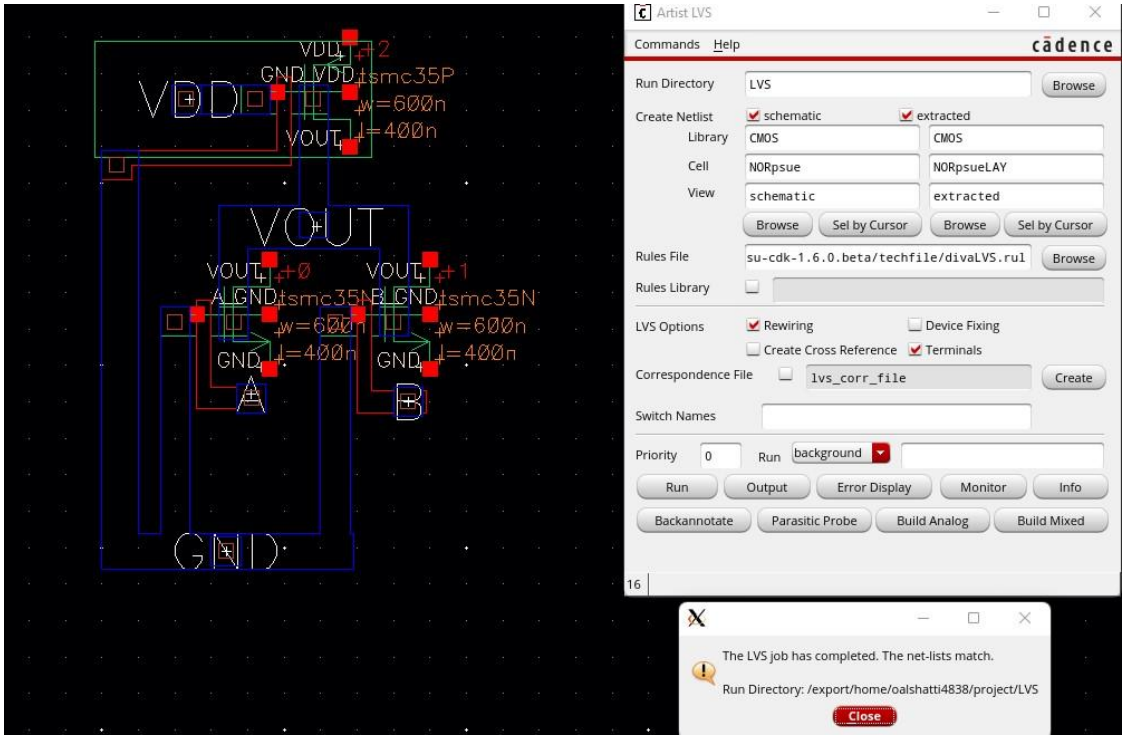
NOR pseudo NMOS Schematic



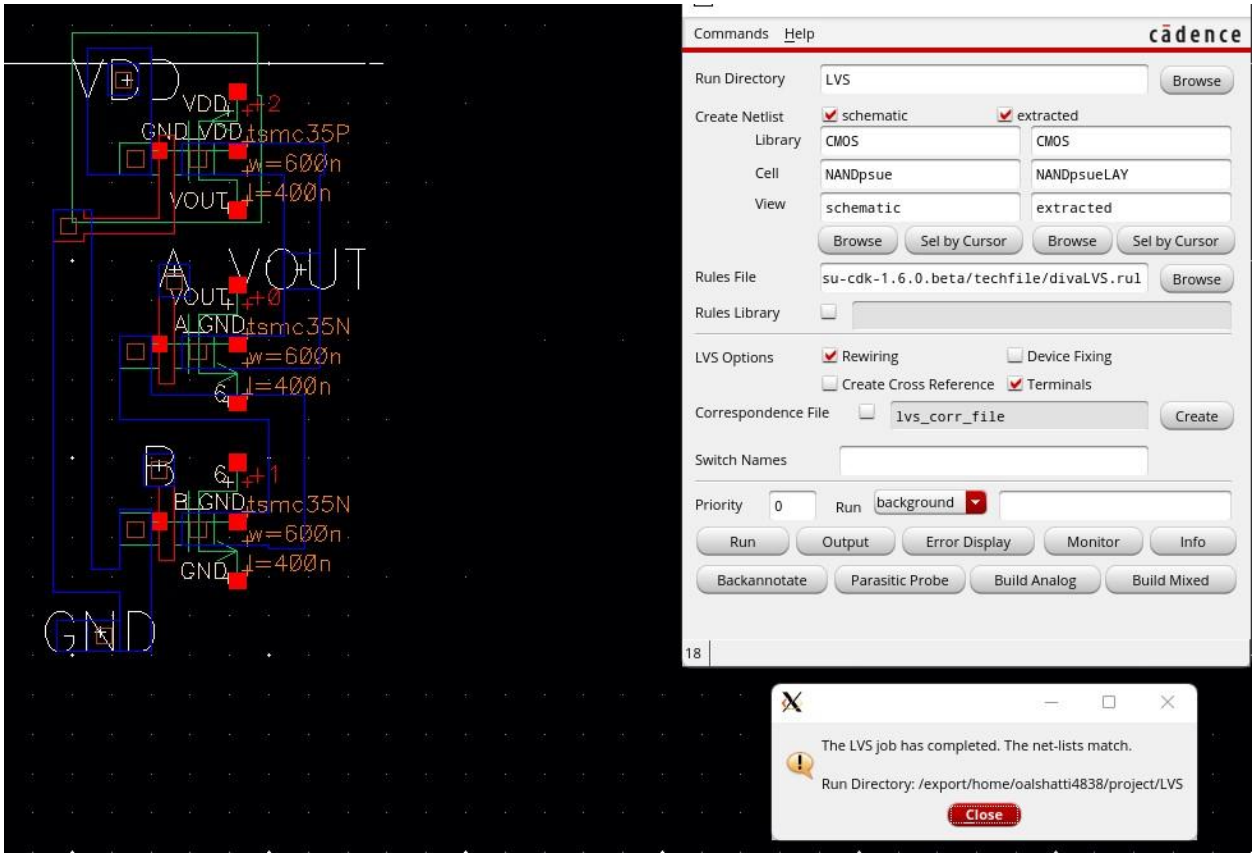
NAND pseudo NMOS Schematic



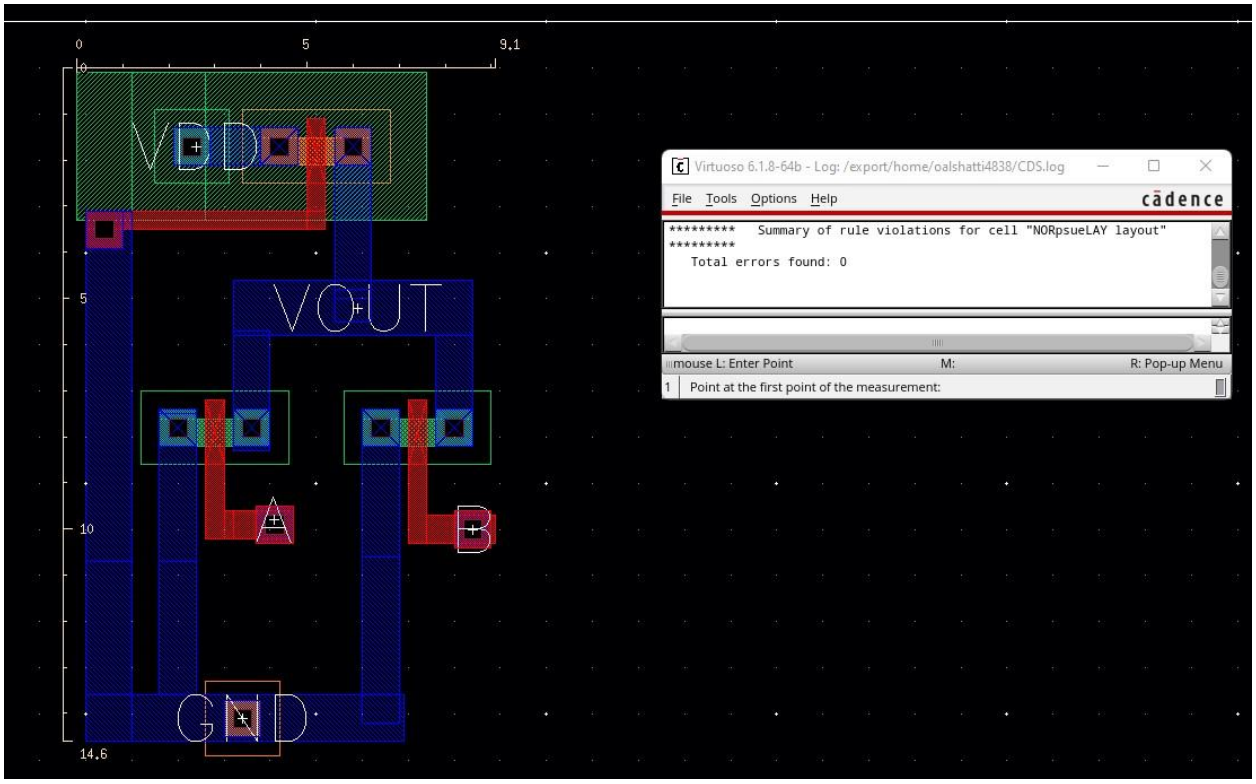
LVS NOR pseudo NMOS



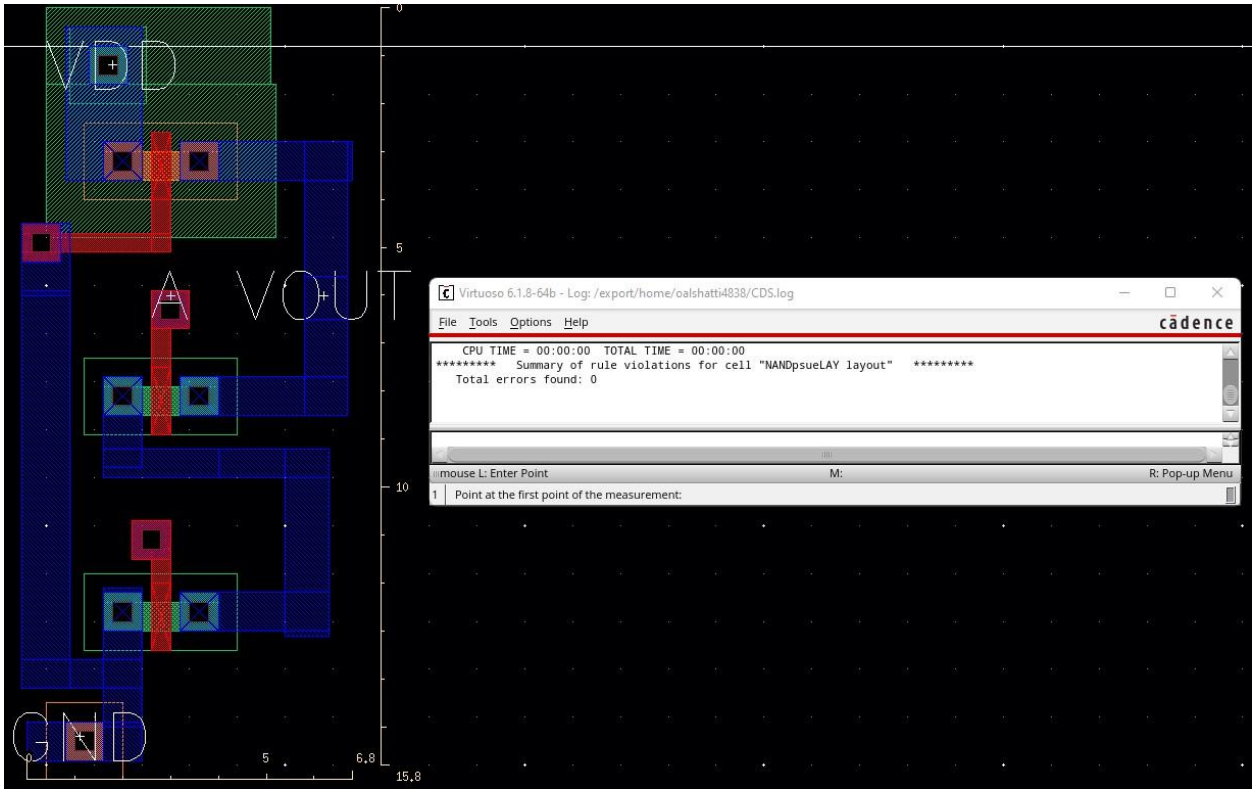
LVS NAND pseudo NMOS



LAYOUT NOR **pseudo NMOS with DRC**



LAYOUT NAND **pseudo NMOS with DRC**

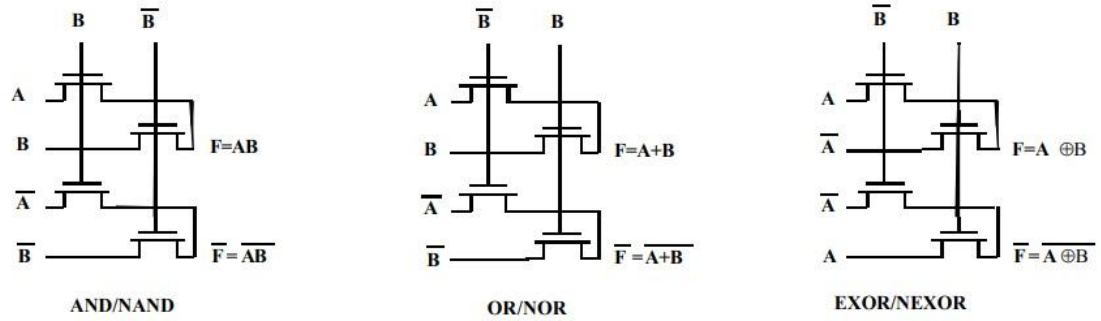


Pseudo NMOS NAND Dimensions: 9.1μmX14.6μm**. Nand Area=**132.86 μm^2

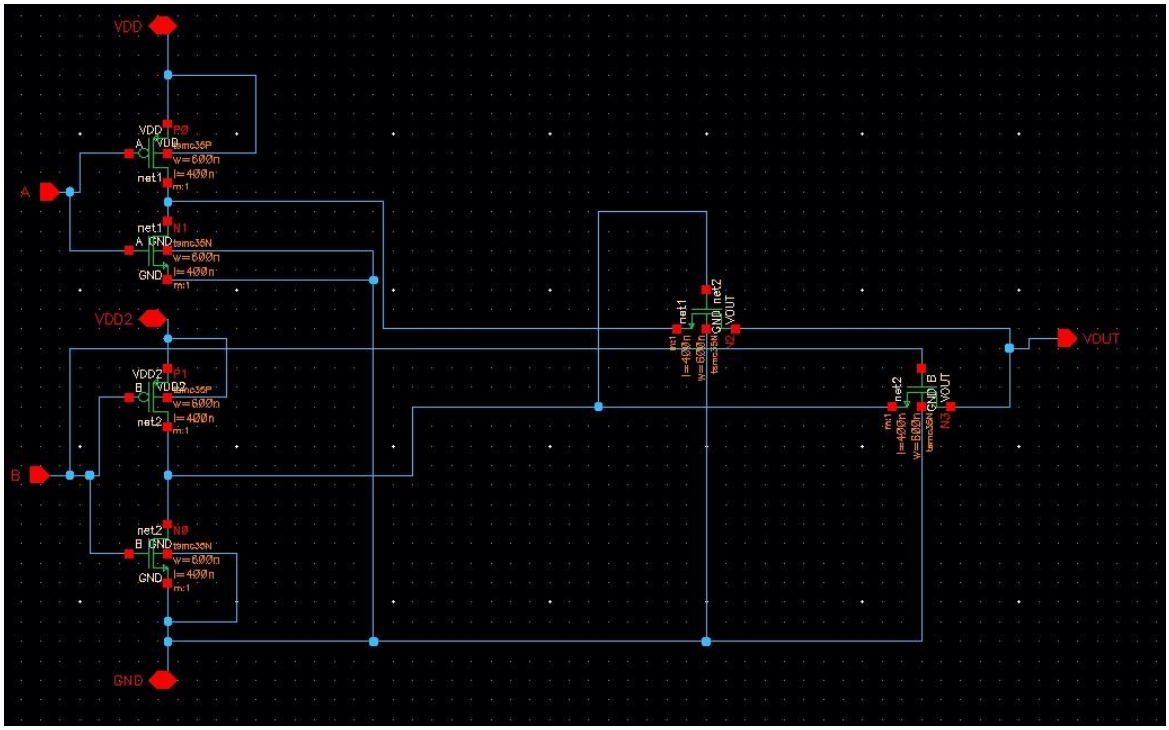
Psuedo NMOSNOR Dimensions: 6.8μmX15.8μm**. NOR Area=**104.44 μm^2

Step 2. 2-Input Nand and NOR Gate with Pass Transistor Logic Technique

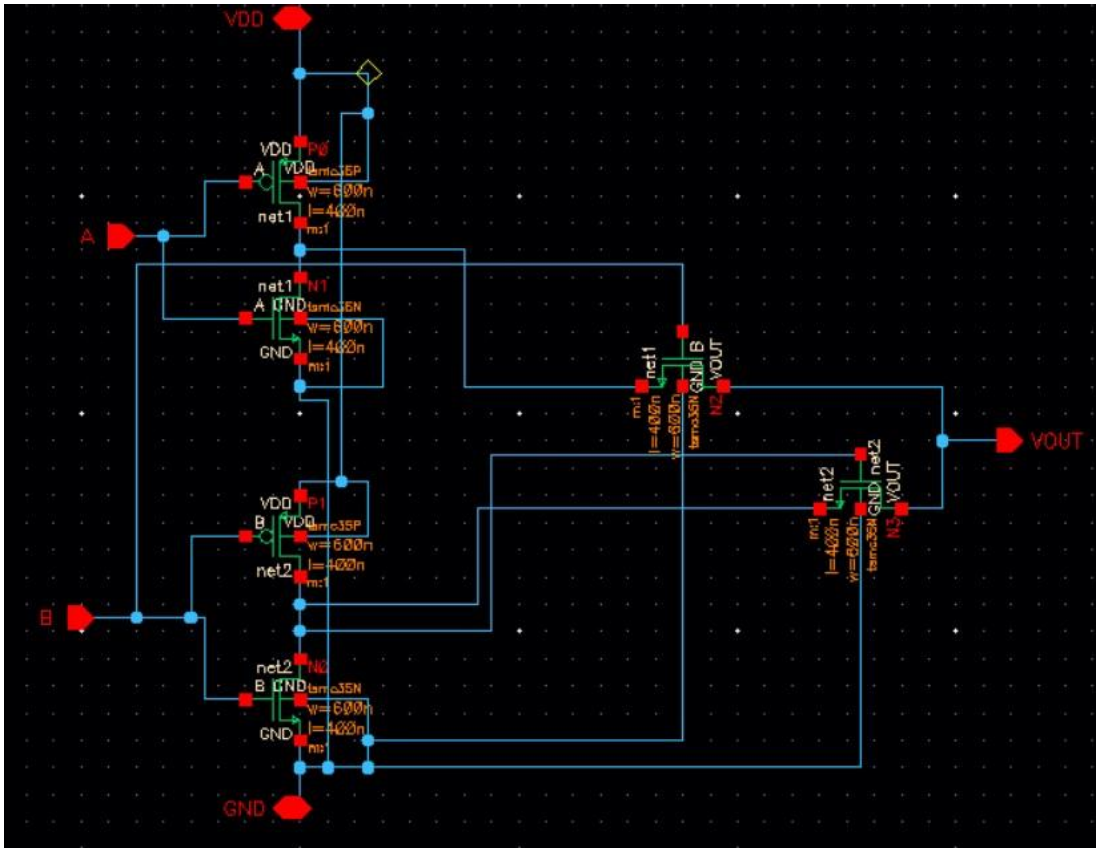
In this step, we implemented the 2-Input NOR and NAND gate using the Pass Transistor Logic Technique (PTL). To implement it, we would first need to have inverters that correspond to our inputs since both NAND and NOR gates need A' and B' to be implemented. Next, used referenced the image below to compute the VOUT.



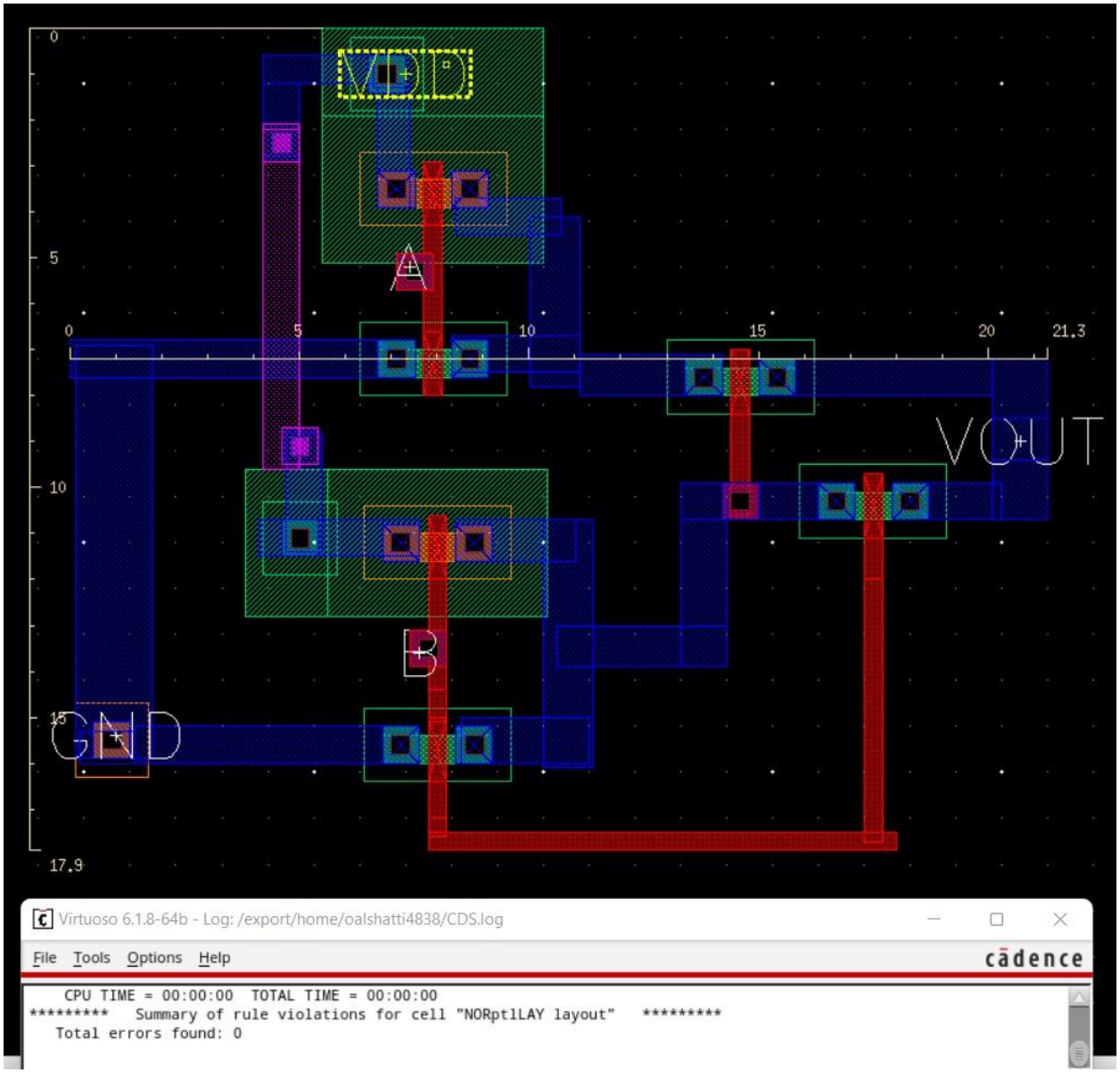
NOR Pass Transistor Logic schematic



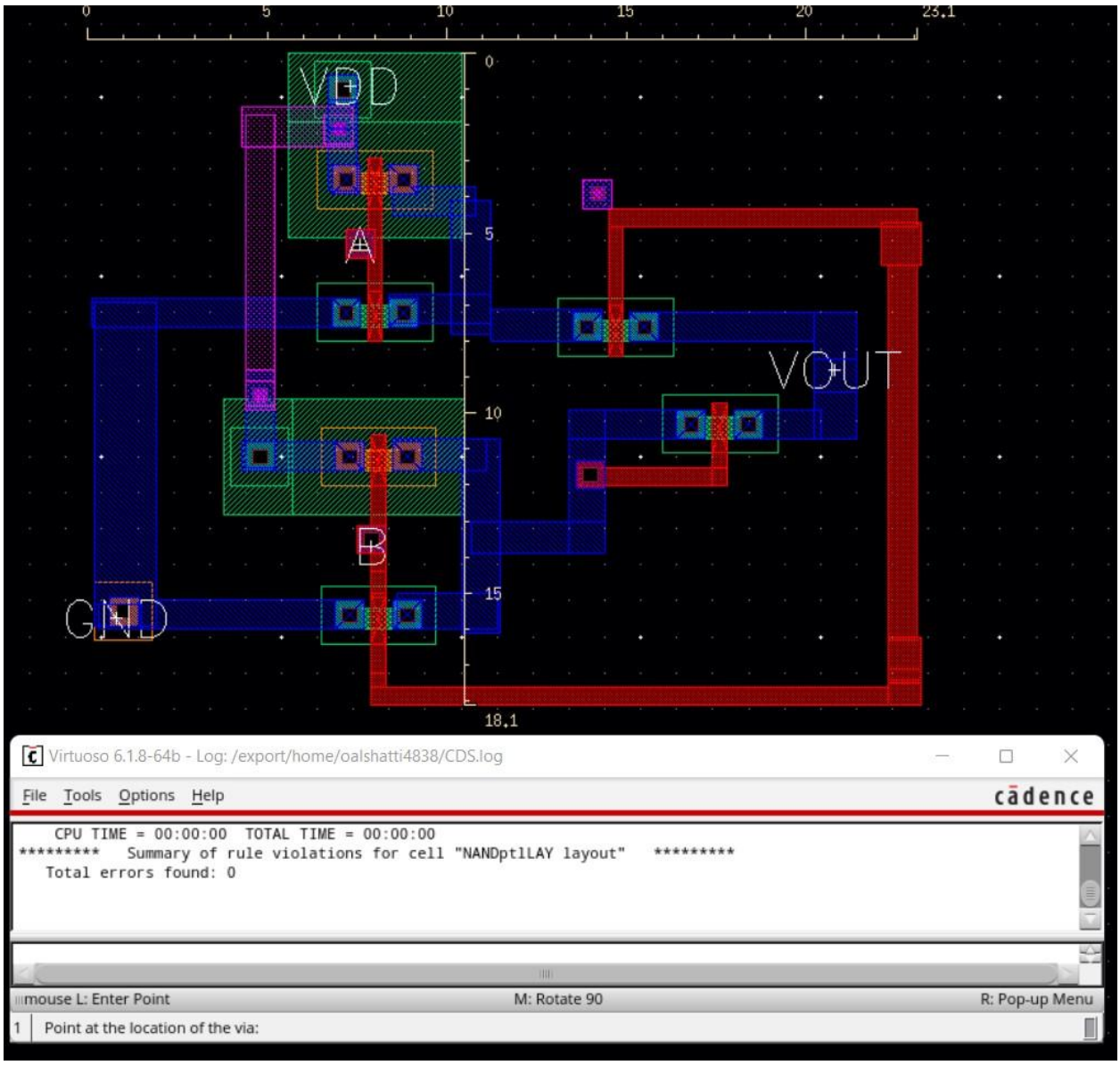
NAND Pass Transistor Logic schematic


VDD
A
B
GND
VOUT
net1
net2
N1
N2
P0
P1
w=600n
l=400n
m:1

NOR Pass Transistor Logic Layout with DRC



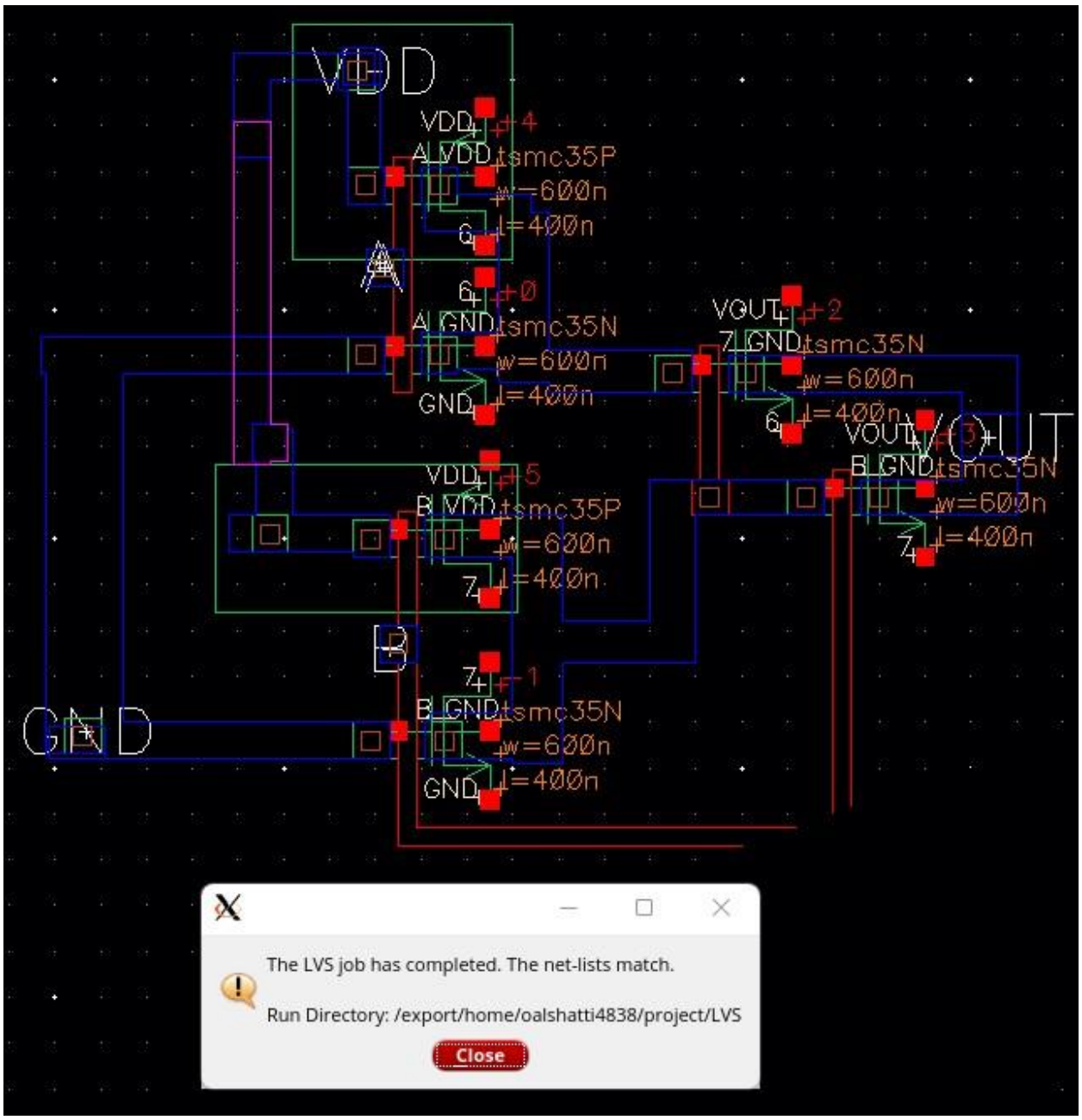
NAND Pass Transistor Logic Layout with DRC



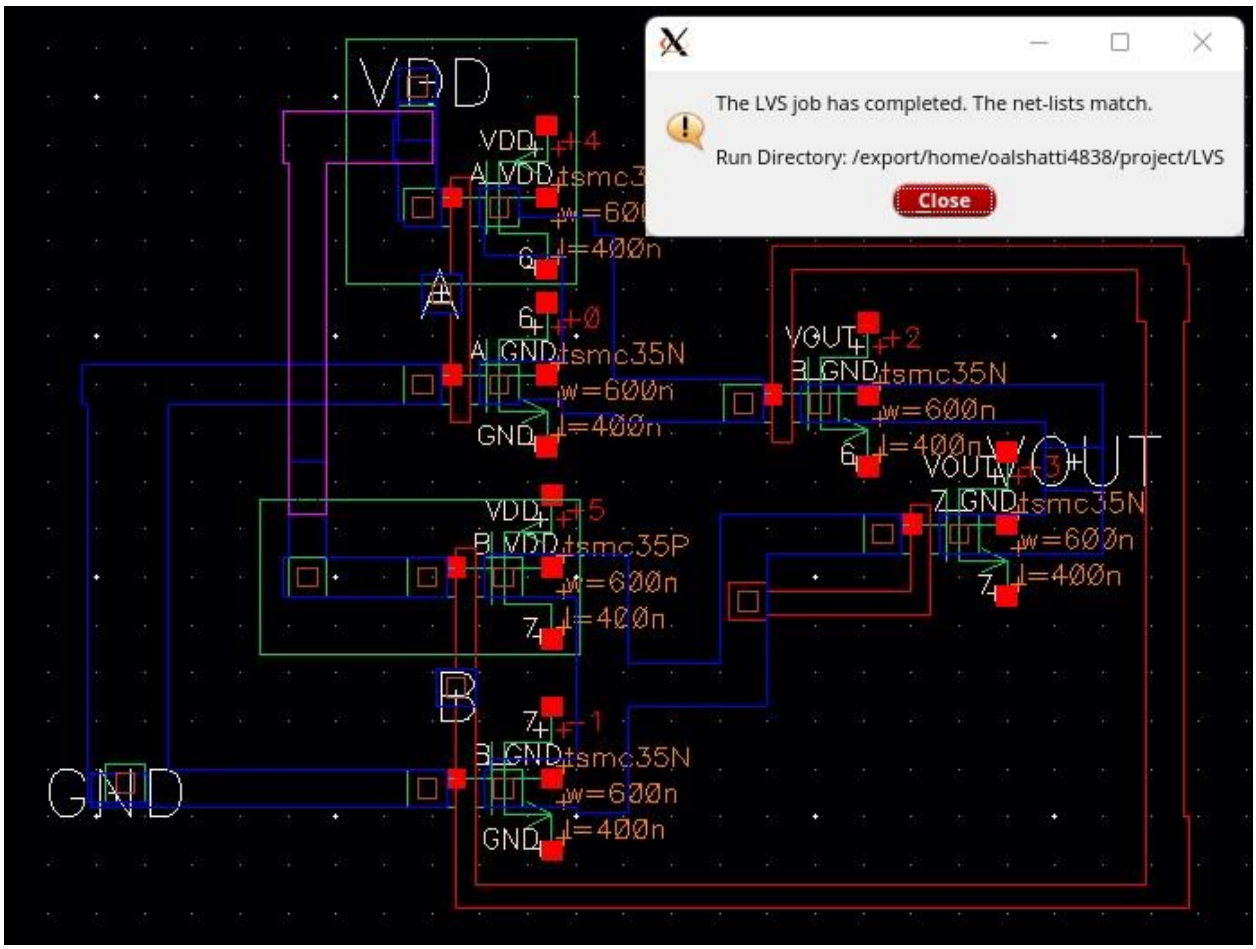
PTL NAND Dimensions: 21.3μmX17.9μm **Nand Area=**381.2**7** μm^2

PTL NOR Dimensions: 23.1μmX18.1μm **NOR Area=** 418.11 μm^2

NOR Pass Transistor Logic LVS



NAND Pass Transistor Logic LVS

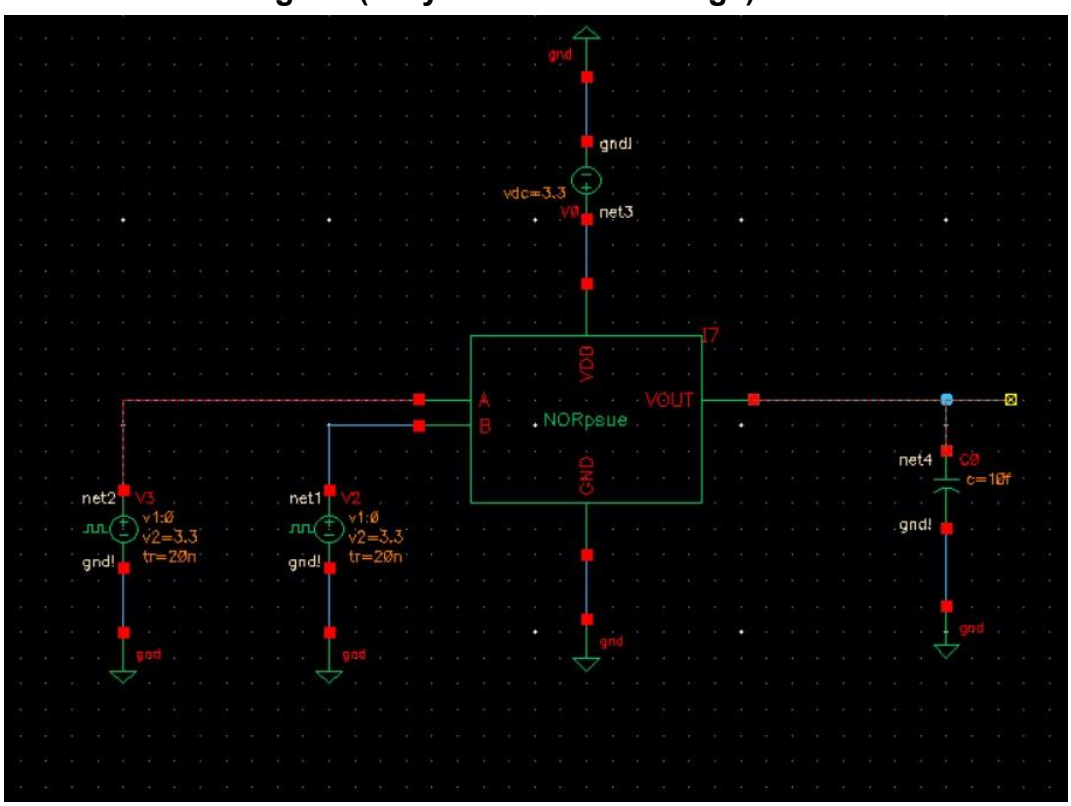


Step 3. Current

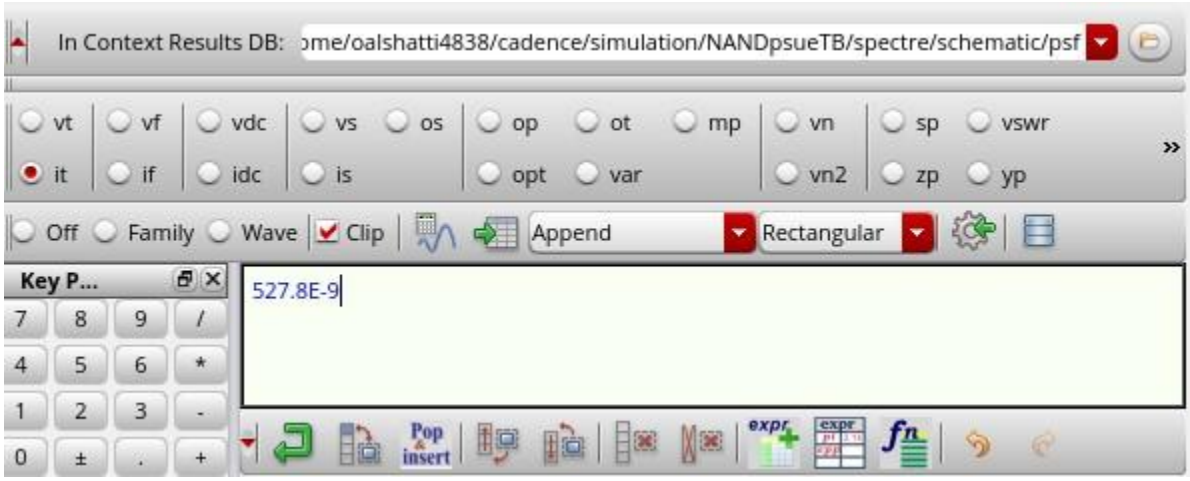
Using the ‘IT’ function found in ADE-L’s calculator, we were able to find the average current for a pulsing voltage of 1us period with a rise and fall time of 20ns. To start, we needed to save the negative terminal of the VDD and the VOUT. Next, we would run a transient simulation to obtain the voltage data of these terminals. Lastly, we would use the built-in function as mentioned above to get the average current and multiply it with VDD (3.3V) to get the average power.

To measure the propagation delay, we placed markers that correspond to VDD/2 in both input and output and subtracted them to get the delay. In the below transient simulations, Net2 was the input and Net4 was the output. Furthermore, we used Project2’s submission for our CMOS delays.

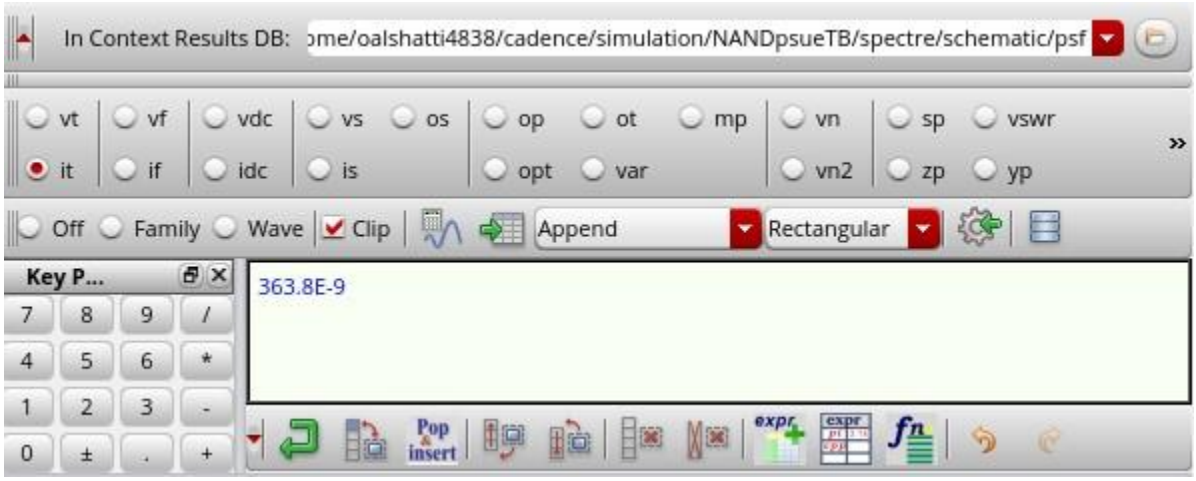
Test Bench for all gates(Only the Cell will change)



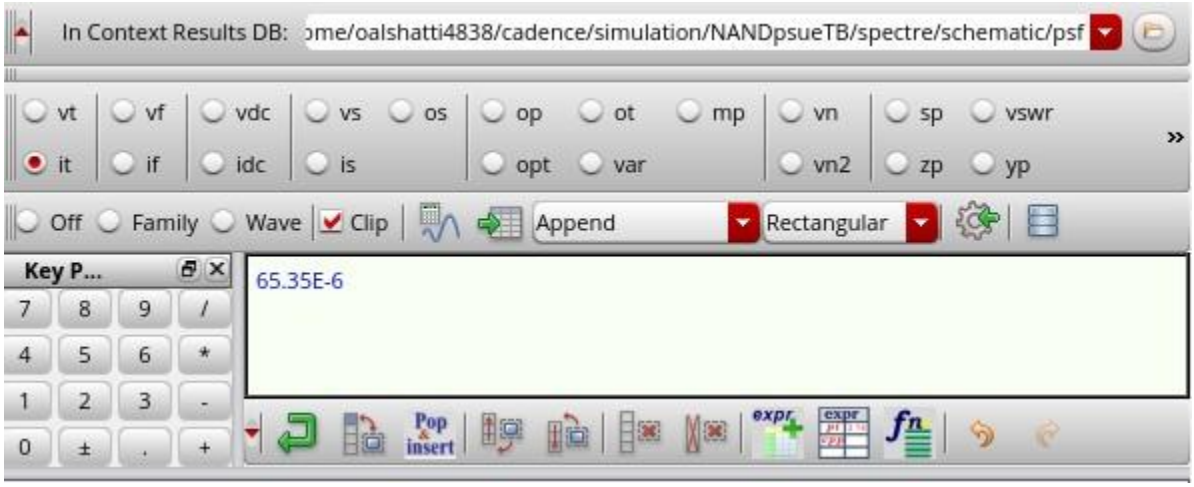
NAND CMOS current calculation using the built in function



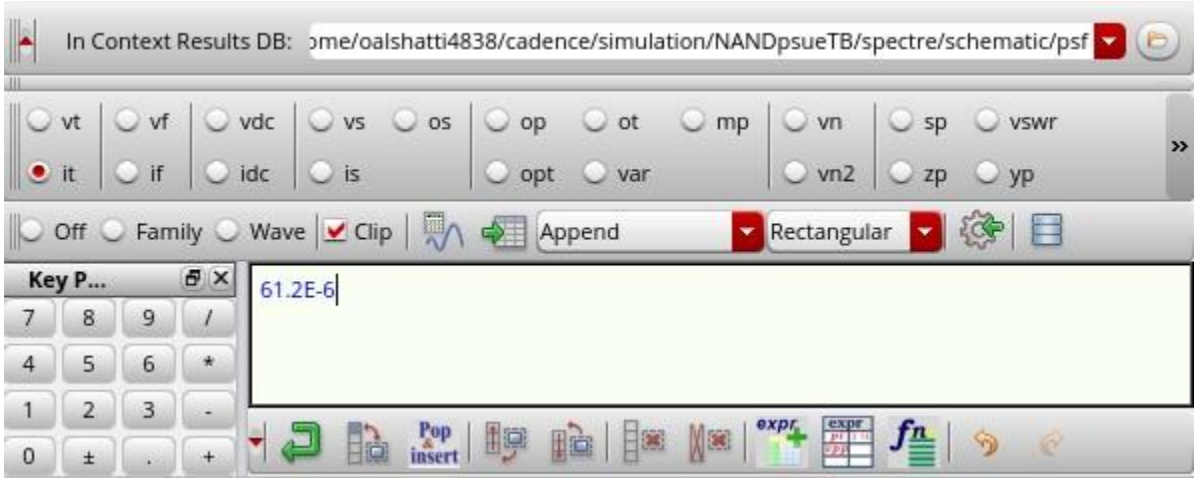
NOR CMOS current calculation using the built in function



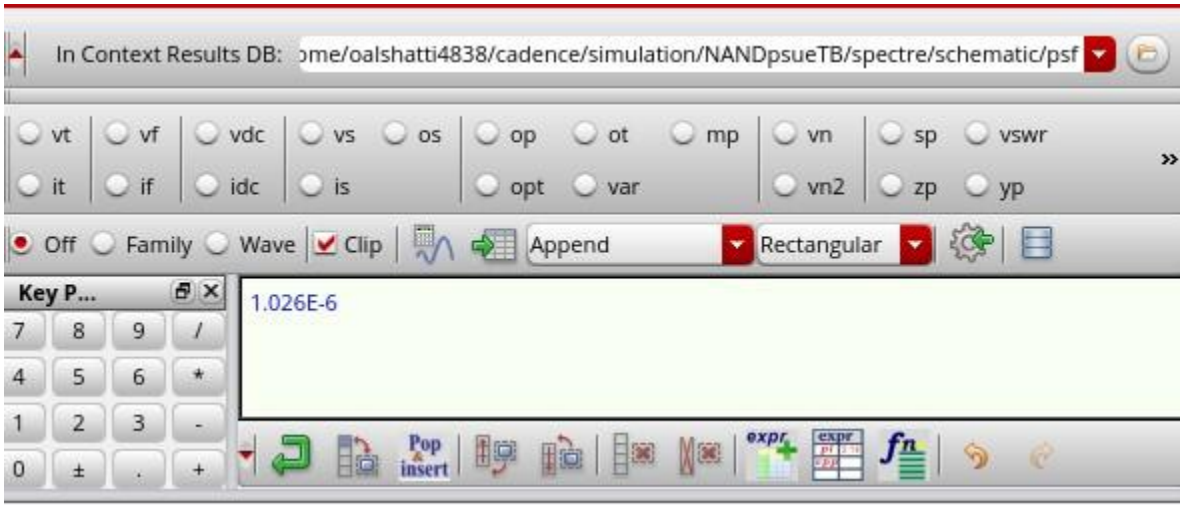
NOR Pseudo-NMOS current calculation using the built in function



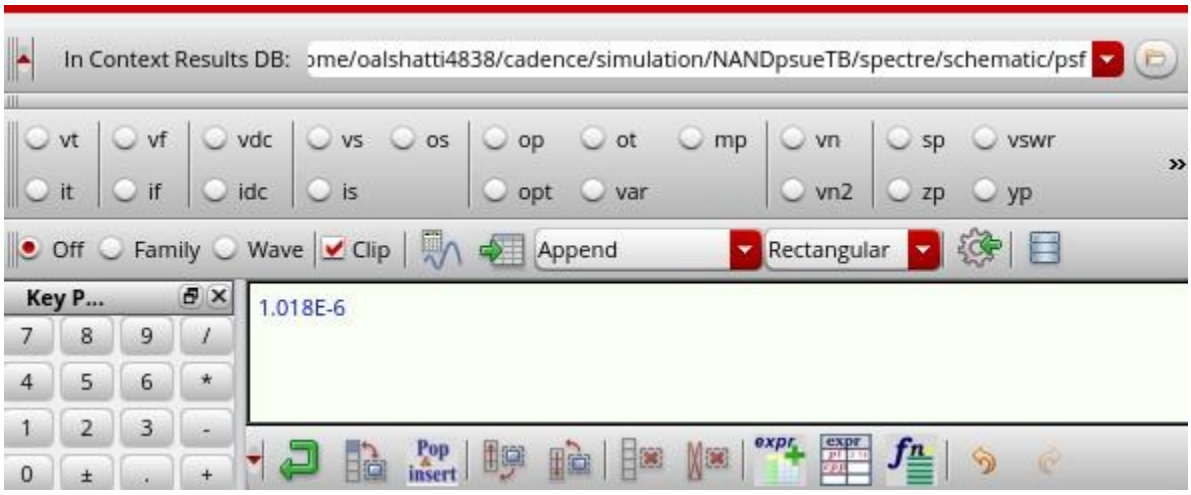
NAND Pseudo-NMOS current calculation using the built in function



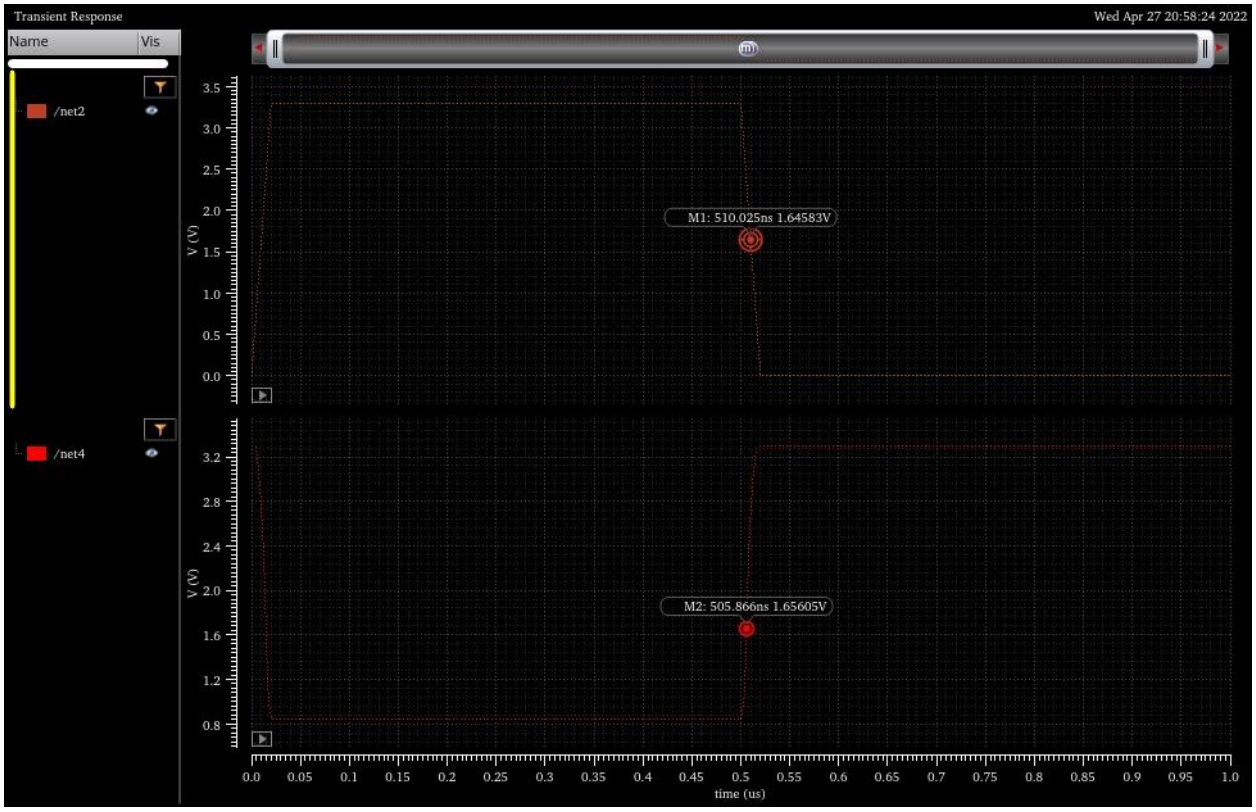
NOR Pass Transistor Logic current calculation using the built in function



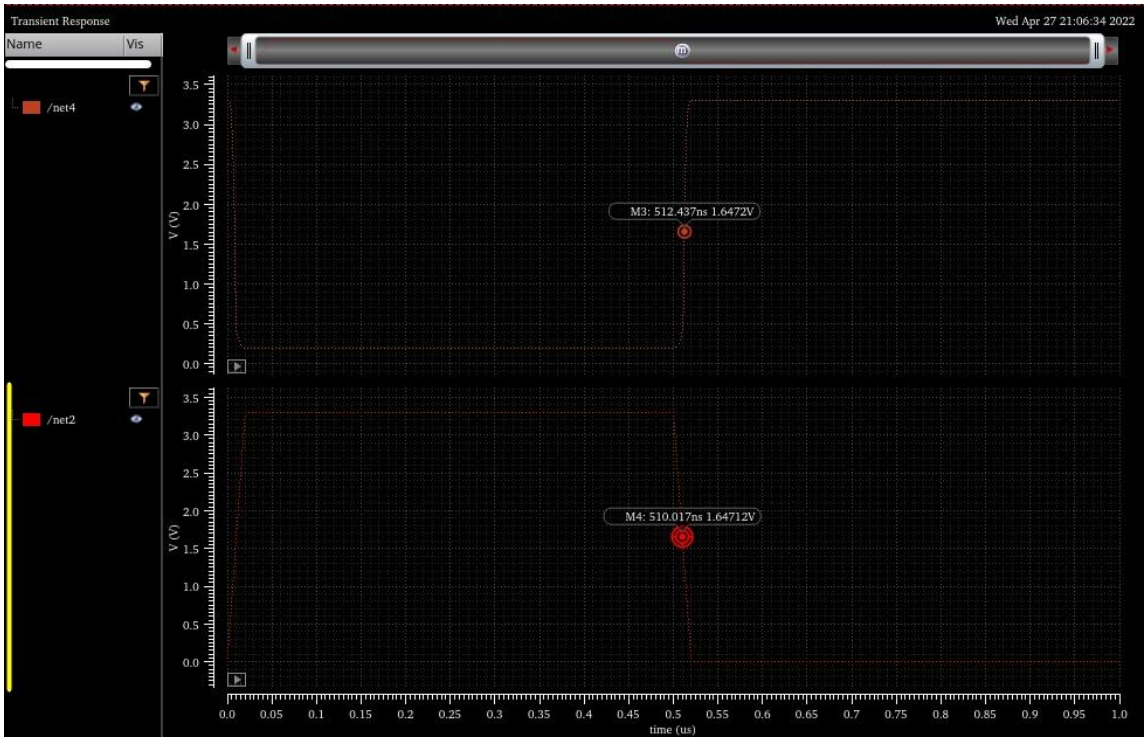
NAND Pass Transistor Logic current calculation using the built in function



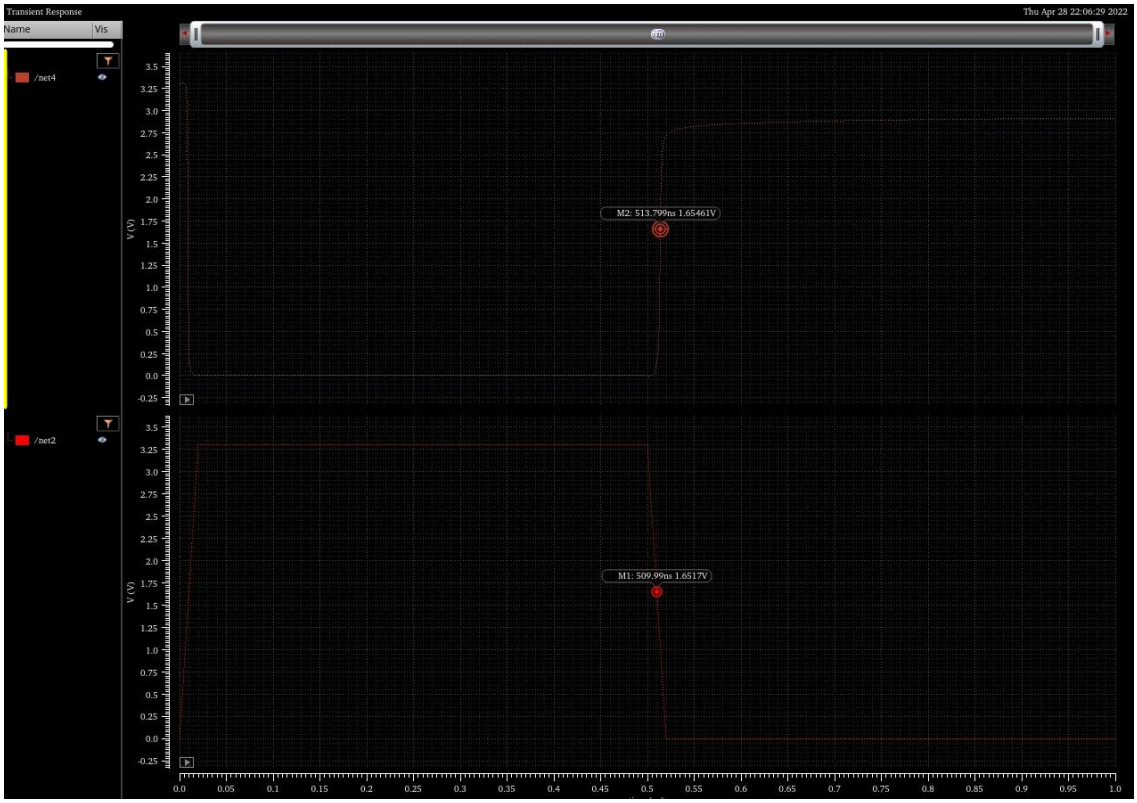
NAND delay Psuedo-NMOS



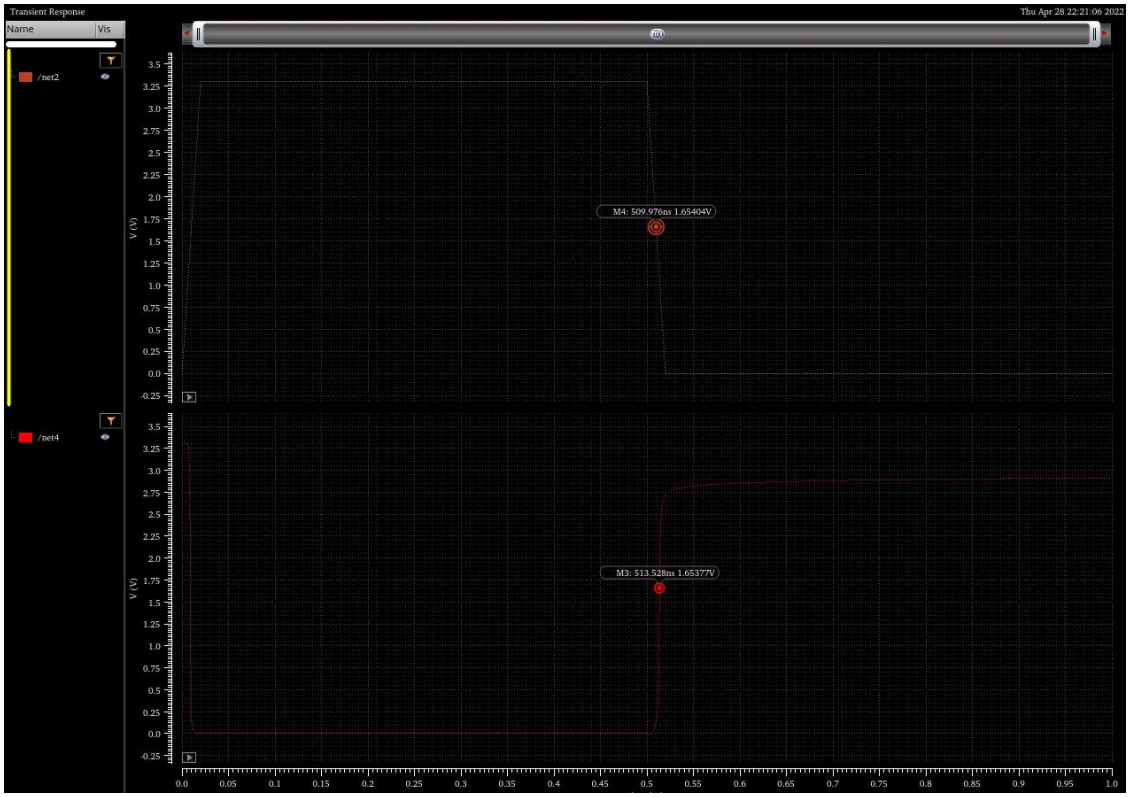
NOR delay Pseudo-NMOS



NAND delay PTL



NOR delay PTL



Power and Delay Table:

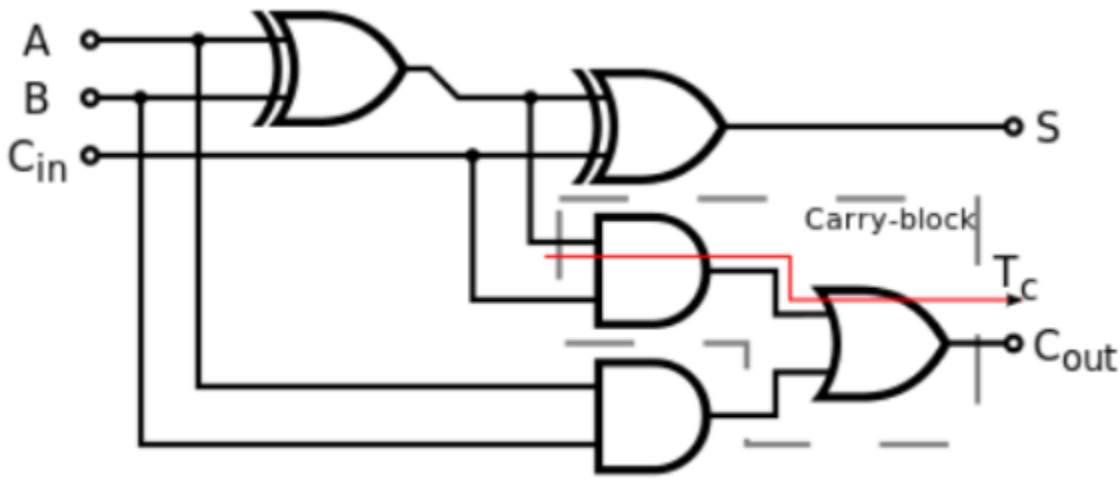
Transistor	Power	Delay
CMOS NAND	1.74 μW	351ps
CMOS NOR	1.2 μW	232.09 ps
Psuedo NAND	20.2 mW	415.9 ps
Psuedo NOR	21.6 mW	242 ps
PTL NAND	3.36 μW	380 ps
PTL NOR	3.36 μW	395.2 ps

Step 4. Specification summary

Transistor	Area	Power	Delay
CMOS NAND	120 $(\mu m)^2$	1.74 μW	351ps
CMOS NOR	133.44 $(\mu m)^2$	1.2 μW	232.09 ps
Psuedo NAND	132.86 $(\mu m)^2$	20.2 mW	415.9 ps
Psuedo NOR	104.44 $(\mu m)^2$	21.6 mW	242 ps
PTL NAND	381.27 $(\mu m)^2$	3.36 μW	380 ps
PTL NOR	418.11 $(\mu m)^2$	3.36 μW	395.2 ps

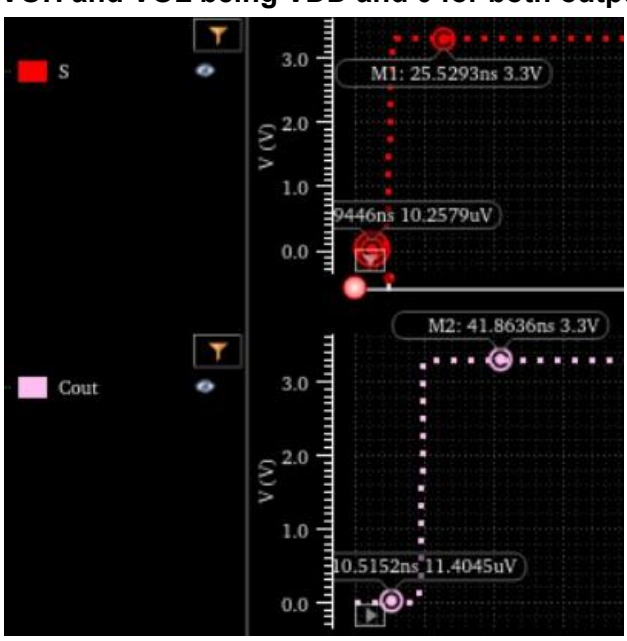
Step 5. 2-bit full adder

Our 2bit full adder was completed with the help of the following schematic:

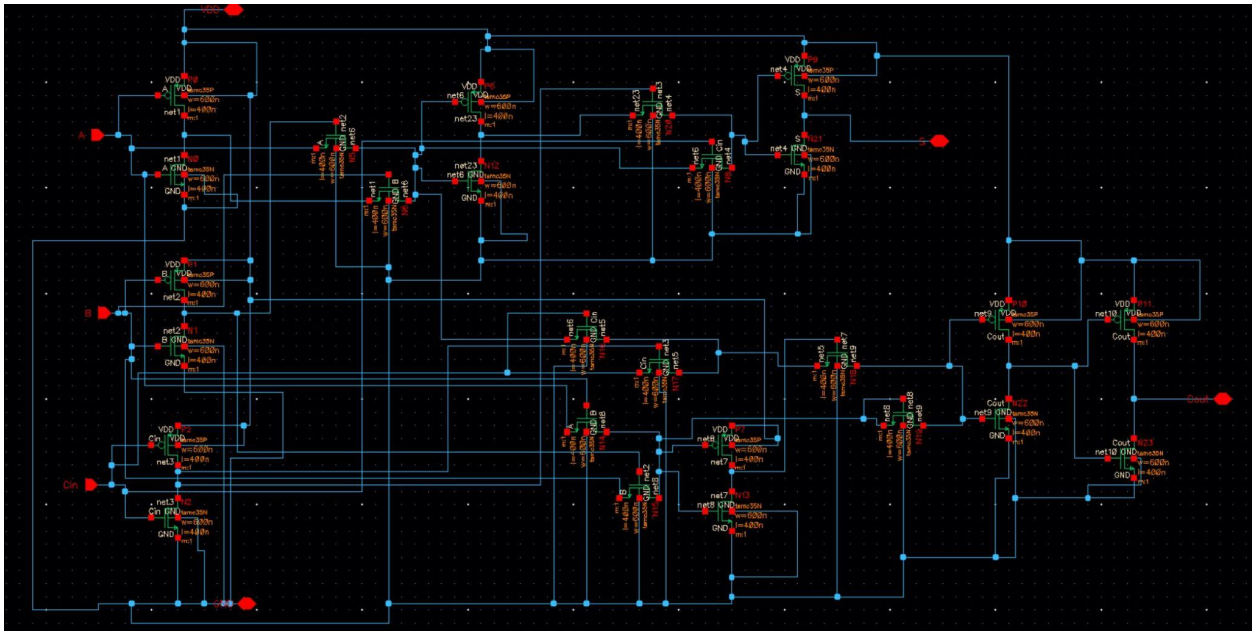


Before we started our design steps, we had to pick a logic family we were all comfortable with, so we picked **PTL**. One of the main problems of PTL is that VOH can't reach VDD and would reach VDD-Vtn. To reach VDD in S, I replaced the XOR gate before S to be an XNOR gate. Next, I would put an inverter which would drive it to VDD or 0 according to the inputs. To reach VDD in Cout, I would insert two inverters that would return the original value of the OR gate. This would allow us to start the schematic implementation.

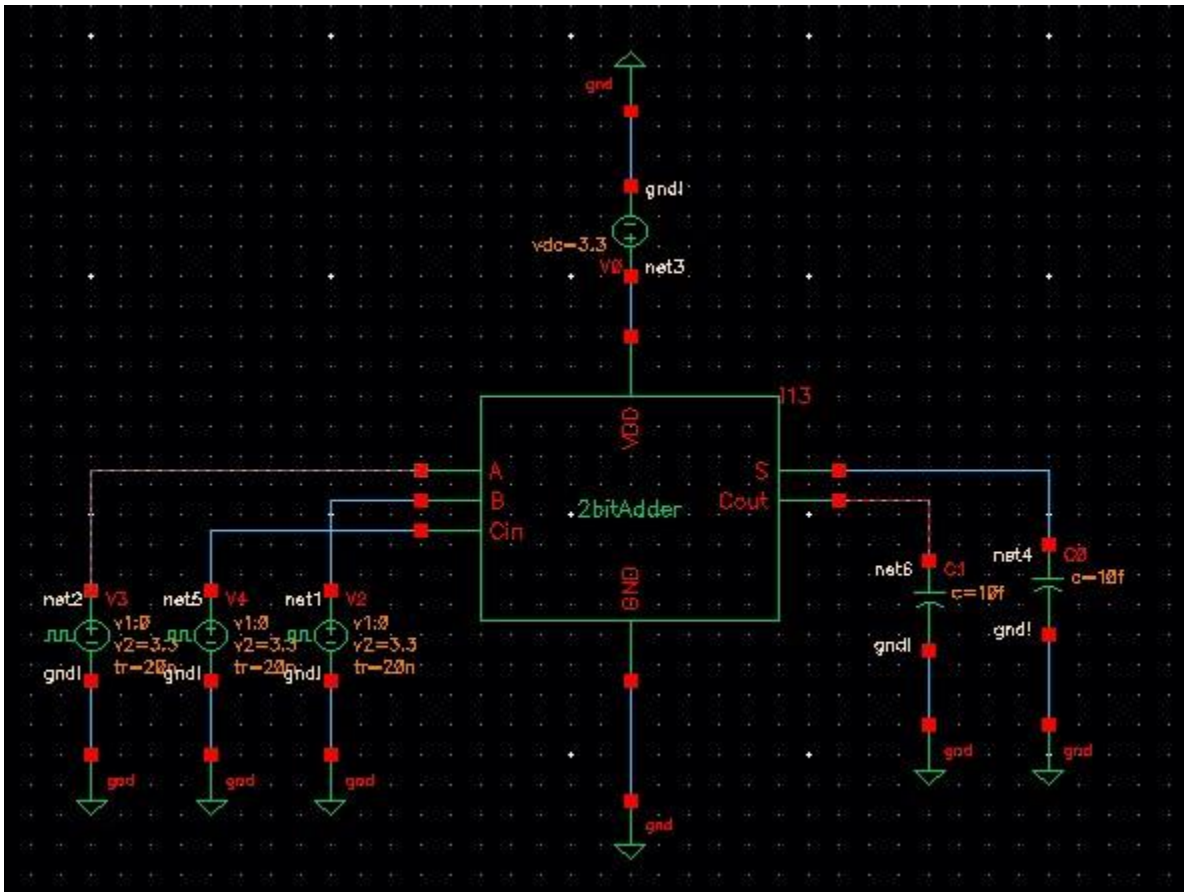
VOH and VOL being VDD and 0 for both outputs



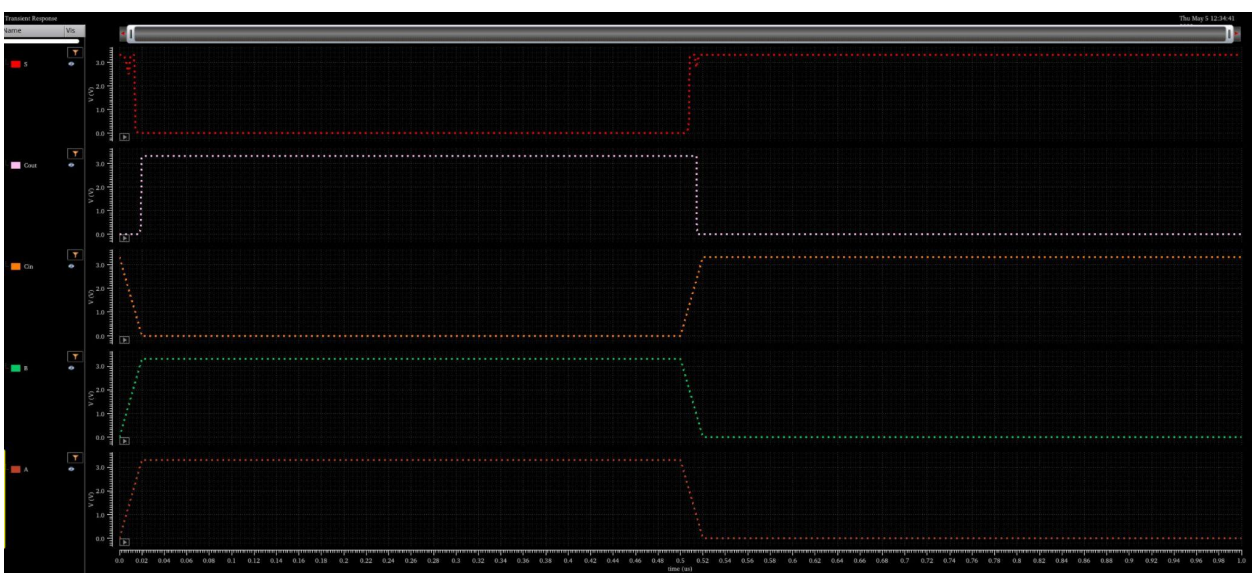
2-Bit Adder Schematic



After completion of the schematic above with no errors we could go ahead and test with the following(**Note: Another Test Bench will be used for power and delay calculations**):

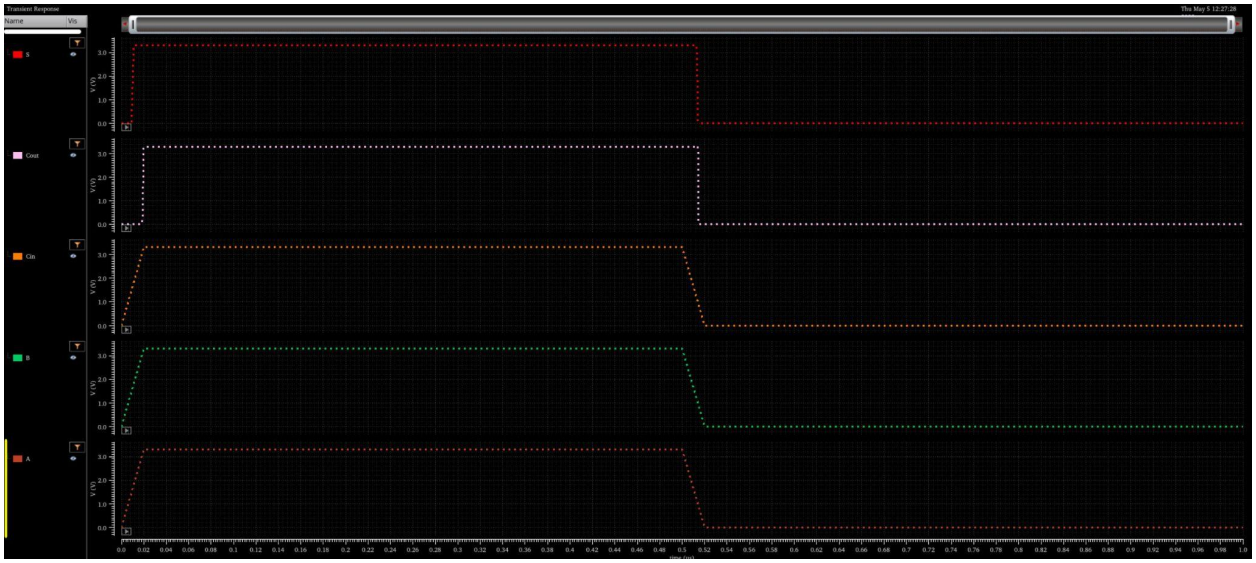


Waveforms which represent functionality of our 2 bit full adder:
001-110

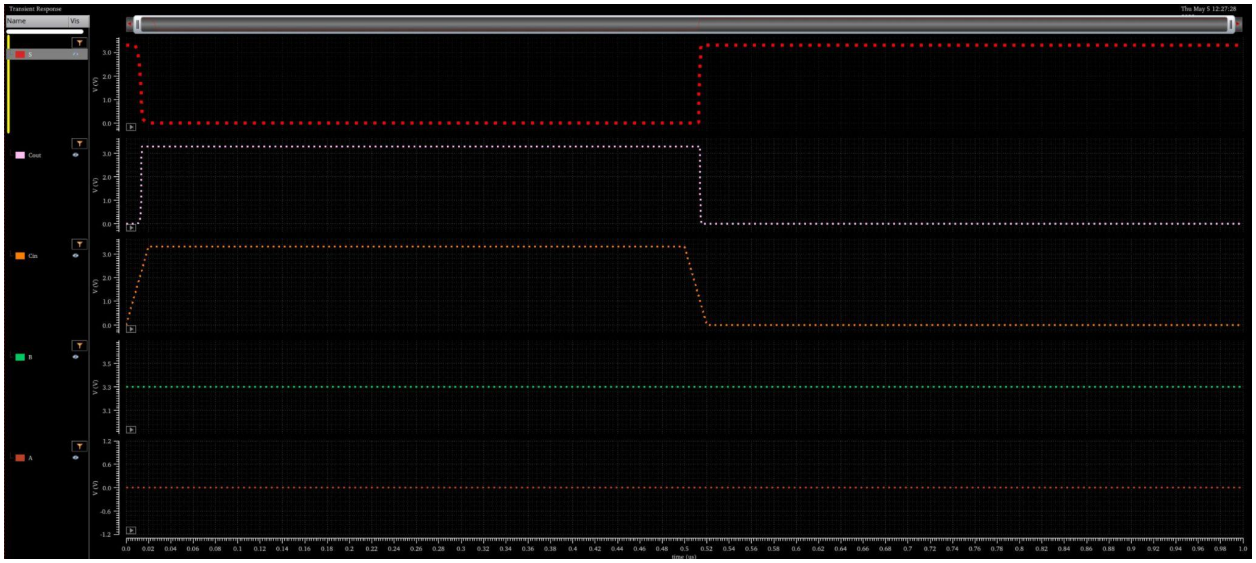


101-100

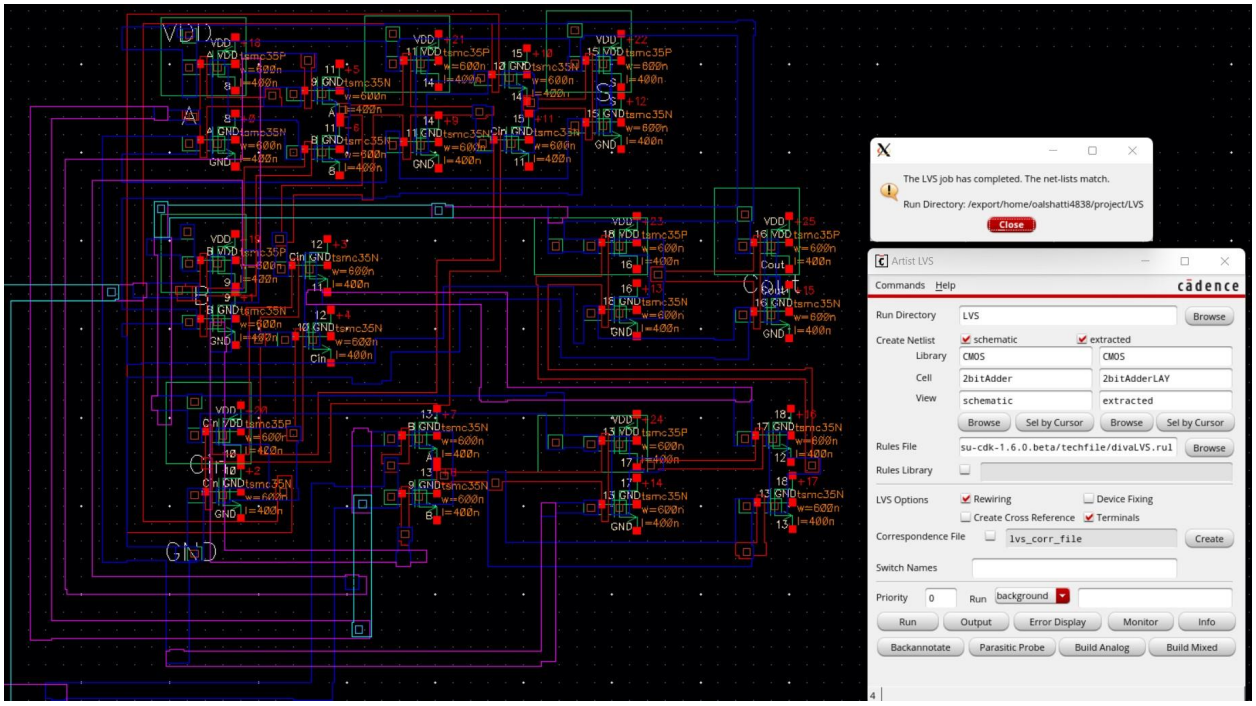
111-000



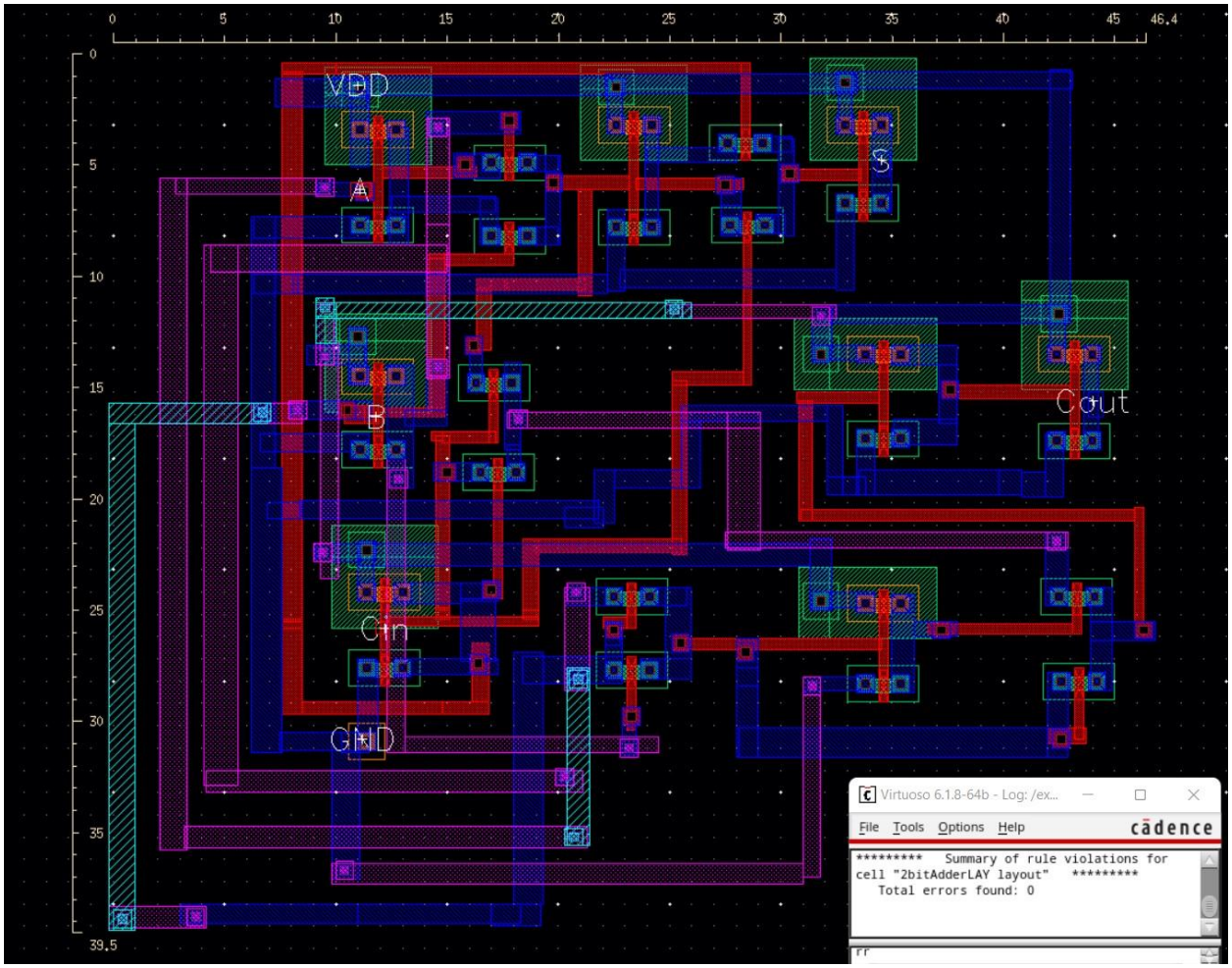
010-011



2bit adder LVS



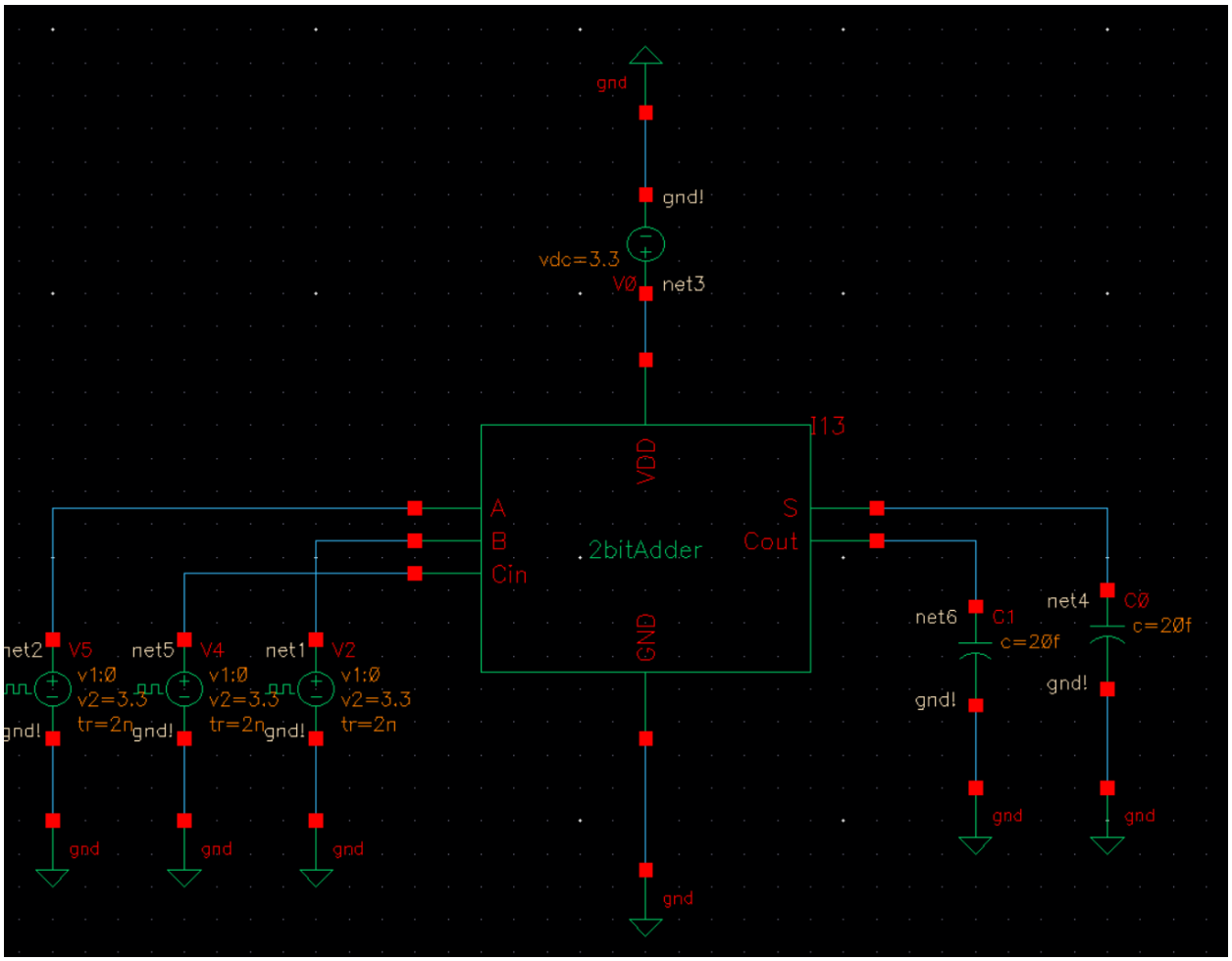
2bit adder DRC



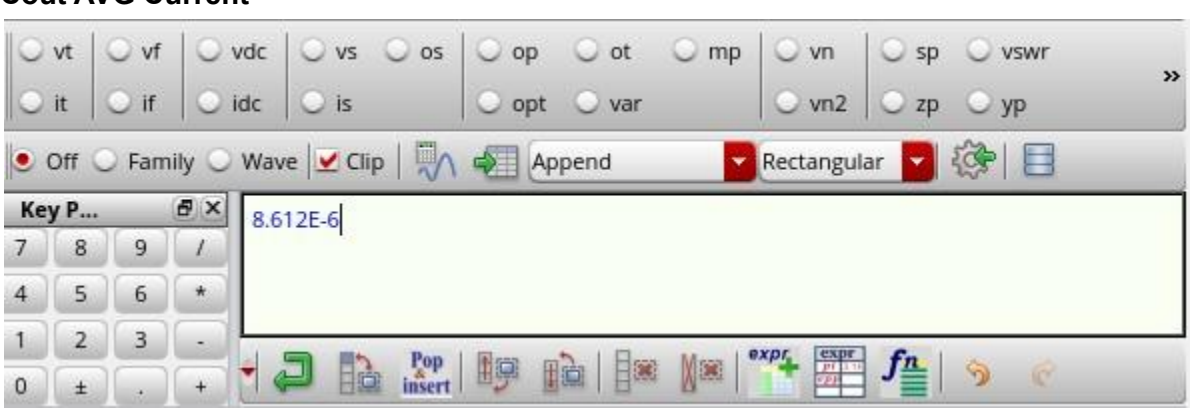
Dimensions: 39.5µm X 46.4µm

Area: 1832.8 μm^2

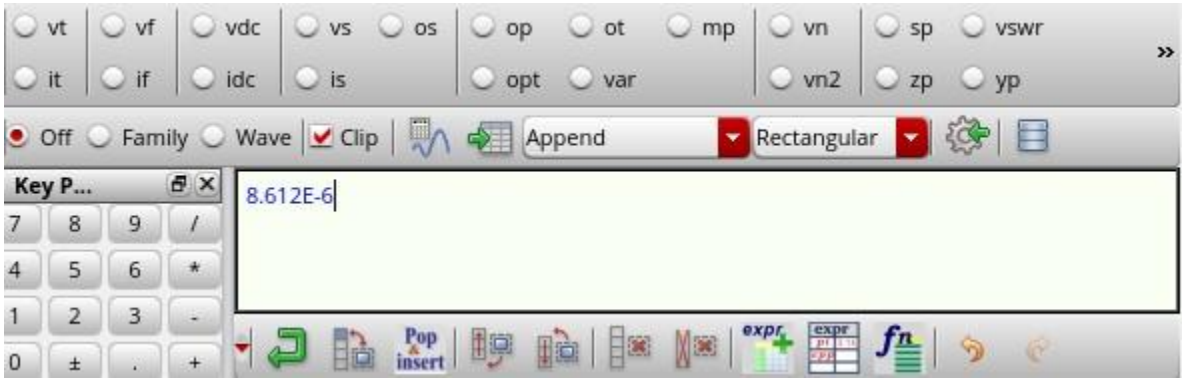
Test Bench for AVG current and delay:



Cout AVG Current



S AVG Current

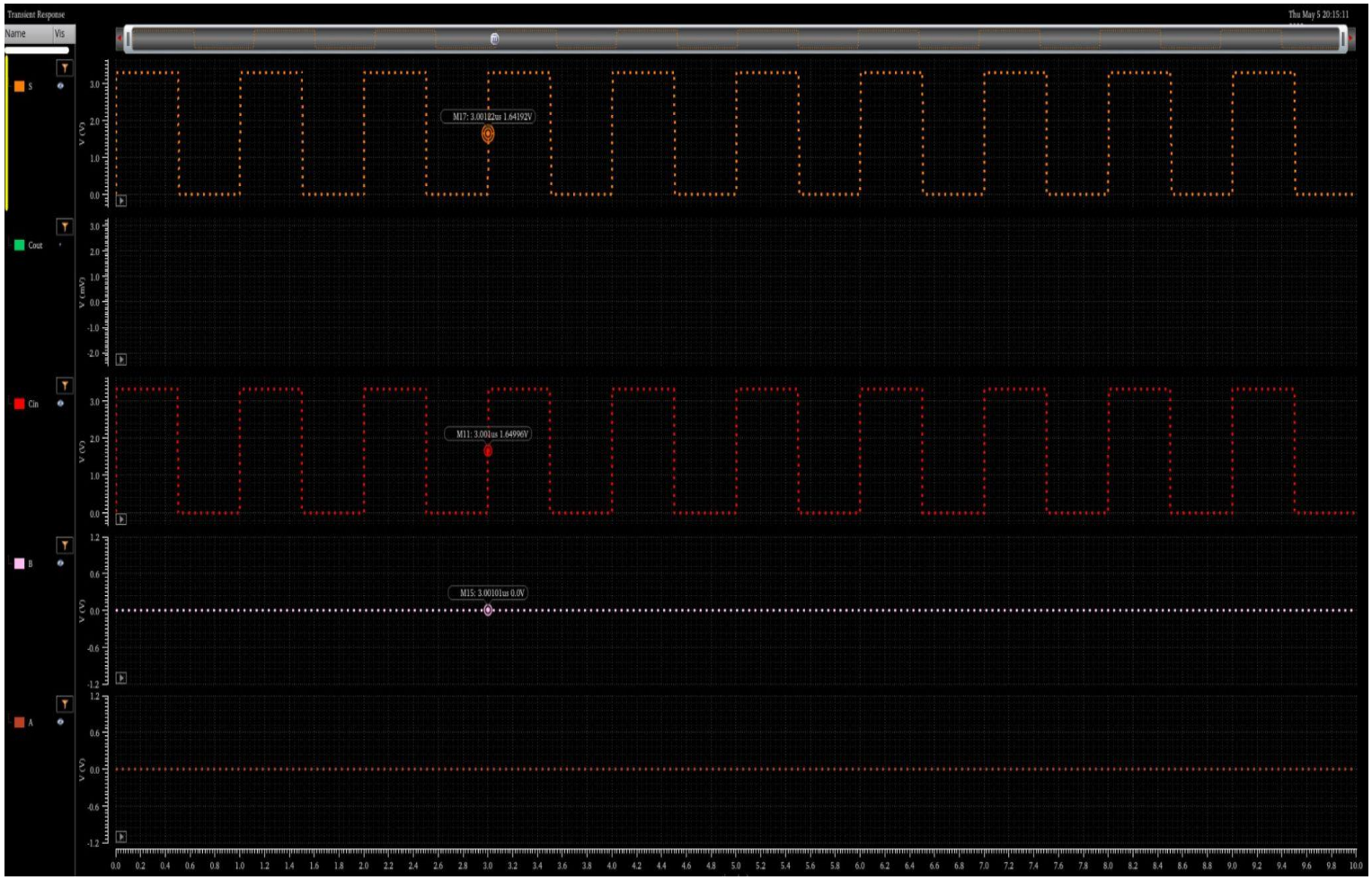


Propagation Delay(With Competition):
In order to measure the propagation delay, I would set the inputs that are supposed to be 0 to go from 0 to 0 to be off. Then, I would wait for the input to be 1 to pulse to VDD/2(1.65V) and measured the time it takes for either S or Cout to go to VDD/2. Lastly, since cadence can't map to a perfect 1.65V in some cases, I took the average of the points closest to 1.65V. Most times these points are equal which would mean both points would give the same propagation delay.

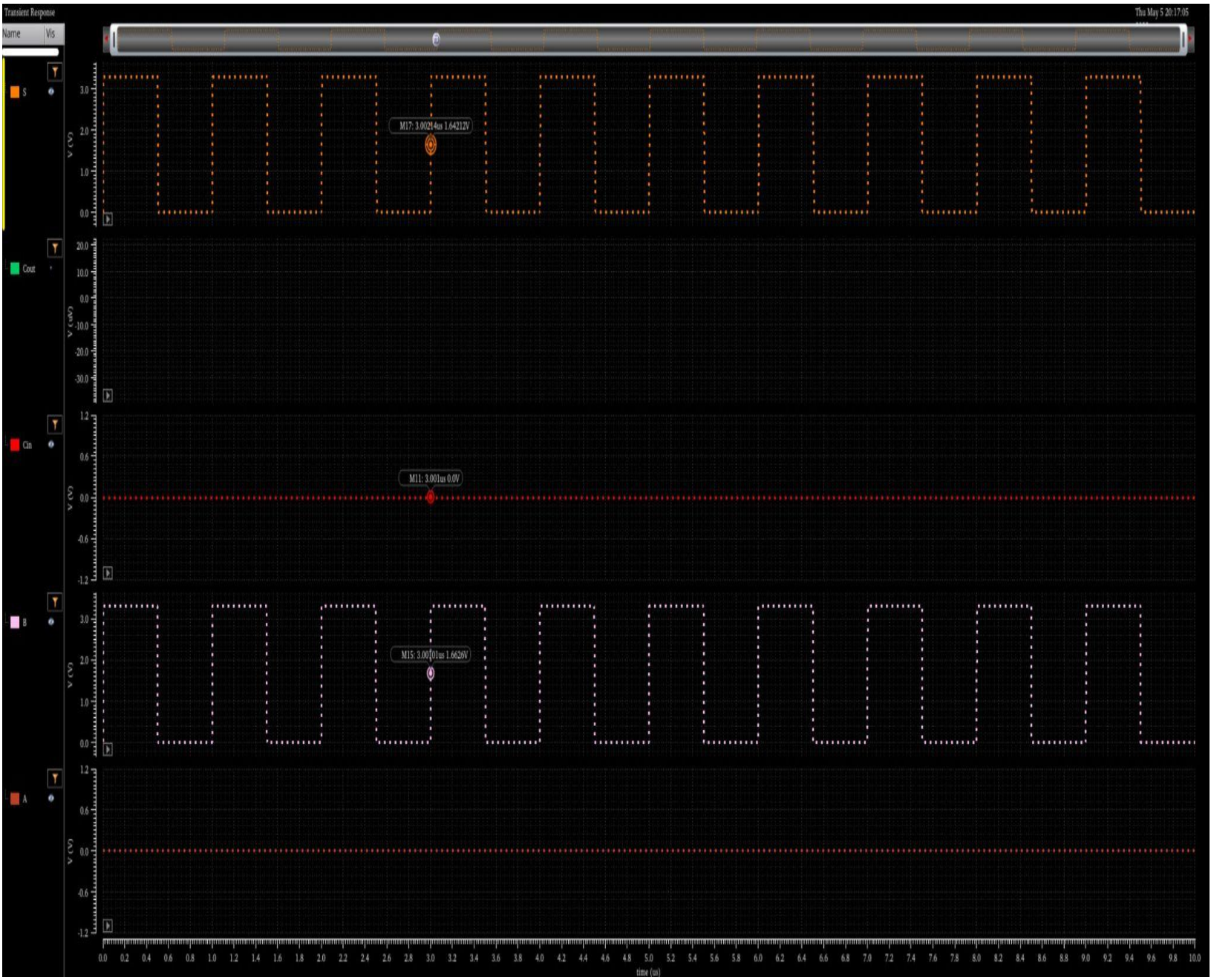
Below is a table with all propagation delays:

Input	Cout delay	S delay
001	N/A	0.22 ns(BEST)
010	N/A	1.14 ns(WORST)
011	1.28 ns	N/A
100	N/A	0.52 ns
101	1.02 ns(BEST)	N/A
110	2.14 ns(WORST)	N/A
111	2.1 ns	0.75 ns

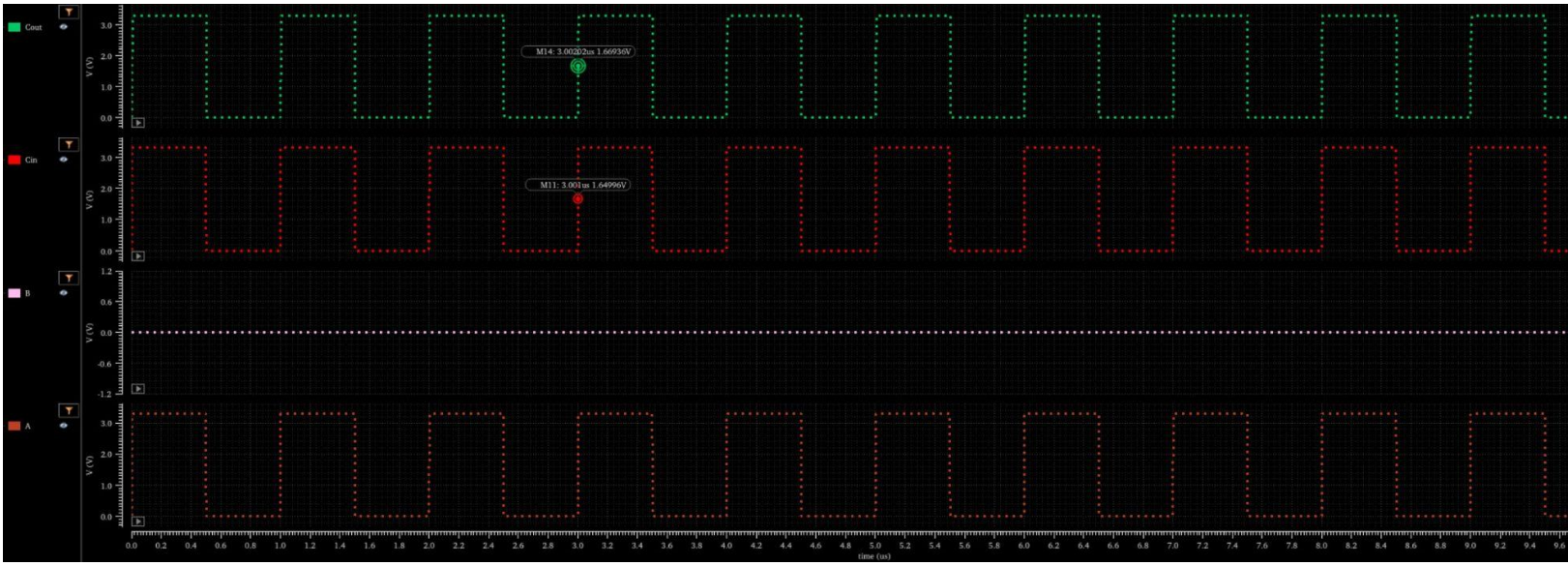
001 Input (Best Delay For S)



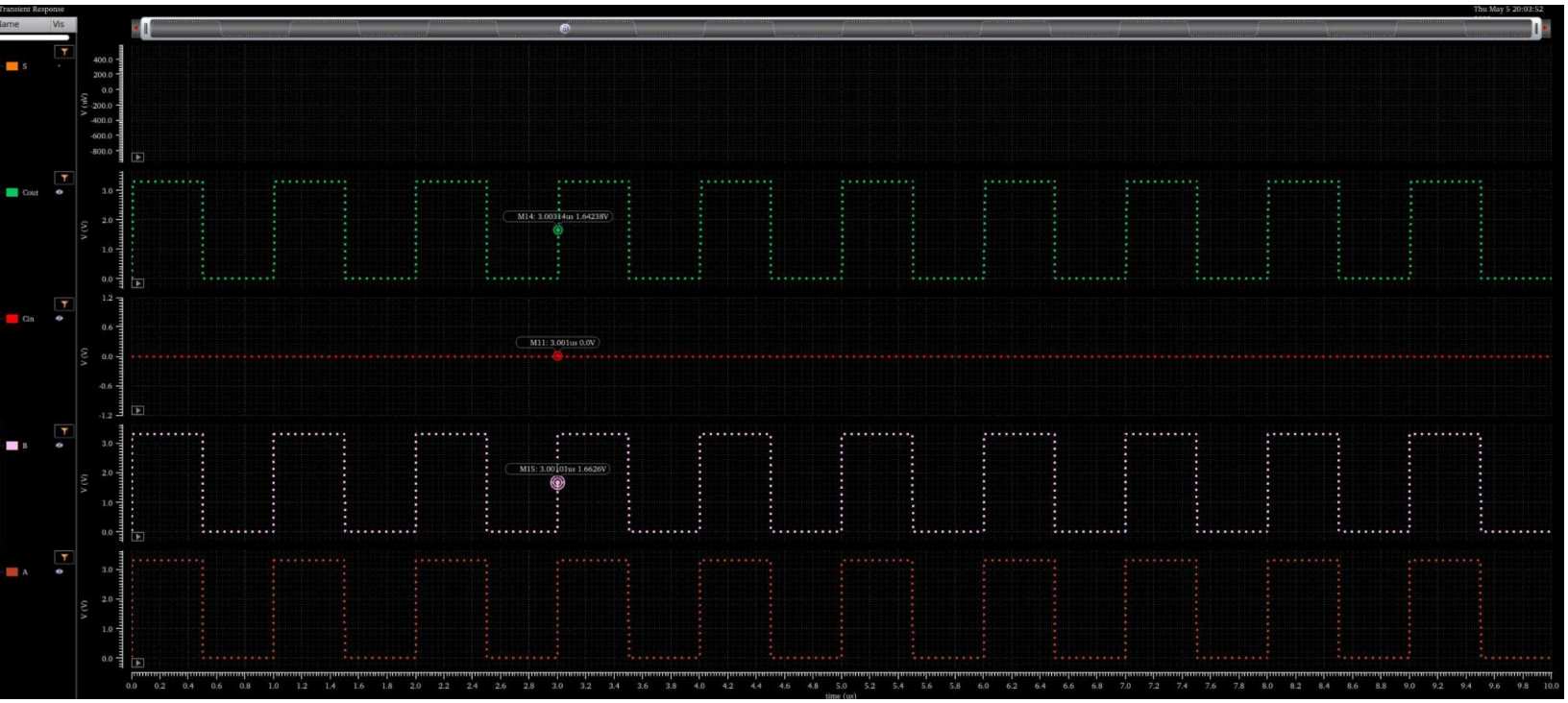
010 Input(Worst delay for S)



101 Input (Best Delay for Cout)



110 Input (Worst Delay for Cout)



Summary Specifications:

Output	Area	Power	Best Delay	Worst Delay
S	1832.8 $(\mu m)^2$	0.284 μW	0.22 ns	1.14 ns
Cout	1832.8 $(\mu m)^2$	0.284 μW	1.02 ns	2.04 ns