Design of a pipelined 8-bit multiplier

Yogesh Kumar Baswal

A Table summarizing the following information

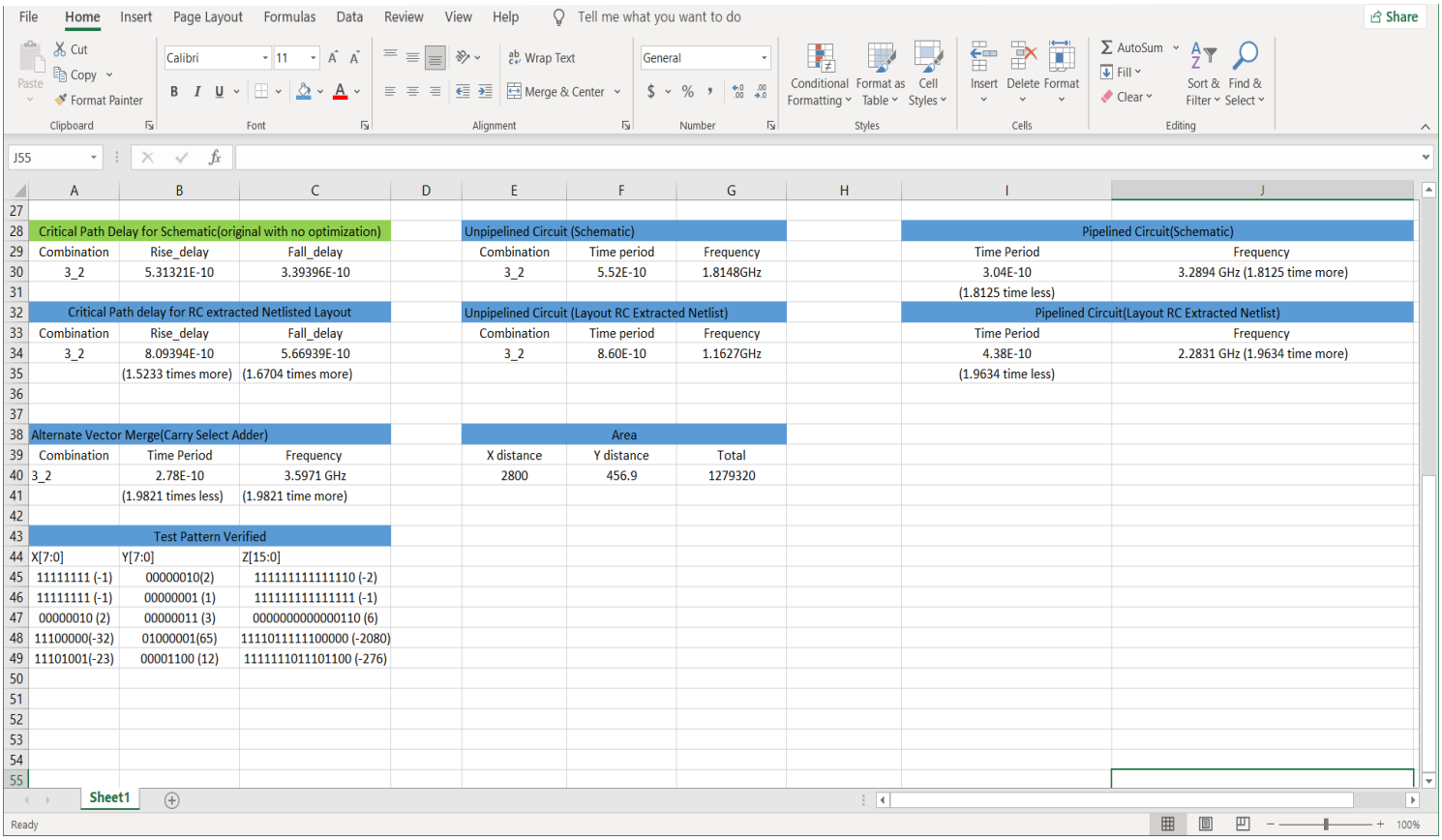
– Maximum clock frequency w/o pipelining for the schematic and the layout extracted netlist

– Maximum clock frequency with pipelining using the layout extracted netlist

– Area of the DRC and LVS clean CSM.

– Number of test patterns you verified the functionality of your CSM against

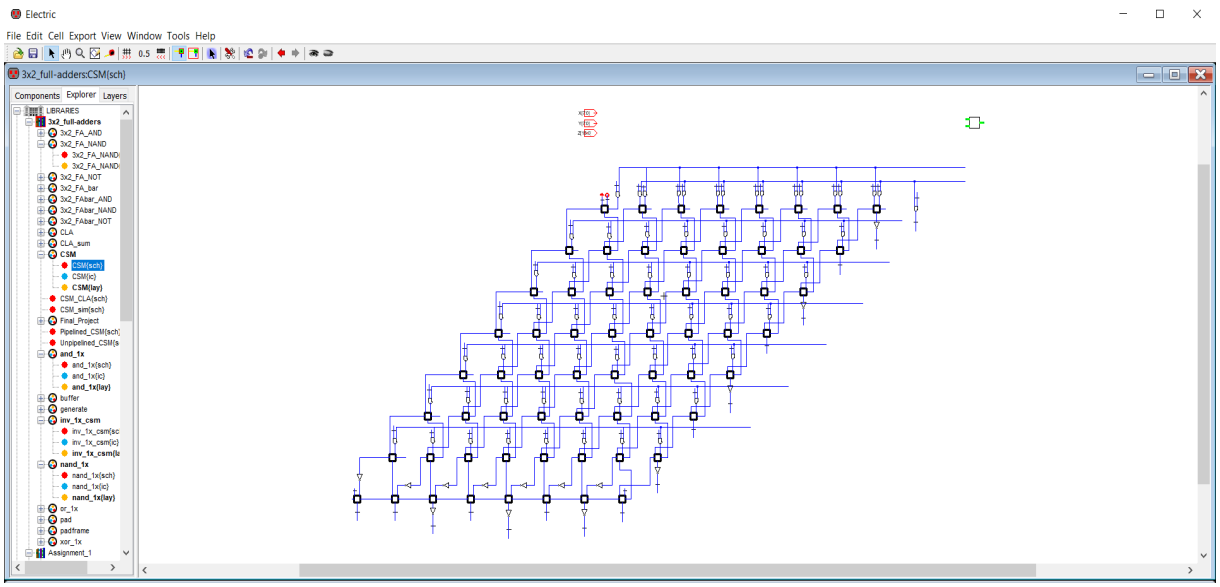
– Number of test patterns you verified the functionality of your CSM against



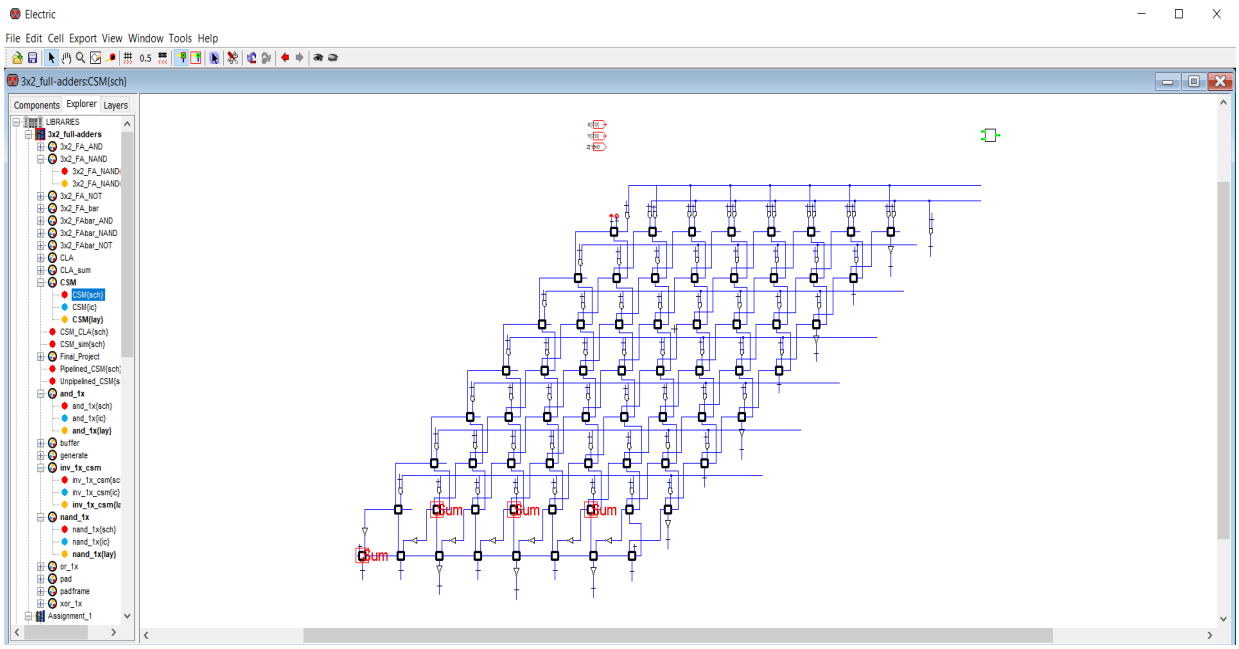
	A	B	C	D	E	F	G	H	I	J
27										
28	Critical Path Delay for Schematic(original with no optimization)				Unpipelined Circuit (Schematic)				Pipelined Circuit(Schematic)	
29	Combination	Rise_delay	Fall_delay		Combination	Time period	Frequency		Time Period	Frequency
30	3_2	5.31321E-10	3.39396E-10		3_2	5.52E-10	1.8148GHz		3.04E-10	3.2894 GHz (1.8125 time more)
31									(1.8125 time less)	
32	Critical Path delay for RC extracted Netlisted Layout				Unpipelined Circuit (Layout RC Extracted Netlist)				Pipelined Circuit(Layout RC Extracted Netlist)	
33	Combination	Rise_delay	Fall_delay		Combination	Time period	Frequency		Time Period	Frequency
34	3_2	8.09394E-10	5.66939E-10		3_2	8.60E-10	1.1627GHz		4.38E-10	2.2831 GHz (1.9634 time more)
35		(1.5233 times more)	(1.6704 times more)						(1.9634 time less)	
36										
37										
38	Alternate Vector Merge(Carry Select Adder)					Area				
39	Combination	Time Period	Frequency		X distance	Y distance	Total			
40	3_2	2.78E-10	3.5971 GHz		2800	456.9	1279320			
41		(1.9821 times less)	(1.9821 time more)							
42										
43	Test Pattern Verified									
44	X[7:0]	Y[7:0]	Z[15:0]							
45	11111111 (-1)	00000010(2)	111111111111110 (-2)							
46	11111111 (-1)	00000001 (1)	111111111111111 (-1)							
47	00000010 (2)	00000011 (3)	0000000000000110 (6)							
48	11100000(-32)	01000001(65)	1111011111100000 (-2080)							
49	11101001(-23)	00001100 (12)	1111111011101100 (-276)							
50										
51										
52										
53										
54										
55										

Block diagram of the 8-bit Carry Save Multiplier (CSM) - Highlight how you used inverting and non-inverting adders to optimize delay

Block Diagram

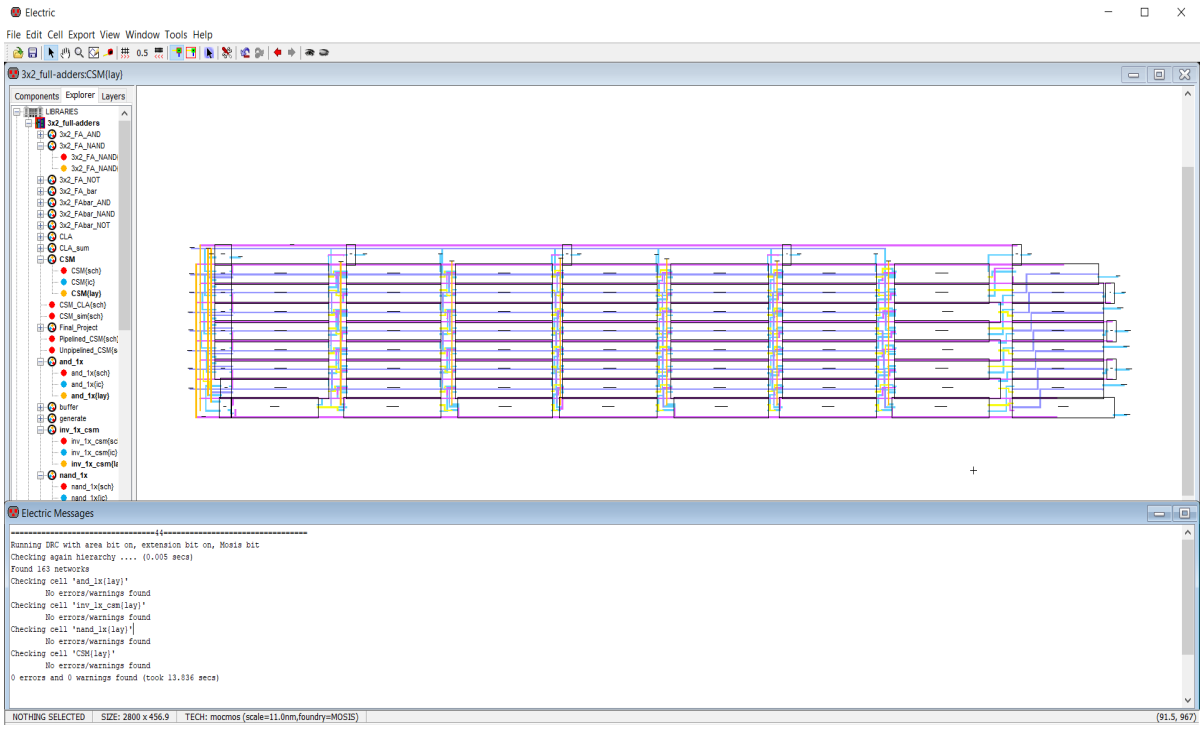


Block Diagram with Non- Inverting Full Adders Highlighted

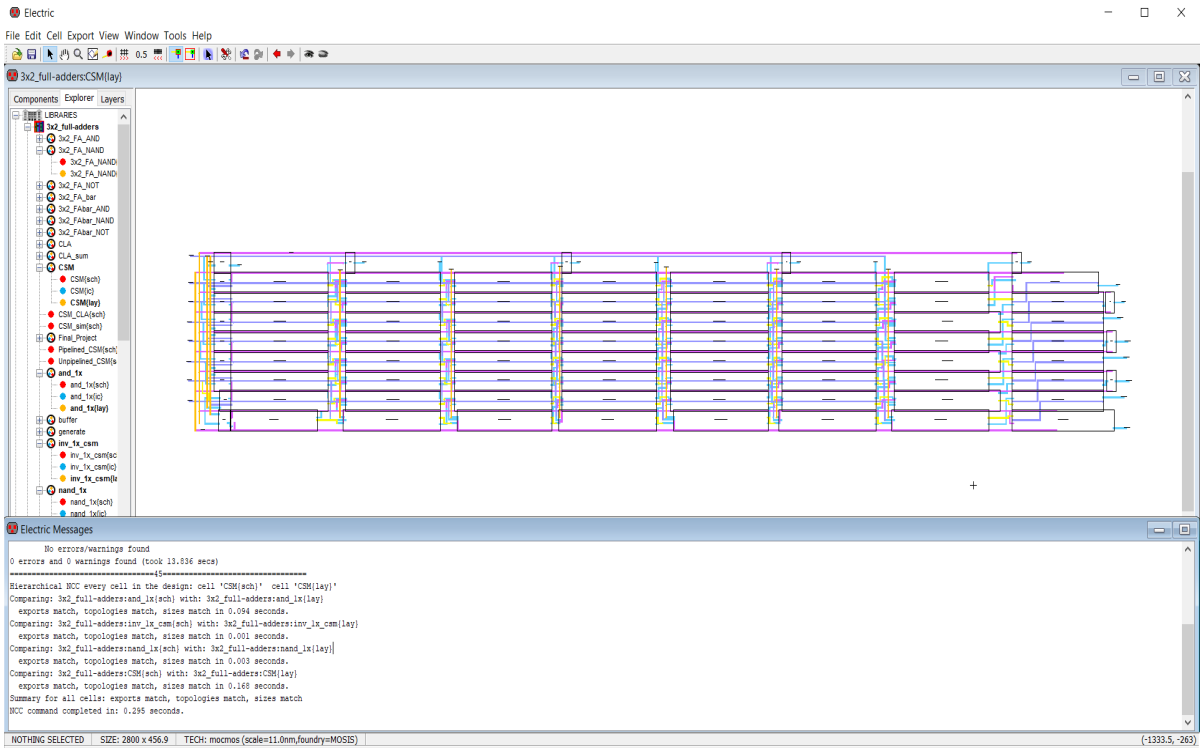


Area of the DRC and LVS clean CSM

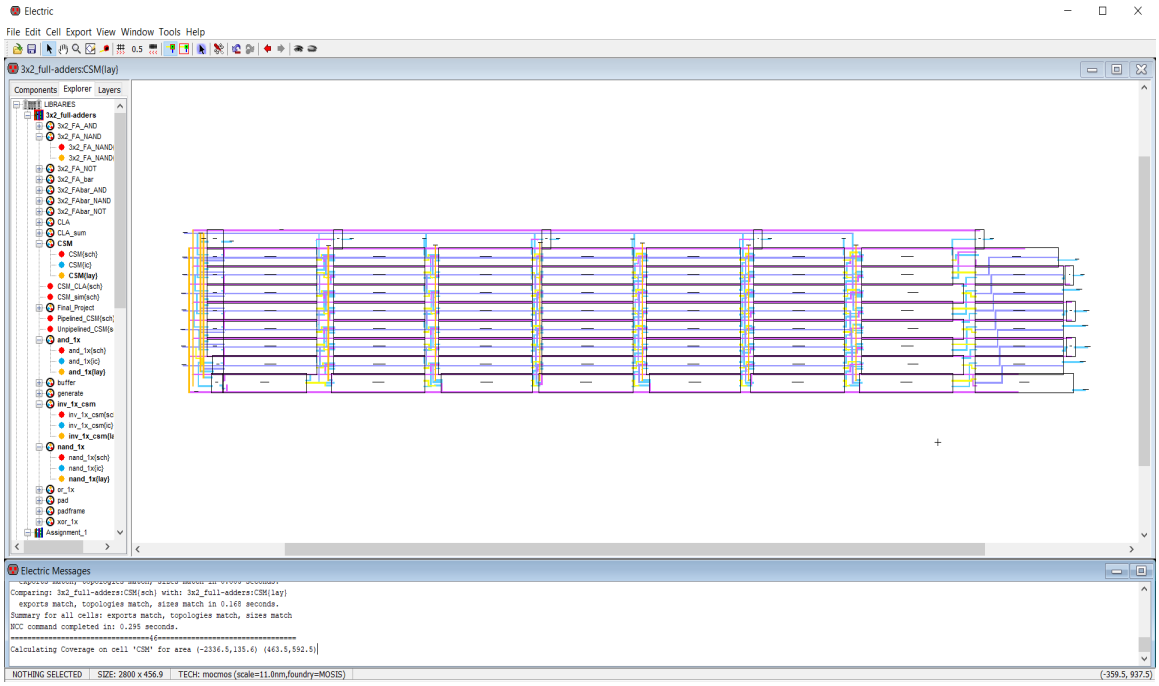
DRC clean



LVS Matched



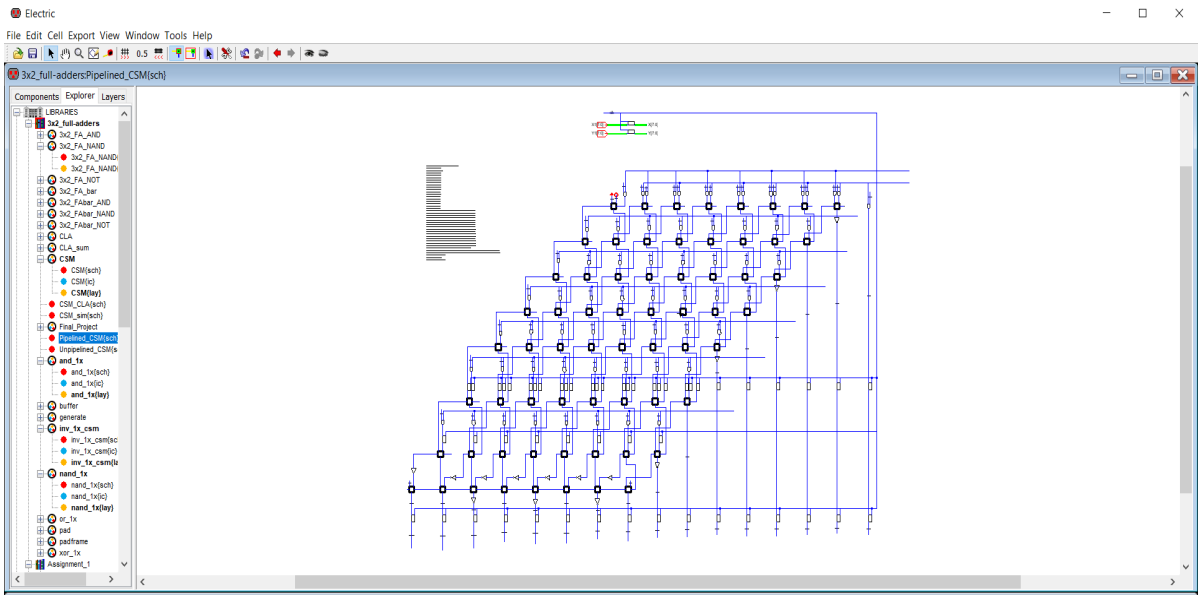
DRC Area check of the Layout



X distance = 2800 Y distance = 456.9

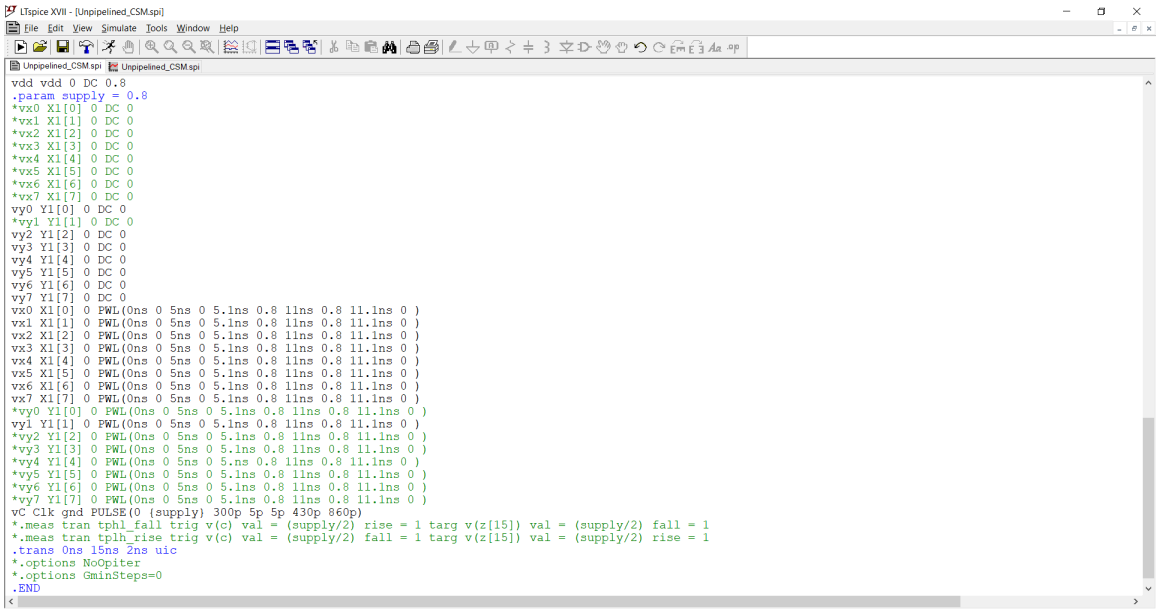
Location of Flip Flop

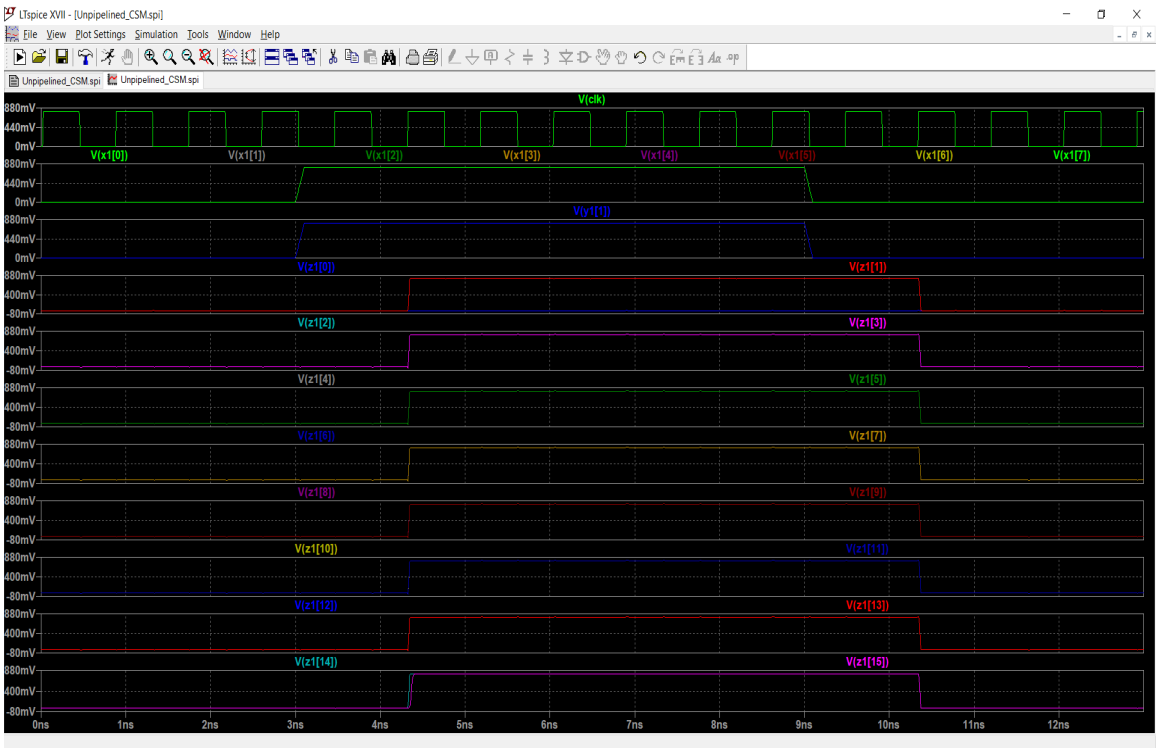
Between 5th and 6th stage of critical path



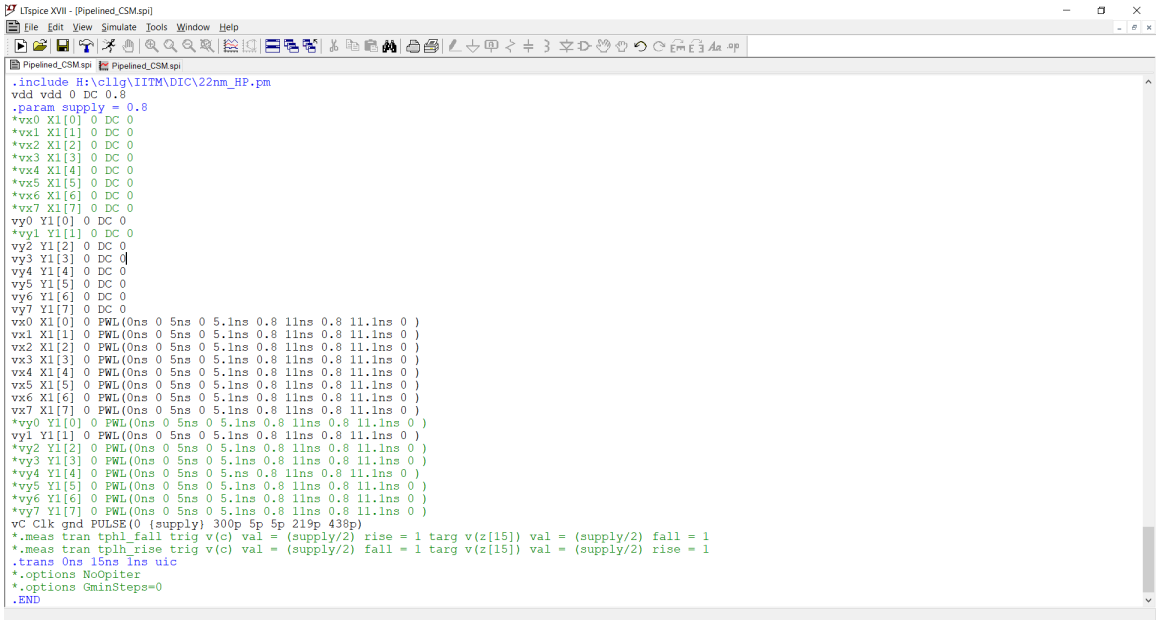
SPICE simulation showing the frequency of operation with and without pipelining using RC extracted netlists for all sub-blocks

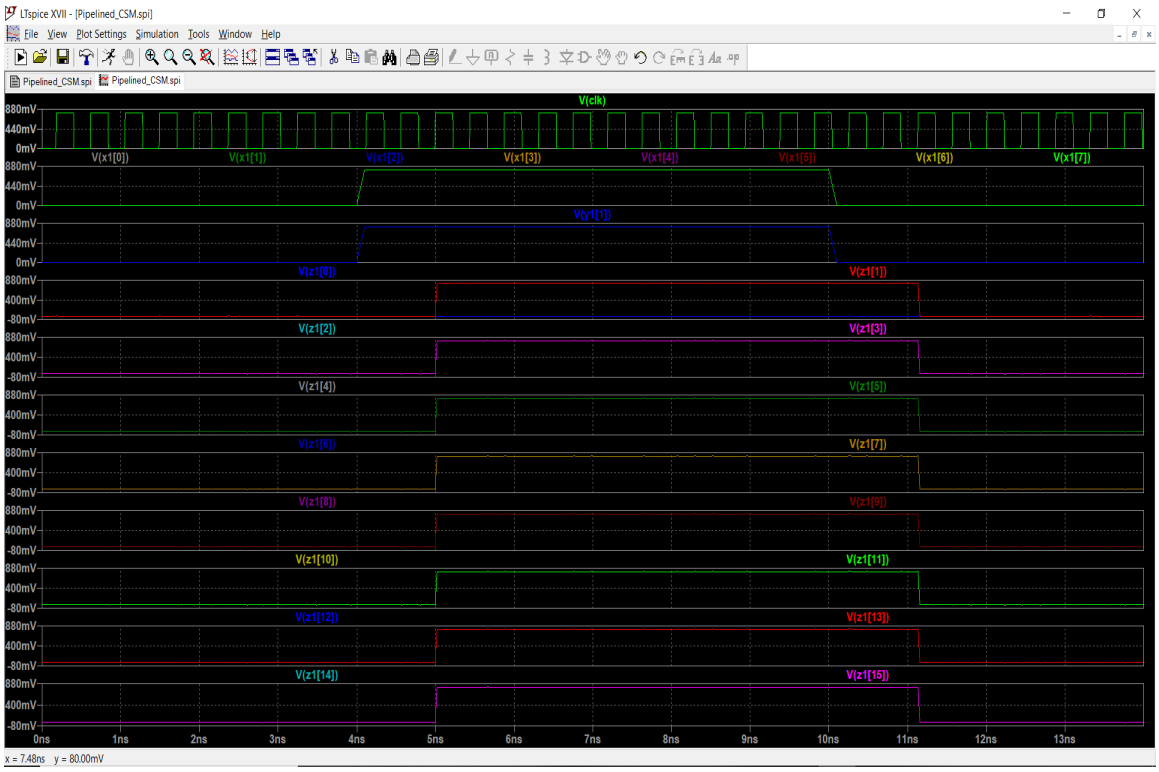
Without Pipeline (T_{clk} = 860 ps and Frequency = 1.1627 GHz)





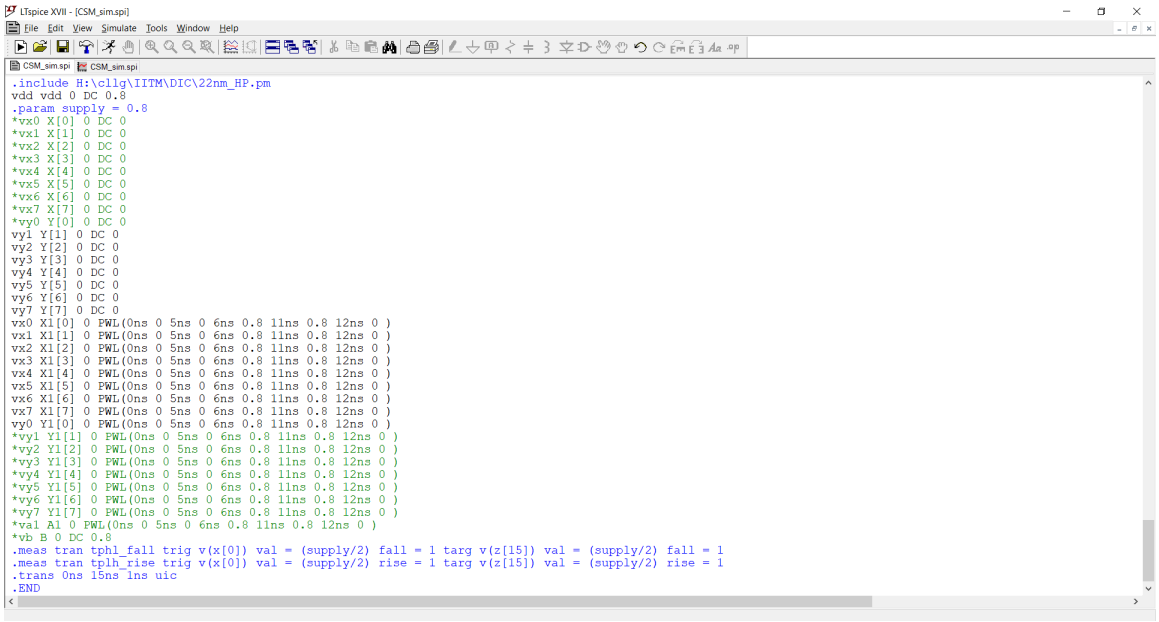
With Pipeline (T_{clk} = 438 ps and Frequency = 2.2831 GHz)

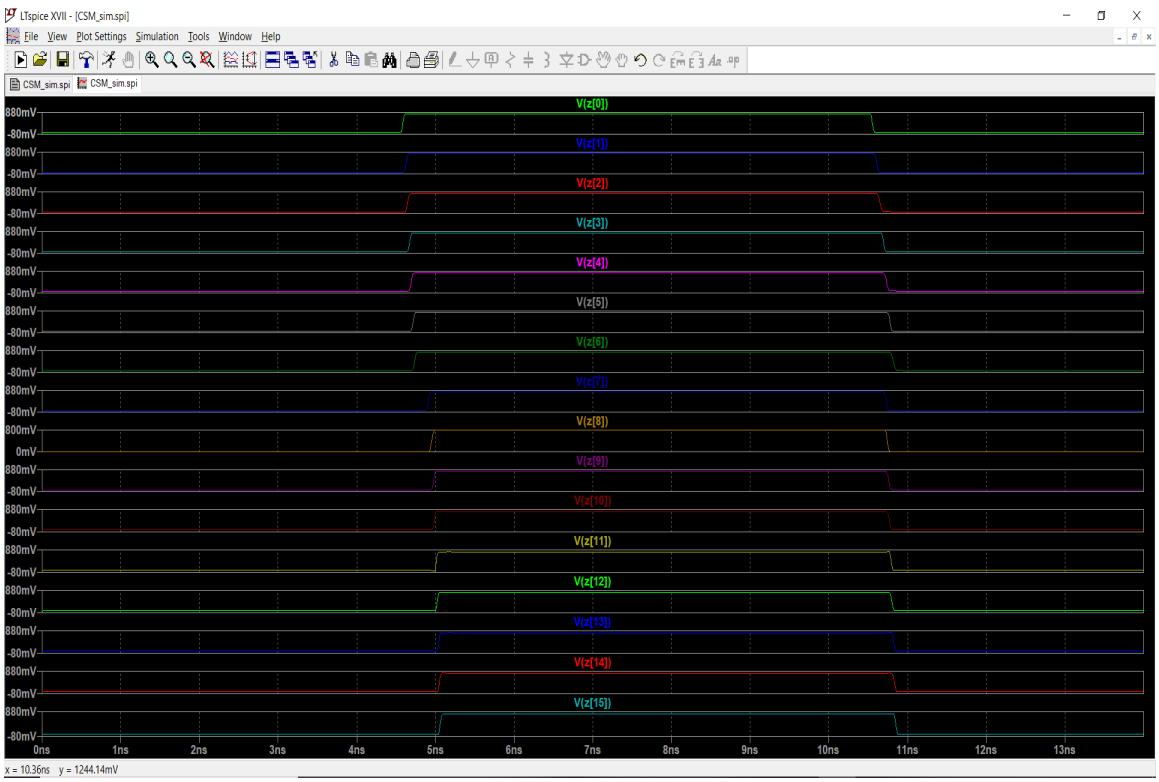




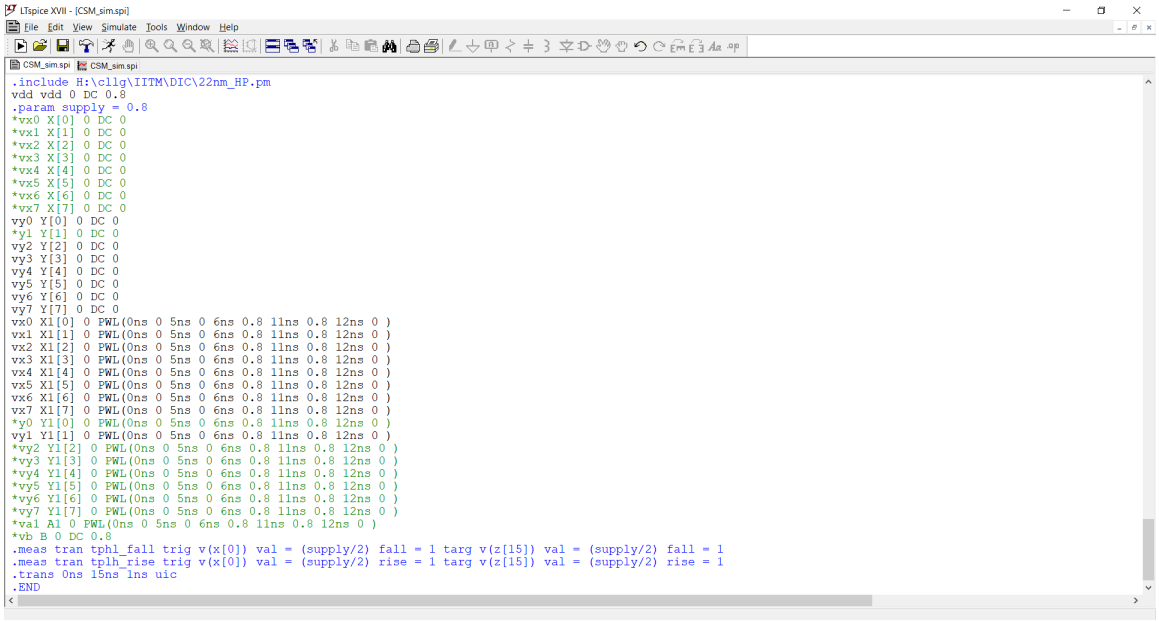
Input combinations that are tested for CSM to see if it's functionally correct - Showing different kinds of multiplications

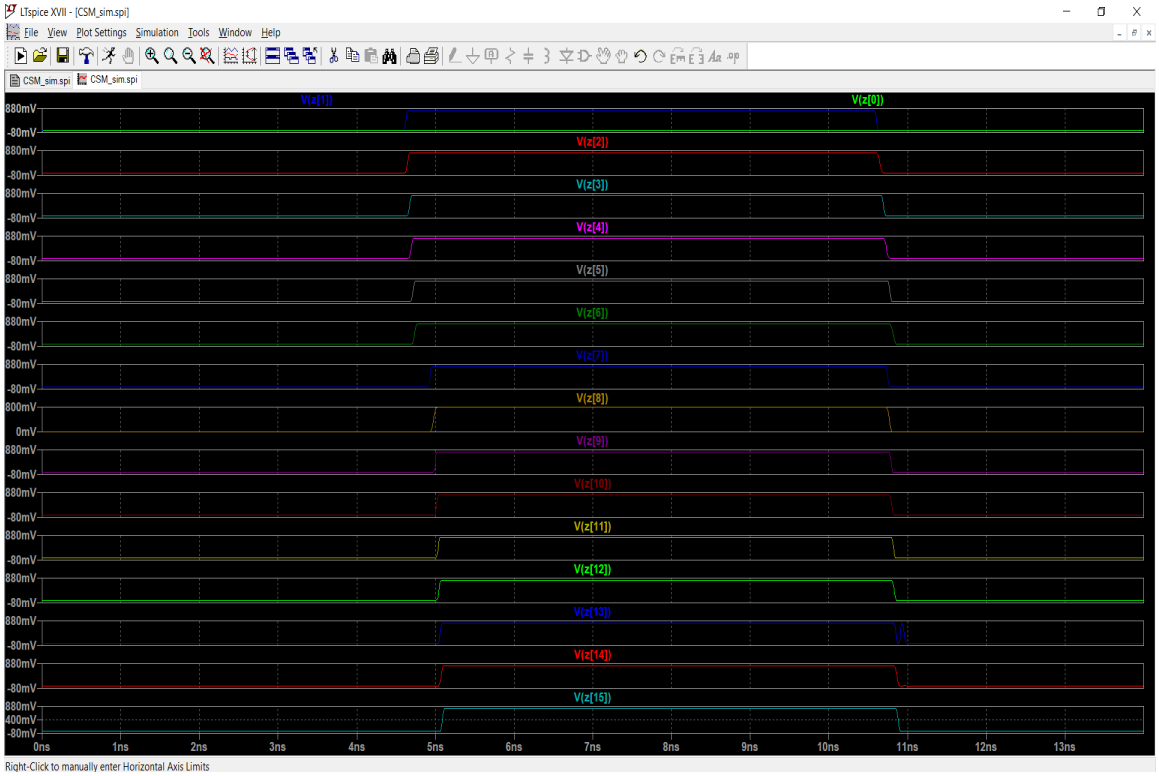
X[7:0]	Y[7:0]	Z[15:0]
11111111 (-1)	00000001(1)	111111111111111 (-1)





X[7:0]	Y[7:0]	Z[15:0]
11111111 (-1)	00000010(2)	111111111111110 (-2)





X[7:0] Y[7:0] Z[15:0]

00000010 (2) 00000011 (3) 0000000000000110 (6)

LTspice XVII - [CSM_sim.spi]

File Edit View Simulate Tools Window Help

CSM_sim.spi CSM_sim.spi

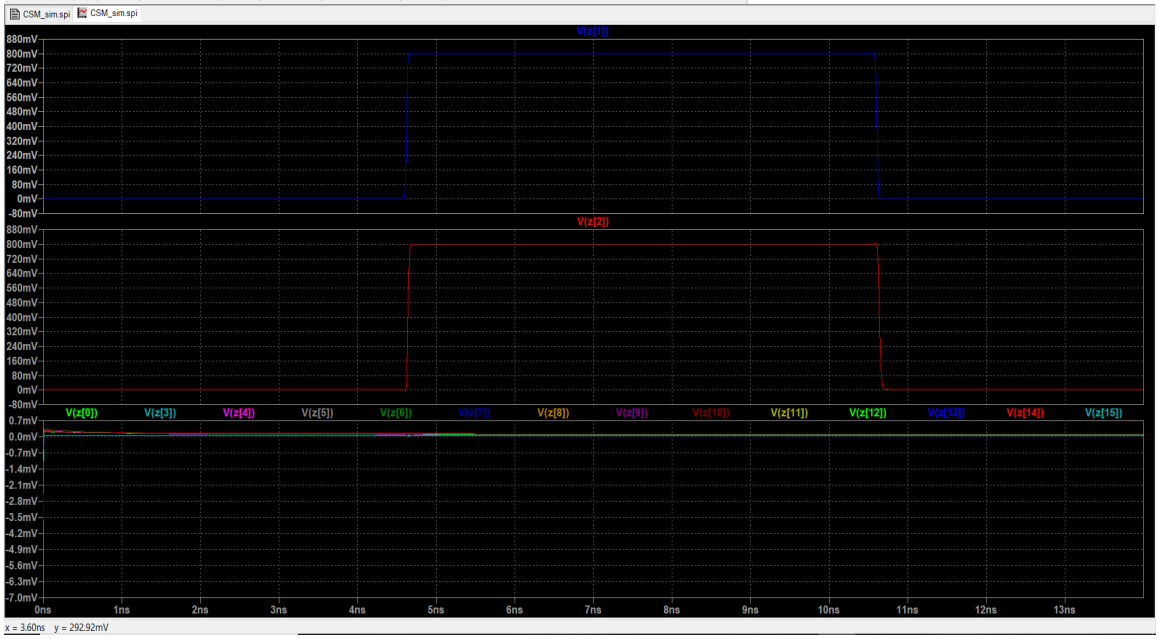
```
.include H:\cllg\IITM\DIC\22nm_HP.pm
vdd vdd 0 DC 0.8
.param supply = 0.8
vx0 X[0] 0 DC 0
*vx1 X[1] 0 DC 0
vx2 X[2] 0 DC 0
vx3 X[3] 0 DC 0
vx4 X[4] 0 DC 0
vx5 X[5] 0 DC 0
vx6 X[6] 0 DC 0
vx7 X[7] 0 DC 0
*vy0 Y[0] 0 DC 0
*vy1 Y[1] 0 DC 0
vy2 Y[2] 0 DC 0
vy3 Y[3] 0 DC 0
vy4 Y[4] 0 DC 0
vy5 Y[5] 0 DC 0
vy6 Y[6] 0 DC 0
vy7 Y[7] 0 DC 0
*vx0 X1[0] 0 PWL(0ns 0 5ns 0 6ns 0.8 11ns 0.8 12ns 0 )
vx1 X1[1] 0 PWL(0ns 0 5ns 0 6ns 0.8 11ns 0.8 12ns 0 )
*vx2 X1[2] 0 PWL(0ns 0 5ns 0 6ns 0.8 11ns 0.8 12ns 0 )
*vx3 X1[3] 0 PWL(0ns 0 5ns 0 6ns 0.8 11ns 0.8 12ns 0 )
*vx4 X1[4] 0 PWL(0ns 0 5ns 0 6ns 0.8 11ns 0.8 12ns 0 )
*vx5 X1[5] 0 PWL(0ns 0 5ns 0 6ns 0.8 11ns 0.8 12ns 0 )
*vx6 X1[6] 0 PWL(0ns 0 5ns 0 6ns 0.8 11ns 0.8 12ns 0 )
*vx7 X1[7] 0 PWL(0ns 0 5ns 0 6ns 0.8 11ns 0.8 12ns 0 )
vy0 Y1[0] 0 PWL(0ns 0 5ns 0 6ns 0.8 11ns 0.8 12ns 0 )
vy1 Y1[1] 0 PWL(0ns 0 5ns 0 6ns 0.8 11ns 0.8 12ns 0 )
*vy2 Y1[2] 0 PWL(0ns 0 5ns 0 6ns 0.8 11ns 0.8 12ns 0 )
*vy3 Y1[3] 0 PWL(0ns 0 5ns 0 6ns 0.8 11ns 0.8 12ns 0 )
*vy4 Y1[4] 0 PWL(0ns 0 5ns 0 6ns 0.8 11ns 0.8 12ns 0 )
*vy5 Y1[5] 0 PWL(0ns 0 5ns 0 6ns 0.8 11ns 0.8 12ns 0 )
*vy6 Y1[6] 0 PWL(0ns 0 5ns 0 6ns 0.8 11ns 0.8 12ns 0 )
*vy7 Y1[7] 0 PWL(0ns 0 5ns 0 6ns 0.8 11ns 0.8 12ns 0 )
*val A1 0 PWL(0ns 0 5ns 0 6ns 0.8 11ns 0.8 12ns 0 )
*vb B 0 DC 0.8
.meas tran tphl_fall trig v(x[0]) val = (supply/2) fall = 1 targ v(z[15]) val = (supply/2) fall = 1
.meas tran tplh_rise trig v(x[0]) val = (supply/2) rise = 1 targ v(z[15]) val = (supply/2) rise = 1
.trans 0ns 15ns 1ns uic
.END
```

x = 1.40ns y = 894.77mV

LTspice XVII - [CSM_sim.spi]

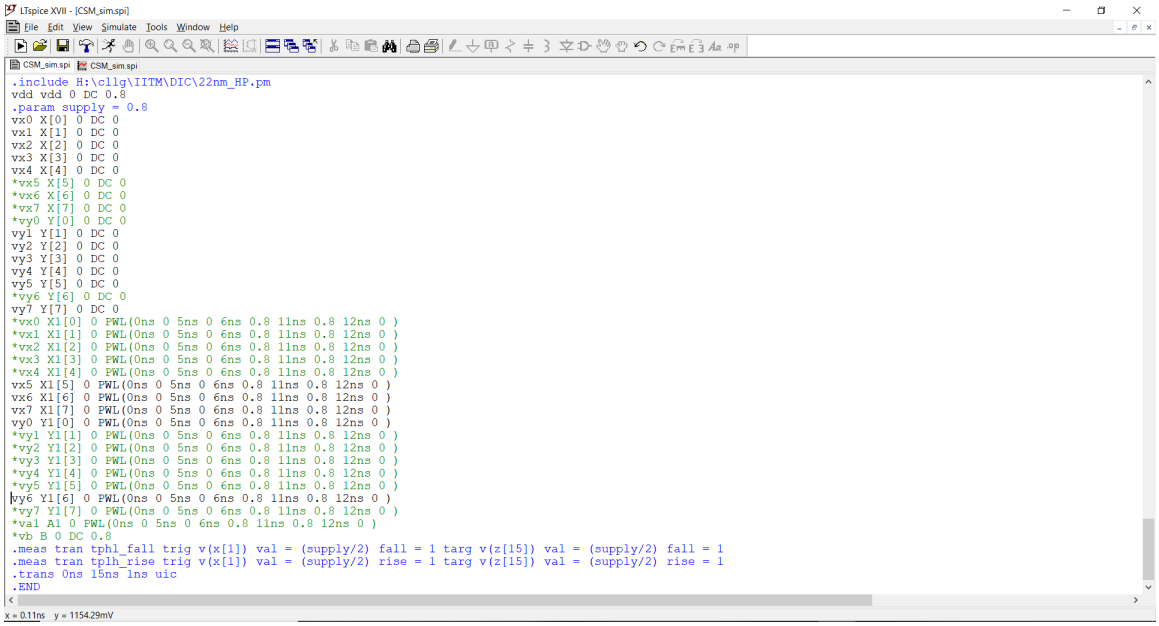
File View Plot Settings Simulation Tools Window Help

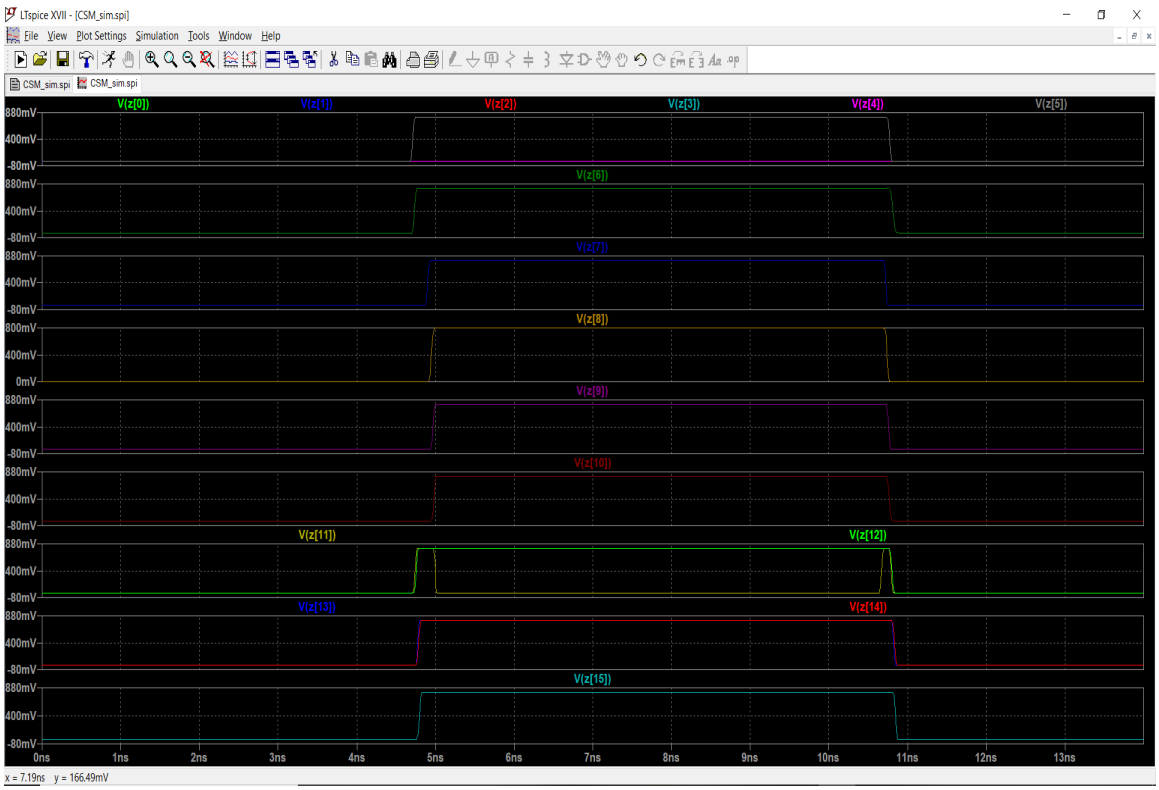
CSM_sim.spi CSM_sim.spi



x = 3.60ns y = 292.92mV

X[7:0]	Y[7:0]	Z[15:0]
11100000(-32)	01000001(65)	1111011111100000 (-2080)





SPICE simulations showing how the use of a faster adder in the Vector merge stage speed up over all delay of the CSM.

T_{clk} = 278 ps and Frequency = 3.5971 GHz