

# Integrated Circuit Design and Simulation

## Assignment 2

EXAM No. Y3900241

January 27, 2022

# Contents

<b>1 LAB 1</b>	<b>1</b>
1.1 Design of Inverter . . . . .	1
1.1.1 Boolean Expression . . . . .	1
1.1.2 CMOS Implementation . . . . .	1
1.1.3 Schematic . . . . .	2
1.1.4 Testbench . . . . .	3
1.1.5 Netlist report . . . . .	4
1.1.6 Functional Simulation . . . . .	5
1.2 Design of NAND2 . . . . .	6
1.2.1 Boolean Expression . . . . .	6
1.2.2 Karnaugh Map . . . . .	6
1.2.3 The Mathematical Method . . . . .	6
1.2.4 CMOS Implementation . . . . .	7
1.2.5 Schematic . . . . .	8
1.2.6 Testbench . . . . .	9
1.2.7 Functional Simulation . . . . .	9
1.2.8 Netlist report . . . . .	10
1.3 Design of NOR2 . . . . .	11
1.3.1 Boolean Expression . . . . .	11
1.3.2 Karnaugh Map . . . . .	11
1.3.3 The Mathematical Method . . . . .	11
1.3.4 CMOS Implementation . . . . .	12
1.3.5 Schematic . . . . .	13
1.3.6 Testbench . . . . .	14
1.3.7 Functional Simulation . . . . .	14
1.3.8 Netlist report . . . . .	15
1.4 Design of XOR . . . . .	16
1.4.1 Boolean Expression . . . . .	16
1.4.2 Karnaugh Map . . . . .	16
1.4.3 The Mathematical Method . . . . .	16
1.4.4 CMOS Implementation . . . . .	17
1.4.5 Schematic . . . . .	18
1.4.6 Testbench . . . . .	19
1.4.7 Functional Simulation . . . . .	19
1.4.8 Netlist report . . . . .	20
1.5 Design of function: $F = (A \cdot B) + (C \cdot D)$ . . . . .	22
1.5.1 Boolean Expression . . . . .	22
1.5.2 Karnaugh Map . . . . .	22

1.5.3	The Mathematical Method . . . . .	23
1.5.4	Boolean Expression for $Y = \overline{F} = \overline{(A \cdot B) + (C \cdot D)}$ . . . . .	23
1.5.5	Karnaugh Map . . . . .	24
1.5.6	The Mathematical Method . . . . .	24
1.5.7	CMOS Implementation . . . . .	25
1.5.8	Schematic . . . . .	26
1.5.9	Testbench . . . . .	27
1.5.10	Functional Simulation . . . . .	27
1.5.11	Netlist report . . . . .	28
1.6	SPICE simulation . . . . .	30
<b>2</b>	<b>LAB 2</b>	<b>31</b>
2.1	DRS, LVS of INVERTER . . . . .	31
2.1.1	LAYOUT . . . . .	31
2.1.2	DRC . . . . .	33
2.1.3	LVS . . . . .	34
2.2	DRS,LVS of NAND2 . . . . .	35
2.2.1	LAYOUT . . . . .	35
2.2.2	DRC . . . . .	36
2.2.3	LVS . . . . .	37
2.3	DRS,LVS of NOR2 . . . . .	38
2.3.1	LAYOUT . . . . .	38
2.3.2	DRC . . . . .	39
2.3.3	LVS . . . . .	40
2.4	DRS,LVS of XOR . . . . .	41
2.4.1	LAYOUT . . . . .	41
2.4.2	DRC . . . . .	41
2.4.3	LVS . . . . .	42
2.5	DRC and LVS . . . . .	42
<b>3</b>	<b>LAB 3</b>	<b>43</b>
3.1	PEX-INVERTER . . . . .	43
3.1.1	Testbench . . . . .	43
3.1.2	Netlist . . . . .	44
3.1.3	Simulation . . . . .	46
3.2	PEX-NAND2 . . . . .	48
3.2.1	Testbench . . . . .	48
3.2.2	Netlist . . . . .	48
3.2.3	Simulation . . . . .	51
3.3	PEX-NOR2 . . . . .	53
3.3.1	Testbench . . . . .	53
3.3.2	Netlist . . . . .	53
3.3.3	Simulation . . . . .	56
3.4	PEX-XOR . . . . .	58
3.4.1	Testbench . . . . .	58
3.4.2	Netlist . . . . .	58
3.4.3	Simulation . . . . .	62
3.5	Inverter layout and calibre view . . . . .	63
3.6	PEX . . . . .	63

<b>4 LAB 4</b>	<b>64</b>
4.1 Configuring NAND2 as Inverter . . . . .	64
4.1.1 Testbench . . . . .	64
4.1.2 Simulation . . . . .	65
4.1.3 Netlist . . . . .	65
4.2 comparison of all 3 inverter designs: . . . . .	68
4.2.1 Testbench . . . . .	68
4.2.2 Pre-layout Simulation . . . . .	69
4.2.3 Pre-layout Simulation netlist . . . . .	69
4.2.4 Post-layout Simulation . . . . .	72
4.2.5 Post-layout Simulation netlist . . . . .	72
4.3 Pre and Post-layout propagation delays . . . . .	77
4.3.1 Pre-layout propagation delay . . . . .	77
4.3.2 Post-layout propagation . . . . .	79
4.4 Standard Cell . . . . .	80
<b>5 LAB 5</b>	<b>81</b>
5.1 Gate level layouts for XOR . . . . .	81
5.1.1 XOR using TSMC180 NAND cell . . . . .	81
5.1.2 XOR using my own NAND cell . . . . .	83
5.2 Simulations . . . . .	85
5.3 Propagation delays . . . . .	86
5.3.1 Pre-layout . . . . .	86
5.3.2 Post-layout . . . . .	89
5.4 Power Consumption . . . . .	91
5.5 Worst-case propagation delay . . . . .	92
5.6 Drive Strength . . . . .	92
<b>6 LAB 6</b>	<b>93</b>
6.1 Gated Inverter . . . . .	93
6.1.1 Schematic . . . . .	93
6.1.2 Layout . . . . .	94
6.1.3 DRC . . . . .	94
6.1.4 LVS . . . . .	95
6.2 Latch . . . . .	95
6.2.1 Schematic . . . . .	95
6.2.2 Layout . . . . .	96
6.2.3 DRC . . . . .	96
6.2.4 LVS . . . . .	97
6.3 DFF . . . . .	97
6.3.1 Schematic . . . . .	97
6.3.2 Layout . . . . .	98
6.3.3 DRC . . . . .	98
6.3.4 LVS . . . . .	99
6.4 clock-to-q delay, setup and hold . . . . .	99
6.5 Simulation results . . . . .	100
6.5.1 clk-to-Q delay . . . . .	101
<b>7 LAB 7</b>	<b>102</b>
7.1 Setup time . . . . .	102

<b>8 LAB 8</b>	<b>104</b>
8.1 Corner Analysis . . . . .	104
8.1.1 Pre and Post-layout simulation of NOR . . . . .	104
8.2 Fast, slow, typical corners . . . . .	105
8.3 Monte Carlo Simulation . . . . .	106
8.3.1 Pre-layout . . . . .	106
8.3.2 Post-layout . . . . .	107
8.3.3 Monte Carlo Simulation . . . . .	107

# List of Figures

1.1	CMOS implementation of Inverter . . . . .	1
1.2	Schematic of Inverter design in Cadence . . . . .	2
1.3	Testbench for Inverter . . . . .	3
1.4	Netlist for Inverter Testbench . . . . .	4
1.5	Transient response simulation of Inverter . . . . .	5
1.6	NAND Karnaugh Map . . . . .	6
1.7	CMOS implementation of NAND2 . . . . .	7
1.8	Cadence Schematic of NAND2 . . . . .	8
1.9	Testbench for NAND2 . . . . .	9
1.10	Transient response simulation of NAND2 . . . . .	9
1.11	Netlist for NAND2 Testbench . . . . .	10
1.12	NOR Karnaugh Map . . . . .	11
1.13	CMOS implementation of NOR2 . . . . .	12
1.14	Cadence Schematic of NOR2 . . . . .	13
1.15	Testbench for NOR2 . . . . .	14
1.16	Transient response simulation of NOR2 . . . . .	14
1.17	Netlist for NOR2 Testbench . . . . .	15
1.18	XOR Karnaugh Map . . . . .	16
1.19	CMOS implementation of XOR . . . . .	17
1.20	Cadence Schematic of XOR . . . . .	18
1.21	Testbench for XOR . . . . .	19
1.22	Transient response simulation of XOR . . . . .	19
1.23	Netlist for XOR Testbench -1 . . . . .	20
1.24	Netlist for XOR Testbench -2 . . . . .	21
1.25	Karnaugh Map . . . . .	22
1.26	Karnaugh Map . . . . .	24
1.27	CMOS implementation of $F = A \cdot B + C \cdot D$ . . . . .	25
1.28	Cadence schematic of $F = A \cdot B + C \cdot D$ . . . . .	26
1.29	Testbench for $F = A \cdot B + C \cdot D$ . . . . .	27
1.30	Transient response simulation of $F = A \cdot B + C \cdot D$ . . . . .	27
1.31	Netlist of $F = A \cdot B + C \cdot D$ Testbench -1 . . . . .	28
1.32	Netlist of $F = A \cdot B + C \cdot D$ Testbench -2 . . . . .	29
2.1	Layout of Inverter . . . . .	31
2.2	DRC of Inverter design . . . . .	33
2.3	LVS of Inverter design . . . . .	34
2.4	Layout of NAND2 . . . . .	35
2.5	DRC of NAND2 design . . . . .	36
2.6	LVS of NAND2 design . . . . .	37

2.7	Layout of NOR2 . . . . .	38
2.8	DRC of NOR2 design . . . . .	39
2.9	LVS of NOR2 design . . . . .	40
2.10	Layout of XOR . . . . .	41
2.11	DRC of XOR design . . . . .	41
2.12	LVS of XOR design . . . . .	42
3.1	Testbench for pre and post-layout of Inverter . . . . .	43
3.2	Netlist for pre and post-layout TB of Inverter -1 . . . . .	44
3.3	Netlist for pre and post-layout TB of Inverter -2 . . . . .	45
3.4	Simulation of pre and post-layout . . . . .	46
3.5	Pre-layout simulation of Inverter . . . . .	46
3.6	Post-layout simulation of Inverter . . . . .	47
3.7	Testbench for pre and post-layout of NAND2 . . . . .	48
3.8	Netlist for pre and post-layout TB of NAND2 -1 . . . . .	49
3.9	Netlist for pre and post-layout TB of NAND2 -2 . . . . .	50
3.10	Simulation of pre and post-layout . . . . .	51
3.11	Simulation of Pre-layout NAND2 . . . . .	51
3.12	Simulation of Post-layout NAND2 . . . . .	52
3.13	Testbench for pre and post-layout of NOR2 . . . . .	53
3.14	Netlist for pre and post-layout TB of NOR2 -1 . . . . .	54
3.15	Netlist for pre and post-layout TB of NOR2 -2 . . . . .	55
3.16	Simulation of Pre and Post-layout NOR2 . . . . .	56
3.17	Simulation of Pre-layout NOR2 . . . . .	56
3.18	Simulation of Post-layout NOR2 . . . . .	57
3.19	Testbench for pre and post-layout of XOR . . . . .	58
3.20	Netlist for pre and post-layout TB of XOR -1 . . . . .	59
3.21	Netlist for pre and post-layout TB of XOR -2 . . . . .	60
3.22	Netlist for pre and post-layout TB of XOR -3 . . . . .	61
3.23	Simulation of Pre and Post-layout XOR . . . . .	62
3.24	Simulation with pre and post-layout . . . . .	62
3.25	Simulation with pre and post-layout . . . . .	63
4.1	NAND2 as Inverter Testbench . . . . .	64
4.2	NAND2 as Inverter Simulation (pre and Post-layout) . . . . .	65
4.3	NAND2 as Inverter Netlist -1 . . . . .	66
4.4	NAND2 as Inverter Netlist -2 . . . . .	67
4.5	3 Inverters Testbench . . . . .	68
4.6	Pre-layout Simulation of 3 designs . . . . .	69
4.7	Pre-layout Simulation Netlist -1 . . . . .	70
4.8	Pre-layout Simulation Netlist -2 . . . . .	71
4.9	Post layout Simulation of 3 designs . . . . .	72
4.10	Post layout Simulation netlist -1 . . . . .	73
4.11	Post layout Simulation netlist -2 . . . . .	74
4.12	Post layout Simulation netlist -3 . . . . .	75
4.13	Post layout Simulation netlist -4 . . . . .	76
4.14	Pre-layout Simulation of NAND2 as Inverter . . . . .	77
4.15	Pre-layout Simulation of Inverter . . . . .	77
4.16	Pre-layout Simulation of TSMC180-cell Inverter . . . . .	78
4.17	Post-layout Simulation of NAND2 as Inverter . . . . .	79

4.18 Post-layout Simulation of Inverter . . . . .	79
4.19 Post-layout Simulation of TSMC180-cell Inverter . . . . .	80
5.1 Schematic of XOR using TSMC180 NAND gates . . . . .	81
5.2 Layout of XOR using TSMC180 NAND gates . . . . .	81
5.3 DRC of XOR using TSMC180 NAND gates . . . . .	82
5.4 LVS of XOR using TSMC180 NAND gates . . . . .	82
5.5 Schematic of XOR using my own NAND gates . . . . .	83
5.6 Layout of XOR using my own NAND gates . . . . .	83
5.7 DRC of XOR using my own NAND gates . . . . .	83
5.8 LVS of XOR using my own NAND gates . . . . .	84
5.9 Testbench with 3 designs . . . . .	84
5.10 Pre-layout simulation comparison . . . . .	85
5.11 Post-layout simulation comparison . . . . .	85
5.12 Pre-layout simulation comparison -1 . . . . .	86
5.13 Pre-layout simulation comparison -2 . . . . .	86
5.14 Pre-layout simulation comparison -3 . . . . .	87
5.15 Pre-layout simulation comparison -4 . . . . .	87
5.16 Post-layout simulation comparison -1 . . . . .	89
5.17 Post-layout simulation comparison -2 . . . . .	89
5.18 Post-layout simulation comparison -3 . . . . .	90
5.19 Post-layout simulation comparison -4 . . . . .	90
6.1 Schematic of Gated Inverter . . . . .	93
6.2 Layout of Gated Inverter . . . . .	94
6.3 DRC of Gated Inverter . . . . .	94
6.4 LVS of Gated Inverter . . . . .	95
6.5 Schematic of Latch . . . . .	95
6.6 Layout of Latch . . . . .	96
6.7 DRC of Latch . . . . .	96
6.8 LVS of Latch . . . . .	97
6.9 Schematic of DFF . . . . .	97
6.10 Layout of DFF . . . . .	98
6.11 DRC of DFF . . . . .	98
6.12 LVS of DFF . . . . .	99
6.13 Testbench for DFF . . . . .	100
6.14 Simulation of DFF when data and clock edge arrives at same time . . . . .	100
6.15 clk-to-Q delay . . . . .	101
7.1 setup time simulation . . . . .	102
7.2 hold time simulation . . . . .	103
8.1 Output high to low trasition . . . . .	104
8.2 Output low to high transition . . . . .	105
8.3 Histogram of propagation delay Pre-layout -1 . . . . .	106
8.4 Histogram of propagation delay Pre-layout -2 . . . . .	106
8.5 Histogram of propagation delay Post-layout -1 . . . . .	107
8.6 Histogram of propagation delay Post-layout -2 . . . . .	107

# Chapter 1

## LAB 1

All the designs shown use NMOS width = 450nm and PMOS width = 600nm. PMOS width is greater than NMOS because NMOS are faster compared to PMOS.

### 1.1 Design of Inverter

In this section the Schematic, Testbench and Simulation results of Inverter design are presented.

#### 1.1.1 Boolean Expression

$$Y = \overline{A}$$

A	Y
1	0
0	1

#### 1.1.2 CMOS Implementation

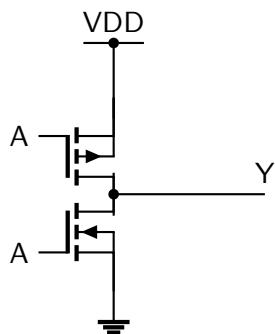


Figure 1.1: CMOS implementation of Inverter

NMOS is connected to the ground (pull-down network) and PMOS is connected to VDD(pull-up network).

### 1.1.3 Schematic

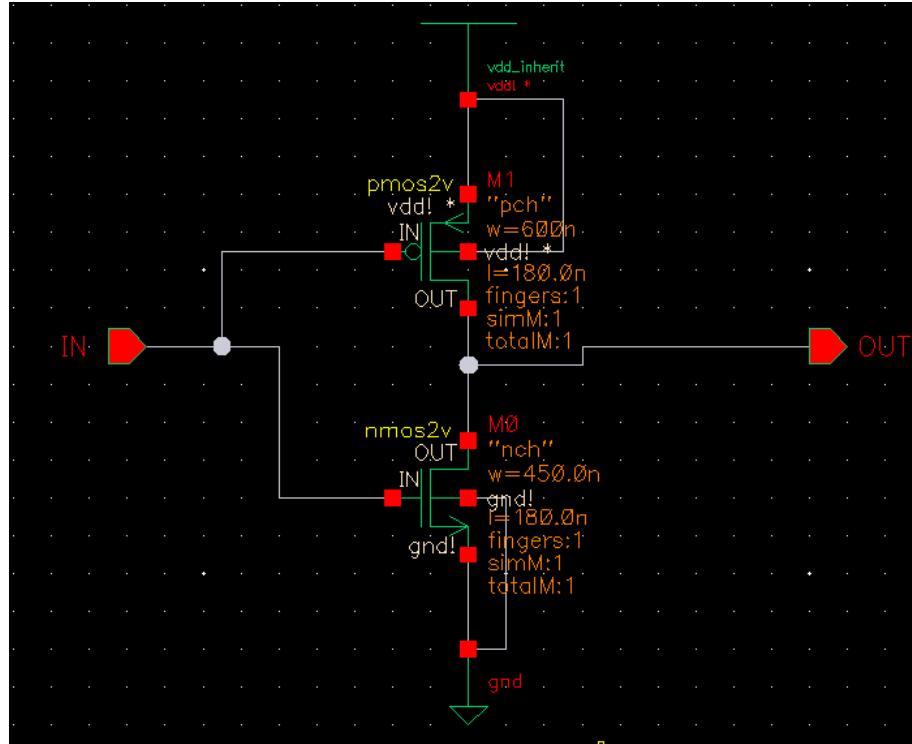


Figure 1.2: Schematic of Inverter design in Cadence

Figure 1.2 is the schematic realised in Cadence Virtuoso for the Inverter design presented in Figure 1.1.

#### 1.1.4 Testbench

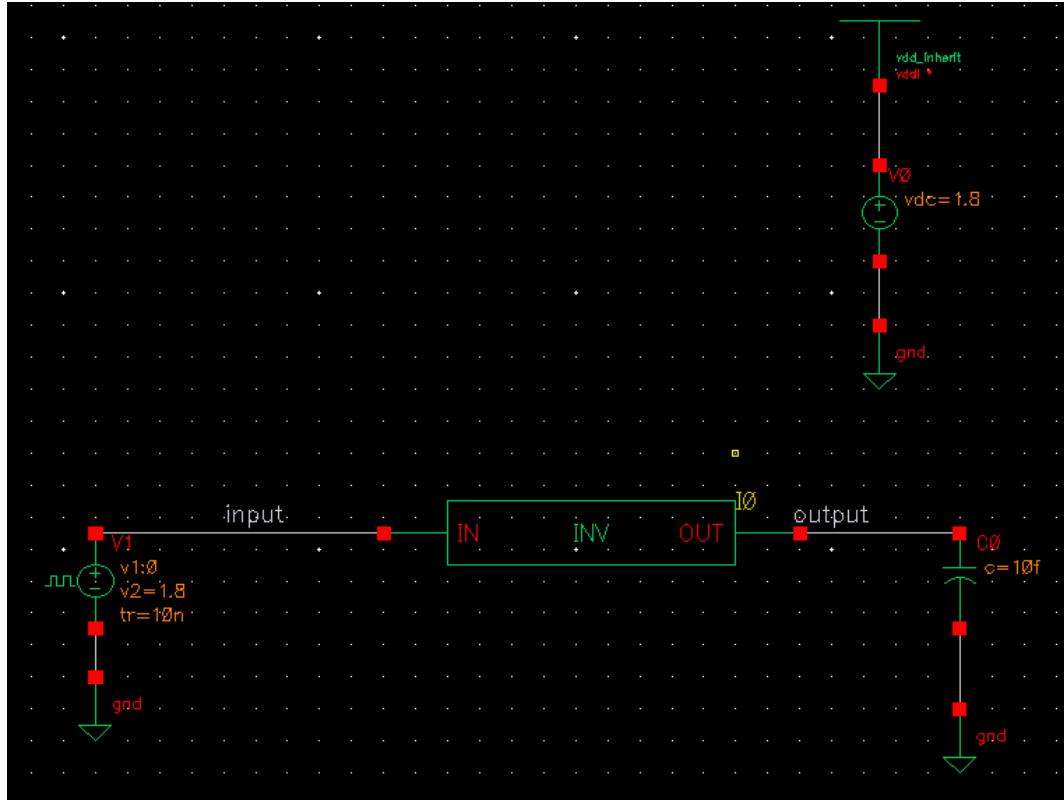


Figure 1.3: Testbench for Inverter

The testbench shown in Figure 1.3 is created for performing functional simulation of created Inverted schematic in Figure 1.3. Inverter is instantiated in the testbench design.

### 1.1.5 Netlist report

```

-----+
include "/home/cohort-2021/sp1822/icds2021/tsmc180/global.scs"

// Library name: LOGIC180
// Cell name: INV
// View name: schematic
subckt INV IN OUT inh_vdd
    M0 (OUT IN 0 0) nch l=180.0n w=450.0n m=1 nf=1 sd=540.0n ad=2.16e-13 \
        as=2.16e-13 pd=1.86u ps=1.86u nrd=0.6 nrs=0.6 sa=480.0n sb=480.0n \
        sca=0 scb=0 scc=0
    M1 (OUT IN inh_vdd inh_vdd) pch l=180.0n w=600n m=1 nf=1 sd=540.0n \
        ad=2.88e-13 as=2.88e-13 pd=2.16u ps=2.16u nrd=0.45 nrs=0.45 \
        sa=480.0n sb=480.0n sca=0 scb=0 scc=0
ends INV
// End of subcircuit definition.

// Library name: LOGIC180
// Cell name: INV_tb
// View name: schematic
I0 (_net0 _net1 vdd!) INV
V0 (vdd! 0) vsource dc=1.8 type=dc
C0 (_net1 0) capacitor c=10f
V1 (_net0 0) vsource dc=1.8 type=pulse val0=0 val1=1.8 period=100n \
    rise=10n fall=10n width=40n
simulatorOptions options reltol=1e-3 vabstol=1e-6 iabstol=1e-12 temp=27 \
    tnom=27 scalem=1.0 scale=1.0 gmin=1e-12 rforce=1 maxnotes=5 maxwarns=5 \
    digits=5 cols=80 pivrel=1e-3 sensfile="../psf/sens.output" \
    checklimitdest=psf
tran tran stop=200n errpreset=conservative write="spectre.ic" \
    writefinal="spectre.fc" annotate=status maxiters=5

```

Figure 1.4: Netlist for Inverter Testbench

Netlist is created for our testbench design in preparation for performing transient analysis simulation of Inverter. The netlist describes the instantiated Inverter design as shown in Figure 1.4 starting with the line subckt.

### 1.1.6 Functional Simulation

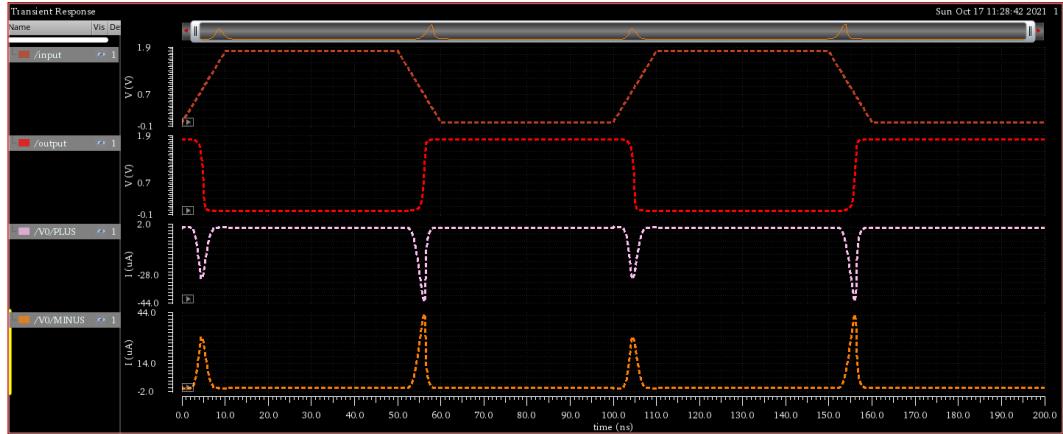


Figure 1.5: Transient response simulation of Inverter

The waveform results are in accordance with the Boolean table for Inverter presented in Subsection 1.1.1.

## 1.2 Design of NAND2

In this section the Schematic, Testbench and Simulation results for NAND2 are presented.

### 1.2.1 Boolean Expression

$$Y = \overline{A \cdot B}$$

A	B	Y
0	0	1
0	1	1
1	0	1
1	1	0

### 1.2.2 Karnaugh Map

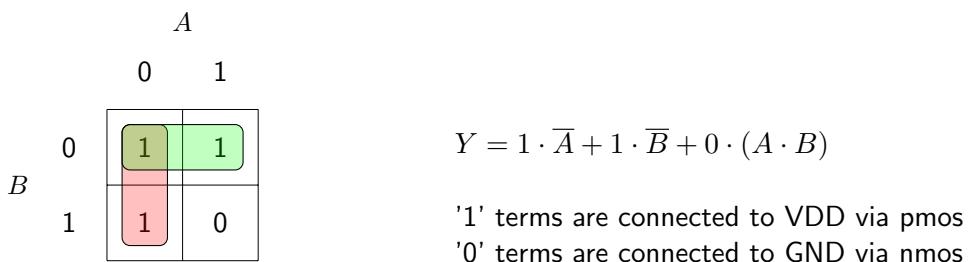


Figure 1.6: NAND Karnaugh Map

### 1.2.3 The Mathematical Method

Alternative method to Karnaugh Map for constructing CMOS logic gates.

**PMOS :**

$$F = \overline{A \cdot B}$$

$$F = \overline{A} + \overline{B} \text{ (Applying Demorgan's Law)}$$

$$F_p = A + B \text{ (invert inputs for pmos)}$$

**NMOS :**

$$F_n = \overline{\overline{A} \cdot \overline{B}} \text{ (Invert output for nmos)}$$

$$F_n = A \cdot B$$

### 1.2.4 CMOS Implementation

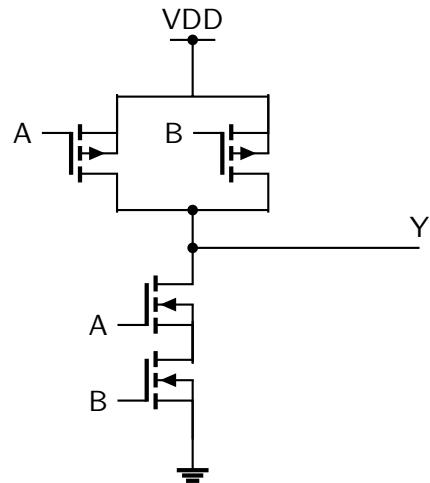


Figure 1.7: CMOS implementation of NAND2

We can observe from the Figure 1.7 that the two PMOS are in parallel and the two NMOS are in series.

### 1.2.5 Schematic

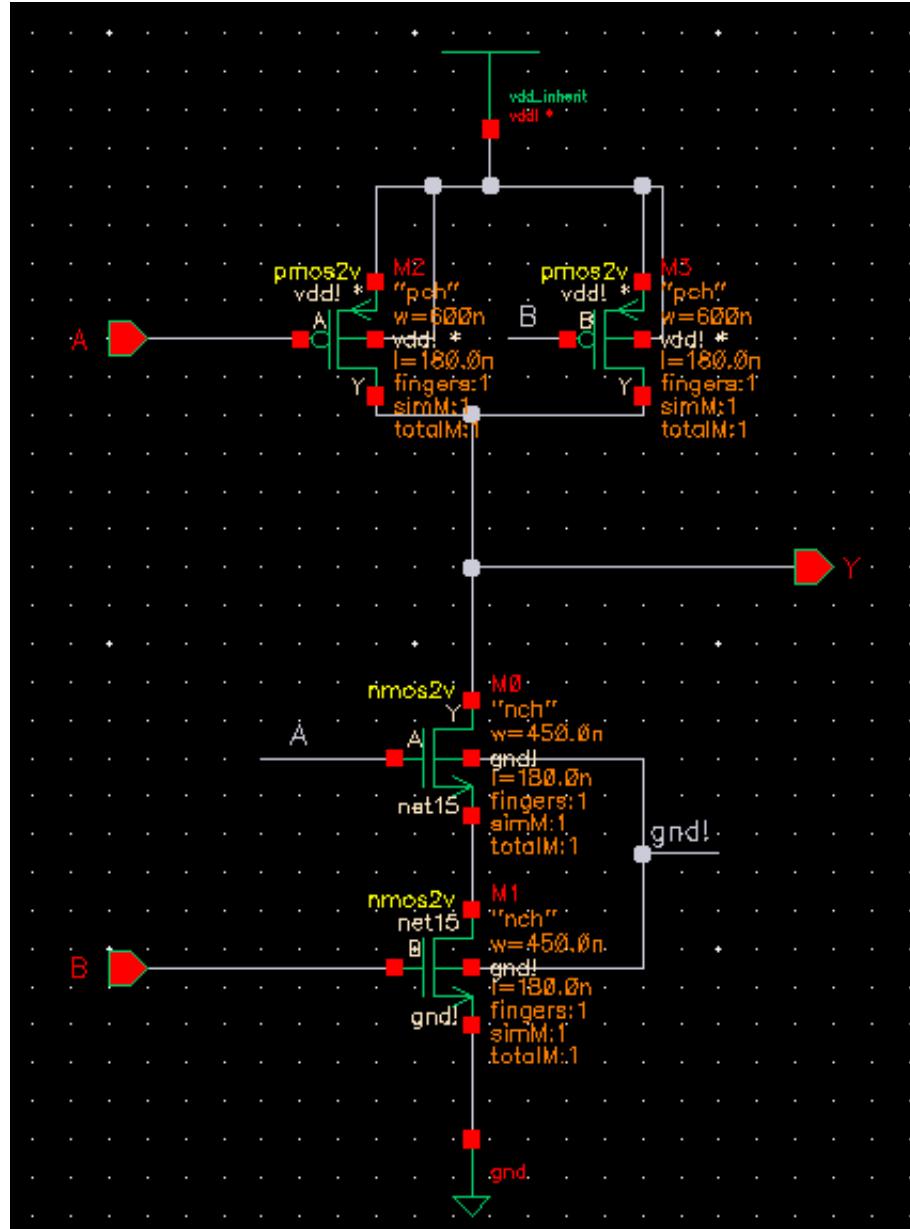


Figure 1.8: Cadence Schematic of NAND2

Figure 1.8 is the schematic realised in Cadence Virtuoso for the NAND2 design presented in Figure 1.7.

### 1.2.6 Testbench

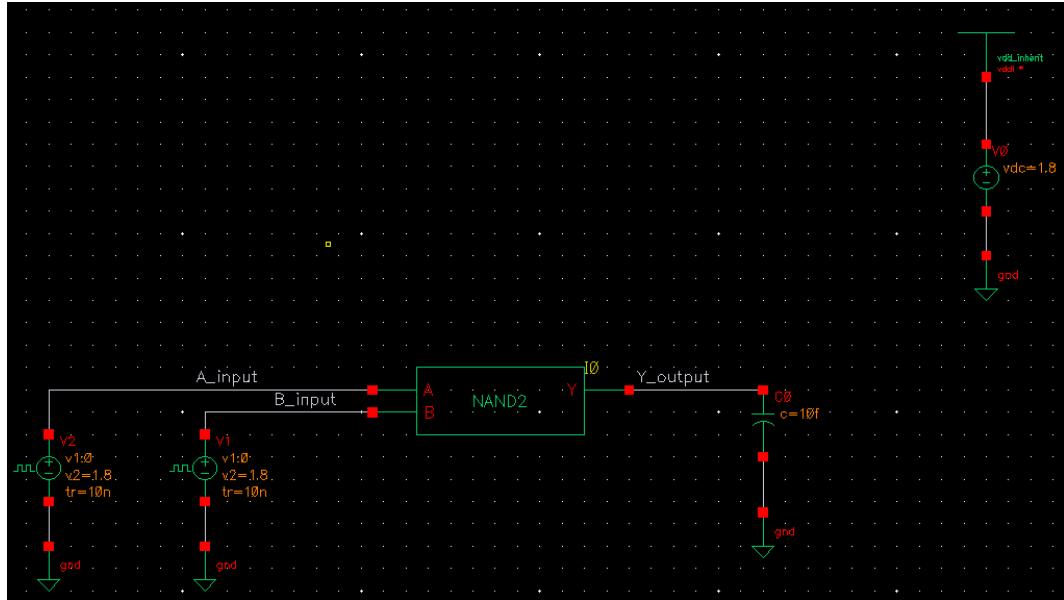


Figure 1.9: Testbench for NAND2

The testbench shown in Figure 1.9 is created for performing functional simulation of created NAND2 schematic in Figure 1.8. NAND2 is instantiated in the testbench design.

### 1.2.7 Functional Simulation

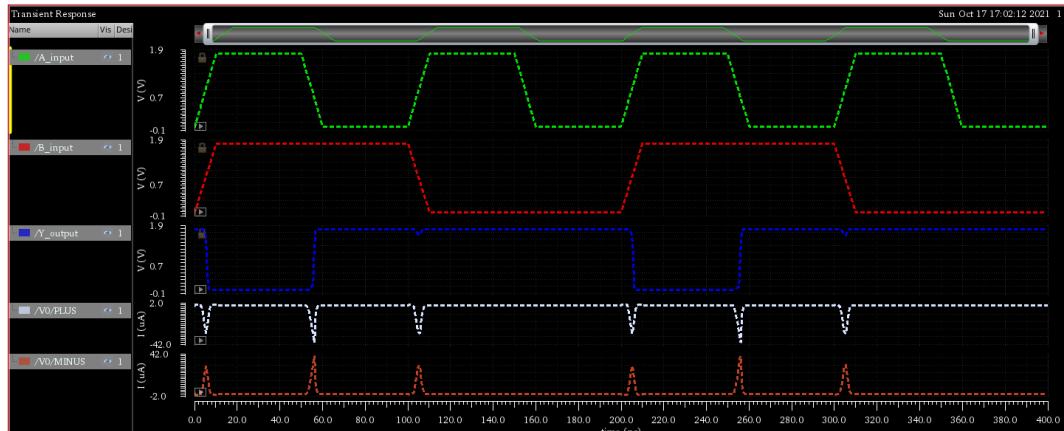


Figure 1.10: Transient response simulation of NAND2

The waveform results in Figure 1.10 are in accordance with the Boolean table for NAND2 presented in Subsection 1.2.1.

### 1.2.8 Netlist report

```

// Library name: LOGIC180
// Cell name: NAND2
// View name: schematic
subckt NAND2 A B Y inh_vdd
    M1 (net15 B 0 0) nch l=180.0n w=450.0n m=1 nf=1 sd=540.0n ad=2.16e-13 \
        as=2.16e-13 pd=1.86u ps=1.86u nrd=0.6 nrs=0.6 sa=480.0n sb=480.0n \
        sca=0 scb=0 scc=0
    M0 (Y A net15 0) nch l=180.0n w=450.0n m=1 nf=1 sd=540.0n ad=2.16e-13 \
        as=2.16e-13 pd=1.86u ps=1.86u nrd=0.6 nrs=0.6 sa=480.0n sb=480.0n \
        sca=0 scb=0 scc=0
    M3 (Y B inh_vdd inh_vdd) pch l=180.0n w=600n m=1 nf=1 sd=540.0n \
        ad=2.88e-13 as=2.88e-13 pd=2.16u ps=2.16u nrd=0.45 nrs=0.45 \
        sa=480.0n sb=480.0n sca=0 scb=0 scc=0
    M2 (Y A inh_vdd inh_vdd) pch l=180.0n w=600n m=1 nf=1 sd=540.0n \
        ad=2.88e-13 as=2.88e-13 pd=2.16u ps=2.16u nrd=0.45 nrs=0.45 \
        sa=480.0n sb=480.0n sca=0 scb=0 scc=0
ends NAND2
// End of subcircuit definition.

// Library name: LOGIC180
// Cell name: NAND2_tb
// View name: schematic
I0 (A_input B_input Y_output vdd!) NAND2
V0 (vdd! 0) vsource dc=1.8 type=dc
V2 (A_input 0) vsource dc=1.8 type=pulse val0=0 val1=1.8 period=100n \
    rise=10n fall=10n width=40n
V1 (B_input 0) vsource dc=1.8 type=pulse val0=0 val1=1.8 period=200n \
    rise=10n fall=10n width=90n
C0 (Y_output 0) capacitor c=10f
simulatorOptions options reltol=1e-3 vabstol=1e-6 iabstol=1e-12 temp=27 \
    tnom=27 scalem=1.0 scale=1.0 gmin=1e-12 rforce=1 maxnotes=5 maxwarns=5 \
    digits=5 cols=80 pivrel=1e-3 sensfile="../psf/sens.output" \
    checklimitdest=psf
tran tran stop=400n write="spectre.ic" writefinal="spectre.fc" \
    annotate=status maxiters=5

```

Figure 1.11: Netlist for NAND2 Testbench

Netlist is created for our testbench design in preparation for performing transient analysis simulation of NAND2. The netlist describes the Testbench and instantiated NAND2 design as shown in Figure 1.11.

## 1.3 Design of NOR2

In this section the CMOS design Schematic, Testbench and Simulation results of NOR2 are presented.

### 1.3.1 Boolean Expression

$$Y = \overline{A + B}$$

A	B	Y
0	0	1
0	1	0
1	0	0
1	1	0

### 1.3.2 Karnaugh Map



Figure 1.12: NOR Karnaugh Map

$$Y = 1 \cdot \overline{A} \cdot \overline{B} + 0 \cdot B + 0 \cdot A$$

'1' terms are connected to VDD via pmos

'0' terms are connected to GND via nmos.

### 1.3.3 The Mathematical Method

Alternative method to Karnaugh Map for constructing CMOS logic gates.

**PMOS :**

$$F = \overline{A + B}$$

$$F = \overline{A} \cdot \overline{B} \text{ (Applying Demorgan's Law)}$$

$$F_p = A \cdot B \text{ (invert inputs for pmos)}$$

**NMOS :**

$$F_n = \overline{\overline{A + B}} \text{ (Invert output for nmos)}$$

$$F_n = A + B$$

### 1.3.4 CMOS Implementation

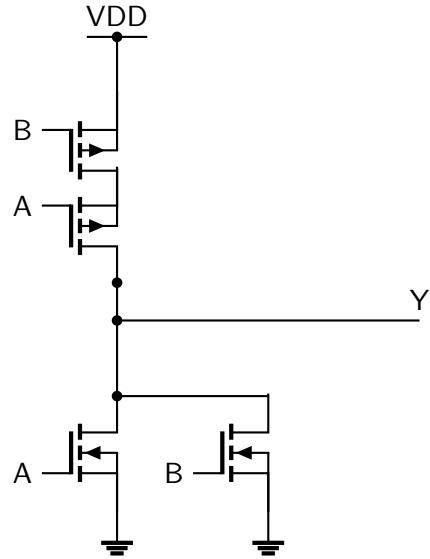


Figure 1.13: CMOS implementation of NOR2

We can observe from the Figure 1.13 the two PMOS are in series and the two NMOS are in parallel.

### 1.3.5 Schematic

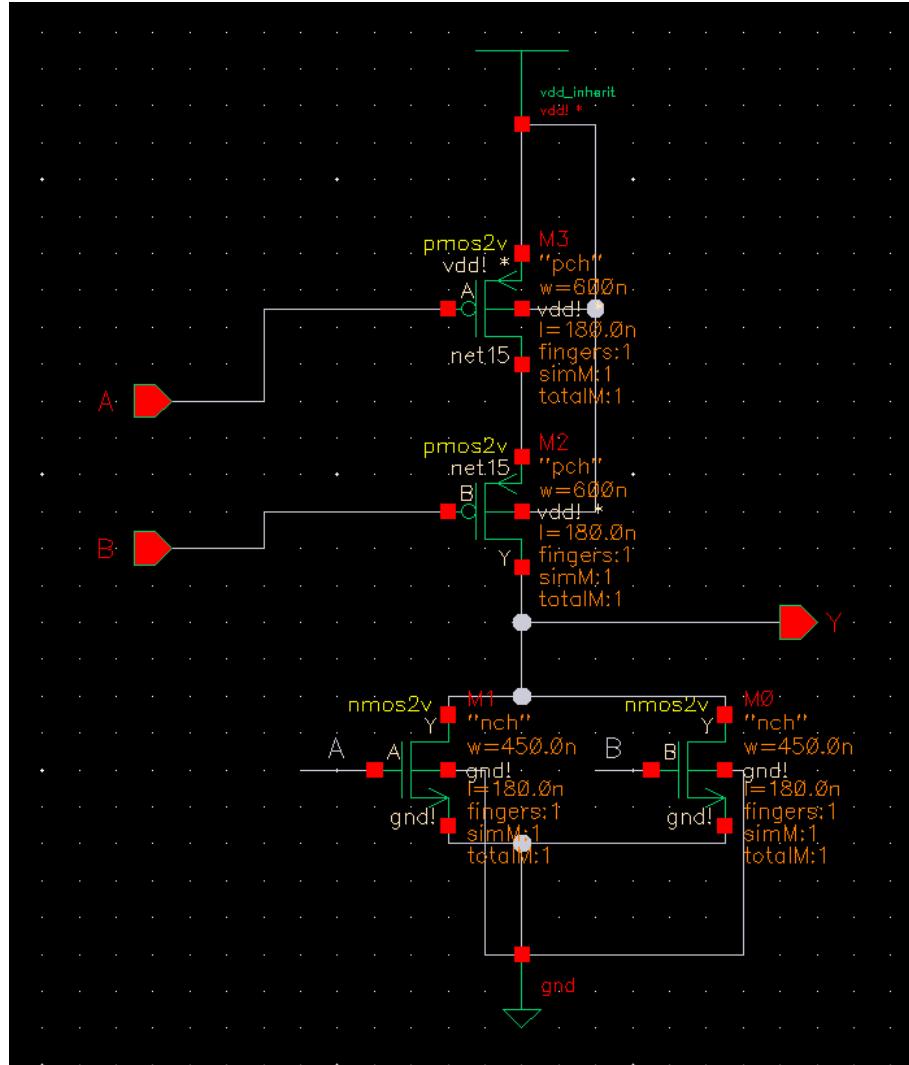


Figure 1.14: Cadence Schematic of NOR2

Figure 1.14 is the schematic realised in Cadence Virtuoso for the NOR2 design presented in Figure 1.13.

### 1.3.6 Testbench

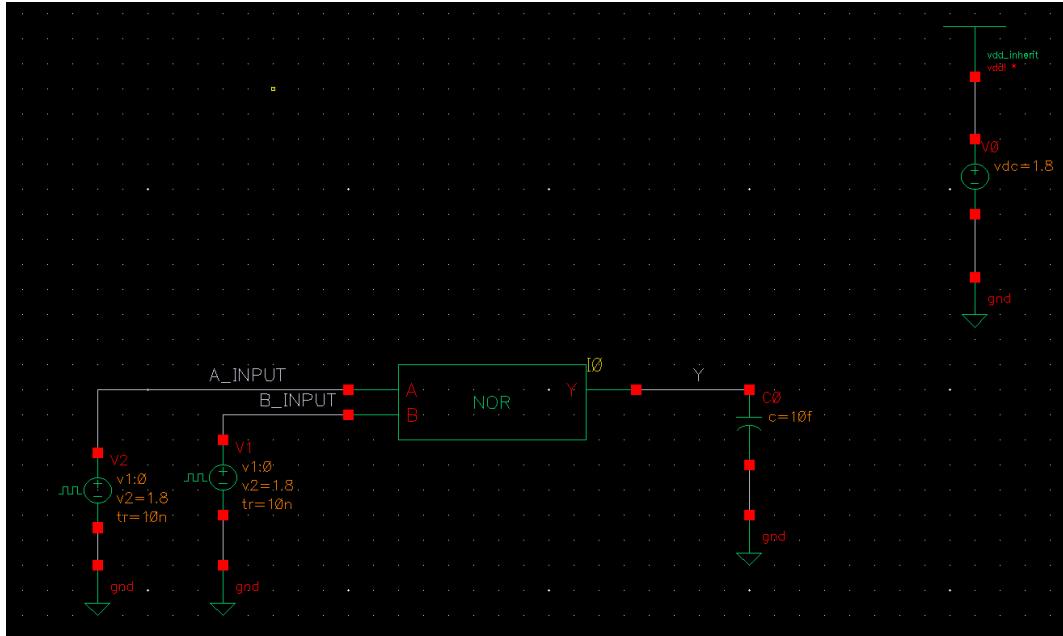


Figure 1.15: Testbench for NOR2

The testbench shown in Figure 1.15 is created for performing functional simulation of created NOR2 schematic in Figure 1.14. NOR2 is instantiated in the testbench design.

### 1.3.7 Functional Simulation

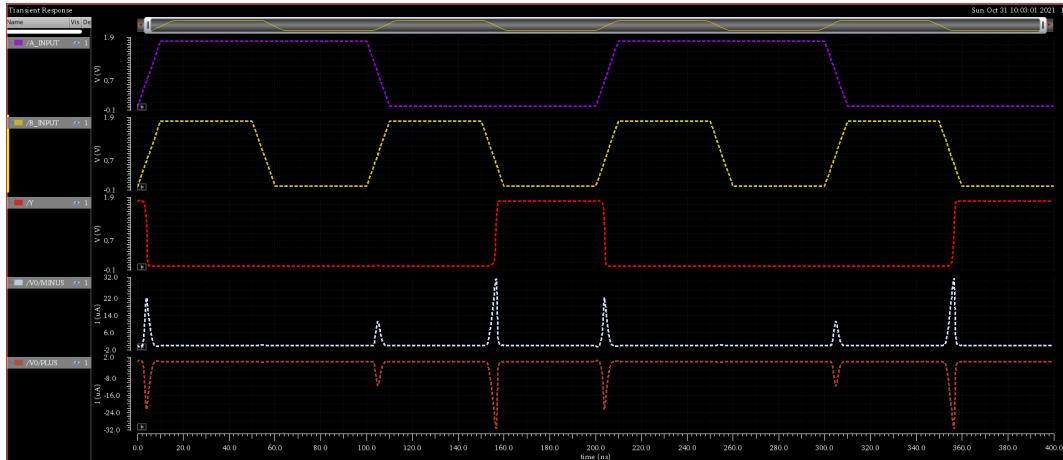


Figure 1.16: Transient response simulation of NOR2

The waveform results in Figure 1.16 are in accordance with the Boolean table for NOR2 presented in Subsection 1.3.1.

### 1.3.8 Netlist report

```

include "/home/cohort-2021/sp1822/icds2021/tsmc180/global.scs"

// Library name: LOGIC180
// Cell name: NOR
// View name: schematic
subckt NOR A B Y inh_vdd
    M1 (Y A 0 0) nch l=180.0n w=450.0n m=1 nf=1 sd=540.0n ad=2.16e-13 \
        as=2.16e-13 pd=1.86u ps=1.86u nrd=0.6 nrs=0.6 sa=480.0n sb=480.0n \
        sca=0 scb=0 scc=0
    M0 (Y B 0 0) nch l=180.0n w=450.0n m=1 nf=1 sd=540.0n ad=2.16e-13 \
        as=2.16e-13 pd=1.86u ps=1.86u nrd=0.6 nrs=0.6 sa=480.0n sb=480.0n \
        sca=0 scb=0 scc=0
    M3 (net15 A inh_vdd inh_vdd) pch l=180.0n w=600n m=1 nf=1 sd=540.0n \
        ad=2.88e-13 as=2.88e-13 pd=2.16u ps=2.16u nrd=0.45 nrs=0.45 \
        sa=480.0n sb=480.0n sca=0 scb=0 scc=0
    M2 (Y B net15 inh_vdd) pch l=180.0n w=600n m=1 nf=1 sd=540.0n \
        ad=2.88e-13 as=2.88e-13 pd=2.16u ps=2.16u nrd=0.45 nrs=0.45 \
        sa=480.0n sb=480.0n sca=0 scb=0 scc=0
ends NOR
// End of subcircuit definition.

// Library name: LOGIC180
// Cell name: NOR_tb
// View name: schematic
I0 (A_INPUT B_INPUT Y vdd!) NOR
V0 (vdd! 0) vsource dc=1.8 type=dc
V2 (A_INPUT 0) vsource dc=1.8 type=pulse val0=0 val1=1.8 period=200n \
    rise=10n fall=10n width=90n
V1 (B_INPUT 0) vsource dc=1.8 type=pulse val0=0 val1=1.8 period=100n \
    rise=10n fall=10n width=40n
C0 (Y 0) capacitor c=10f
simulatorOptions options reltol=1e-3 vabstol=1e-6 iabstol=1e-12 temp=27 \
    tnom=27 scalem=1.0 scale=1.0 gmin=1e-12 rforce=1 maxnotes=5 maxwarns=5 \
    digits=5 cols=80 pivrel=1e-3 sensfile="../psf/sens.output" \
    checklimitdest=psf
tran tran stop=400n write="spectre.ic" writefinal="spectre.fc" \
    annotate=status maxiters=5
finalTimeOP info what=oppoint where=rawfile
modelParameter info what=models where=rawfile
element info what=inst where=rawfile
outputParameter info what=output where=rawfile
designParamVals info what=parameters where=rawfile
primitives info what=primitives where=rawfile
subckts info what=subckts where=rawfile
save V0:p V0:p
saveOptions options save=allpub

```

Figure 1.17: Netlist for NOR2 Testbench

Netlist is created for our testbench design in preparation for performing transient analysis simulation of NOR2. The netlist describes the Testbench and instantiated NOR2 design as shown in Figure 1.17.

## 1.4 Design of XOR

In this section the CMOS design Schematic, Testbench and Simulation results of XOR are presented.

### 1.4.1 Boolean Expression

$$Y = A \cdot \overline{B} + B \cdot \overline{A}$$

A	B	Y
0	0	0
0	1	1
1	0	1
1	1	0

### 1.4.2 Karnaugh Map



Figure 1.18: XOR Karnaugh Map

$$Y = 1 \cdot \overline{A} \cdot B + 1 \cdot \overline{B} \cdot A + 0 \cdot A \cdot B + 0 \cdot \overline{A} \cdot \overline{B}$$

'1' terms are connected to VDD via pmos

'0' terms are connected to GND via nmos.

### 1.4.3 The Mathematical Method

Alternative method to Karnaugh Map for constructing CMOS logic gates.

**PMOS :**

$$F_p = \overline{A} \cdot B + \overline{B} \cdot A$$

**NMOS :**

$$F_n = \overline{\overline{A} \cdot B + \overline{B} \cdot A} \text{ (Invert output for nmos)}$$

$$F_p = \overline{B} \cdot A + \overline{A} \cdot B \text{ (invert inputs for pmos)} \quad F_n = A \cdot B + \overline{A} \cdot \overline{B}$$

#### 1.4.4 CMOS Implementation

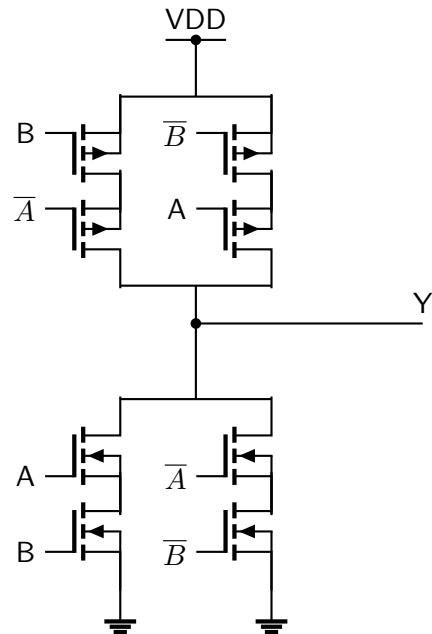


Figure 1.19: CMOS implementation of XOR

We can observe from the Figure 1.19 that four PMOS transistors and four NMOS transistors are used, but you need 2 Inverters to generate  $\bar{A}$  and  $\bar{B}$ . In total 12 transistors(6 PMOS and 6 NMOS) are used to realise the logic.

### 1.4.5 Schematic

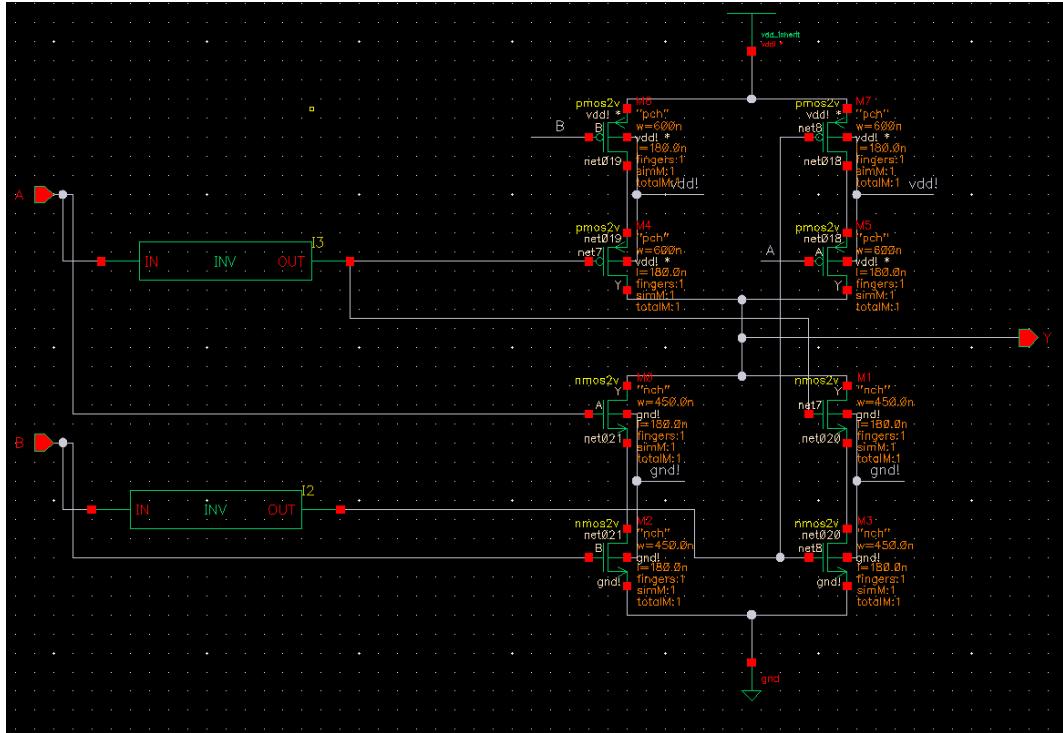


Figure 1.20: Cadence Schematic of XOR

Figure 1.20 is the schematic realised in Cadence Virtuoso for the XOR design presented in Figure 1.19. It is obvious from the Figure 1.20 that 2 Inverters (Inverter Schematic presented in Figure 1.2) are used for generating  $\bar{A}$  and  $\bar{B}$  signals.

### 1.4.6 Testbench

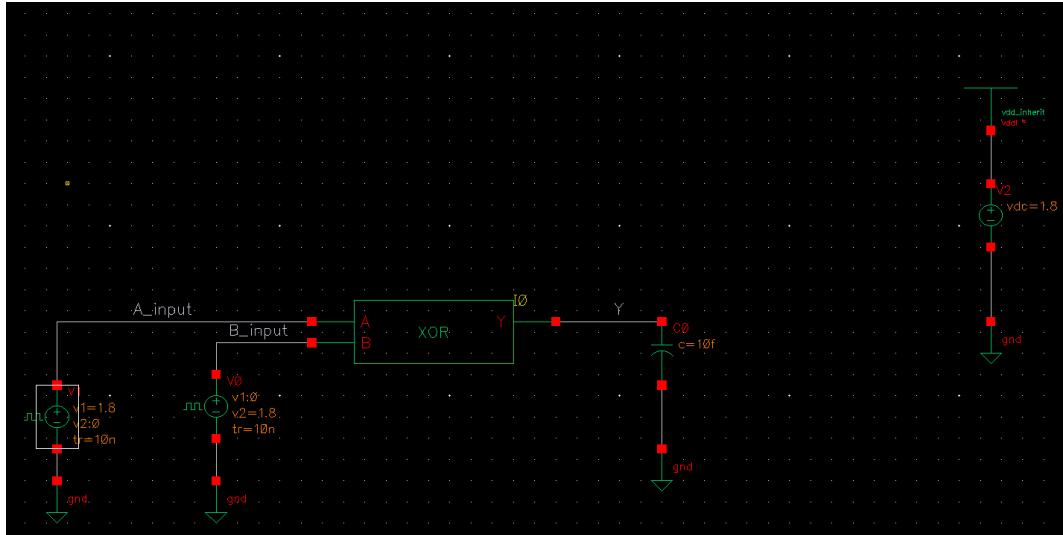


Figure 1.21: Testbench for XOR

The testbench shown in Figure 1.21 is created for performing functional simulation of created XOR schematic in Figure 1.20. XOR is instantiated in the testbench design.

### 1.4.7 Functional Simulation

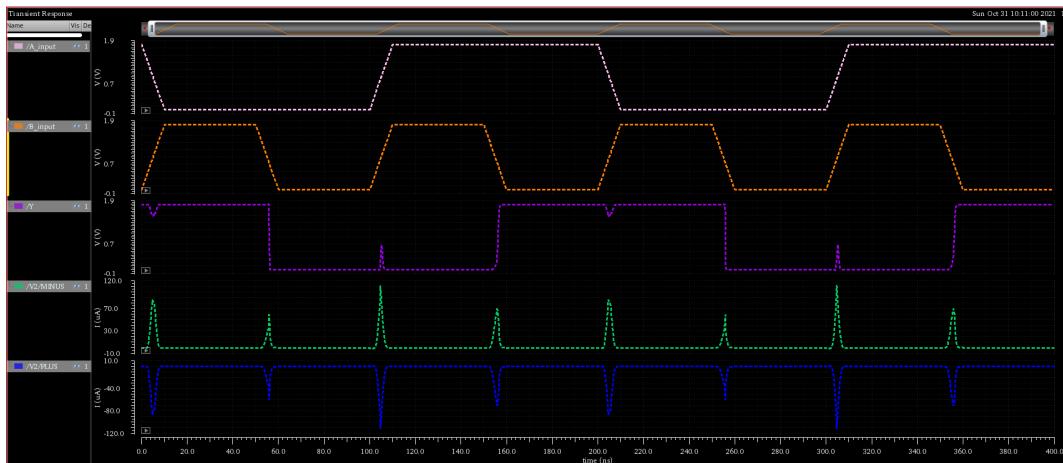


Figure 1.22: Transient response simulation of XOR

The waveform results in Figure 1.22 are in accordance with the Boolean table for XOR presented in Subsection 1.4.1.

### 1.4.8 Netlist report

```

include "/home/cohort-2021/sp1822/icds2021/tsmc180/global.scs"

// Library name: LOGIC180
// Cell name: INV
// View name: schematic
subckt INV IN OUT inh_vdd
    M0 (OUT IN 0 0) nch l=180.0n w=450.0n m=1 nf=1 sd=540.0n ad=2.16e-13 \
        as=2.16e-13 pd=1.86u ps=1.86u nrd=0.6 nrs=0.6 sa=480.0n sb=480.0n \
        sca=0 scb=0 scc=0
    M1 (OUT IN inh_vdd inh_vdd) pch l=180.0n w=600n m=1 nf=1 sd=540.0n \
        ad=2.88e-13 as=2.88e-13 pd=2.16u ps=2.16u nrd=0.45 nrs=0.45 \
        sa=480.0n sb=480.0n sca=0 scb=0 scc=0
ends INV
// End of subcircuit definition.

// Library name: LOGIC180
// Cell name: XOR
// View name: schematic
subckt XOR A B Y inh_vdd
    M5 (Y A net018 inh_vdd) pch l=180.0n w=600n m=1 nf=1 sd=540.0n \
        ad=2.88e-13 as=2.88e-13 pd=2.16u ps=2.16u nrd=0.45 nrs=0.45 \
        sa=480.0n sb=480.0n sca=0 scb=0 scc=0
    M7 (net018 net8 inh_vdd inh_vdd) pch l=180.0n w=600n m=1 nf=1 \
        sd=540.0n ad=2.88e-13 as=2.88e-13 pd=2.16u ps=2.16u nrd=0.45 \
        nrs=0.45 sa=480.0n sb=480.0n sca=0 scb=0 scc=0
    M6 (net019 B inh_vdd inh_vdd) pch l=180.0n w=600n m=1 nf=1 sd=540.0n \
        ad=2.88e-13 as=2.88e-13 pd=2.16u ps=2.16u nrd=0.45 nrs=0.45 \
        sa=480.0n sb=480.0n sca=0 scb=0 scc=0
    M4 (Y net7 net019 inh_vdd) pch l=180.0n w=600n m=1 nf=1 sd=540.0n \
        ad=2.88e-13 as=2.88e-13 pd=2.16u ps=2.16u nrd=0.45 nrs=0.45 \
        sa=480.0n sb=480.0n sca=0 scb=0 scc=0
    M8 (Y A net021 0) nch l=180.0n w=450.0n m=1 nf=1 sd=540.0n ad=2.16e-13 \
        as=2.16e-13 pd=1.86u ps=1.86u nrd=0.6 nrs=0.6 sa=480.0n sb=480.0n \
        sca=0 scb=0 scc=0
    M3 (net020 net8 0 0) nch l=180.0n w=450.0n m=1 nf=1 sd=540.0n \
        ad=2.16e-13 as=2.16e-13 pd=1.86u ps=1.86u nrd=0.6 nrs=0.6 \
        sa=480.0n sb=480.0n sca=0 scb=0 scc=0
    M2 (net021 B 0 0) nch l=180.0n w=450.0n m=1 nf=1 sd=540.0n ad=2.16e-13 \
        as=2.16e-13 pd=1.86u ps=1.86u nrd=0.6 nrs=0.6 sa=480.0n sb=480.0n \
        sca=0 scb=0 scc=0
    M1 (Y net7 net020 0) nch l=180.0n w=450.0n m=1 nf=1 sd=540.0n \
        ad=2.16e-13 as=2.16e-13 pd=1.86u ps=1.86u nrd=0.6 nrs=0.6 \
        sa=480.0n sb=480.0n sca=0 scb=0 scc=0
    I3 (A net7 inh_vdd) INV
    I2 (B net8 inh_vdd) INV
ends XOR
// End of subcircuit definition.

```

Figure 1.23: Netlist for XOR Testbench -1

Netlist is created for our testbench design in preparation for performing transient analysis simulation of XOR. The netlist describes the Testbench and instantiated XOR design as shown in Figure 1.23 and 1.24. The Netlist is broken into two parts to increase readability.

```
// Library name: LOGIC180
// Cell name: XOR_tb
// View name: schematic
I0 (A_input B_input Y vdd!) XOR
V1 (A_input 0) vsource dc=1.8 type=pulse val0=1.8 val1=0 period=200n \
    rise=10n fall=10n width=90n
V0 (B_input 0) vsource dc=1.8 type=pulse val0=0 val1=1.8 period=100n \
    rise=10n fall=10n width=40n
C0 (Y 0) capacitor c=10f
V2 (vdd! 0) vsource dc=1.8 type=dc
simulatorOptions options reltol=1e-3 vabstol=1e-6 iabstol=1e-12 temp=27 \
    tnom=27 scalem=1.0 scale=1.0 gmin=1e-12 rforce=1 maxnotes=5 maxwarns=5 \
    digits=5 cols=80 pivrel=1e-3 sensfile="../psf/sens.output" \
    checklimitdest=psf
tran tran stop=400n errpreset=conservative write="spectre.ic" \
    writefinal="spectre.fc" annotate=status maxiters=5
finalTimeOP info what=oppont where=rawfile
modelParameter info what=models where=rawfile
element info what=inst where=rawfile
outputParameter info what=output where=rawfile
designParamVals info what=parameters where=rawfile
primitives info what=primitives where=rawfile
subckts info what=subckts where=rawfile
save V2:p V2:p
saveOptions options save=allpub
```

Figure 1.24: Netlist for XOR Testbench -2

## 1.5 Design of function: $F = (A \cdot B) + (C \cdot D)$

In this section the CMOS design Schematic, Testbench and Simulation results for  $F = (A \cdot B) + (C \cdot D)$  are presented.

### 1.5.1 Boolean Expression

$$F = A \cdot B + C \cdot D$$

A	B	C	D	F
0	0	0	0	0
0	0	0	1	0
0	0	1	0	0
0	0	1	1	1
0	1	0	0	0
0	1	0	1	0
0	1	1	0	0
0	1	1	1	1
1	0	0	0	0
1	0	0	1	0
1	0	1	0	0
1	0	1	1	1
1	1	0	0	1
1	1	0	1	1
1	1	1	0	1
1	1	1	1	1

### 1.5.2 Karnaugh Map

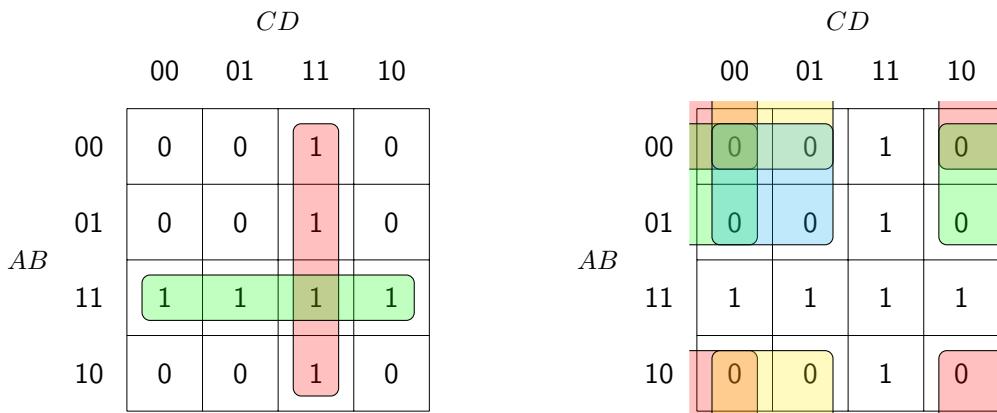


Figure 1.25: Karnaugh Map

$$F = 1 \cdot A \cdot B + 1 \cdot C \cdot D + 0 \cdot \bar{C} \cdot \bar{A} + 0 \cdot \bar{B} \cdot \bar{D} + 0 \cdot \bar{B} \cdot \bar{C} + 0 \cdot \bar{D} \cdot \bar{A}$$

'1' terms are connected to VDD via pmos

'0' terms are connected to GND via nmos.

### 1.5.3 The Mathematical Method

Alternative method to Karnaugh Map for constructing CMOS logic gates.

**PMOS :**

$$F = A \cdot B + C \cdot D$$

$$F_p = \overline{A} \cdot \overline{B} + \overline{C} \cdot \overline{D} \text{ (invert inputs for pmos)}$$

$$F_n = \overline{\overline{A} \cdot B + C \cdot D} \text{ (Invert output for nmos)}$$

$$F_n = (\overline{A} \cdot \overline{B}) \cdot (\overline{C} \cdot \overline{D})$$

$$F_n = \overline{C} \cdot \overline{A} + \overline{B} \cdot \overline{D} + \overline{B} \cdot \overline{C} + \overline{D} \cdot \overline{A}$$

From the above equations it is evident that you need four inverters for constructing our CMOS design of the function  $F = (A \cdot B) + (C \cdot D)$ .

Now consider the alternate representation of the above function:

$$\overline{\overline{F}} = (A \cdot B) + (C \cdot D) = F$$

If we construct CMOS logic for  $\overline{\overline{F}} = \overline{(A \cdot B) + (C \cdot D)}$  and add an Inverter at the end , it represents our original function.

### 1.5.4 Boolean Expression for $Y = \overline{\overline{F}} = \overline{(A \cdot B) + (C \cdot D)}$

$$Y = \overline{(A \cdot B) + (C \cdot D)}$$

A	B	C	D	Y
0	0	0	0	1
0	0	0	1	1
0	0	1	0	1
0	0	1	1	0
0	1	0	0	1
0	1	0	1	1
0	1	1	0	1
0	1	1	1	0
1	0	0	0	1
1	0	0	1	1
1	0	1	0	1
1	0	1	1	0
1	1	0	0	0
1	1	0	1	0
1	1	1	0	0
1	1	1	1	0

### 1.5.5 Karnaugh Map

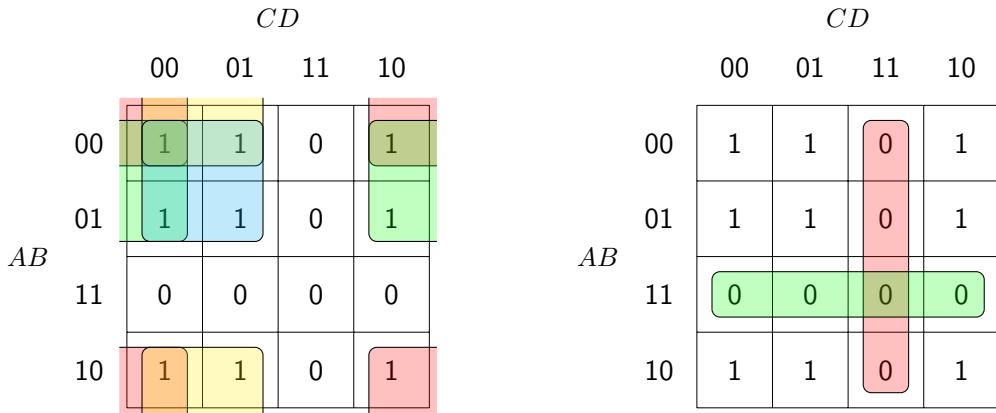


Figure 1.26: Karnaugh Map

$$Y = 0 \cdot A \cdot B + 0 \cdot C \cdot D + 1 \cdot \bar{C} \cdot \bar{A} + 1 \cdot \bar{B} \cdot \bar{D} + 1 \cdot \bar{B} \cdot \bar{C} + 1 \cdot \bar{D} \cdot \bar{A}$$

'1' terms are connected to VDD via pmos

'0' terms are connected to GND via nmos.

### 1.5.6 The Mathematical Method

Alternative method to Karnaugh Map for constructing CMOS logic gates.

**PMOS :**

$$Y = \overline{(A \cdot B) + (C \cdot D)}$$

$$Y = \overline{(A \cdot B)} \cdot \overline{(C \cdot D)} \text{ (Applying Demorgan's law)}$$

$$Y = (\bar{A} + \bar{B}) \cdot (\bar{C} + \bar{D}) \text{ (Applying Demorgan's law)}$$

$$Y_p = (A+B) \cdot (C+D) \text{ (invert inputs for pmos)}$$

**NMOS :**

$$Y_n = \overline{\overline{(A \cdot B) + (C \cdot D)}} \text{ (Invert output for nmos)}$$

$$Y_n = A \cdot B + C \cdot D$$

By implementing the CMOS design this way, there is a reduction of three inverters( six transistors).

### 1.5.7 CMOS Implementation

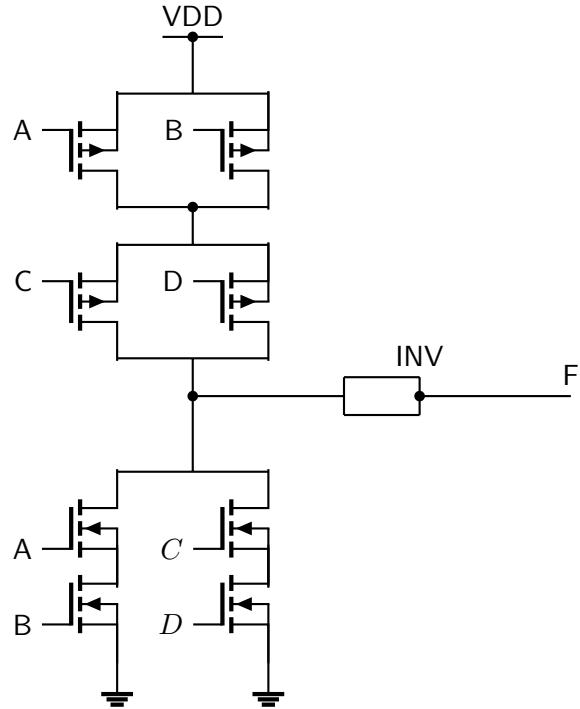


Figure 1.27: CMOS implementation of  $F = A \cdot B + C \cdot D$

It is obvious from the Figure 1.27 that total 10 transistors (5 NMOS transistors and 5 PMOS transistors) are required in realising the logic. There is a reduction of 6 transistors when compared to the direct implementation of the logic function.

### 1.5.8 Schematic

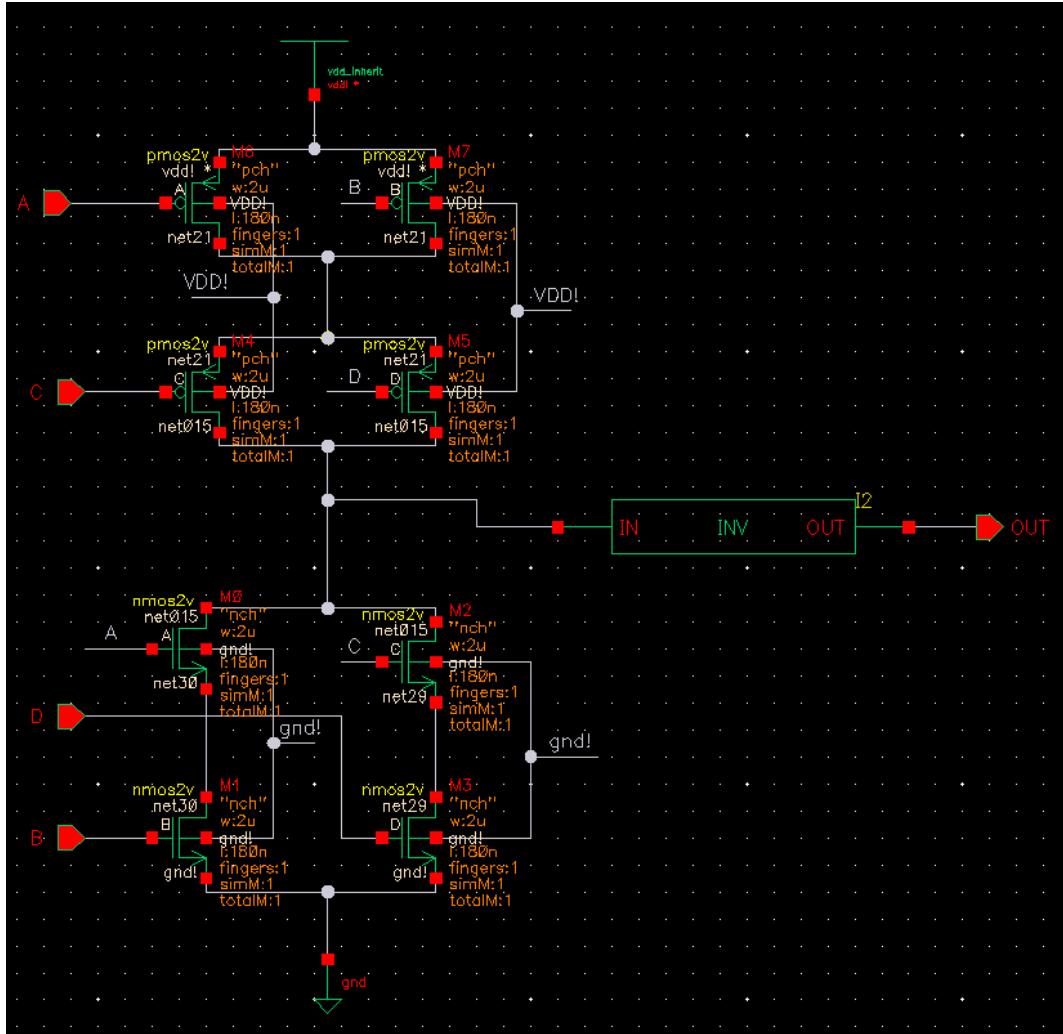


Figure 1.28: Cadence schematic of  $F = A \cdot B + C \cdot D$

Figure 1.28 is the schematic realised in Cadence Virtuoso for the function design presented in Figure 1.27. An inverter has been instantiated (Inverter Schematic presented in Figure 1.2).

### 1.5.9 Testbench

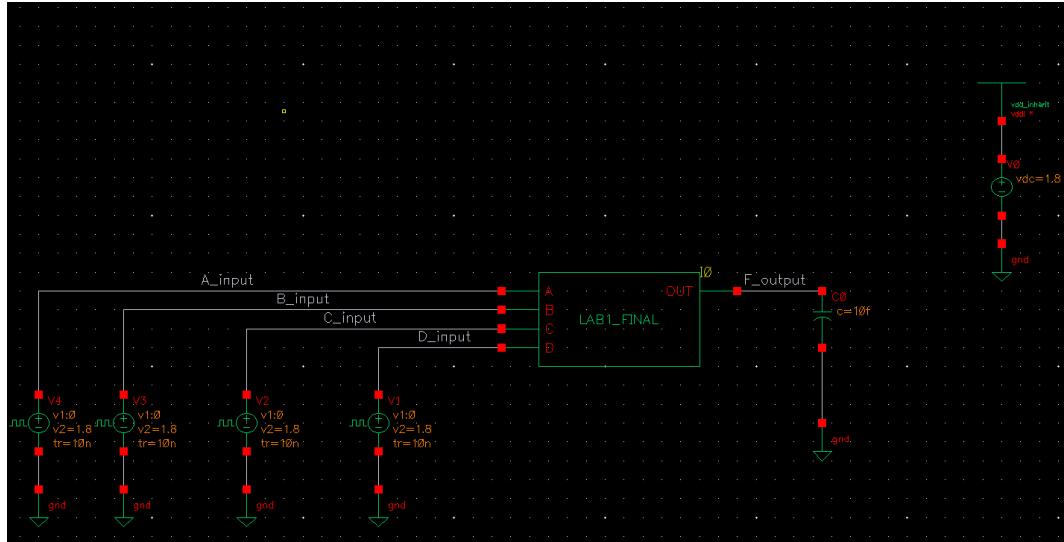


Figure 1.29: Testbench for  $F = A \cdot B + C \cdot D$

The testbench shown in Figure 1.29 is created for performing functional simulation of created function schematic in Figure 1.28. The function schematic is instantiated in the testbench design.

### 1.5.10 Functional Simulation

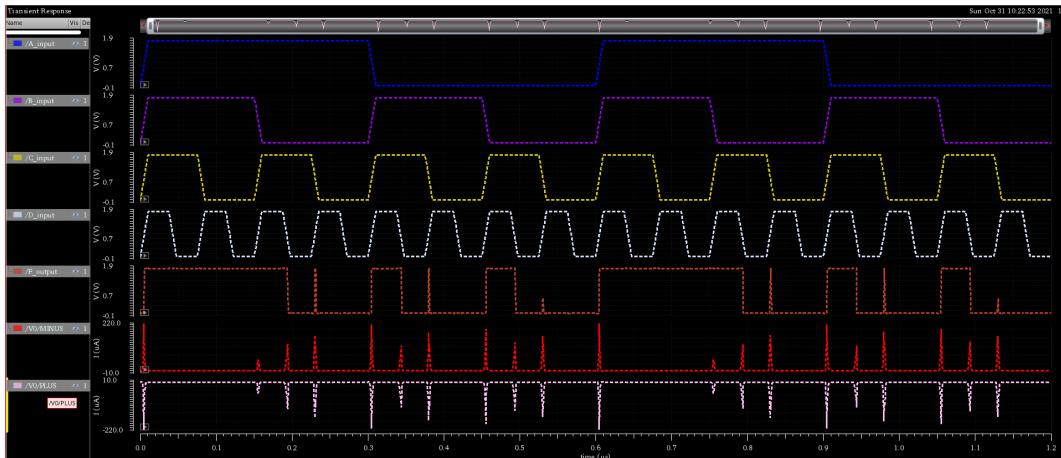


Figure 1.30: Transient response simulation of  $F = A \cdot B + C \cdot D$

The waveform results in Figure 1.30 are in accordance with the Boolean table for the function presented in Subsection 1.5.1.

### 1.5.11 Netlist report

```

include "/home/cohort-2021/sp1822/icds2021/tsmc180/global.scs"

// Library name: LOGIC180
// Cell name: INV
// View name: schematic
subckt INV IN OUT inh_vdd
    M0 (OUT IN 0 0) nch l=180.0n w=450.0n m=1 nf=1 sd=540.0n ad=2.16e-13 \
        as=2.16e-13 pd=1.86u ps=1.86u nrd=0.6 nrs=0.6 sa=480.0n sb=480.0n \
        sca=0 scb=0 scc=0
    M1 (OUT IN inh_vdd inh_vdd) pch l=180.0n w=600n m=1 nf=1 sd=540.0n \
        ad=2.88e-13 as=2.88e-13 pd=2.16u ps=2.16u nrd=0.45 nrs=0.45 \
        sa=480.0n sb=480.0n sca=0 scb=0 scc=0
ends INV
// End of subcircuit definition.

// Library name: LOGIC180
// Cell name: LAB1_FINAL
// View name: schematic
subckt LAB1_FINAL A B C D OUT inh_vdd
    M3 (net29 D 0 0) nch l=180n w=2u m=1 nf=1 sd=540n ad=9.6e-13 \
        as=9.6e-13 pd=4.96u ps=4.96u nrd=0.135 nrs=0.135 sa=480n sb=480n \
        sca=0 scb=0 scc=0
    M2 (net015 C net29 0) nch l=180n w=2u m=1 nf=1 sd=540n ad=9.6e-13 \
        as=9.6e-13 pd=4.96u ps=4.96u nrd=0.135 nrs=0.135 sa=480n sb=480n \
        sca=0 scb=0 scc=0
    M1 (net30 B 0 0) nch l=180n w=2u m=1 nf=1 sd=540n ad=9.6e-13 \
        as=9.6e-13 pd=4.96u ps=4.96u nrd=0.135 nrs=0.135 sa=480n sb=480n \
        sca=0 scb=0 scc=0
    M0 (net015 A net30 0) nch l=180n w=2u m=1 nf=1 sd=540n ad=9.6e-13 \
        as=9.6e-13 pd=4.96u ps=4.96u nrd=0.135 nrs=0.135 sa=480n sb=480n \
        sca=0 scb=0 scc=0
    M7 (net21 B inh_vdd VDD!) pch l=180n w=2u m=1 nf=1 sd=540n ad=9.6e-13 \
        as=9.6e-13 pd=4.96u ps=4.96u nrd=0.135 nrs=0.135 sa=480n sb=480n \
        sca=0 scb=0 scc=0
    M6 (net21 A inh_vdd VDD!) pch l=180n w=2u m=1 nf=1 sd=540n ad=9.6e-13 \
        as=9.6e-13 pd=4.96u ps=4.96u nrd=0.135 nrs=0.135 sa=480n sb=480n \
        sca=0 scb=0 scc=0
    M5 (net015 D net21 VDD!) pch l=180n w=2u m=1 nf=1 sd=540n ad=9.6e-13 \
        as=9.6e-13 pd=4.96u ps=4.96u nrd=0.135 nrs=0.135 sa=480n sb=480n \
        sca=0 scb=0 scc=0
    M4 (net015 C net21 VDD!) pch l=180n w=2u m=1 nf=1 sd=540n ad=9.6e-13 \
        as=9.6e-13 pd=4.96u ps=4.96u nrd=0.135 nrs=0.135 sa=480n sb=480n \
        sca=0 scb=0 scc=0
    I2 (net015 OUT inh_vdd) INV
ends LAB1_FINAL
// End of subcircuit definition.

```

Figure 1.31: Netlist of  $F = A \cdot B + C \cdot D$  Testbench -1

Netlist is created for our testbench design in preparation for performing transient analysis simulation of the function. The netlist describes the Testbench and instantiated function design as shown in Figure 1.31 and 1.32. The Netlist is broken into two parts to increase readability.

```
// Library name: LOGIC180
// Cell name: LAB1_FINAL_tb
// View name: schematic
I0 (A_input B_input C_input D_input F_output vdd!) LAB1_FINAL
C0 (F_output 0) capacitor c=10f
V0 (vdd! 0) vsource dc=1.8 type=dc
V4 (A_input 0) vsource dc=1.8 type=pulse val0=0 val1=1.8 period=600n \
    rise=10n fall=10n width=290n
V3 (B_input 0) vsource dc=1.8 type=pulse val0=0 val1=1.8 period=300n \
    rise=10n fall=10n width=140n
V2 (C_input 0) vsource dc=1.8 type=pulse val0=0 val1=1.8 period=150n \
    rise=10n fall=10n width=65n
V1 (D_input 0) vsource dc=1.8 type=pulse val0=0 val1=1.8 period=75n \
    rise=10n fall=10n width=27.5n
simulatorOptions options reltol=1e-3 vabstol=1e-6 iabstol=1e-12 temp=27 \
    tnom=27 scalem=1.0 scale=1.0 gmin=1e-12 rforce=1 maxnotes=5 maxwarns=5 \
    digits=5 cols=80 pivrel=1e-3 sensfile="../psf/sens.output" \
    checklimitdest=psf
tran tran stop=1200n errpreset=conservative write="spectre.ic" \
    writefinal="spectre.fc" annotate=status maxiters=5
finalTimeOP info what=oppont where=rawfile
modelParameter info what=models where=rawfile
element info what=inst where=rawfile
outputParameter info what=output where=rawfile
designParamVals info what=parameters where=rawfile
primitives info what=primitives where=rawfile
subckts info what=subckts where=rawfile
save V0:p V0:p
saveOptions options save=allpub
```

Figure 1.32: Netlist of  $F = A \cdot B + C \cdot D$  Testbench -2

## 1.6 SPICE simulation

SPICE is a computer simulation and modeling program used to mathematically predict the behavior of electronics circuits. It stands for "Simulation Program with Integrated Circuit Emphasis". A circuit needs to be converted into a netlist which is used by SPICE for simulating the behaviour of the circuit. This is the reason we create a netlist of all our designs before simulation.

# Chapter 2

## LAB 2

### 2.1 DRS, LVS of INVERTER

In this section the results of Layout, Design Rule Check (DRC) and Layout vs Schematic(LVS) of Inverter are presented.

#### 2.1.1 LAYOUT

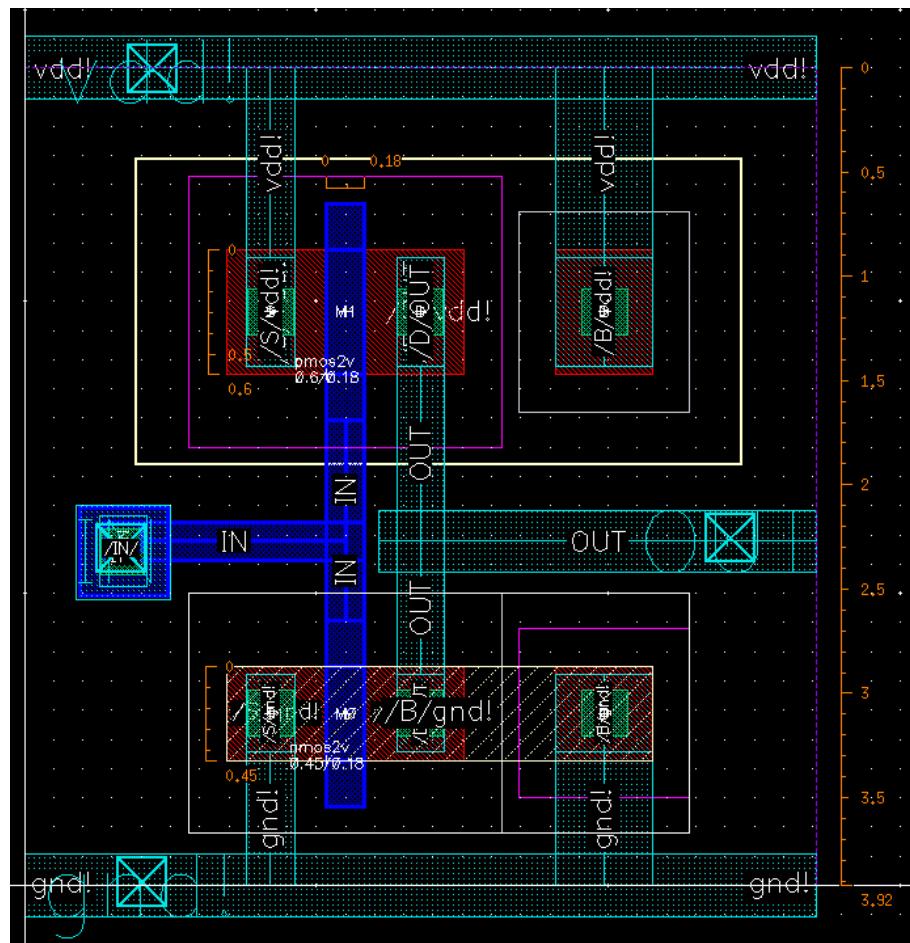


Figure 2.1: Layout of Inverter

A layout view is created from the Inverter design schematic shown in 1.2. Layout view is presented in Figure 2.1. The errors are related to the density.

### 2.1.2 DRC

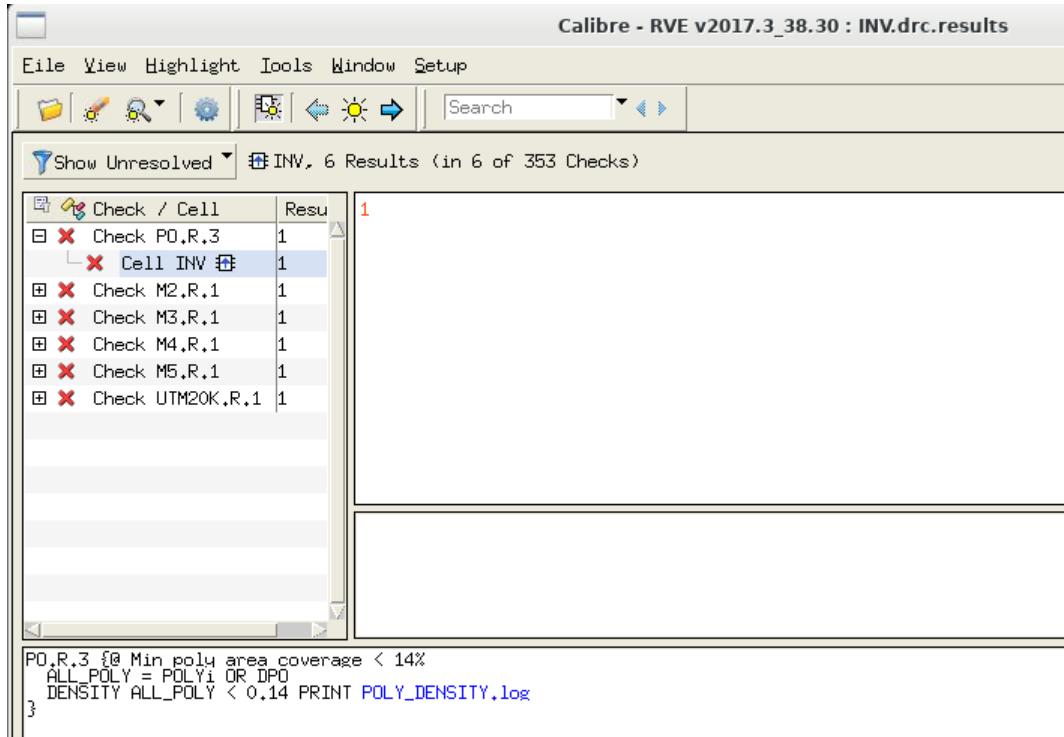


Figure 2.2: DRC of Inverter design

Design rule check is performed on the layout Figure 2.1 of the Inverter. The resulting errors are presented in Figure 2.2.

### 2.1.3 LVS

Comparison Results x }				
Layout Cell / Type	Source Cell	Nets	Instances	Ports
INV	INV	4L, 4S	1L, 1S	4L, 4S
Cell INV Summary (Clean)				
# #####				
LAYOUT CELL NAME:	INV			
SOURCE CELL NAME:	INV			
-----				
INITIAL NUMBERS OF OBJECTS				
-----				
Ports:	4	4		
Nets:	4	4		
Instances:	1	1	MN (4 pins) MP (4 pins)	
Total Inst:	2	2		
-----				
NUMBERS OF OBJECTS AFTER TRANSFORMATION				
-----				
Ports:	4	4		
Nets:	4	4		
Instances:	1	1	INV (2 pins): output input	
Total Inst:	1	1		
*****				

Figure 2.3: LVS of Inverter design

Layout vs Schematic is performed on the layout Figure 2.1 of the Inverter. The results of LVS are presented in 2.3.

## 2.2 DRS,LVS of NAND2

In this section the results of Layout, Design Rule Check (DRC) and Layout vs Schematic(LVS) of NAND2 design are presented.

### 2.2.1 LAYOUT

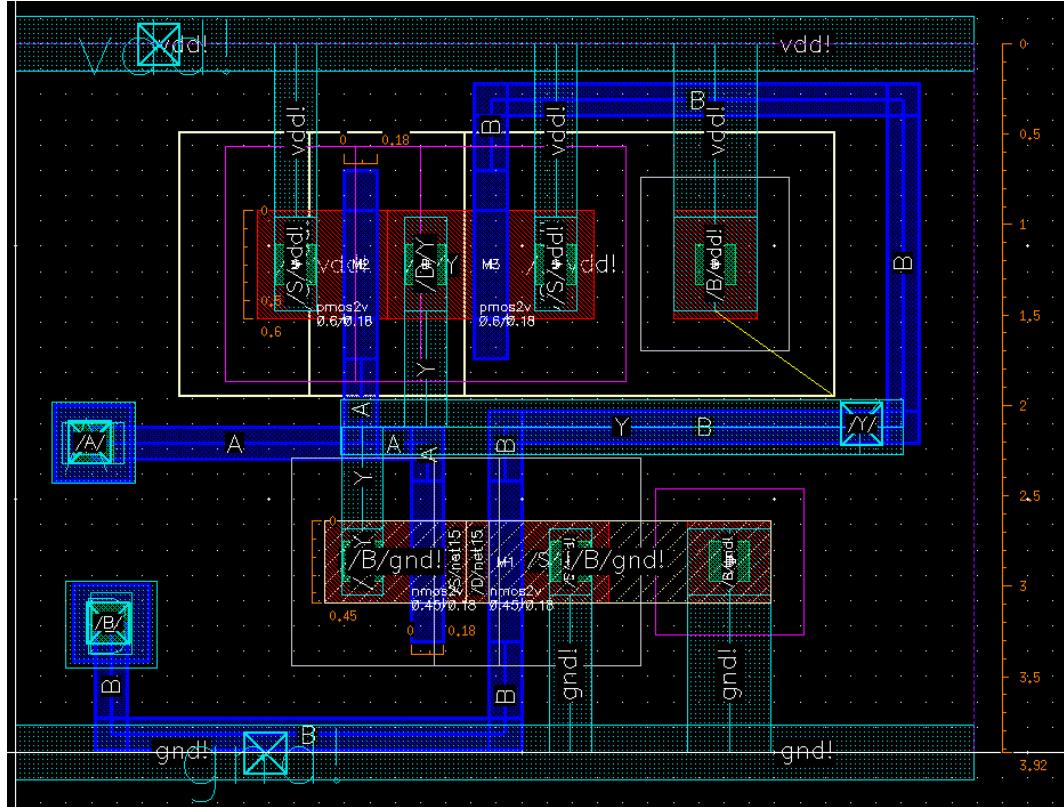


Figure 2.4: Layout of NAND2

A layout view is created from the NAND2 design schematic shown in 1.8. Layout view is presented in Figure 2.4.

### 2.2.2 DRC

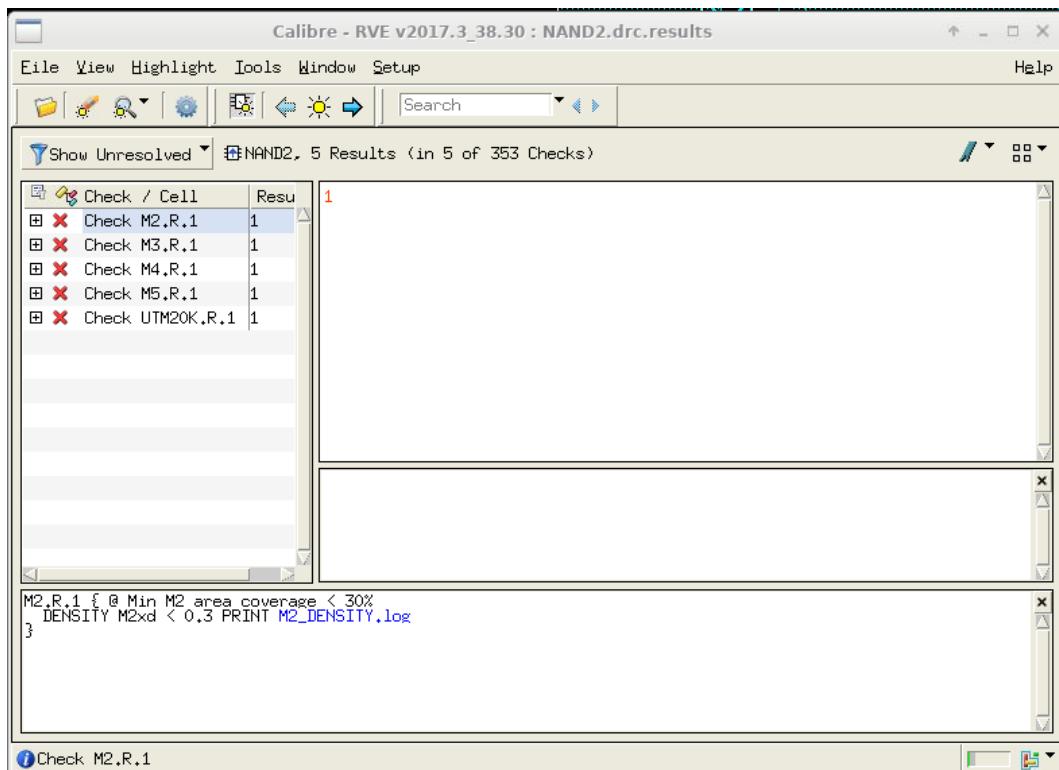


Figure 2.5: DRC of NAND2 design

Design rule check is performed on the layout Figure 2.4 of the NAND2. The resulting errors are presented in Figure 2.5.

### 2.2.3 LVS

Comparison Results		LVS Report		
Layout Cell / Type	Source Cell	Nets	Instances	Ports
NAND2	NAND2	5L, 5S	1L, 1S	5L, 5S

Cell NAND2 Summary (Clean)				
INITIAL NUMBERS OF OBJECTS				
Ports:	5	5	Component	Type
Nets:	6	6		
Instances:	2	2	MN (4 pins)	
	2	2	MP (4 pins)	
Total Inst:	4	4		

NUMBERS OF OBJECTS AFTER TRANSFORMATION				
Ports:	5	5	Component	Type
Nets:	5	5		
Instances:	1	1	NAND2 (3 pins): output input input	
Total Inst:	1	1		

Figure 2.6: LVS of NAND2 design

Layout vs Schematic is performed on the layout Figure 2.4 of the NAND2. The results of LVS are presented in 2.6.

## 2.3 DRS,LVS of NOR2

In this section the results of Layout, Design Rule Check (DRC) and Layout vs Schematic(LVS) of NOR2 are presented.

### 2.3.1 LAYOUT

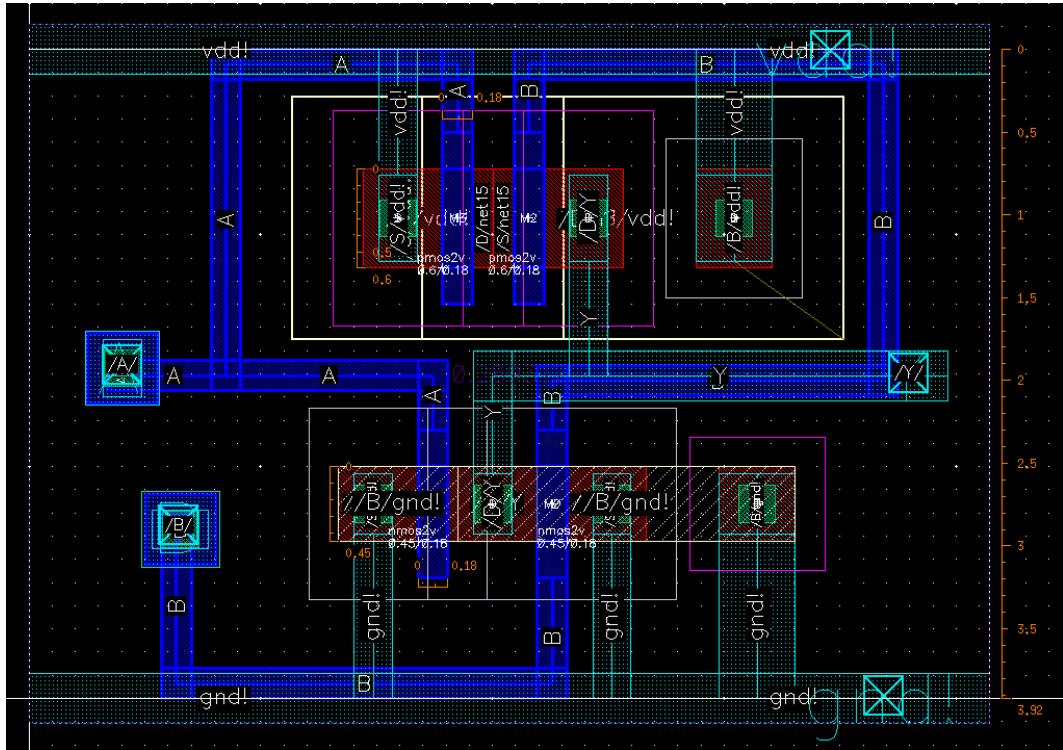


Figure 2.7: Layout of NOR2

A layout view is created from the NOR2 design schematic shown in 1.14. Layout view is presented in Figure 2.7.

### 2.3.2 DRC

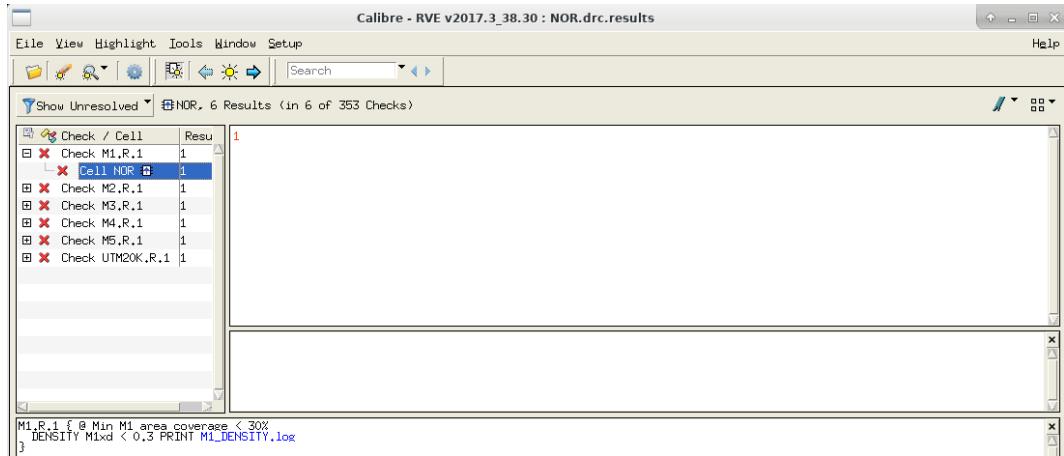


Figure 2.8: DRC of NOR2 design

Design rule check is performed on the layout Figure 2.7 of the NOR2. The resulting errors are presented in Figure 2.8.

### 2.3.3 LVS

Comparison Results x				
Layout Cell / Type	Source Cell	Nets	Instances	Ports
NOR #	NOR	5L, 5S	1L, 1S	5L, 5S

Cell NOR Summary (Clean)				
LAYOUT CELL NAME:	NOR	SOURCE CELL NAME:	NOR	

---

INITIAL NUMBERS OF OBJECTS				
	Layout	Source	Component	Type
Ports:	5	5		
Nets:	6	6		
Instances:	2	2	MN (4 pins) MP (4 pins)	
Total Inst:	4	4		

---

NUMBERS OF OBJECTS AFTER TRANSFORMATION				
	Layout	Source	Component	Type
Ports:	5	5		
Nets:	5	5		
Instances:	1	1	NOR2 (3 pins)	
Total Inst:	1	1		

---

Figure 2.9: LVS of NOR2 design

Layout vs Schematic is performed on the layout Figure 2.7 of the NOR2. The results of LVS are presented in 2.9.

## 2.4 DRS,LVS of XOR

In this section the results of Layout, Design Rule Check (DRC) and Layout vs Schematic(LVS) of XOR are presented.

### 2.4.1 LAYOUT

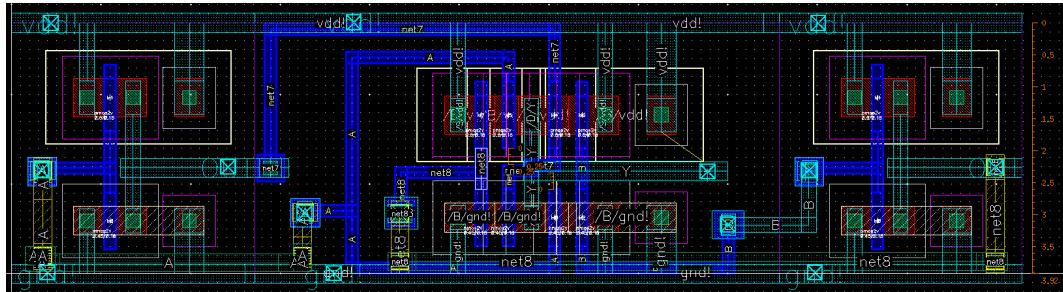


Figure 2.10: Layout of XOR

A layout view is created from the XOR design schematic shown in 1.20. Layout view is presented in Figure 2.10.

### 2.4.2 DRC



Figure 2.11: DRC of XOR design

Design rule check is performed on the layout Figure 2.10 of the XOR. The resulting errors are presented in Figure 2.11.

### 2.4.3 LVS

Comparison Results x ]				
Layout Cell / Type	Source Cell	Nets	Instances	Ports
INV	INV	4L, 4S	1L, 1S	4L, 4S
XOR	XOR	7L, 7S	6L, 6S	5L, 5S

Cell XOR Summary (Clean)						
SOURCE CELL NAME: XOR						
<hr/>						
INITIAL NUMBERS OF OBJECTS						
<hr/>						
Ports:	Layout	Source	Component Type			
	5	5				
Nets:	11	11				
Instances:	4	4	MN (4 pins)			
	4	4	MP (4 pins)			
	2	2	INV (4 pins): gnd! in out inh_vdd			
Total Inst:	10	10				
<hr/>						
NUMBERS OF OBJECTS AFTER TRANSFORMATION						
<hr/>						
Ports:	Layout	Source	Component Type			
	5	5				
Nets:	7	7				
Instances:	2	2	SDW2 (3 pins)			
	2	2	SUP2 (3 pins)			
	2	2	INV (4 pins): gnd! in out inh_vdd			
Total Inst:	6	6				
<hr/>						
***** INFORMATION AND WARNINGS *****						

Figure 2.12: LVS of XOR design

Layout vs Schematic is performed on the layout Figure 2.10 of the XOR. The results of LVS are presented in 2.12.

## 2.5 DRC and LVS

DRC stands for Design Rule Check. It is used for checking whether the layout is following the rules like spacing between two different poly etc or not.

LVS stands for Layout vs Schematic. It is used check whether the layout created is similar to design schematic or not.

# Chapter 3

## LAB 3

### 3.1 PEX-INVERTER

In this section, post-layout simulation results of Inverter are presented. DRC and LVS results were already presented in Figure 2.2 and in Figure 2.3.

#### 3.1.1 Testbench

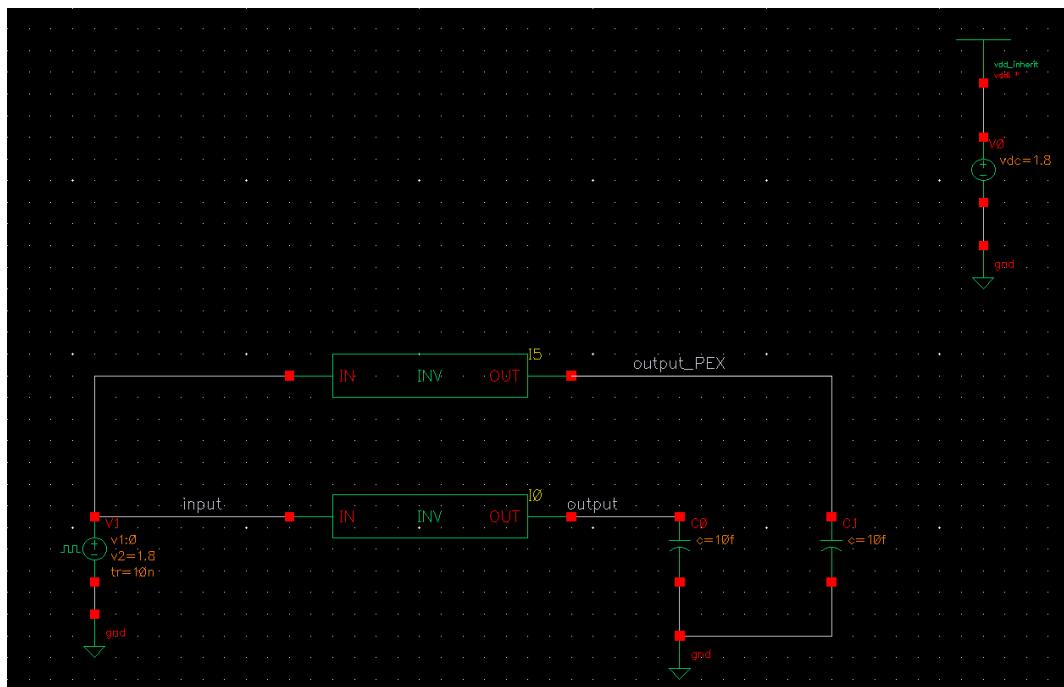


Figure 3.1: Testbench for pre and post-layout of Inverter

The testbench in Figure 1.3 has been modified to have both pre-layout and post-layout Inverters.

### 3.1.2 Netlist

```

// Library name: LOGIC180
// Cell name: INV
// View name: calibre
// Inherited view list: spectre cmos_sch cmos.sch schematic veriloga ahdl
//pspice dspf calibre
subckt INV IN OUT
    MM0 (MM0_d MM0_g MM0_s MM0_b) nch l=1.8e-07 w=4.5e-07 m=1 nf=1 \
        sd=540.0n ad=2.16e-13 as=2.16e-13 pd=1.86e-06 ps=1.86e-06 \
        nrd=1.066667 nrs=1.066667 sa=4.8e-07 sb=4.8e-07 sca=2.44077 \
        scb=0.000112946 scc=2.51784e-08
    MM1 (MM1_d MM1_g MM1_s MM1_b) pch l=1.8e-07 w=6e-07 m=1 nf=1 sd=540.0n \
        ad=2.88e-13 as=2.88e-13 pd=2.16e-06 ps=2.16e-06 nrd=0.8 nrs=0.8 \
        sa=4.8e-07 sb=4.8e-07 sca=5.80149 scb=0.00243491 scc=1.47315e-05
    cc_22 (OUT c_40_n) capacitor c=0.0290444f
    cc_21 (c_23_n c_40_n) capacitor c=0.0202772f
    cc_20 (c_23_n c_39_n) capacitor c=0.0145222f
    cc_19 (c_23_n c_37_n) capacitor c=0.0211383f
    cc_18 (c_18_n OUT) capacitor c=0.0216886f
    cc_17 (c_18_n c_24_n) capacitor c=0.0284977f
    cc_16 (c_17_n c_24_n) capacitor c=0.00995304f
    cc_15 (c_15_n c_24_n) capacitor c=0.0297079f
    cc_14 (MM0_g c_40_n) capacitor c=0.00549381f
    cc_13 (MM0_g c_39_n) capacitor c=0.00230988f
    cc_12 (MM0_g c_37_n) capacitor c=0.0142967f
    cc_11 (c_1_p c_37_n) capacitor c=0.00637946f
    cc_10 (c_10_p c_25_n) capacitor c=0.00688577f
    cc_9 (MM1_g c_25_n) capacitor c=0.00162394f
    cc_8 (MM0_g c_25_n) capacitor c=0.00162394f
    cc_7 (IN c_25_n) capacitor c=0.0099802f
    cc_6 (MM1_g c_24_n) capacitor c=0.0296066f
    cc_5 (MM0_g c_23_n) capacitor c=0.0210336f
    cc_4 (MM1_g c_18_n) capacitor c=0.00579981f
    cc_3 (MM1_g c_17_n) capacitor c=0.00187313f
    cc_2 (MM1_g c_15_n) capacitor c=0.0169982f
    cc_1 (c_1_p c_15_n) capacitor c=0.00547307f
    ciGND!_13 (GND!_2 0) capacitor c=0.10882f
    ciGND!_12 (c_37_n 0) capacitor c=0.0322802f

```

Figure 3.2: Netlist for pre and post-layout TB of Inverter -1

The netlist generated for testbench in Figure 3.1 is shown in Figure 3.2 and in Figure 3.3. It is divided into two parts to increase readability.

```

rVDD!_17 (c_17_n VDD!_20) resistor r=0.0463781
rVDD!_16 (VDD!_11 VDD!_19) resistor r=0.0263361
rVDD!_15 (VDD!_12 c_18_n) resistor r=0.172957
rVDD!_14 (VDD!_12 VDD!_20) resistor r=0.00090057
rIN_27 (IN IN_11) resistor r=7.8
rIN_26 (IN_11 IN_30) resistor r=6.58242
rIN_25 (c_1_p IN_30) resistor r=33.575
rIN_24 (c_1_p c_10_p) resistor r=0.869364
rIN_23 (IN_19 MM0_g) resistor r=32.2583
rIN_22 (IN_19 c_10_p) resistor r=3.34522
rIN_21 (IN_23 MM1_g) resistor r=44.3278
rIN_20 (IN_23 c_10_p) resistor r=3.34522
ends INV
// End of subcircuit definition.

// Library name: LOGIC180
// Cell name: INV
// View name: schematic
// Inherited view list: spectre cmos_sch cmos.sch schematic veriloga ahdl
// pspice dspf calibre
subckt INV_schematic IN OUT inh_vdd
    M0 (OUT IN 0 0) nch l=180.0n w=450.0n m=1 nf=1 sd=540.0n ad=2.16e-13 \
        as=2.16e-13 pd=1.86u ps=1.86u nrd=0.6 nrs=0.6 sa=480.0n sb=480.0n \
        sca=0 scb=0 scc=0
    M1 (OUT IN inh_vdd inh_vdd) pch l=180.0n w=600n m=1 nf=1 sd=540.0n \
        ad=2.88e-13 as=2.88e-13 pd=2.16u ps=2.16u nrd=0.45 nrs=0.45 \
        sa=480.0n sb=480.0n sca=0 scb=0 scc=0
ends INV_schematic
// End of subcircuit definition.

// Library name: LOGIC180
// Cell name: INV_tb
// View name: schematic
// Inherited view list: spectre cmos_sch cmos.sch schematic veriloga ahdl
// pspice dspf calibre
I5 (_net0 output_PEX) INV
I0 (_net0 _net1 vdd!) INV_schematic
V0 (vdd! 0) vsource dc=1.8 type=dc
C1 (output_PEX 0) capacitor c=10f
C0 (_net1 0) capacitor c=10f
V1 (_net0 0) vsource dc=1.8 type=pulse val0=0 val1=1.8 period=100n \
    rise=10n fall=10n width=40n
simulatorOptions options reltol=1e-3 vabstol=1e-6 iabstol=1e-12 temp=27 \
    tnom=27 scalem=1.0 scale=1.0 gmin=1e-12 rforce=1 maxnotes=5 maxwarns=5 \
    digits=5 cols=80 pivrel=1e-3 sensfile="../psf/sens.output" \
    checklimitdest=psf
tran tran stop=200n errpreset=conservative write="spectre.ic" \
    writefinal="spectre.fc" annotate=status maxiters=5
finalTimeOP info what=oppont where=rawfile
modelParameter info what=models where=rawfile
element info what=inst where=rawfile
outputParameter info what=output where=rawfile
designParamVals info what=parameters where=rawfile
primitives info what=primitives where=rawfile
subckts info what=subckts where=rawfile
save V0:p V0:p
saveOptions options save=allpub

```

Figure 3.3: Netlist for pre and post-layout TB of Inverter -2

### 3.1.3 Simulation

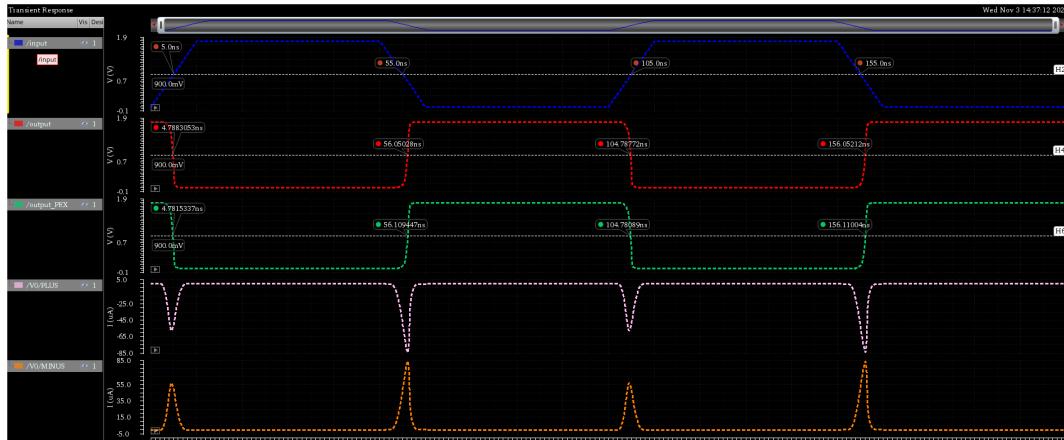


Figure 3.4: Simulation of pre and post-layout

Figure 3.4 shows the simulation of Inverter design before layout(uses schematic view) and after layout(uses calibre view).

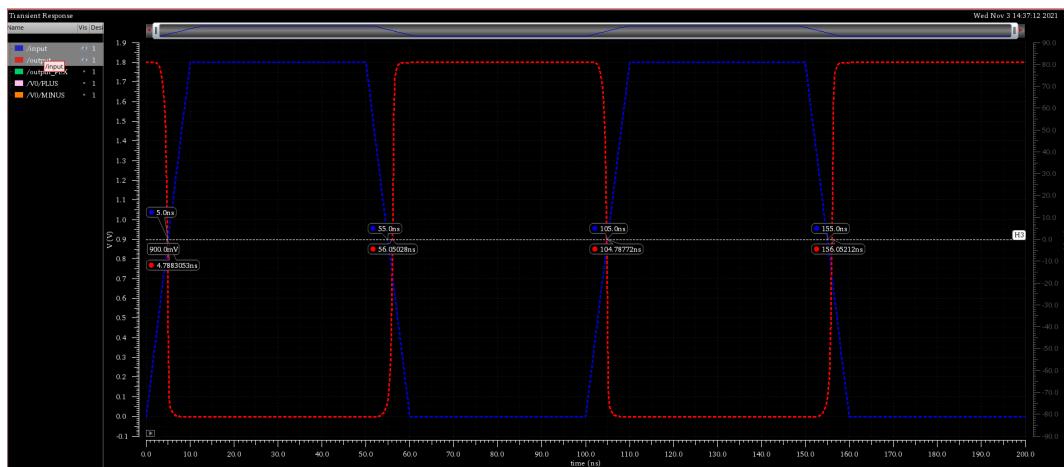


Figure 3.5: Pre-layout simulation of Inverter

Figure 3.5 shows the transient behaviour of Pre-layout Inverter design.

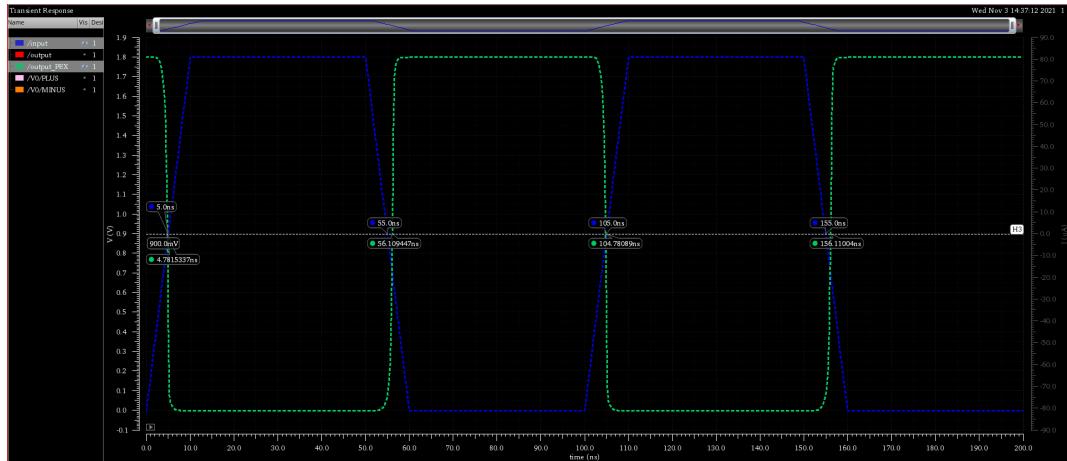


Figure 3.6: Post-layout simulation of Inverter

Figure 3.6 shows the transient behaviour of Post-layout Inverter design.

## 3.2 PEX-NAND2

In this section, post-layout simulation results of NAND2 are presented. DRC and LVS results were already presented in Figure 2.5 and in Figure 2.6.

### 3.2.1 Testbench

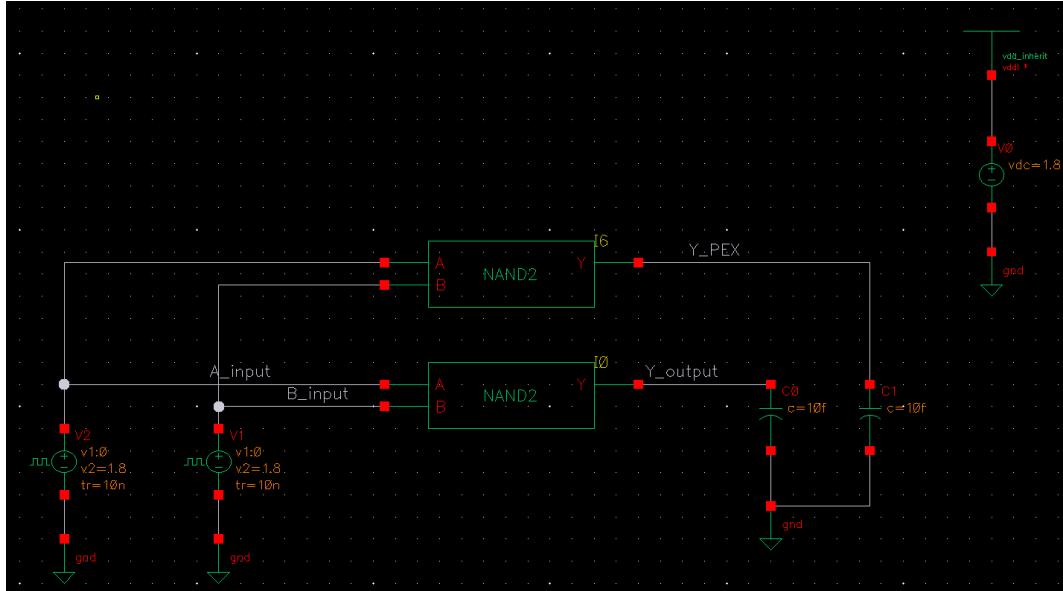


Figure 3.7: Testbench for pre and post-layout of NAND2

The testbench in Figure 1.9 has been modified to have both pre-layout and post-layout NAND2.

### 3.2.2 Netlist

The netlist generated for testbench in Figure 3.7 is shown in Figure 3.8 and in Figure 3.9. It is divided into two parts to increase readability.

---

```

// Library name: LOGIC180
// Cell name: NAND2
// View name: calibre
// Inherited view list: spectre cmos_sch cmos.sch schematic veriloga ahdl
//pspice dspf calibre
subckt NAND2 A B Y
    MM1 (MM1_d MM1_g MM1_s MM1_b) nch l=1.8e-07 w=4.5e-07 m=1 nf=1 \
        sd=540.0n ad=5.625e-14 as=2.16e-13 pd=7e-07 ps=1.86e-06 \
        nrd=0.277778 nrs=1.06667 sa=9.1e-07 sb=4.8e-07 sca=2.26326 \
        scb=0.000186355 scc=8.38108e-08
    MM0 (MM0_d MM0_g MM0_s MM0_b) nch l=1.8e-07 w=4.5e-07 m=1 nf=1 \
        sd=540.0n ad=2.16e-13 as=5.625e-14 pd=1.86e-06 ps=7e-07 \
        nrd=1.06667 nrs=0.277778 sa=4.8e-07 sb=9.1e-07 sca=2.27997 \
        scb=0.000186356 scc=8.38108e-08
    MM3 (MM3_d MM3_g MM3_s MM3_b) pch l=1.8e-07 w=6e-07 m=1 nf=1 sd=540.0n \
        ad=1.62e-13 as=2.88e-13 pd=1.14e-06 ps=2.16e-06 nrd=0.45 nrs=0.8 \
        sa=1.2e-06 sb=4.8e-07 sca=5.13227 scb=0.00238452 scc=1.47283e-05
    MM2 (MM2_d MM2_g MM2_s MM2_b) pch l=1.8e-07 w=6e-07 m=1 nf=1 sd=540.0n \
        ad=1.62e-13 as=2.88e-13 pd=1.14e-06 ps=2.16e-06 nrd=0.45 nrs=0.8 \
        sa=4.8e-07 sb=1.2e-06 sca=5.66971 scb=0.0024349 scc=1.47315e-05
    cc_60 (Y c_116_n) capacitor c=0.0336702f
    cc_59 (Y c_113_n) capacitor c=0.0168351f
    cc_58 (c_82_n c_113_n) capacitor c=0.0116964f
    cc_57 (c_82_n c_108_n) capacitor c=0.00974388f
    cc_56 (c_73_n Y) capacitor c=0.0290377f
    cc_55 (c_70_n Y) capacitor c=0.0145188f
    cc_54 (c_70_n c_86_n) capacitor c=0.0297079f
    cc_53 (c_66_n c_86_n) capacitor c=0.00939528f
    cc_52 (c_64_n c_86_n) capacitor c=0.0297079f
    cc_51 (c_43_p c_116_n) capacitor c=0.00462831f
    cc_50 (c_43_p c_113_n) capacitor c=0.00231415f
    cc_49 (MM1_g c_113_n) capacitor c=0.0276109f
    cc_48 (c_32_n c_113_n) capacitor c=0.00156675f
    cc_47 (c_59_p c_108_n) capacitor c=0.00931551f
    cc_46 (c_32_n c_108_n) capacitor c=0.128331f
    cc_45 (c_29_n c_108_n) capacitor c=0.00208606f
    cc_44 (B c_108_n) capacitor c=0.0426398f
    cc_43 (c_46_p Y) capacitor c=0.00804244f
    ...
    ...

```

Figure 3.8: Netlist for pre and post-layout TB of NAND2 -1

---

```

rA_39 (c_2_p c_3_p) resistor r=51.7889
rA_38 (c_2_p c_19_p) resistor r=3.34522
rA_37 (A_19 MM2_g) resistor r=39.5
rA_36 (A_19 c_19_p) resistor r=0.869364
rA_35 (c_9_p A_24) resistor r=8.33889
rA_34 (c_9_p A_25) resistor r=4.41396
rA_33 (A_24 c_19_p) resistor r=3.34522
rA_32 (A_25 MM0_g) resistor r=24.7972
ends NAND2
// End of subcircuit definition.

// Library name: LOGIC180
// Cell name: NAND2
// View name: schematic
// Inherited view list: spectre cmos_sch cmos.sch schematic veriloga ahdl
// pspice dspf calibre
subckt NAND2_schematic A B Y inh_vdd
    M1 (net15 B 0 0) nch l=180.0n w=450.0n m=1 nf=1 sd=540.0n ad=2.16e-13 \
        as=2.16e-13 pd=1.86u ps=1.86u nrd=0.6 nrs=0.6 sa=480.0n sb=480.0n \
        sca=0 scb=0 scc=0
    M0 (Y A net15 0) nch l=180.0n w=450.0n m=1 nf=1 sd=540.0n ad=2.16e-13 \
        as=2.16e-13 pd=1.86u ps=1.86u nrd=0.6 nrs=0.6 sa=480.0n sb=480.0n \
        sca=0 scb=0 scc=0
    M3 (Y B inh_vdd inh_vdd) pch l=180.0n w=600n m=1 nf=1 sd=540.0n \
        ad=2.88e-13 as=2.88e-13 pd=2.16u ps=2.16u nrd=0.45 nrs=0.45 \
        sa=480.0n sb=480.0n sca=0 scb=0 scc=0
    M2 (Y A inh_vdd inh_vdd) pch l=180.0n w=600n m=1 nf=1 sd=540.0n \
        ad=2.88e-13 as=2.88e-13 pd=2.16u ps=2.16u nrd=0.45 nrs=0.45 \
        sa=480.0n sb=480.0n sca=0 scb=0 scc=0
ends NAND2_schematic
// End of subcircuit definition.

// Library name: LOGIC180
// Cell name: NAND2_tb
// View name: schematic
// Inherited view list: spectre cmos_sch cmos.sch schematic veriloga ahdl
// pspice dspf calibre
I6 (A_input B_input Y_PEX) NAND2
I0 (A_input B_input Y_output vdd!) NAND2_schematic
V0 (vdd! 0) vsource dc=1.8 type=dc
V3 (B_input 0) vsource dc=1.8 type=pulse val0=0 val1=1.8 period=100n \
    rise=10n fall=10n width=40n
V2 (A_input 0) vsource dc=1.8 type=pulse val0=0 val1=1.8 period=200n \
    rise=10n fall=10n width=90n
C1 (Y_PEX 0) capacitor c=10f
C0 (Y_output 0) capacitor c=10f
simulatorOptions options reltol=1e-3 vabstol=1e-6 iabstol=1e-12 temp=27 \
    tnom=27 scalem=1.0 scale=1.0 gmin=1e-12 rforce=1 maxnotes=5 maxwarns=5 \
    digits=5 cols=80 pivrel=1e-3 sensfile="../psf/sens.output" \
    checklimitdest=psf
tran tran stop=400n write="spectre.ic" writefinal="spectre.fc" \
    annotate=status maxiters=5
finalTimeOP info what=oppont where=rawfile
modelParameter info what=models where=rawfile
element info what=inst where=rawfile
outputParameter info what=output where=rawfile
designParamVals info what=parameters where=rawfile
primitives info what=primitives where=rawfile
subckts info what=subckts where=rawfile
save V0:p V0:p
saveOptions options save=allpub

```

---

Figure 3.9: Netlist for pre and post-layout TB of NAND2 -2

### 3.2.3 Simulation

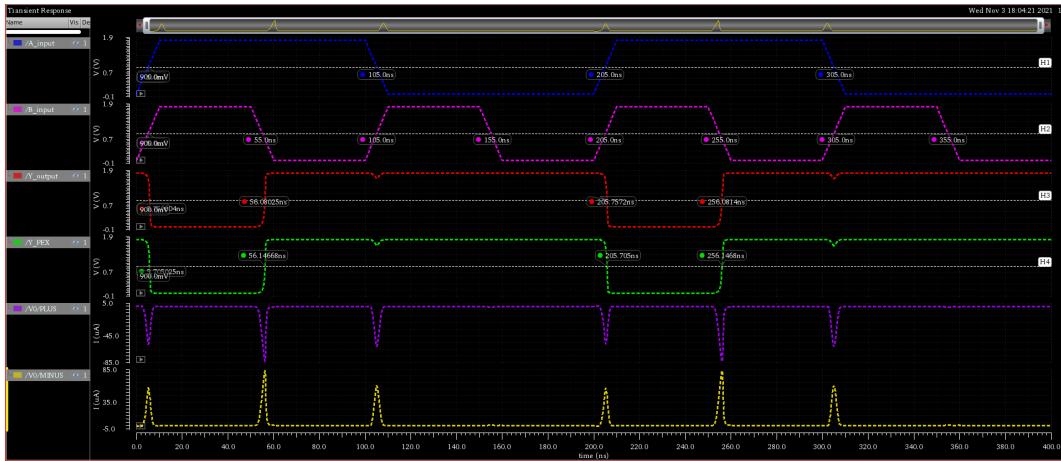


Figure 3.10: Simulation of pre and post-layout

Figure 3.10 shows the simulation of NAND2 design before layout(uses schematic view) and after layout(uses calibre view).

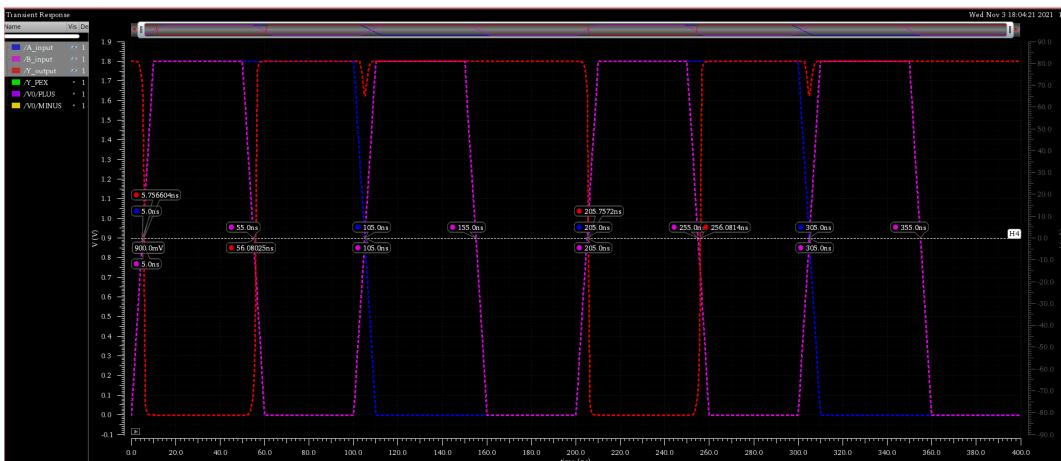


Figure 3.11: Simulation of Pre-layout NAND2

Figure 3.11 shows the transient behaviour of Pre-layout NAND2 design.

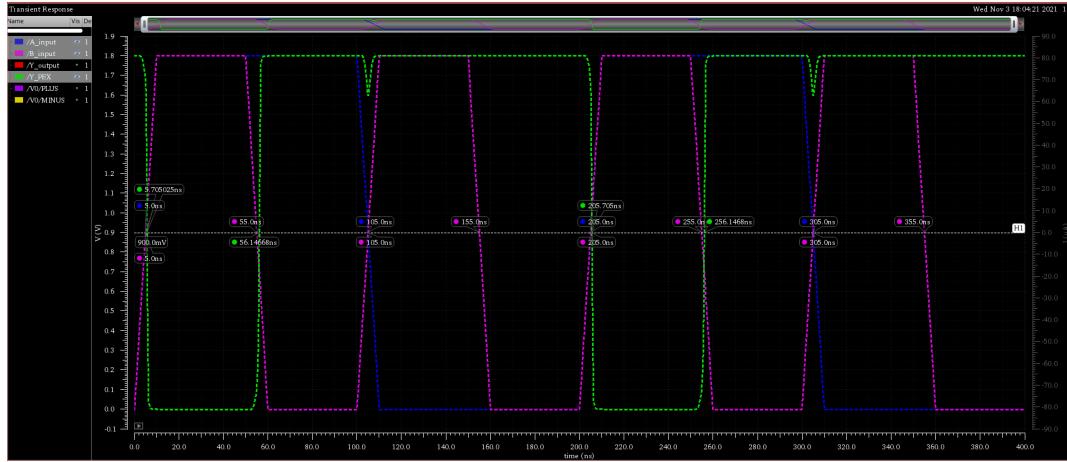


Figure 3.12: Simulation of Post-layout NAND2

Figure 3.12 shows the transient behaviour of Post-layout NAND2 design.

### 3.3 PEX-NOR2

In this section, post-layout simulation results of NOR2 are presented. DRC and LVS results were already presented in Figure 2.8 and in Figure 2.9.

#### 3.3.1 Testbench

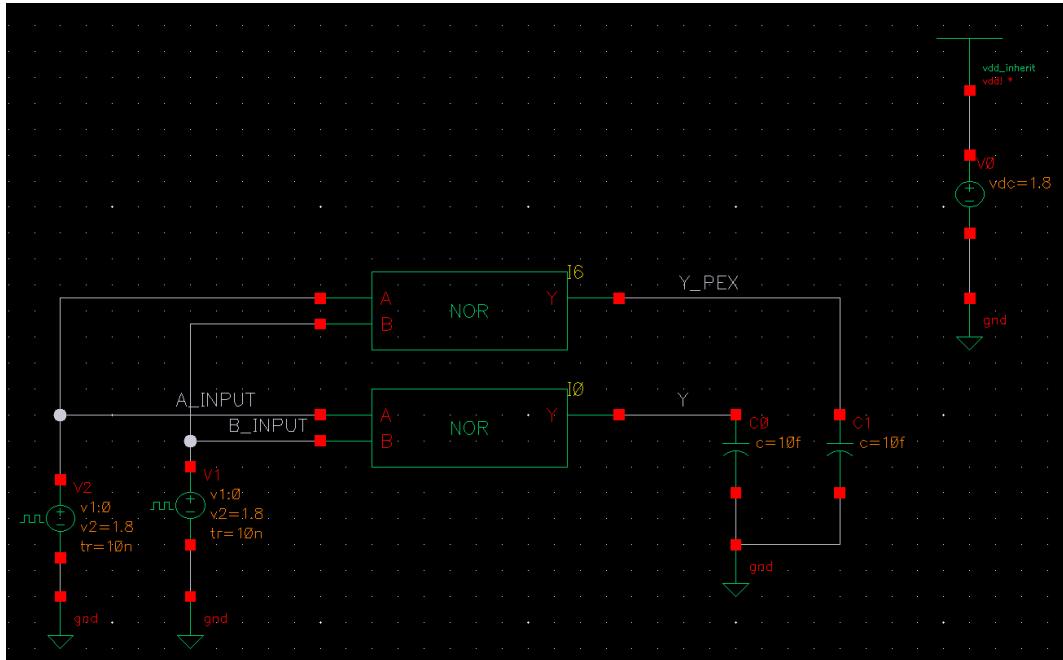


Figure 3.13: Testbench for pre and post-layout of NOR2

The testbench in Figure 1.15 has been modified to have both pre-layout and post-layout NOR2.

#### 3.3.2 Netlist

The netlist generated for testbench in Figure 3.13 is shown in Figure 3.14 and in Figure 3.15. It is divided into two parts to increase readability.

```

// Library name: LOGIC180
// Cell name: NOR
// View name: calibre
// Inherited view list: spectre cmos_sch cmos.sch schematic veriloga ahdl
//pspice dspf calibre
subckt NOR A B Y
    MM0 (MM0_d MM0_g MM0_s MM0_b) nch l=1.8e-07 w=4.5e-07 m=1 nf=1 \
        sd=540.0n ad=1.215e-13 as=2.16e-13 pd=9.9e-07 ps=1.86e-06 nrd=0.6 \
        nrs=1.06667 sa=1.2e-06 sb=4.8e-07 sca=1.8895 scb=9.02603e-05 \
        scc=1.87114e-08
    MM1 (MM1_d MM1_g MM1_s MM1_b) nch l=1.8e-07 w=4.5e-07 m=1 nf=1 \
        sd=540.0n ad=1.215e-13 as=2.16e-13 pd=9.9e-07 ps=1.86e-06 nrd=0.6 \
        nrs=1.06667 sa=4.8e-07 sb=1.2e-06 sca=1.90237 scb=9.02604e-05 \
        scc=1.87114e-08
    MM2 (MM2_d MM2_g MM2_s MM2_b) pch l=1.8e-07 w=6e-07 m=1 nf=1 sd=540.0n \
        ad=2.88e-13 as=7.5e-14 pd=2.16e-06 ps=8.5e-07 nrd=0.8 nrs=0.208333 \
        sa=9.1e-07 sb=4.8e-07 sca=5.28429 scb=0.00238544 scc=1.47283e-05
    MM3 (MM3_d MM3_g MM3_s MM3_b) pch l=1.8e-07 w=6e-07 m=1 nf=1 sd=540.0n \
        ad=7.5e-14 as=2.88e-13 pd=8.5e-07 ps=2.16e-06 nrd=0.208333 nrs=0.8 \
        sa=4.8e-07 sb=9.1e-07 sca=5.70833 scb=0.0024349 scc=1.47315e-05
    cc_69 (Y c_129_n) capacitor c=0.0317563f
    cc_68 (c_107_n c_126_n) capacitor c=0.0162056f
    cc_67 (c_95_n c_126_n) capacitor c=0.0211383f
    cc_66 (c_95_n c_119_n) capacitor c=0.00838666f
    cc_65 (c_95_n c_117_n) capacitor c=0.0211383f
    cc_64 (c_85_n Y) capacitor c=0.0267654f
    cc_63 (c_85_n c_103_n) capacitor c=0.0284977f
    cc_62 (c_79_n c_103_n) capacitor c=0.0113714f
    cc_61 (c_75_n c_103_n) capacitor c=0.0164383f
    cc_60 (c_74_p c_130_n) capacitor c=0.0111495f
    cc_59 (c_35_n c_130_n) capacitor c=0.0642872f
    cc_58 (c_31_n c_130_n) capacitor c=0.00168307f
    cc_57 (B c_130_n) capacitor c=0.0219413f
    cc_56 (c_46_p c_129_n) capacitor c=0.00473414f
    cc_55 (c_46_p c_126_n) capacitor c=0.00236053f
    cc_54 (MM0_g c_126_n) capacitor c=0.0304297f
    cc_53 (c_35_n c_126_n) capacitor c=0.001171117f
    cc_52 (c_35_n c_119_n) capacitor c=0.0492564f
    cc_51 (c_65_p c_117_n) capacitor c=0.00957216f
    cc_50 (MM0_g c_117_n) capacitor c=0.00969058f
    cc_49 (c_35_n c_117_n) capacitor c=0.00667913f
    cc_48 (c_31_n c_117_n) capacitor c=0.00312255f
    cc_47 (B c_117_n) capacitor c=0.018146f
    cc_46 (c_46_p c_107_n) capacitor c=0.00966494f
    cc_45 (c_47_p Y) capacitor c=0.0119781f
    cc_44 (c_46_p Y) capacitor c=0.0629914f
    cc_43 (c_42_p c_103_n) capacitor c=0.00329813f
    cc_42 (MM2_g c_103_n) capacitor c=0.0184004f
    cc_41 (c_39_n c_99_n) capacitor c=0.00621099f
    cc_40 (c_46_p c_99_n) capacitor c=4.61474e-19
    cc_39 (MM0_g c_99_n) capacitor c=0.00900959f
    cc_38 (MM2_g c_99_n) capacitor c=0.00539808f
    cc_37 (MM0_g c_95_n) capacitor c=0.011659f
    cc_36 (c_35_n c_95_n) capacitor c=0.00291964f

```

Figure 3.14: Netlist for pre and post-layout TB of NOR2 -1

```

rA_44 (A_19 c_3_p) resistor r=0.869364
rA_43 (c_15_p c_21_p) resistor r=4.41396
rA_42 (c_13_p A_22) resistor r=46.9611
rA_41 (c_13_p A_25) resistor r=4.41396
rA_40 (A_22 c_3_p) resistor r=3.34522
rA_39 (c_12_p c_21_p) resistor r=53.5444
rA_38 (c_12_p A_29) resistor r=4.41396
rA_37 (A_25 MM1_g) resistor r=30.0639
rA_36 (A_29 MM3_g) resistor r=36.8667
ends NOR
// End of subcircuit definition.

// Library name: LOGIC180
// Cell name: NOR
// View name: schematic
// Inherited view list: spectre cmos_sch cmos.sch schematic veriloga ahdl
// pspice dspf calibre
subckt NOR_schematic A B Y inh_vdd
    M1 (Y A 0 0) nch l=180.0n w=450.0n m=1 nf=1 sd=540.0n ad=2.16e-13 \
        as=2.16e-13 pd=1.86u ps=1.86u nrd=0.6 nrs=0.6 sa=480.0n sb=480.0n \
        sca=0 scb=0 scc=0
    M0 (Y B 0 0) nch l=180.0n w=450.0n m=1 nf=1 sd=540.0n ad=2.16e-13 \
        as=2.16e-13 pd=1.86u ps=1.86u nrd=0.6 nrs=0.6 sa=480.0n sb=480.0n \
        sca=0 scb=0 scc=0
    M3 (net15 A inh_vdd inh_vdd) pch l=180.0n w=600n m=1 nf=1 sd=540.0n \
        ad=2.88e-13 as=2.88e-13 pd=2.16u ps=2.16u nrd=0.45 nrs=0.45 \
        sa=480.0n sb=480.0n sca=0 scb=0 scc=0
    M2 (Y B net15 inh_vdd) pch l=180.0n w=600n m=1 nf=1 sd=540.0n \
        ad=2.88e-13 as=2.88e-13 pd=2.16u ps=2.16u nrd=0.45 nrs=0.45 \
        sa=480.0n sb=480.0n sca=0 scb=0 scc=0
ends NOR_schematic
// End of subcircuit definition.

// Library name: LOGIC180
// Cell name: NOR_tb
// View name: schematic
// Inherited view list: spectre cmos_sch cmos.sch schematic veriloga ahdl
// pspice dspf calibre
I6 (A_INPUT B_INPUT Y_PEX) NOR
I0 (A_INPUT B_INPUT Y vdd!) NOR_schematic
V0 (vdd! 0) vsource dc=1.8 type=dc
V2 (A_INPUT 0) vsource dc=1.8 type=pulse val0=0 val1=1.8 period=200n \
    rise=10n fall=10n width=90n
V1 (B_INPUT 0) vsource dc=1.8 type=pulse val0=0 val1=1.8 period=100n \
    rise=10n fall=10n width=40n
C1 (Y_PEX 0) capacitor c=10f
C0 (Y 0) capacitor c=10f
simulatorOptions options reltol=1e-3 vabstol=1e-6 iabstol=1e-12 temp=27 \
    tnom=27 scalem=1.0 scale=1.0 gmin=1e-12 rforce=1 maxnotes=5 maxwarns=5 \
    digits=5 cols=80 pivrel=1e-3 sensfile="../psf/sens.output" \
    checklimitdest=psf
tran tran stop=400n write="spectre.ic" writefinal="spectre.fc" \
    annotate=status maxiters=5
finalTimeOP info what=oppont where=rawfile
modelParameter info what=models where=rawfile
element info what=inst where=rawfile
outputParameter info what=output where=rawfile
designParamVals info what=parameters where=rawfile
primitives info what=primitives where=rawfile
subckts info what=subckts where=rawfile
save V0:p V0:p
saveOptions options save=allpub

```

Figure 3.15: Netlist for pre and post-layout TB of NOR2 -2

### 3.3.3 Simulation

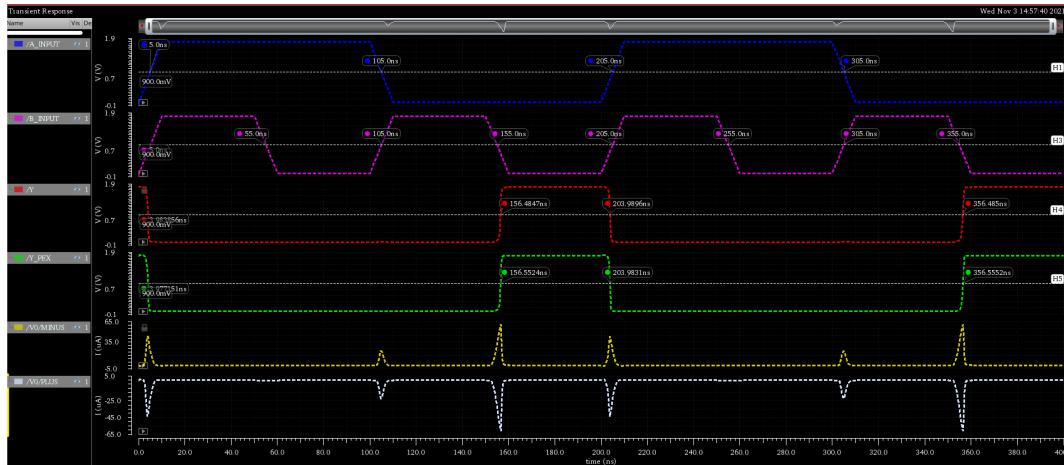


Figure 3.16: Simulation of Pre and Post-layout NOR2

Figure 3.16 shows the simulation of NOR2 design before layout(uses schematic view) and after layout(uses calibre view).

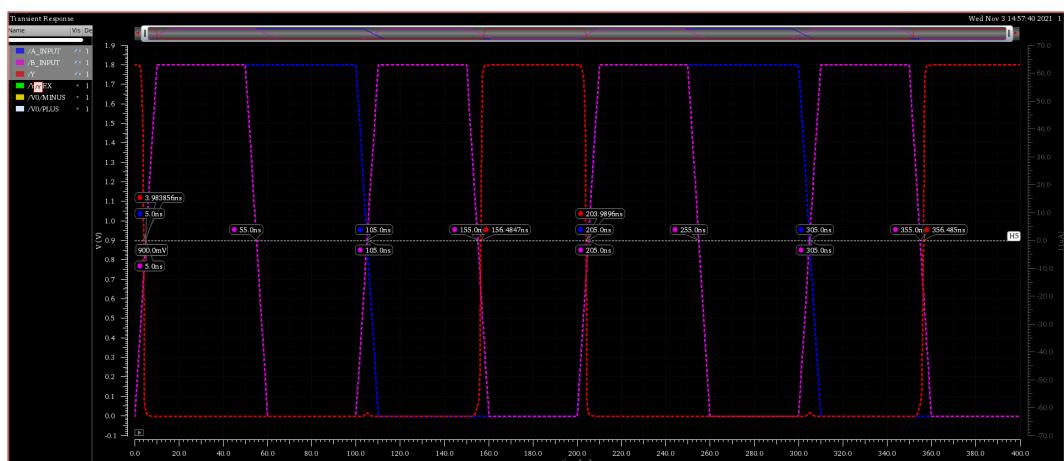


Figure 3.17: Simulation of Pre-layout NOR2

Figure 3.17 shows the transient behaviour of Pre-layout NOR2 design.

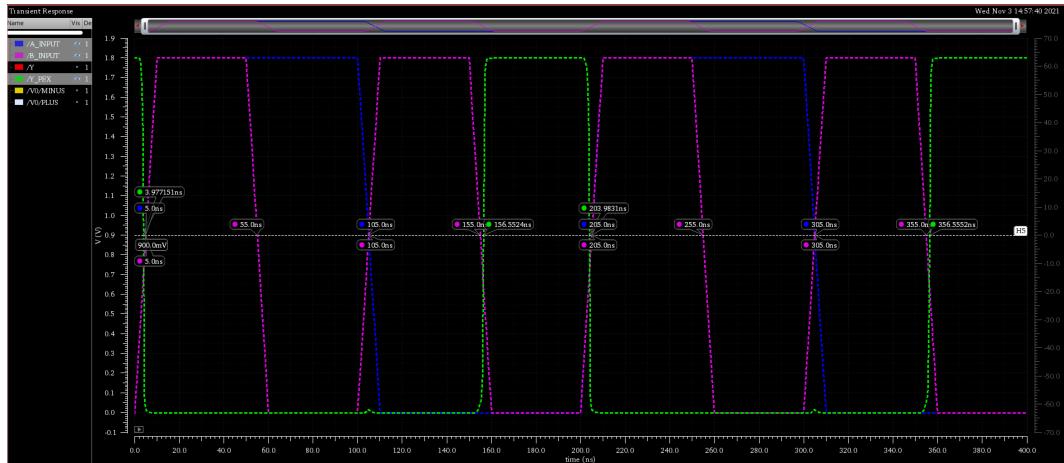


Figure 3.18: Simulation of Post-layout NOR2

Figure 3.18 shows the transient behaviour of Post-layout NOR2 design.

## 3.4 PEX-XOR

In this section, post-layout simulation results of XOR are presented. DRC and LVS results were already presented in Figure 2.11 and in Figure 2.12.

### 3.4.1 Testbench

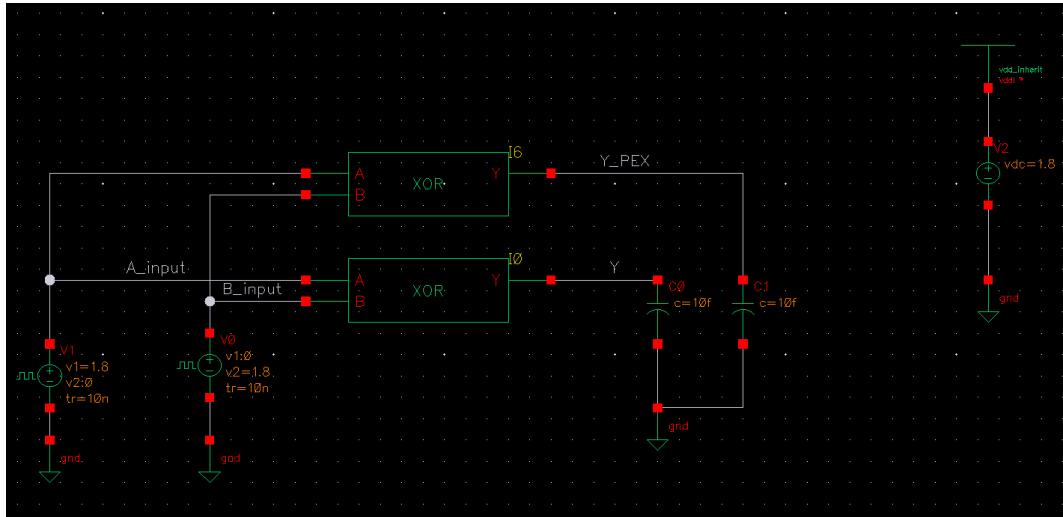


Figure 3.19: Testbench for pre and post-layout of XOR

The testbench in Figure 1.21 has been modified to have both pre-layout and post-layout XOR.

### 3.4.2 Netlist

The netlist generated for testbench in Figure 3.19 is shown in Figure 3.20, Figure 3.21 and in Figure 3.22. It is divided into three parts to increase readability.

---

```

// Library name: LOGIC180
// Cell name: XOR
// View name: calibre
// Inherited view list: spectre cmos_sch cmos.sch schematic veriloga ahd1
//pspice dspf calibre
subckt XOR A B Y
    M5_noxref (M5_noxref_d M5_noxref_g M5_noxref_s M5_noxref_b) nch \
        l=1.8e-07 w=4.5e-07 m=1 nf=1 sd=540n ad=2.16e-13 as=2.16e-13 \
        pd=1.86e-06 ps=1.86e-06 nrd=1.06667 nrs=1.06667 sa=4.8e-07 \
        sb=4.8e-07 sca=2.05902 scb=0.000112608 scc=2.51783e-08
    M4_noxref (M4_noxref_d M4_noxref_g M4_noxref_s M4_noxref_b) nch \
        l=1.8e-07 w=4.5e-07 m=1 nf=1 sd=540n ad=2.16e-13 as=2.16e-13 \
        pd=1.86e-06 ps=1.86e-06 nrd=1.06667 nrs=1.06667 sa=4.8e-07 \
        sb=4.8e-07 sca=2.28415 scb=0.000112946 scc=2.51784e-08
    MM2 (MM2_d MM2_g MM2_s MM2_b) nch l=1.8e-07 w=4.5e-07 m=1 nf=1 \
        sd=540.0n ad=5.625e-14 as=2.16e-13 pd=7e-07 ps=1.86e-06 \
        nrd=0.277778 nrs=1.06667 sa=2.06e-06 sb=4.8e-07 sca=2.46565 \
        scb=0.000309227 scc=1.85439e-07
    MM0 (MM0_d MM0_g MM0_s MM0_b) nch l=1.8e-07 w=4.5e-07 m=1 nf=1 \
        sd=540.0n ad=1.215e-13 as=5.625e-14 pd=9.9e-07 ps=7e-07 nrd=0.6 \
        nrs=0.277778 sa=1.63e-06 sb=9.1e-07 sca=2.46565 scb=0.000309227 \
        scc=1.85439e-07
    MM1 (MM1_d MM1_g MM1_s MM1_b) nch l=1.8e-07 w=4.5e-07 m=1 nf=1 \
        sd=540.0n ad=1.215e-13 as=5.625e-14 pd=9.9e-07 ps=7e-07 nrd=0.6 \
        nrs=0.277778 sa=9.1e-07 sb=1.63e-06 sca=2.46565 scb=0.000309227 \
        scc=1.85439e-07
    MM3 (MM3_d MM3_g MM3_s MM3_b) nch l=1.8e-07 w=4.5e-07 m=1 nf=1 \
        sd=540.0n ad=5.625e-14 as=2.16e-13 pd=7e-07 ps=1.86e-06 \
        nrd=0.277778 nrs=1.06667 sa=4.8e-07 sb=2.06e-06 sca=2.46565 \
        scb=0.000309227 scc=1.85439e-07
    I2_MM1 (I2_MM1_d I2_MM1_g I2_MM1_s I2_MM1_b) pch l=1.8e-07 w=6e-07 m=1 \
        nf=1 sd=540.0n ad=2.88e-13 as=2.88e-13 pd=2.16e-06 ps=2.16e-06 \
        nrd=0.8 nrs=0.8 sa=4.8e-07 sb=4.8e-07 sca=5.80149 scb=0.00243491 \
        scc=1.47315e-05
    I3_MM1 (I3_MM1_d I3_MM1_g I3_MM1_s I3_MM1_b) pch l=1.8e-07 w=6e-07 m=1 \
        nf=1 sd=540.0n ad=2.88e-13 as=2.88e-13 pd=2.16e-06 ps=2.16e-06 \
        nrd=0.8 nrs=0.8 sa=4.8e-07 sb=4.8e-07 sca=5.80149 scb=0.00243491 \
        scc=1.47315e-05
    MM6 (MM6_d MM6_g MM6_s MM6_b) pch l=1.8e-07 w=6e-07 m=1 nf=1 sd=540.0n \
        ad=7.5e-14 as=2.88e-13 pd=8.5e-07 ps=2.16e-06 nrd=0.208333 nrs=0.8 \
        sa=2.06e-06 sb=4.8e-07 sca=4.94374 scb=0.00238446 scc=1.47283e-05
    MM4 (MM4_d MM4_g MM4_s MM4_b) pch l=1.8e-07 w=6e-07 m=1 nf=1 sd=540.0n \
        ad=1.62e-13 as=7.5e-14 pd=1.14e-06 ps=8.5e-07 nrd=0.45 \
        nrs=0.208333 sa=1.63e-06 sb=9.1e-07 sca=4.91688 scb=0.00238444 \
        scc=1.47283e-05
    MM5 (MM5_d MM5_g MM5_s MM5_b) pch l=1.8e-07 w=6e-07 m=1 nf=1 sd=540.0n \
        ad=1.62e-13 as=7.5e-14 pd=1.14e-06 ps=8.5e-07 nrd=0.45 \
        nrs=0.208333 sa=9.1e-07 sb=1.63e-06 sca=5.11425 scb=0.00238543 \
        scc=1.47283e-05
    MM7 (MM7_d MM7_g MM7_s MM7_b) pch l=1.8e-07 w=6e-07 m=1 nf=1 sd=540.0n \
        ad=7.5e-14 as=2.88e-13 pd=8.5e-07 ps=2.16e-06 nrd=0.208333 nrs=0.8 \
        sa=4.8e-07 sb=2.06e-06 sca=5.60649 scb=0.0024349 scc=1.47315e-05
    cc_204 (Y c_349_n) capacitor c=0.0350974f
    cc_203 (Y c_347_n) capacitor c=0.0175487f
    cc_202 (c_294_n c_347_n) capacitor c=0.0116964f
    cc_201 (c_294_n c_345_n) capacitor c=0.0123657f
    cc_200 (c_294_n c_339_n) capacitor c=0.0116964f
    cc_199 (c_293_p Y) capacitor c=0.0300712f
    cc_198 (c_272_n Y) capacitor c=0.0150356f
    cc_197 (c_272_n c_295_n) capacitor c=0.0164383f
    cc_196 (c_260_n c_295_n) capacitor c=0.0164383f
    cc_195 (c_215_n c_352_n) capacitor c=0.0108805f

```

Figure 3.20: Netlist for pre and post-layout TB of XOR -1

---

```

rNET7_64 (NET7_44 NET7) resistor r=0.796811
rNET7_63 (NET7_45 MM4_g) resistor r=54.8611
rNET7_62 (c_83_n NET7) resistor r=29.0204
rNET8_120 (NET8_1 c_36_p) resistor r=0.228913
rNET8_119 (NET8_1 c_26_p) resistor r=0.0409537
rNET8_118 (c_36_p NET8_4) resistor r=11
rNET8_117 (NET8_4 M5_noxref_d) resistor r=4.092
rNET8_116 (NET8_6 c_28_p) resistor r=0.322174
rNET8_115 (NET8_6 c_26_p) resistor r=0.0409537
rNET8_114 (c_28_p NET8_9) resistor r=10
rNET8_113 (NET8_9 I2_MM1_d) resistor r=3.492
rNET8_112 (c_26_p NET8_24) resistor r=0.00259375
rNET8_111 (NET8_24 c_42_p) resistor r=0.3185
rNET8_110 (c_42_p NET8_64) resistor r=0.0333918
rNET8_109 (NET8_33 c_12_p) resistor r=14.4833
rNET8_108 (NET8_33 c_10_p) resistor r=6.58242
rNET8_107 (c_12_p c_13_p) resistor r=4.41396
rNET8_106 (c_43_p c_13_p) resistor r=46.9611
rNET8_105 (c_43_p c_2_p) resistor r=0.869364
rNET8_104 (NET8_37 MM3_g) resistor r=26.9917
rNET8_103 (NET8_37 c_2_p) resistor r=3.34522
rNET8_102 (NET8_41 MM7_g) resistor r=35.55
rNET8_101 (NET8_41 c_2_p) resistor r=3.34522
rNET8_100 (NET8_47 c_8_p) resistor r=6.4
rNET8_99 (NET8_47 c_3_p) resistor r=2.5935
rNET8_98 (c_8_p NET8_60) resistor r=0.208929
rNET8_97 (c_3_p c_56_p) resistor r=6.4
rNET8_96 (c_56_p NET8) resistor r=0.410893
rNET8_95 (c_10_p c_7_p) resistor r=7.8
rNET8_94 (c_7_p NET8_60) resistor r=6.4
rNET8_93 (NET8_64 NET8) resistor r=6.4
ends XOR
// End of subcircuit definition.

// Library name: LOGIC180
// Cell name: INV
// View name: schematic
// Inherited view list: spectre cmos_sch cmos.sch schematic verilog-a ahdl
//pspice dspf calibre
subckt INV IN OUT inh_vdd
    M0 (OUT IN 0 0) nch l=180.0n w=450.0n m=1 nf=1 sd=540.0n ad=2.16e-13 \
        as=2.16e-13 pd=1.86u ps=1.86u nrd=0.6 nrs=0.6 sa=480.0n sb=480.0n \
        sca=0 scb=0 scc=0
    M1 (OUT IN inh_vdd inh_vdd) pch l=180.0n w=600n m=1 nf=1 sd=540.0n \
        ad=2.88e-13 as=2.88e-13 pd=2.16u ps=2.16u nrd=0.45 nrs=0.45 \
        sa=480.0n sb=480.0n sca=0 scb=0 scc=0
ends INV
// End of subcircuit definition.

```

Figure 3.21: Netlist for pre and post-layout TB of XOR -2

```

// Cell name: XOR
// View name: schematic
// Inherited view list: spectre cmos_sch cmos.sch schematic veriloga ahd1
// pspice dspf calibre
subckt XOR_schematic A B Y inh_vdd
    M5 (Y A net018 inh_vdd) pch l=180.0n w=600n m=1 nf=1 sd=540.0n \
        ad=2.88e-13 as=2.88e-13 pd=2.16u ps=2.16u nrd=0.45 nrs=0.45 \
        sa=480.0n sb=480.0n sca=0 scb=0 scc=0
    M7 (net018 net8 inh_vdd inh_vdd) pch l=180.0n w=600n m=1 nf=1 \
        sd=540.0n ad=2.88e-13 as=2.88e-13 pd=2.16u ps=2.16u nrd=0.45 \
        nrs=0.45 sa=480.0n sb=480.0n sca=0 scb=0 scc=0
    M6 (net019 B inh_vdd inh_vdd) pch l=180.0n w=600n m=1 nf=1 sd=540.0n \
        ad=2.88e-13 as=2.88e-13 pd=2.16u ps=2.16u nrd=0.45 nrs=0.45 \
        sa=480.0n sb=480.0n sca=0 scb=0 scc=0
    M4 (Y net7 net019 inh_vdd) pch l=180.0n w=600n m=1 nf=1 sd=540.0n \
        ad=2.88e-13 as=2.88e-13 pd=2.16u ps=2.16u nrd=0.45 nrs=0.45 \
        sa=480.0n sb=480.0n sca=0 scb=0 scc=0
    M8 (Y A net021 0) nch l=180.0n w=450.0n m=1 nf=1 sd=540.0n ad=2.16e-13 \
        as=2.16e-13 pd=1.86u ps=1.86u nrd=0.6 nrs=0.6 sa=480.0n sb=480.0n \
        sca=0 scb=0 scc=0
    M3 (net020 net8 0 0) nch l=180.0n w=450.0n m=1 nf=1 sd=540.0n \
        ad=2.16e-13 as=2.16e-13 pd=1.86u ps=1.86u nrd=0.6 nrs=0.6 \
        sa=480.0n sb=480.0n sca=0 scb=0 scc=0
    M2 (net021 B 0 0) nch l=180.0n w=450.0n m=1 nf=1 sd=540.0n ad=2.16e-13 \
        as=2.16e-13 pd=1.86u ps=1.86u nrd=0.6 nrs=0.6 sa=480.0n sb=480.0n \
        sca=0 scb=0 scc=0
    M1 (Y net7 net020 0) nch l=180.0n w=450.0n m=1 nf=1 sd=540.0n \
        ad=2.16e-13 as=2.16e-13 pd=1.86u ps=1.86u nrd=0.6 nrs=0.6 \
        sa=480.0n sb=480.0n sca=0 scb=0 scc=0
    I3 (A net7 inh_vdd) INV
    I2 (B net8 inh_vdd) INV
ends XOR_schematic
// End of subcircuit definition.

// Library name: LOGIC180
// Cell name: XOR_tb
// View name: schematic
// Inherited view list: spectre cmos_sch cmos.sch schematic veriloga ahd1
// pspice dspf calibre
I6 (A_input B_input Y_PEX) XOR
I0 (A_input B_input Y vdd!) XOR_schematic
V1 (A_input 0) vsource dc=1.8 type=pulse val0=1.8 val1=0 period=200n \
    rise=10n fall=10n width=90n
V0 (B_input 0) vsource dc=1.8 type=pulse val0=0 val1=1.8 period=100n \
    rise=10n fall=10n width=40n
C1 (Y_PEX 0) capacitor c=10f
C0 (Y 0) capacitor c=10f
V2 (vdd! 0) vsource dc=1.8 type=dc
simulatorOptions options reltol=1e-3 vabstol=1e-6 iabstol=1e-12 temp=27 \
    tnom=27 scalem=1.0 scale=1.0 gmin=1e-12 rforce=1 maxnotes=5 maxwarns=5 \
    digits=5 cols=80 pivrel=1e-3 sensfile="../psf/sens.output" \
    checklimitdest=psf
tran tran stop=400n errpreset=conservative write="spectre.ic" \
    writefinal="spectre.fc" annotate=status maxiters=5
finalTimeOP info what=oppoint where=rawfile
modelParameter info what=models where=rawfile
element info what=inst where=rawfile
outputParameter info what=output where=rawfile
designParamVals info what=parameters where=rawfile
primitives info what=primitives where=rawfile
subckts info what=subckts where=rawfile
save V2:p V2:p

```

Figure 3.22: Netlist for pre and post-layout TB of XOR -3

### 3.4.3 Simulation

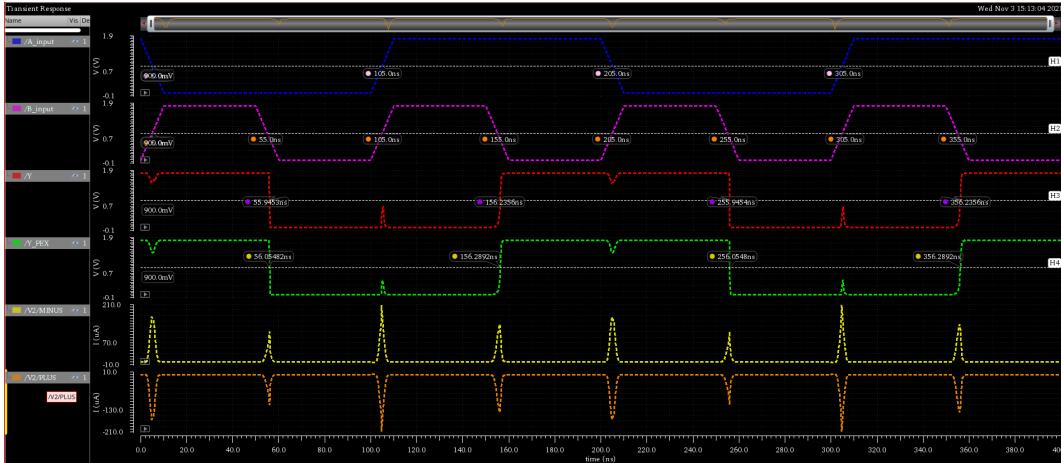


Figure 3.23: Simulation of Pre and Post-layout XOR

Figure 3.23 shows the simulation of XOR design before layout(uses schematic view) and after layout(uses calibre view).

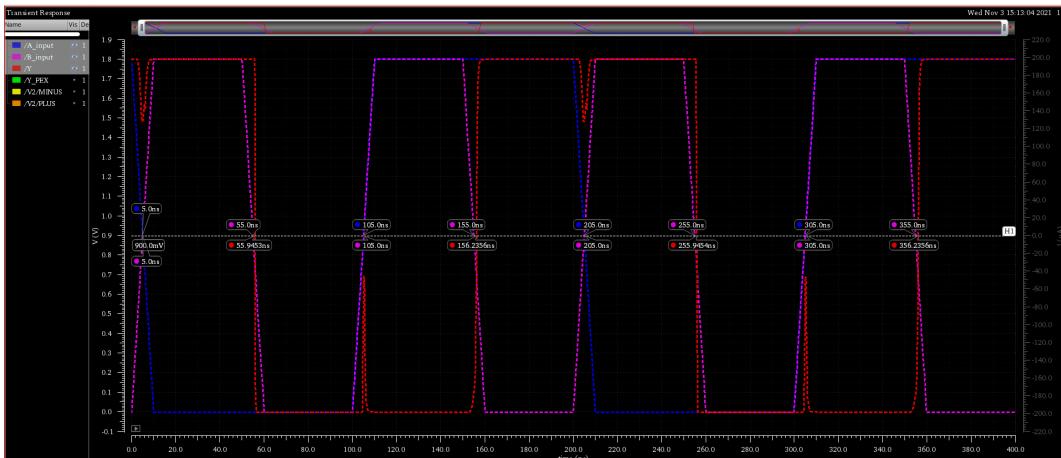


Figure 3.24: Simulation with pre and post-layout

Figure 3.24 shows the transient behaviour of Pre-layout XOR design.

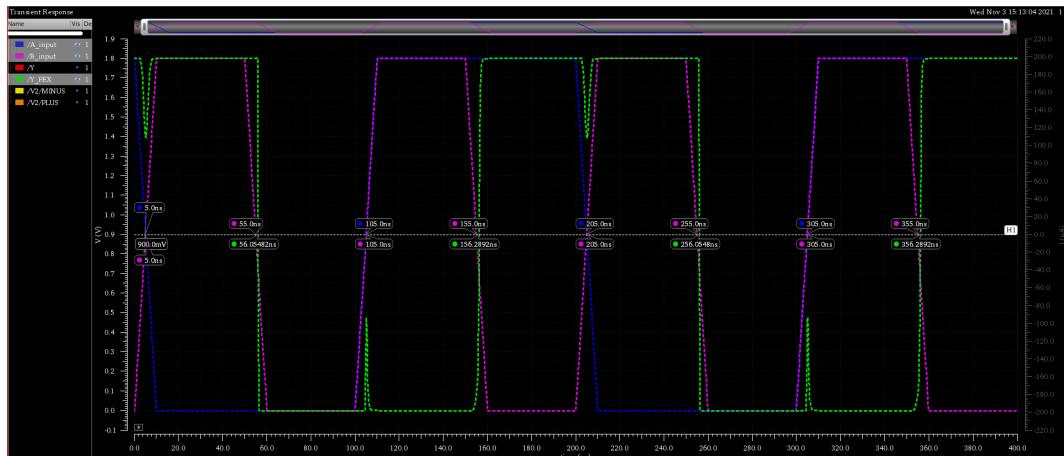


Figure 3.25: Simulation with pre and post-layout

Figure 3.25 shows the transient behaviour of Post-layout XOR design.

### 3.5 Inverter layout and calibre view

The calibre view is created by the PEX. It generates a bunch of capacitors and resistors for all the contacts and routes used in layout. For Inverter calibre view contains of 1 PMOS and 1 NMOS, power sources and contains lot of capacitors and resistors.

### 3.6 PEX

PEX stands for Parasitic Extraction. It is used for creating parasitic devices like capacitors and resistors from the layout design to create a realistic model that can be used for evaluating the performance of the design. PEX transforms all the contacts and wiring in the layout to parasitic devices.

# Chapter 4

## LAB 4

In the lab 4 performance of 3 different Inverter designs are evaluated. The following are the three inverter designs:

- 1) Inverter design schematic in Figure 1.2.
- 2) Connecting both the inputs of NAND2 design schematic in Figure 1.8.
- 3) Inverter from TSMC180-CELLS library.

### 4.1 Configuring NAND2 as Inverter

In this section we configure out NAND2 design schematic shown in Figure 1.8 to act as an Inverter by connecting the both inputs A and B. The updated testbench and simulation results are presented here.

#### 4.1.1 Testbench

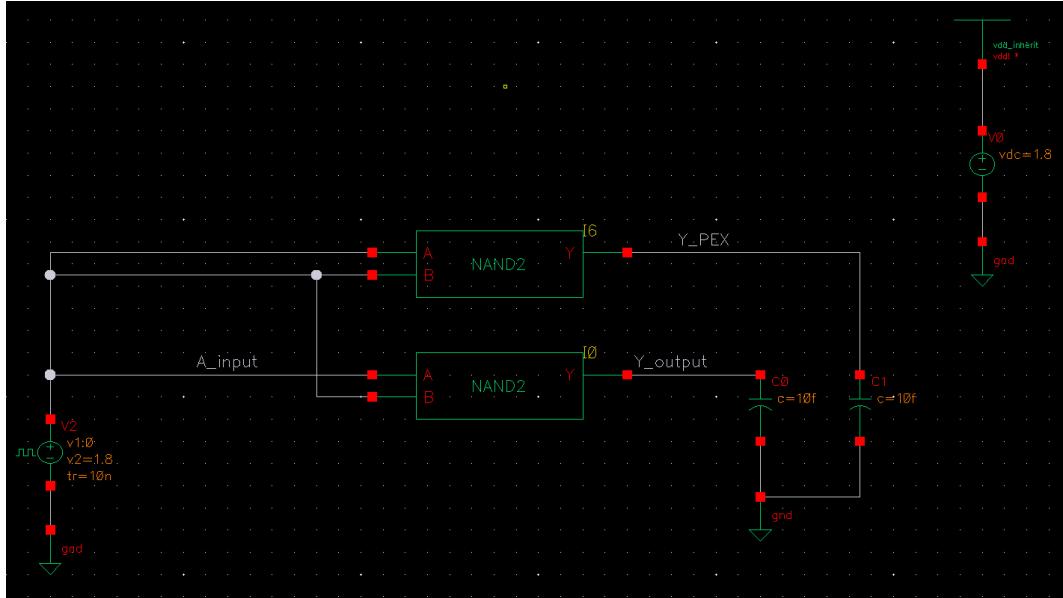


Figure 4.1: NAND2 as Inverter Testbench

In the Figure 4.1 the two inputs of the NAND2 are connected.

### 4.1.2 Simulation

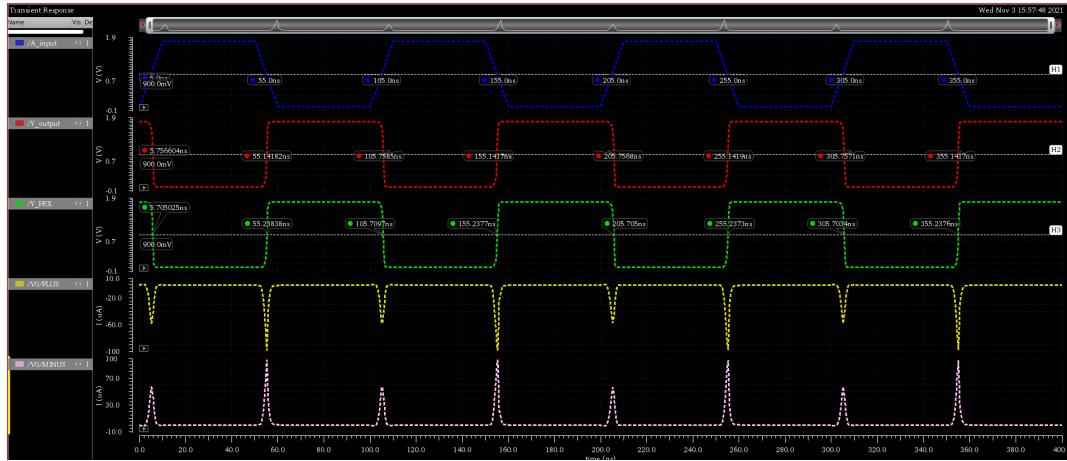


Figure 4.2: NAND2 as Inverter Simulation (pre and Post-layout)

### 4.1.3 Netlist

The netlist is divided into two parts to increase readability.

```

// Library name: LOGIC180
// Cell name: NAND2
// View name: calibre
// Inherited view list: spectre cmos_sch cmos.sch schematic veriloga ahdl
//pspice dspf calibre
subckt NAND2 A B Y
    MM1 (MM1_d MM1_g MM1_s MM1_b) nch l=1.8e-07 w=4.5e-07 m=1 nf=1 \
        sd=540.0n ad=5.625e-14 as=2.16e-13 pd=7e-07 ps=1.86e-06 \
        nrd=0.277778 nrs=1.06667 sa=9.1e-07 sb=4.8e-07 sca=2.26326 \
        scb=0.000186355 scc=8.38108e-08
    MM0 (MM0_d MM0_g MM0_s MM0_b) nch l=1.8e-07 w=4.5e-07 m=1 nf=1 \
        sd=540.0n ad=2.16e-13 as=5.625e-14 pd=1.86e-06 ps=7e-07 \
        nrd=1.06667 nrs=0.277778 sa=4.8e-07 sb=9.1e-07 sca=2.27997 \
        scb=0.000186356 scc=8.38108e-08
    MM3 (MM3_d MM3_g MM3_s MM3_b) pch l=1.8e-07 w=6e-07 m=1 nf=1 sd=540.0n \
        ad=1.62e-13 as=2.88e-13 pd=1.14e-06 ps=2.16e-06 nrd=0.45 nrs=0.8 \
        sa=1.2e-06 sb=4.8e-07 sca=5.13227 scb=0.00238452 scc=1.47283e-05
    MM2 (MM2_d MM2_g MM2_s MM2_b) pch l=1.8e-07 w=6e-07 m=1 nf=1 sd=540.0n \
        ad=1.62e-13 as=2.88e-13 pd=1.14e-06 ps=2.16e-06 nrd=0.45 nrs=0.8 \
        sa=4.8e-07 sb=1.2e-06 sca=5.66971 scb=0.0024349 scc=1.47315e-05
    cc_60 (Y c_116_n) capacitor c=0.0336702f
    cc_59 (Y c_113_n) capacitor c=0.0168351f
    cc_58 (c_82_n c_113_n) capacitor c=0.0116964f
    cc_57 (c_82_n c_108_n) capacitor c=0.00974388f
    cc_56 (c_73_n Y) capacitor c=0.0290377f
    cc_55 (c_70_n Y) capacitor c=0.0145188f
    cc_54 (c_70_n c_86_n) capacitor c=0.0297079f
    cc_53 (c_66_n c_86_n) capacitor c=0.00939528f
    cc_52 (c_64_n c_86_n) capacitor c=0.0297079f
    cc_51 (c_43_p c_116_n) capacitor c=0.00462831f
    cc_50 (c_43_p c_113_n) capacitor c=0.00231415f
    cc_49 (MM1_g c_113_n) capacitor c=0.0276109f
    cc_48 (c_32_n c_113_n) capacitor c=0.00156675f
    cc_47 (c_59_p c_108_n) capacitor c=0.00931551f
    cc_46 (c_32_n c_108_n) capacitor c=0.128331f
    cc_45 (c_29_n c_108_n) capacitor c=0.00208606f
    cc_44 (B c_108_n) capacitor c=0.0426398f
    cc_43 (c_46_p Y) capacitor c=0.00804244f
    cc_42 (c_36_n Y) capacitor c=0.00543562f
    cc_41 (c_43_p Y) capacitor c=0.0821191f
    cc_40 (MM1_g Y) capacitor c=0.0100685f
    cc_39 (MM3_g Y) capacitor c=0.00445057f
    cc_38 (MM3_g c_86_n) capacitor c=0.00938435f
    cc_37 (MM1_g c_82_n) capacitor c=0.00116471f
    cc_36 (c_32_n c_82_n) capacitor c=0.00314663f
    cc_35 (c_42_p c_76_n) capacitor c=0.0313777f
    cc_34 (c_46_p c_73_n) capacitor c=0.0229049f
    cc_33 (c_43_p c_73_n) capacitor c=0.00446034f

```

Figure 4.3: NAND2 as Inverter Netlist -1

```

rA_41 (A A_11) resistor r=7.8
rA_40 (A_11 c_3_p) resistor r=6.58242
rA_39 (c_2_p c_3_p) resistor r=51.7889
rA_38 (c_2_p c_19_p) resistor r=3.34522
rA_37 (A_19 MM2_g) resistor r=39.5
rA_36 (A_19 c_19_p) resistor r=0.869364
rA_35 (c_9_p A_24) resistor r=8.33889
rA_34 (c_9_p A_25) resistor r=4.41396
rA_33 (A_24 c_19_p) resistor r=3.34522
rA_32 (A_25 MM0_g) resistor r=24.7972
ends NAND2
// End of subcircuit definition.

// Library name: LOGIC180
// Cell name: NAND2
// View name: schematic
// Inherited view list: spectre cmos_sch cmos.sch schematic veriloga ahd1
// pspice dspf calibre
subckt NAND2_schematic A B Y inh_vdd
    M1 (net15 B 0 0) nch l=180.0n w=450.0n m=1 nf=1 sd=540.0n ad=2.16e-13 \
        as=2.16e-13 pd=1.86u ps=1.86u nrd=0.6 nrs=0.6 sa=480.0n sb=480.0n \
        sca=0 scb=0 scc=0
    M0 (Y A net15 0) nch l=180.0n w=450.0n m=1 nf=1 sd=540.0n ad=2.16e-13 \
        as=2.16e-13 pd=1.86u ps=1.86u nrd=0.6 nrs=0.6 sa=480.0n sb=480.0n \
        sca=0 scb=0 scc=0
    M3 (Y B inh_vdd inh_vdd) pch l=180.0n w=600n m=1 nf=1 sd=540.0n \
        ad=2.88e-13 as=2.88e-13 pd=2.16u ps=2.16u nrd=0.45 nrs=0.45 \
        sa=480.0n sb=480.0n sca=0 scb=0 scc=0
    M2 (Y A inh_vdd inh_vdd) pch l=180.0n w=600n m=1 nf=1 sd=540.0n \
        ad=2.88e-13 as=2.88e-13 pd=2.16u ps=2.16u nrd=0.45 nrs=0.45 \
        sa=480.0n sb=480.0n sca=0 scb=0 scc=0
ends NAND2_schematic
// End of subcircuit definition.

// Library name: LOGIC180
// Cell name: NAND2_tb
// View name: schematic
// Inherited view list: spectre cmos_sch cmos.sch schematic veriloga ahd1
// pspice dspf calibre
I6 (A_input A_input Y_PEX) NAND2
I0 (A_input A_input Y_output vdd!) NAND2_schematic
V0 (vdd! 0) vsource dc=1.8 type=dc
V2 (A_input 0) vsource dc=1.8 type=pulse val0=0 val1=1.8 period=100n \
    rise=10n fall=10n width=40n
C1 (Y_PEX 0) capacitor c=10f
C0 (Y_output 0) capacitor c=10f
simulatorOptions options reltol=1e-3 vabstol=1e-6 iabstol=1e-12 temp=27 \
    tnom=27 scalem=1.0 scale=1.0 gmin=1e-12 rforce=1 maxnotes=5 maxwarns=5 \
    digits=5 cols=80 pivrel=1e-3 sensfile="../psf/sens.output" \
    checklimitdest=psf
tran tran stop=400n write="spectre.ic" writefinal="spectre.fc" \
    annotate=status maxiters=5
finalTimeOP info what=oppoint where=rawfile
modelParameter info what=models where=rawfile
element info what=inst where=rawfile
outputParameter info what=output where=rawfile
designParamVals info what=parameters where=rawfile
primitives info what=primitives where=rawfile
subckts info what=subckts where=rawfile
save V0:p V0:p
saveOptions options save=allpub

```

Figure 4.4: NAND2 as Inverter Netlist -2

## 4.2 comparison of all 3 inverter designs:

All the 3 inverters are instantiated in the testbench shown in Figure 4.5.

### 4.2.1 Testbench

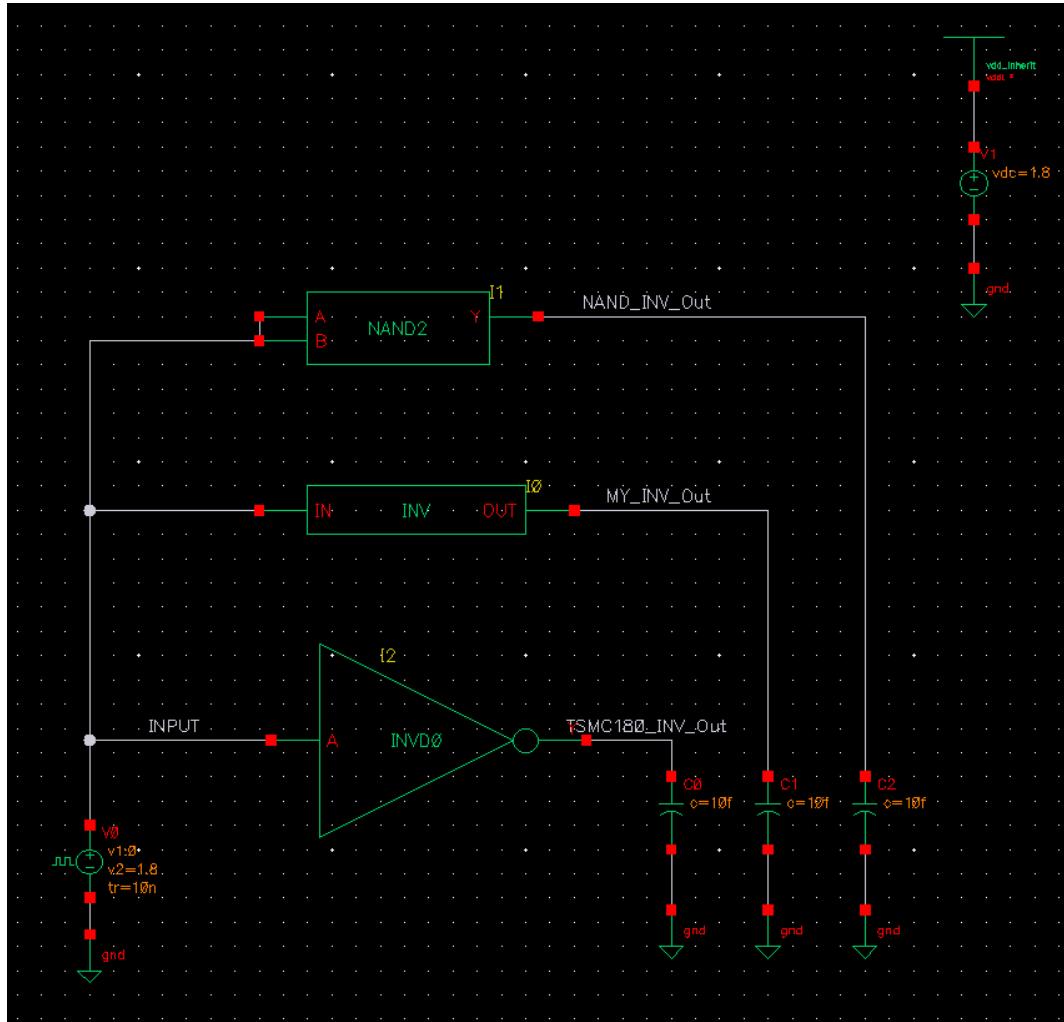


Figure 4.5: 3 Inverters Testbench

### 4.2.2 Pre-layout Simulation

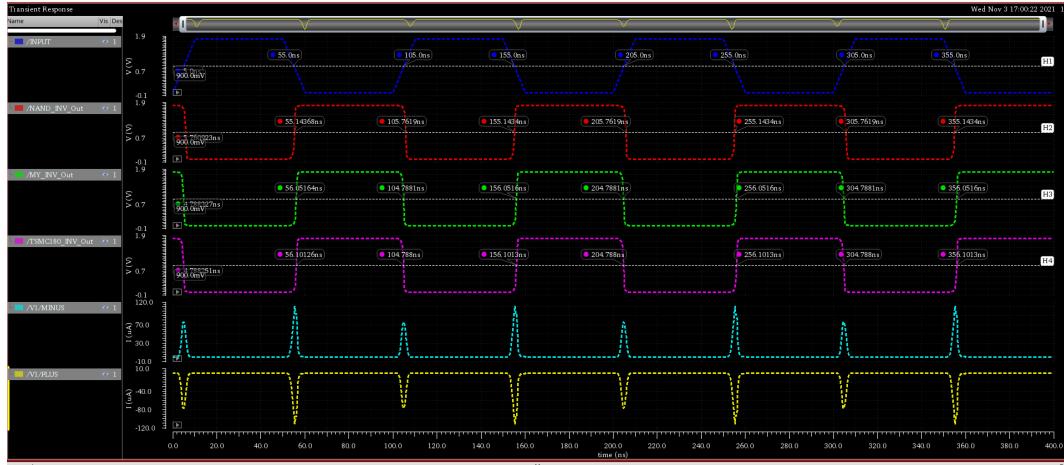


Figure 4.6: Pre-layout Simulation of 3 designs

The Simulation in Figure 4.6 shows the Pre-layout functional simulation of all the 3 designs.

### 4.2.3 Pre-layout Simulation netlist

The netlist is divided into two parts to increase readability.

```

// Library name: LOGIC180
// Cell name: INV
// View name: schematic
// Inherited view list: spectre cmos_sch cmos.sch schematic veriloga ahdl
//pspice dspf calibre
subckt INV_schematic IN OUT inh_vdd
    M0 (OUT IN 0 0) nch l=180.0n w=450.0n m=1 nf=1 sd=540.0n ad=2.16e-13 \
        as=2.16e-13 pd=1.86u ps=1.86u nrd=0.6 nrs=0.6 sa=480.0n sb=480.0n \
        sca=0 scb=0 scc=0
    M1 (OUT IN inh_vdd inh_vdd) pch l=180.0n w=600n m=1 nf=1 sd=540.0n \
        ad=2.88e-13 as=2.88e-13 pd=2.16u ps=2.16u nrd=0.45 nrs=0.45 \
        sa=480.0n sb=480.0n sca=0 scb=0 scc=0
ends INV_schematic
// End of subcircuit definition.

// Library name: LOGIC180
// Cell name: NAND2
// View name: schematic
// Inherited view list: spectre cmos_sch cmos.sch schematic veriloga ahdl
//pspice dspf calibre
subckt NAND2_schematic A B Y inh_vdd
    M1 (net15 B 0 0) nch l=180.0n w=450.0n m=1 nf=1 sd=540.0n ad=2.16e-13 \
        as=2.16e-13 pd=1.86u ps=1.86u nrd=0.6 nrs=0.6 sa=480.0n sb=480.0n \
        sca=0 scb=0 scc=0
    M0 (Y A net15 0) nch l=180.0n w=450.0n m=1 nf=1 sd=540.0n ad=2.16e-13 \
        as=2.16e-13 pd=1.86u ps=1.86u nrd=0.6 nrs=0.6 sa=480.0n sb=480.0n \
        sca=0 scb=0 scc=0
    M3 (Y B inh_vdd inh_vdd) pch l=180.0n w=600n m=1 nf=1 sd=540.0n \
        ad=2.88e-13 as=2.88e-13 pd=2.16u ps=2.16u nrd=0.45 nrs=0.45 \
        sa=480.0n sb=480.0n sca=0 scb=0 scc=0
    M2 (Y A inh_vdd inh_vdd) pch l=180.0n w=600n m=1 nf=1 sd=540.0n \
        ad=2.88e-13 as=2.88e-13 pd=2.16u ps=2.16u nrd=0.45 nrs=0.45 \
        sa=480.0n sb=480.0n sca=0 scb=0 scc=0
ends NAND2_schematic
// End of subcircuit definition.

// Library name: TSMC180_CELLS
// Cell name: INV0
// View name: schematic
// Inherited view list: spectre cmos_sch cmos.sch schematic veriloga ahdl
//pspice dspf calibre
subckt INV0_schematic A Y inh_vdd
    M1 (Y A 0 0) nch l=180.0n w=400n m=1 nf=1 sd=620.0n ad=2.164e-13 \
        as=2.164e-13 pd=1.88u ps=1.88u nrd=0.775 nrs=0.775 sa=520.0n \
        sb=520.0n sca=0 scb=0 scc=0
    M0 (Y A inh_vdd inh_vdd) pch l=180.0n w=540.0n m=1 nf=1 sd=540.0n \
        ad=2.592e-13 as=2.592e-13 pd=2.04u ps=2.04u nrd=0.5 nrs=0.5 \
        sa=480.0n sb=480.0n sca=0 scb=0 scc=0
ends INV0_schematic
// End of subcircuit definition.

```

Figure 4.7: Pre-layout Simulation Netlist -1

```
// Library name: LOGIC180
// Cell name: TASK4_tb
// View name: schematic
// Inherited view list: spectre cmos_sch cmos.sch schematic veriloga ahdl
// pspice dspf calibre
I0 (INPUT MY_INV_Out vdd!) INV_schematic
I1 (INPUT INPUT NAND_INV_Out vdd!) NAND2_schematic
I2 (INPUT TSMC180_INV_Out vdd!) INVD0_schematic
V0 (INPUT 0) vsource dc=1.8 type=pulse val0=0 val1=1.8 period=100n \
    rise=10n fall=10n width=40n
C2 (NAND_INV_Out 0) capacitor c=10f
C1 (MY_INV_Out 0) capacitor c=10f
C0 (TSMC180_INV_Out 0) capacitor c=10f
V1 (vdd! 0) vsource dc=1.8 type=dc
simulatorOptions options reltol=1e-3 vabstol=1e-6 iabstol=1e-12 temp=27 \
    tnom=27 scalem=1.0 scale=1.0 gmin=1e-12 rforce=1 maxnotes=5 maxwarns=5 \
    digits=5 cols=80 pivrel=1e-3 sensfile="../psf/sens.output" \
    checklimitdest=psf
tran tran stop=400n write="spectre.ic" writefinal="spectre.fc" \
    annotate=status maxiters=5
finalTimeOP info what=oppont where=rawfile
modelParameter info what=models where=rawfile
element info what=inst where=rawfile
outputParameter info what=output where=rawfile
designParamVals info what=parameters where=rawfile
primitives info what=primitives where=rawfile
subckts info what=subckts where=rawfile
save V1:p V1:p
saveOptions options save=allpub
```

Figure 4.8: Pre-layout Simulation Netlist -2

#### 4.2.4 Post-layout Simulation



Figure 4.9: Post layout Simulation of 3 designs

The Simulation in Figure 4.9 shows the Post-layout simulation of all the 3 designs.

#### 4.2.5 Post-layout Simulation netlist

The netlist is divided into four parts to increase readability.

---

```

// Library name: LOGIC180
// Cell name: INV
// View name: calibre
// Inherited view list: spectre cmos_sch cmos.sch schematic veriloga ahdl
//pspice dspf calibre
subckt INV IN OUT
    MM0 (MM0_d MM0_g MM0_s MM0_b) nch l=1.8e-07 w=4.5e-07 m=1 nf=1 \
        sd=540.0n ad=2.16e-13 as=2.16e-13 pd=1.86e-06 ps=1.86e-06 \
        nrd=1.06667 nrs=1.06667 sa=4.8e-07 sb=4.8e-07 sca=2.44077 \
        scb=0.000112946 scc=2.51784e-08
    MM1 (MM1_d MM1_g MM1_s MM1_b) pch l=1.8e-07 w=6e-07 m=1 nf=1 sd=540.0n \
        ad=2.88e-13 as=2.88e-13 pd=2.16e-06 ps=2.16e-06 nrd=0.8 nrs=0.8 \
        sa=4.8e-07 sb=4.8e-07 sca=5.80149 scb=0.00243491 scc=1.47315e-05
    cc_22 (OUT c_40_n) capacitor c=0.0290444f
    cc_21 (c_23_n c_40_n) capacitor c=0.0202772f
    cc_20 (c_23_n c_39_n) capacitor c=0.0145222f
    cc_19 (c_23_n c_37_n) capacitor c=0.0211383f
    cc_18 (c_18_n OUT) capacitor c=0.0216886f
    cc_17 (c_18_n c_24_n) capacitor c=0.0284977f
    cc_16 (c_17_n c_24_n) capacitor c=0.00995304f
    cc_15 (c_15_n c_24_n) capacitor c=0.0297079f
    cc_14 (MM0_g c_40_n) capacitor c=0.00549381f
    cc_13 (MM0_g c_39_n) capacitor c=0.00230988f
    cc_12 (MM0_g c_37_n) capacitor c=0.0142967f
    cc_11 (c_1_p c_37_n) capacitor c=0.00637946f
    cc_10 (c_10_p c_25_n) capacitor c=0.00688577f
    cc_9 (MM1_g c_25_n) capacitor c=0.00162394f
    cc_8 (MM0_g c_25_n) capacitor c=0.00162394f
    cc_7 (IN c_25_n) capacitor c=0.0099802f
    cc_6 (MM1_g c_24_n) capacitor c=0.0296066f
    cc_5 (MM0_g c_23_n) capacitor c=0.0210336f
    cc_4 (MM1_g c_18_n) capacitor c=0.00579981f
    cc_3 (MM1_g c_17_n) capacitor c=0.00187313f
    cc_2 (MM1_g c_15_n) capacitor c=0.0169982f
    cc_1 (c_1_p c_15_n) capacitor c=0.00547307f

```

Figure 4.10: Post layout Simulation netlist -1

```

rVDD!_15 (VDD!_12 c_18_n) resistor r=0.172957
rVDD!_14 (VDD!_12 VDD!_20) resistor r=0.00090057
rIN_27 (IN IN_11) resistor r=7.8
rIN_26 (IN_11 IN_30) resistor r=6.58242
rIN_25 (c_1_p IN_30) resistor r=33.575
rIN_24 (c_1_p c_10_p) resistor r=0.869364
rIN_23 (IN_19 MM0_g) resistor r=32.2583
rIN_22 (IN_19 c_10_p) resistor r=3.34522
rIN_21 (IN_23 MM1_g) resistor r=44.3278
rIN_20 (IN_23 c_10_p) resistor r=3.34522
ends INV
// End of subcircuit definition.

// Library name: LOGIC180
// Cell name: NAND2
// View name: calibre
// Inherited view list: spectre cmos_sch cmos.sch schematic veriloga ahdl
//pspice dspf calibre
subckt NAND2 A B Y
    MM1 (MM1_d MM1_g MM1_s MM1_b) nch l=1.8e-07 w=4.5e-07 m=1 nf=1 \
        sd=540.0n ad=5.625e-14 as=2.16e-13 pd=7e-07 ps=1.86e-06 \
        nrd=0.277778 nrs=1.06667 sa=9.1e-07 sb=4.8e-07 sca=2.26326 \
        scb=0.000186355 scc=8.38108e-08
    MM0 (MM0_d MM0_g MM0_s MM0_b) nch l=1.8e-07 w=4.5e-07 m=1 nf=1 \
        sd=540.0n ad=2.16e-13 as=5.625e-14 pd=1.86e-06 ps=7e-07 \
        nrd=1.06667 nrs=0.277778 sa=4.8e-07 sb=9.1e-07 sca=2.27997 \
        scb=0.000186356 scc=8.38108e-08
    MM3 (MM3_d MM3_g MM3_s MM3_b) pch l=1.8e-07 w=6e-07 m=1 nf=1 sd=540.0n \
        ad=1.62e-13 as=2.88e-13 pd=1.14e-06 ps=2.16e-06 nrd=0.45 nrs=0.8 \
        sa=1.2e-06 sb=4.8e-07 sca=5.13227 scb=0.00238452 scc=1.47283e-05
    MM2 (MM2_d MM2_g MM2_s MM2_b) pch l=1.8e-07 w=6e-07 m=1 nf=1 sd=540.0n \
        ad=1.62e-13 as=2.88e-13 pd=1.14e-06 ps=2.16e-06 nrd=0.45 nrs=0.8 \
        sa=4.8e-07 sb=1.2e-06 sca=5.66971 scb=0.0024349 scc=1.47315e-05
    cc_60 (Y c_116_n) capacitor c=0.0336702f
    cc_59 (Y c_113_n) capacitor c=0.0168351f
    cc_58 (c_82_n c_113_n) capacitor c=0.0116964f
    cc_57 (c_82_n c_108_n) capacitor c=0.00974388f
    cc_56 (c_73_n Y) capacitor c=0.0290377f
    cc_55 (c_70_n Y) capacitor c=0.0145188f
    cc_54 (c_70_n c_86_n) capacitor c=0.0297079f
    cc_53 (c_66_n c_86_n) capacitor c=0.00939528f
    cc_52 (c_64_n c_86_n) capacitor c=0.0297079f
    cc_51 (c_43_p c_116_n) capacitor c=0.00462831f
    cc_50 (c_43_p c_113_n) capacitor c=0.00231415f
    cc_49 (MM1_g c_113_n) capacitor c=0.0276109f
    cc_48 (c_32_n c_113_n) capacitor c=0.00156675f
    cc_47 (c_59_p c_108_n) capacitor c=0.00931551f

```

Figure 4.11: Post layout Simulation netlist -2

```

rA_40 (A_11 c_3_p) resistor r=6.58242
rA_39 (c_2_p c_3_p) resistor r=51.7889
rA_38 (c_2_p c_19_p) resistor r=3.34522
rA_37 (A_19 MM2_g) resistor r=39.5
rA_36 (A_19 c_19_p) resistor r=0.869364
rA_35 (c_9_p A_24) resistor r=8.33889
rA_34 (c_9_p A_25) resistor r=4.41396
rA_33 (A_24 c_19_p) resistor r=3.34522
rA_32 (A_25 MM0_g) resistor r=24.7972
ends NAND2
// End of subcircuit definition.

// Library name: TSMC180_CELLS
// Cell name: INVD0
// View name: calibre
// Inherited view list: spectre cmos_sch cmos.sch schematic veriloga ahdl
//pspice dspf calibre
subckt INVD0 A Y
    MM1 (MM1_d MM1_g MM1_s MM1_b) nch l=1.8e-07 w=400n m=1 nf=1 sd=620.0n \
        ad=2.164e-13 as=2.164e-13 pd=1.88e-06 ps=1.88e-06 nrd=1.3525 \
        nrs=1.3525 sa=5.2e-07 sb=5.2e-07 sca=2.6068 scb=7.92728e-05 \
        scc=6.40404e-09
    MM0 (MM0_d MM0_g MM0_s MM0_b) pch l=1.8e-07 w=540.0n m=1 nf=1 \
        sd=540.0n ad=2.592e-13 as=2.592e-13 pd=2.04e-06 ps=2.04e-06 \
        nrd=0.888889 nrs=0.888889 sa=4.8e-07 sb=4.8e-07 sca=6.08081 \
        scb=0.00268964 scc=1.63675e-05
    cc_21 (c_39_n c_25_n) capacitor c=0.0187354f
    cc_20 (c_38_n c_25_n) capacitor c=0.017727f
    cc_19 (c_35_n c_25_n) capacitor c=0.0170204f
    cc_18 (c_24_n c_20_n) capacitor c=0.0252095f
    cc_17 (c_24_n c_19_n) capacitor c=0.0131534f
    cc_16 (c_24_n c_16_n) capacitor c=0.02628f
    cc_15 (c_39_n MM1_g) capacitor c=0.00520076f
    cc_14 (c_38_n MM1_g) capacitor c=0.00435309f
    cc_13 (c_35_n c_3_p) capacitor c=0.00158058f
    cc_12 (c_35_n MM1_g) capacitor c=0.0131957f
    cc_11 (c_35_n c_1_p) capacitor c=0.0110803f
    cc_10 (Y c_3_p) capacitor c=0.0090338f

```

Figure 4.12: Post layout Simulation netlist -3

```

----- V----- / ----- -----
rVDD!_18 (VDD! c_19_n) resistor r=0.129447
rVDD!_17 (VDD!_21 c_19_n) resistor r=0.0324416
rVDD!_16 (VDD! VDD!_10) resistor r=0.0788298
rVDD!_15 (VDD!_20 VDD!_10) resistor r=0.0176132
rVDD!_14 (c_20_n VDD!_13) resistor r=0.134804
rVDD!_13 (VDD!_21 VDD!_13) resistor r=0.00907945
rA_25 (c_1_p A) resistor r=0.0663
rA_24 (A_12 c_1_p) resistor r=7.8
rA_23 (c_3_p A_12) resistor r=4.32619
rA_22 (MM1_g A_18) resistor r=35.55
rA_21 (c_3_p A_18) resistor r=3.70884
rA_20 (MM0_g A_22) resistor r=37.7444
rA_19 (c_3_p A_22) resistor r=3.70884
ends INV0
// End of subcircuit definition.

// Library name: LOGIC180
// Cell name: TASK4_tb
// View name: schematic
// Inherited view list: spectre cmos_sch cmos.sch schematic veriloga ahdl
// pspice dspf calibre
I0 (INPUT MY_INV_Out) INV
I1 (INPUT INPUT NAND_INV_Out) NAND2
I2 (INPUT TSMC180_INV_Out) INV0
V0 (INPUT 0) vsource dc=1.8 type=pulse val0=0 val1=1.8 period=100n \
    rise=10n fall=10n width=40n
C2 (NAND_INV_Out 0) capacitor c=10f
C1 (MY_INV_Out 0) capacitor c=10f
C0 (TSMC180_INV_Out 0) capacitor c=10f
V1 (vdd! 0) vsource dc=1.8 type=dc
simulatorOptions options reltol=1e-3 vabstol=1e-6 iabstol=1e-12 temp=27 \
    tnom=27 scalem=1.0 scale=1.0 gmin=1e-12 rforce=1 maxnotes=5 maxwarns=5 \
    digits=5 cols=80 pivrel=1e-3 sensfile="../psf/sens.output" \
    checklimitdest=psf
tran tran stop=400n write="spectre.ic" writefinal="spectre.fc" \
    annotate=status maxiters=5
finalTimeOP info what=oppoint where=rawfile
modelParameter info what=models where=rawfile
element info what=inst where=rawfile
outputParameter info what=output where=rawfile
designParamVals info what=parameters where=rawfile
primitives info what=primitives where=rawfile
subckts info what=subckts where=rawfile
save V1:p V1:p
saveOptions options save=allpub

```

Figure 4.13: Post layout Simulation netlist -4

## 4.3 Pre and Post-layout propagation delays

### 4.3.1 Pre-layout propagation delay

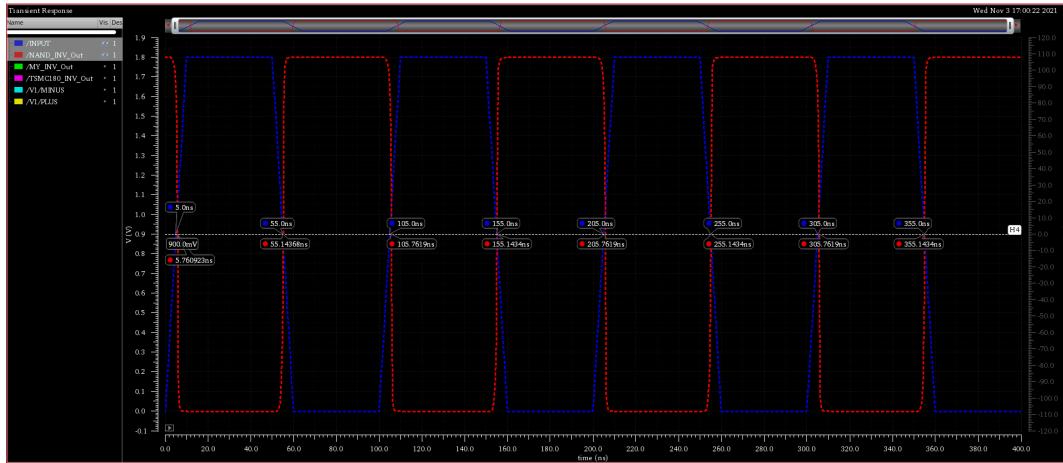


Figure 4.14: Pre-layout Simulation of NAND2 as Inverter

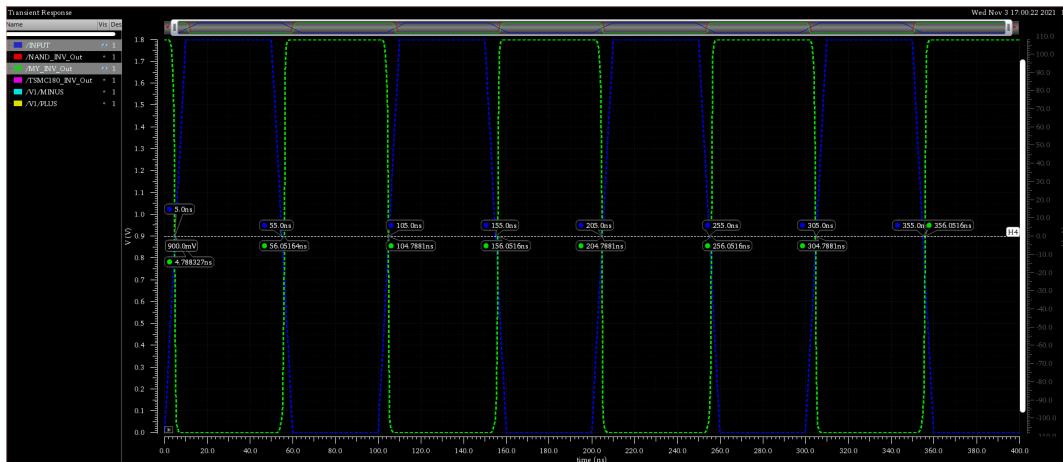


Figure 4.15: Pre-layout Simulation of Inverter

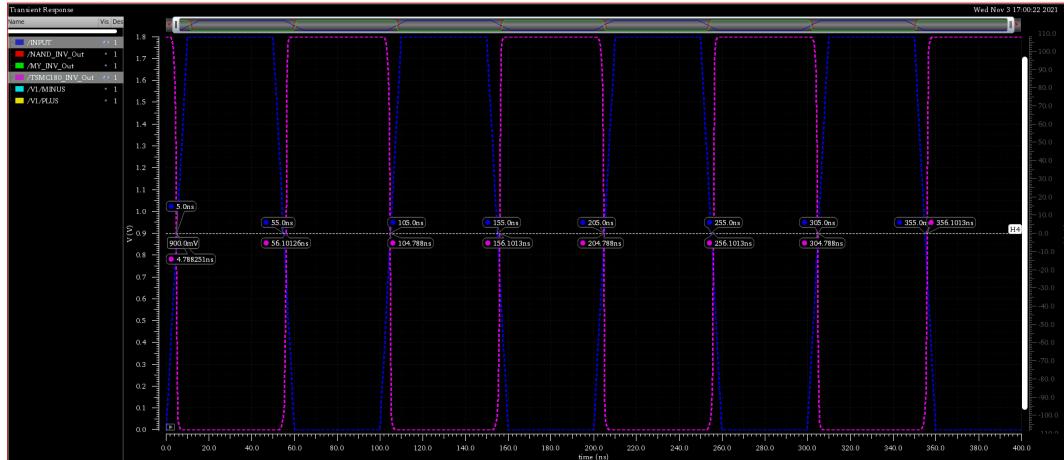


Figure 4.16: Pre-layout Simulation of TSMC180-cell Inverter

Pre-layout Propagation delay of NAND2 as Inverter design is calculated from the simulation shown in Figure 4.14.

- 1)  $t_{phl} = 105.7619 - 105 = 0.7619\text{ns}$  ( transition in between 100 - 120ns region)
- 2)  $t_{plh} = 155.1434 - 155 = 0.1434\text{ns}$  (transition in between 140 - 160ns region)

Pre-layout Propagation delay of Inverter design is calculated from the simulation shown in Figure 4.15.

- 1)  $t_{phl} = 104.7881 - 105 = -0.2119\text{ns}$  ( transition in between 100 - 120ns region)
- 2)  $t_{plh} = 156.0516 - 155 = 1.0516\text{ns}$  (transition in between 140 - 160ns region)

Pre-layout Propagation delay of TSMC180 Inverter design is calculated from the simulation shown in Figure 4.16.

- 1)  $t_{phl} = 104.788 - 105 = -0.212\text{ns}$  ( transition in between 100 - 120ns region)
- 2)  $t_{plh} = 156.1013 - 155 = 1.1013\text{ns}$  (transition in between 140 - 160ns region)

### 4.3.2 Post-layout propagation

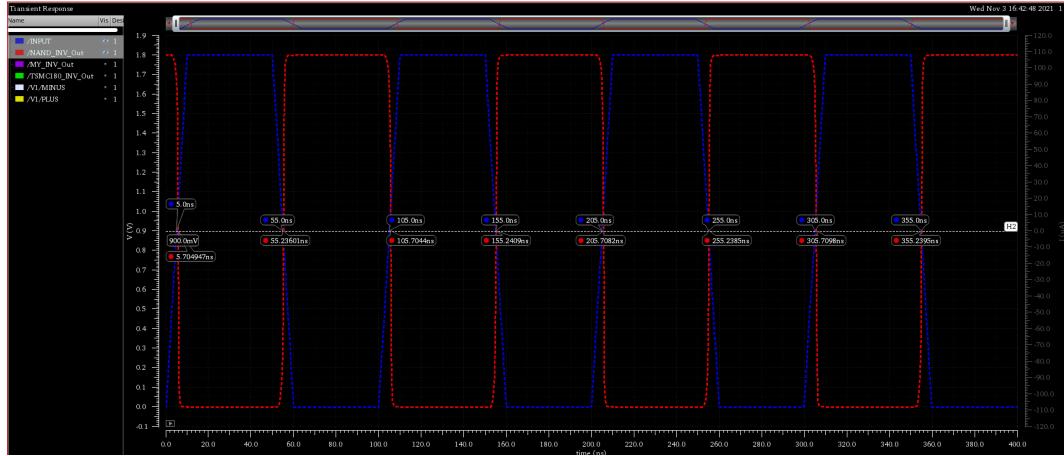


Figure 4.17: Post-layout Simulation of NAND2 as Inverter

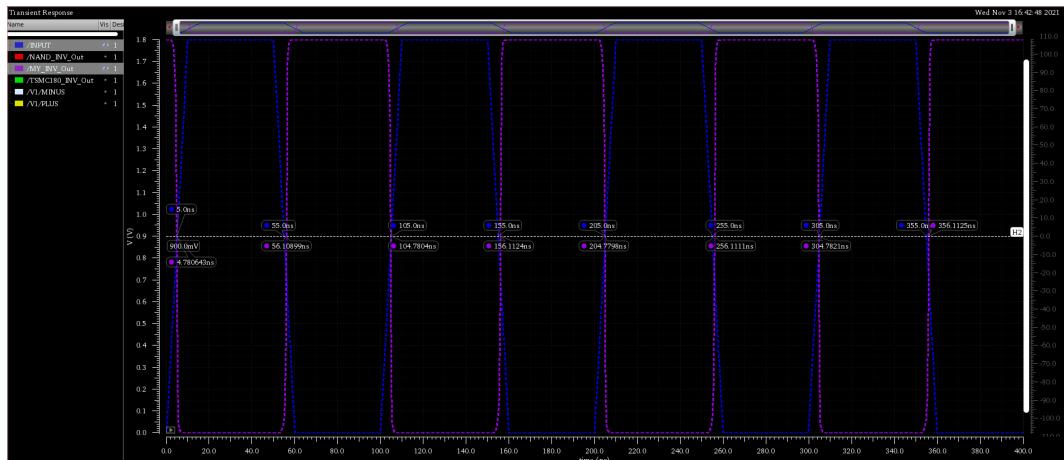


Figure 4.18: Post-layout Simulation of Inverter

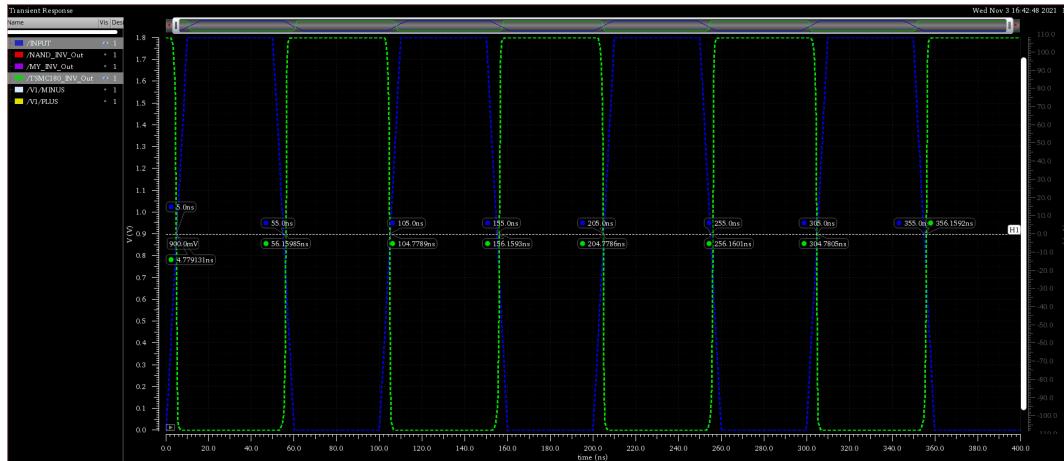


Figure 4.19: Post-layout Simulation of TSMC180-cell Inverter

Post-layout Propagation delay of NAND2 as Inverter design is calculated from the simulation shown in Figure 4.17.

- 1)  $t_{phl} = 105.7044 - 105 = 0.7044\text{ns}$  ( transition in between 100 - 120ns region)
- 2)  $t_{plh} = 155.2409 - 155 = 0.2409\text{ns}$  (transition in between 140 - 160ns region)

Post-layout Propagation delay of Inverter design is calculated from the simulation shown in Figure 4.18.

- 1)  $t_{phl} = 104.7804 - 105 = -0.2196\text{ns}$  ( transition in between 100 - 120ns region)
- 2)  $t_{plh} = 156.1124 - 155 = 1.1124\text{ns}$  (transition in between 140 - 160ns region)

Post-layout Propagation delay of TSMC180 Inverter design is calculated from the simulation shown in Figure 4.19.

- 1)  $t_{phl} = 104.7789 - 105 = -0.2211\text{ns}$  ( transition in between 100 - 120ns region)
- 2)  $t_{plh} = 156.1593 - 155 = 1.1593\text{ns}$  (transition in between 140 - 160ns region)

NMOS are faster compared to PMOS because electrons are the carriers in the NMOS and they are faster than holes that are the carriers in PMOS. Even though the width of PMOS (600nm in the design) is greater than NMOS width of NMOS(450nm in the design), it is not enough. This is the reason transition of output from 1 to 0 is faster than transition from 0 to 1.

## 4.4 Standard Cell

A standard cell is a region in which transistors are used to implement a function. A standard cell has a standard height(in the lab design 3.92 um) so that it is easier to interface different functions implemented using standard cells to construct even more complex functions.

# Chapter 5

## LAB 5

### 5.1 Gate level layouts for XOR

#### 5.1.1 XOR using TSMC180 NAND cell

In this section schematics,layout,DRC,LVS of XOR using TSMC180 NAND gates are presented. Five NAND gates were used for constructing XOR gate.

#### Schematic

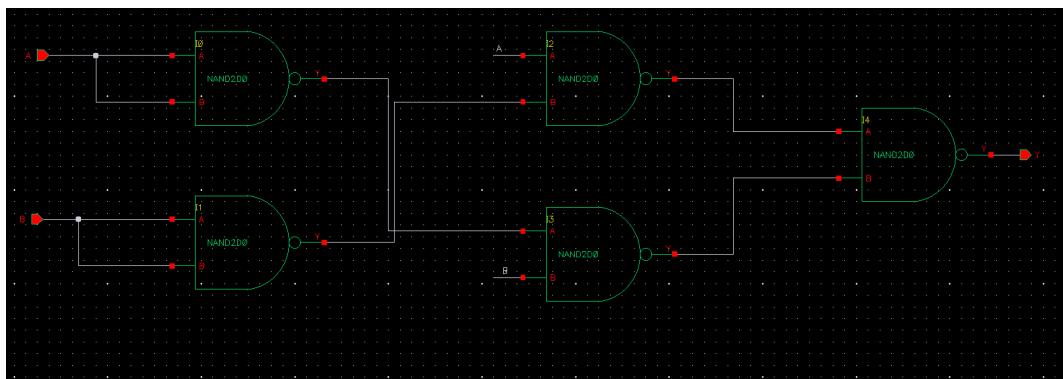


Figure 5.1: Schematic of XOR using TSMC180 NAND gates

#### Layout

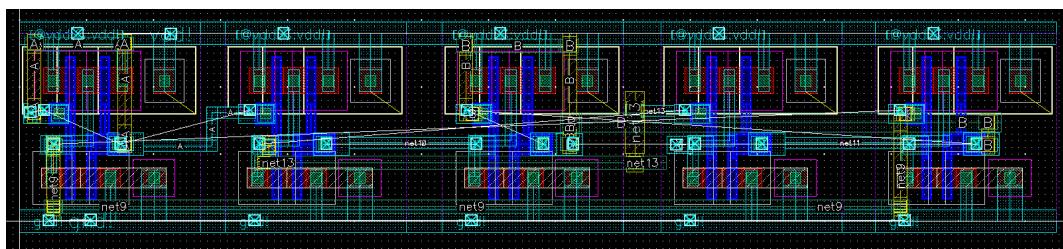


Figure 5.2: Layout of XOR using TSMC180 NAND gates

## DRC

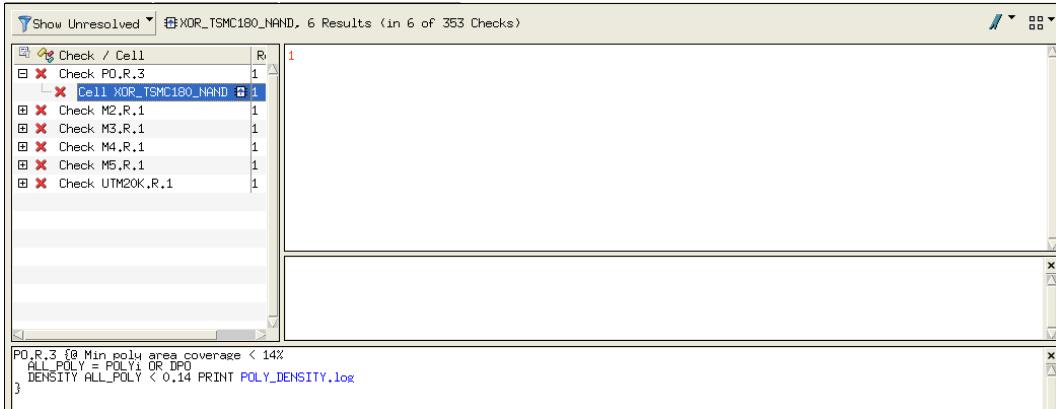


Figure 5.3: DRC of XOR using TSMC180 NAND gates

## LVS

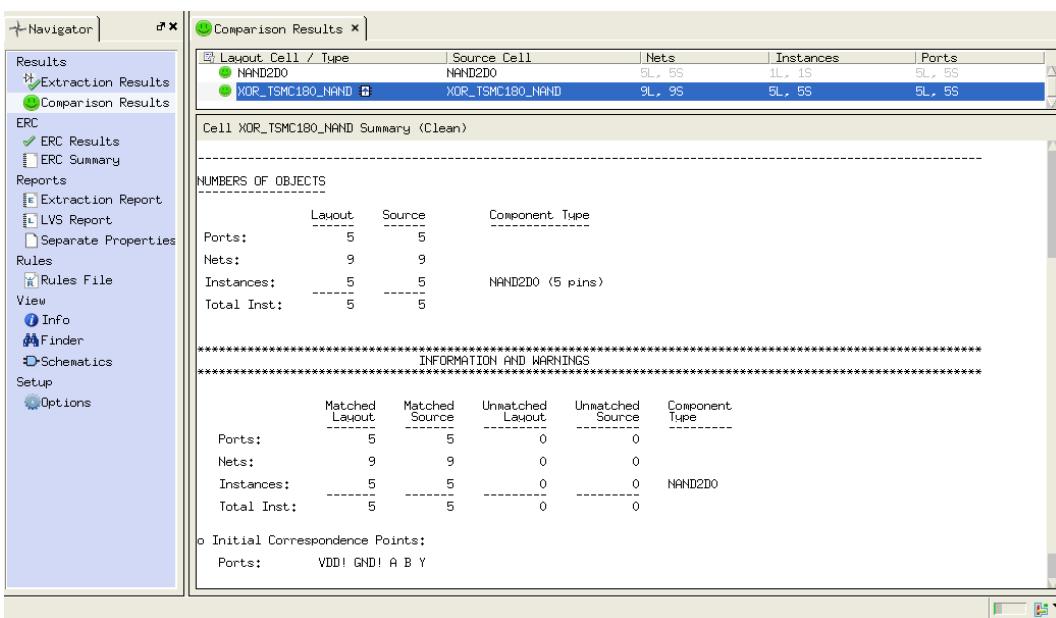


Figure 5.4: LVS of XOR using TSMC180 NAND gates

### 5.1.2 XOR using my own NAND cell

In this section schematics,layout,DRC,LVS of XOR using my own NAND gates are presented. Five NAND gates were used for constructing XOR gate.

#### Schematic

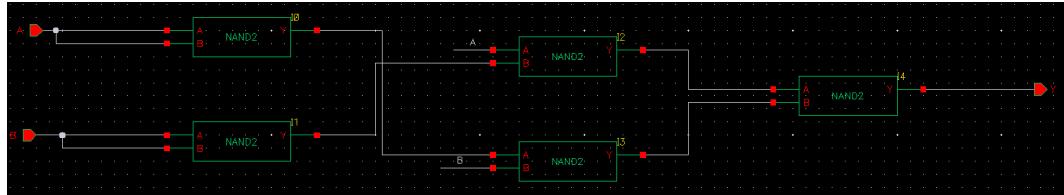


Figure 5.5: Schematic of XOR using my own NAND gates

#### Layout

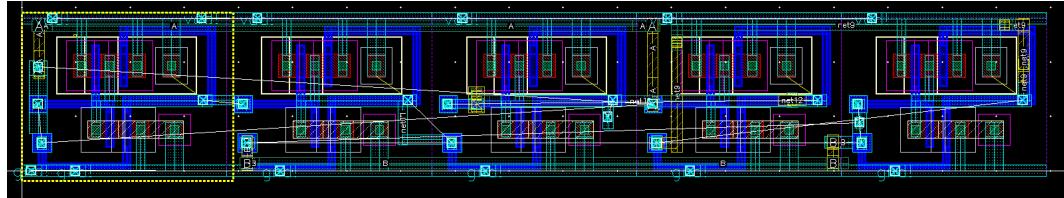


Figure 5.6: Layout of XOR using my own NAND gates

#### DRC

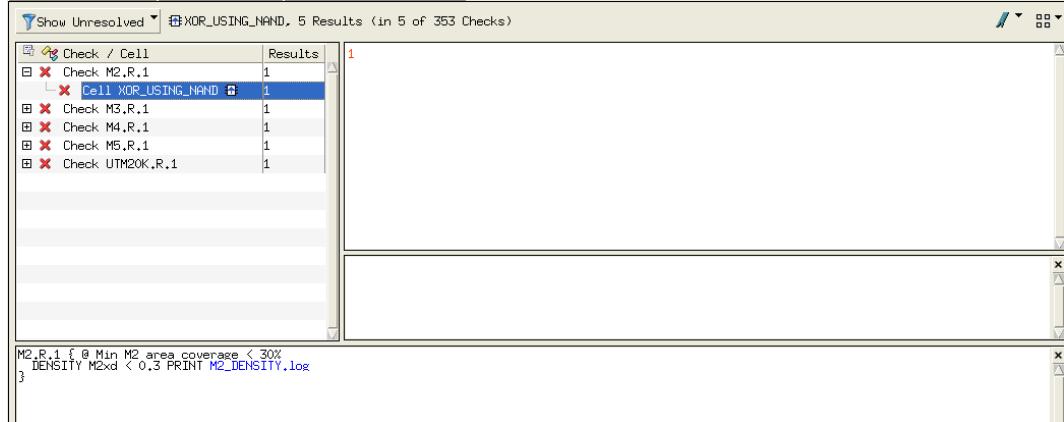


Figure 5.7: DRC of XOR using my own NAND gates

#### LVS

#### TestBench

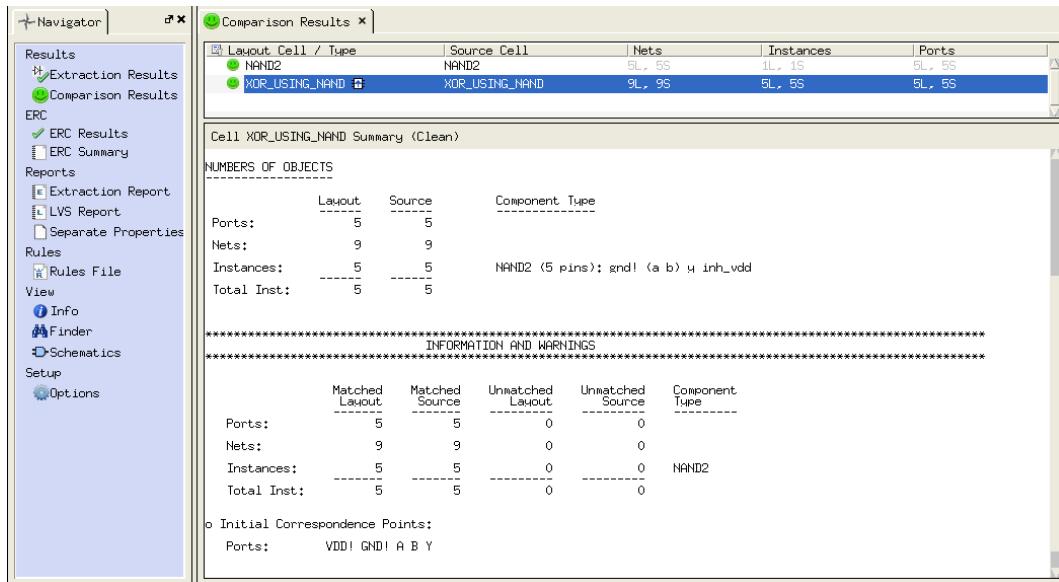


Figure 5.8: LVS of XOR using my own NAND gates

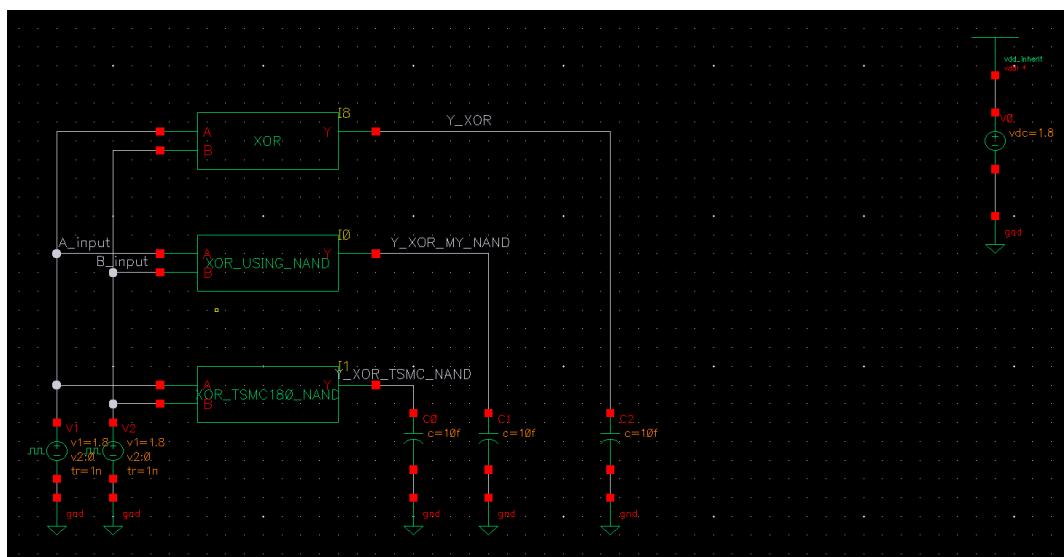


Figure 5.9: Testbench with 3 designs

## 5.2 Simulations

Fig 5.10 and 5.11 shows the simulation output of all 3 designs pre-layout and post-layout. Propagation delays are shown in next section. The propagation delays are different between standard cell-based implementation and schematic XOR design because the propagation delay in a standard cell implementation depends on drive strength of the cell.

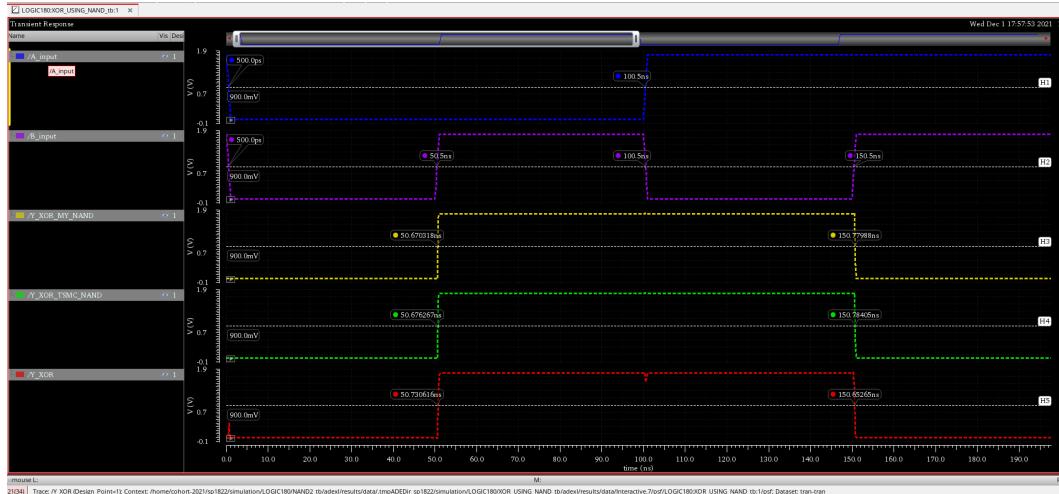


Figure 5.10: Pre-layout simulation comparison

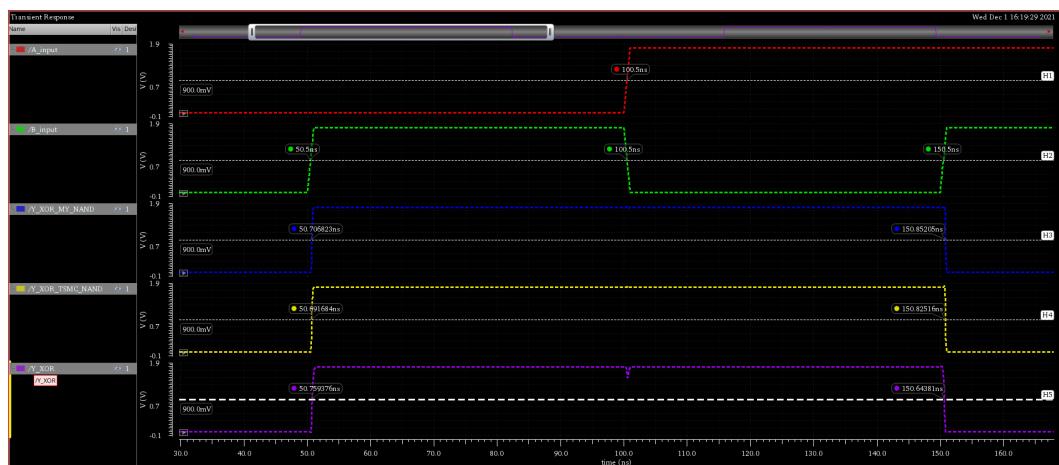


Figure 5.11: Post-layout simulation comparison

## 5.3 Propagation delays

### 5.3.1 Pre-layout

Propagation delay between output and input B are calculated from the Fig 5.12 and 5.13.

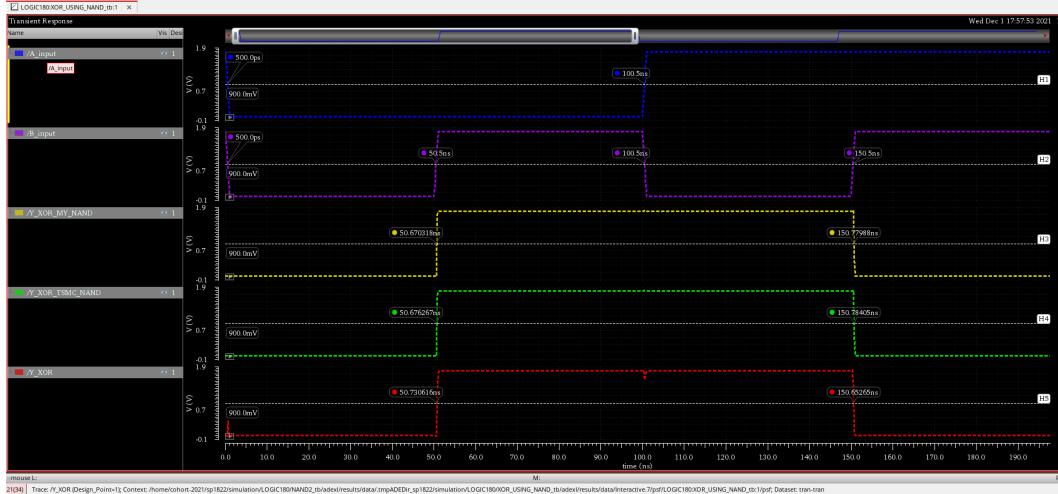


Figure 5.12: Pre-layout simulation comparison -1

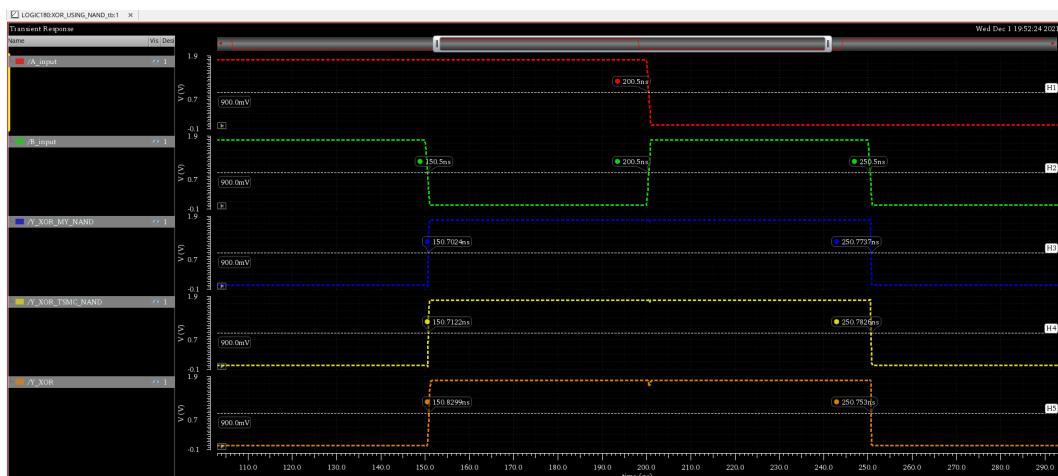


Figure 5.13: Pre-layout simulation comparison -2

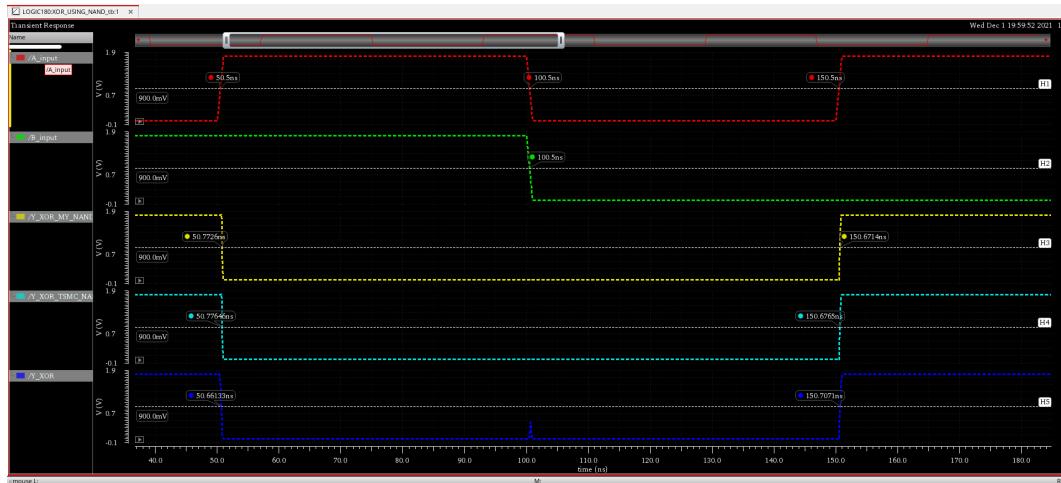


Figure 5.14: Pre-layout simulation comparison -3

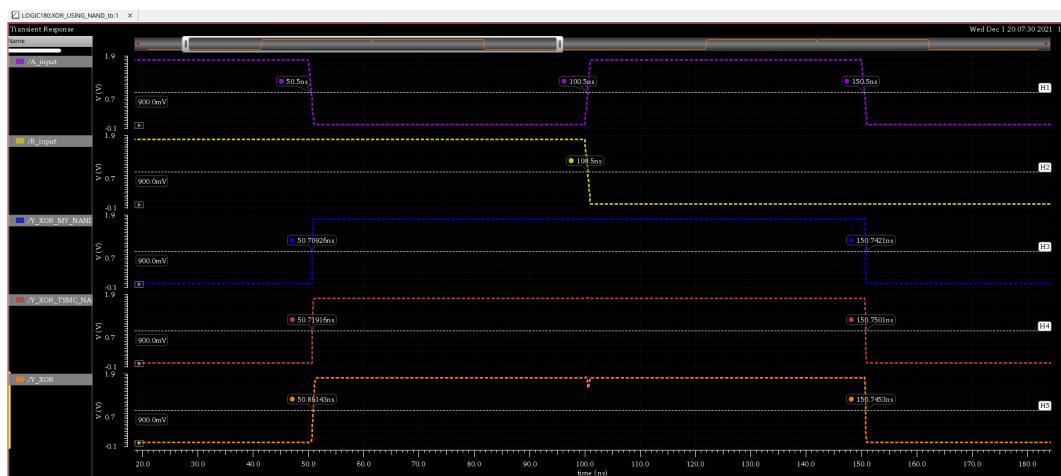


Figure 5.15: Pre-layout simulation comparison -4

Propagation delay between output and input A are calculated from the Fig 5.14 and 5.15.

### Pre-layout Propagation delays:

#### 1. using TSMC180\_cells:

- $t_{phl} = 150.78405\text{ns} - 150.5\text{ns} = 284.05 \text{ ps}$  (B goes low to high)
- $t_{phl} = 250.7826\text{ns} - 250.5\text{ns} = 282.6 \text{ ps}$  (B goes high to low)
- $t_{phl} = 50.77646\text{ns} - 50.5\text{ns} = 276.46 \text{ ps}$  (A goes low to high)
- $t_{phl} = 150.7501\text{ns} - 150.5\text{ns} = 250.1 \text{ ps}$  (A goes high to low)
- $t_{plh} = 50.676267\text{ns} - 50.5\text{ns} = 176.267 \text{ ps}$  (B goes low to high)
- $t_{plh} = 150.7122\text{ns} - 150.5\text{ns} = 212.2 \text{ ps}$  (B goes high to low)
- $t_{plh} = 150.6765\text{ns} - 150.5\text{ns} = 176.5 \text{ ps}$  (A goes low to high)
- $t_{plh} = 50.71916\text{ns} - 50.5\text{ns} = 219.16 \text{ ps}$  (A goes high to low)

2. XOR using my own NAND cell:

- (a)  $t_{phl} = 150.77988\text{ns} - 150.5\text{ns} = 279.88 \text{ ps}$  (B goes low to high)
- (b)  $t_{phl} = 250.7737\text{ns} - 250.5\text{ns} = 273.7 \text{ ps}$  (B goes high to low)
- (c)  $t_{phl} = 50.7726\text{ns} - 50.5\text{ns} = 272.6 \text{ ps}$  (A goes low to high)
- (d)  $t_{phl} = 150.7421\text{ns} - 150.5\text{ns} = 242.1 \text{ ps}$  (A goes high to low)
- (e)  $t_{plh} = 50.670318\text{ns} - 50.5\text{ns} = 170.318 \text{ ps}$  (B goes low to high)
- (f)  $t_{plh} = 150.7024\text{ns} - 150.5\text{ns} = 202.4 \text{ ps}$  (B goes high to low)
- (g)  $t_{plh} = 150.6714\text{ns} - 150.5\text{ns} = 171.4 \text{ ps}$  (A goes low to high)
- (h)  $t_{plh} = 50.70926 - 50.5\text{ns} = 209.26 \text{ ps}$  (A goes high to low)

3. XOR transistor level design:

- (a)  $t_{phl} = 150.65265\text{ns} - 150.5\text{ns} = 152.65 \text{ ps}$  (B goes low to high)
- (b)  $t_{phl} = 250.753\text{ns} - 250.5\text{ns} = 253 \text{ ps}$  (B goes high to low)
- (c)  $t_{phl} = 50.66133\text{ns} - 50.5\text{ns} = 161.33 \text{ ps}$  (A goes low to high)
- (d)  $t_{phl} = 150.7453\text{ns} - 150.5\text{ns} = 245.3 \text{ ps}$  (A goes high to low)
- (e)  $t_{plh} = 50.730616\text{ns} - 50.5\text{ns} = 230.616 \text{ ps}$  (B goes low to high)
- (f)  $t_{plh} = 150.8299\text{ns} - 150.5\text{ns} = 329.9 \text{ ps}$  (B goes high to low)
- (g)  $t_{plh} = 150.7071\text{ns} - 150.5\text{ns} = 207.1 \text{ ps}$  (A goes low to high)
- (h)  $t_{plh} = 50.86143\text{ns} - 50.5\text{ns} = 361.43 \text{ ps}$  (A goes high to low)

### 5.3.2 Post-layout

Propagation delay between output and input B are calculated from the Fig 5.16 and 5.17.



Figure 5.16: Post-layout simulation comparison -1



Figure 5.17: Post-layout simulation comparison -2

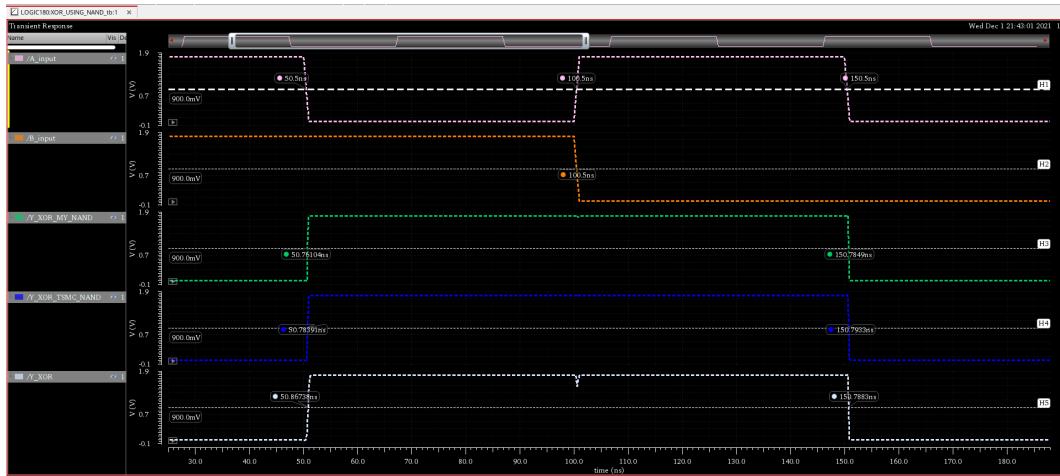


Figure 5.18: Post-layout simulation comparison -3

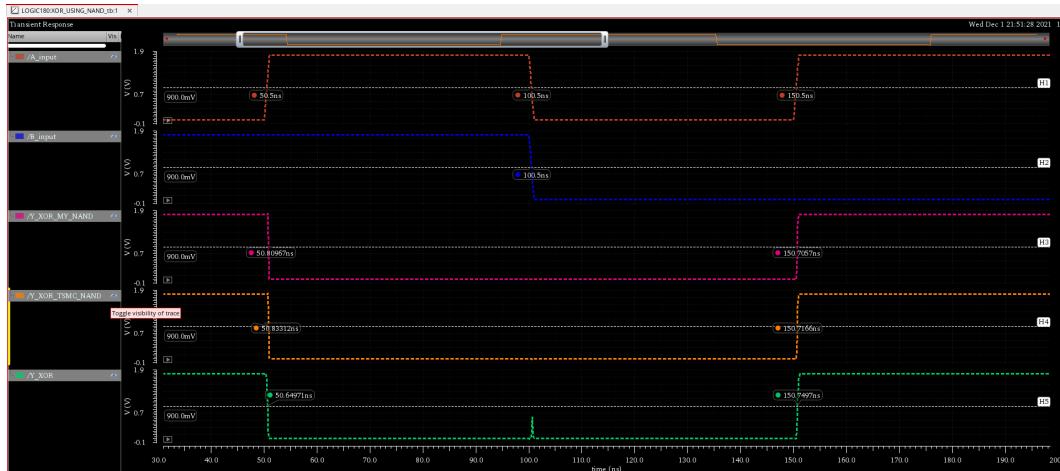


Figure 5.19: Post-layout simulation comparison -4

Propagation delay between output and input A are calculated from the Fig 5.18 and 5.19.

### Post-layout Propagation delays:

1. using TSMC180\_cells:
  - (a)  $t_{phl} = 150.82516\text{ns} - 150.5\text{ns} = 325.16\text{ ps}$  (B goes low to high)
  - (b)  $t_{phl} = 250.8021\text{ns} - 250.5\text{ns} = 302.1\text{ ps}$  (B goes high to low)
  - (c)  $t_{phl} = 50.83312\text{ns} - 50.5\text{ns} = 333.12\text{ ps}$  (A goes low to high)
  - (d)  $t_{phl} = 150.7933\text{ns} - 150.5\text{ns} = 293.3\text{ ps}$  (A goes high to low)
  - (e)  $t_{plh} = 50.691684\text{ns} - 50.5\text{ns} = 191.684\text{ ps}$  (B goes low to high)
  - (f)  $t_{plh} = 150.76567\text{ns} - 150.5\text{ns} = 265.67\text{ ps}$  (B goes high to low)
  - (g)  $t_{plh} = 150.7166\text{ns} - 150.5\text{ns} = 216.6\text{ ps}$  (A goes low to high)
  - (h)  $t_{plh} = 50.78391\text{ns} - 50.5\text{ns} = 383.91\text{ ps}$  (A goes high to low)

2. XOR using my own NAND cell:

- (a)  $t_{phl} = 150.85205\text{ns} - 150.5\text{ns} = 352.05 \text{ ps}$  (B goes low to high)
- (b)  $t_{phl} = 250.81268\text{ns} - 250.5\text{ns} = 312.68 \text{ ps}$  (B goes high to low)
- (c)  $t_{phl} = 50.80967\text{ns} - 50.5\text{ns} = 309.67 \text{ ps}$  (A goes low to high)
- (d)  $t_{phl} = 150.7849\text{ns} - 150.5\text{ns} = 284.9 \text{ ps}$  (A goes high to low)
- (e)  $t_{plh} = 50.706823\text{ns} - 50.5\text{ns} = 206.823 \text{ ps}$  (B goes low to high)
- (f)  $t_{plh} = 150.77803\text{ns} - 150.5\text{ns} = 278.03 \text{ ps}$  (B goes high to low)
- (g)  $t_{plh} = 150.7057\text{ns} - 150.5\text{ns} = 205.7 \text{ ps}$  (A goes low to high)
- (h)  $t_{plh} = 50.76104 - 50.5\text{ns} = 261.04 \text{ ps}$  (A goes high to low)

3. XOR transistor level design:

- (a)  $t_{phl} = 150.64381\text{ns} - 150.5\text{ns} = 143.81 \text{ ps}$  (B goes low to high)
- (b)  $t_{phl} = 250.78632\text{ns} - 250.5\text{ns} = 286.32 \text{ ps}$  (B goes high to low)
- (c)  $t_{phl} = 50.64971\text{ns} - 50.5\text{ns} = 149.71 \text{ ps}$  (A goes low to high)
- (d)  $t_{phl} = 150.7883\text{ns} - 150.5\text{ns} = 288.3 \text{ ps}$  (A goes high to low)
- (e)  $t_{plh} = 50.759376\text{ns} - 50.5\text{ns} = 259.376 \text{ ps}$  (B goes low to high)
- (f)  $t_{plh} = 150.83627\text{ns} - 150.5\text{ns} = 336.27 \text{ ps}$  (B goes high to low)
- (g)  $t_{plh} = 150.7497\text{ns} - 150.5\text{ns} = 249.7 \text{ ps}$  (A goes low to high)
- (h)  $t_{plh} = 50.86738\text{ns} - 50.5\text{ns} = 367.38 \text{ ps}$  (A goes high to low)

## 5.4 Power Consumption

## 1. Power consumption of XOR transistor implementation:

- (a) average power = 39.29 micro W
- (b) static power = 1.042 nano W
- (c) dynamic power =  $39.29 - 0.001042 \text{ micro W} = 39.2889 \text{ micro W}$

## 2. Power consumption of XOR using my own NAND cells:

- (a) average power = 46.85 micro W
- (b) static power = 20 nano W
- (c) dynamic power =  $46.85 - 0.02 \text{ micro W} = 46.83 \text{ micro W}$

## 3. Power consumption of XOR using TSMC180 cells:

- (a) average power = 39.18 micro W
- (b) static power = 1.159 nano W
- (c) dynamic power =  $39.18 - 0.001159 \text{ micro W} = 39.1788 \text{ micro W}$

The average is calculated by finding the area under the current curve between the times 10% and 90% of the output signal transition and multiplied by VDD and divided by the time between the two points. Static power is calculated when there are no transitions. Dynamic power is the difference between the average power and the static power.

## 5.5 Worst-case propagation delay

The worst-case propagation delay for my XOR transistor-level implementation is 367.38 ps for post-layout and 361.43 ps for pre-layout. The worst-case propagation delay happens during the transition  $(A=1, B=1)$  to  $(A=0, B=1)$ .

$(A=1, B=0, Y=1) \rightarrow (A=1, B=1, Y=0)$  (a)  
 $(A=1, B=1, Y=0) \rightarrow (A=0, B=1, Y=1)$  (h)  
 $(A=0, B=1, Y=1) \rightarrow (A=0, B=0, Y=0)$  (b)  
 $(A=0, B=0, Y=0) \rightarrow (A=1, B=0, Y=1)$  (g)  
 $(A=1, B=0, Y=1) \rightarrow (A=0, B=0, Y=0)$  (d)  
 $(A=0, B=0, Y=0) \rightarrow (A=0, B=1, Y=1)$  (e)  
 $(A=0, B=1, Y=1) \rightarrow (A=1, B=1, Y=0)$  (c)  
 $(A=1, B=1, Y=0) \rightarrow (A=1, B=0, Y=1)$  (f)

The above sequence covers all the possible transitions and the alphabet at the end of each transition line represents the propagation delay values shown in subsection 5.3.

## 5.6 Drive Strength

Drive strength of a standard cell is its capacity to drive other cells connected at its output. The more the drive strength, the delay to drive the output connected cell is short. Increase in width of the transistor increases the current which in turn drives the output connected cells faster. In this task 5 NAND standard cells were used, so it is possible to generate different propagation delays by changing the size of the transistors in our NAND standard cells.

# Chapter 6

## LAB 6

In this lab we construct a D-type flip flop using inverters,gated inverters and latches.

### 6.1 Gated Inverter

In this section, the schematic design, layout of Gated Inverter are presented.

#### 6.1.1 Schematic

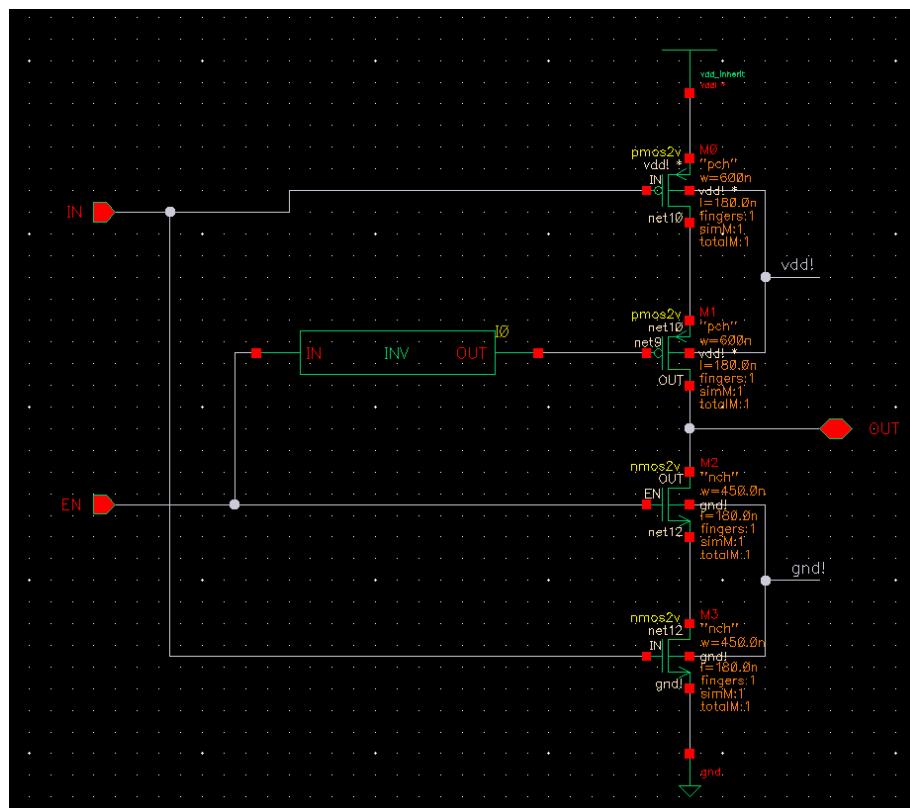


Figure 6.1: Schematic of Gated Inverter

Figure 6.1 shows the schematic of Gated Inverter and Figure 6.2 shows the layout of Gated Inverter.

### 6.1.2 Layout

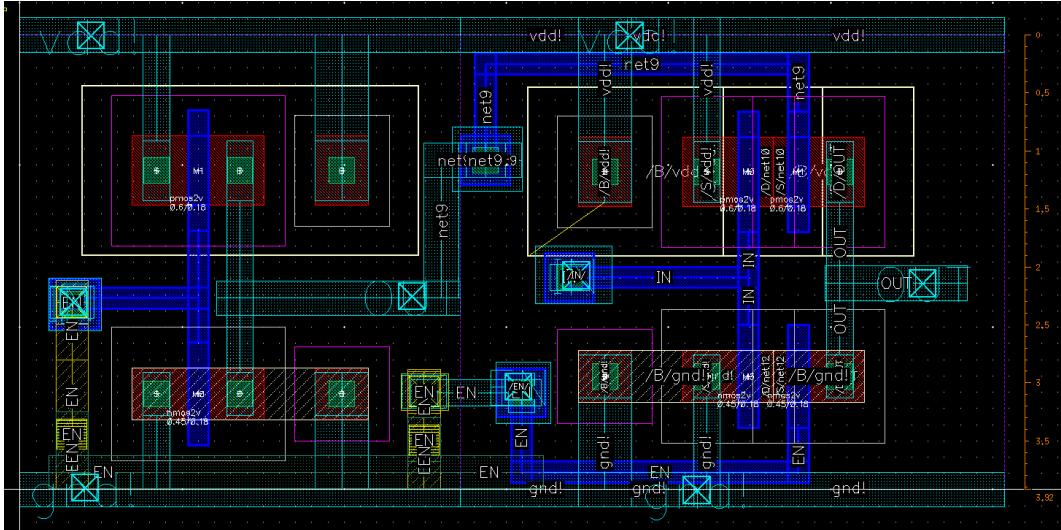


Figure 6.2: Layout of Gated Inverter

### 6.1.3 DRC

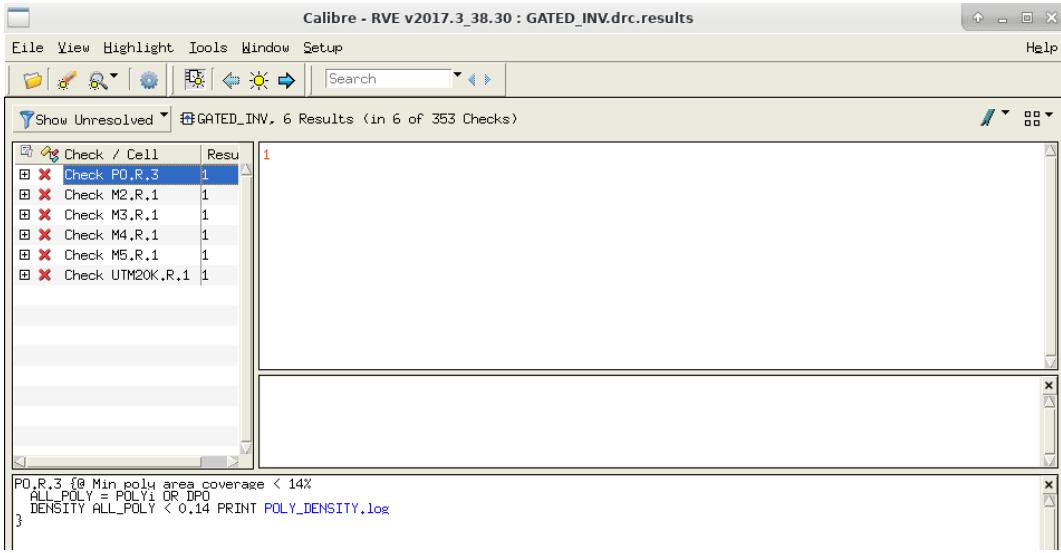


Figure 6.3: DRC of Gated Inverter

Figure 6.3 and 6.4 represents the drc and lvs results of the Gated Inverter layout.

### 6.1.4 LVS

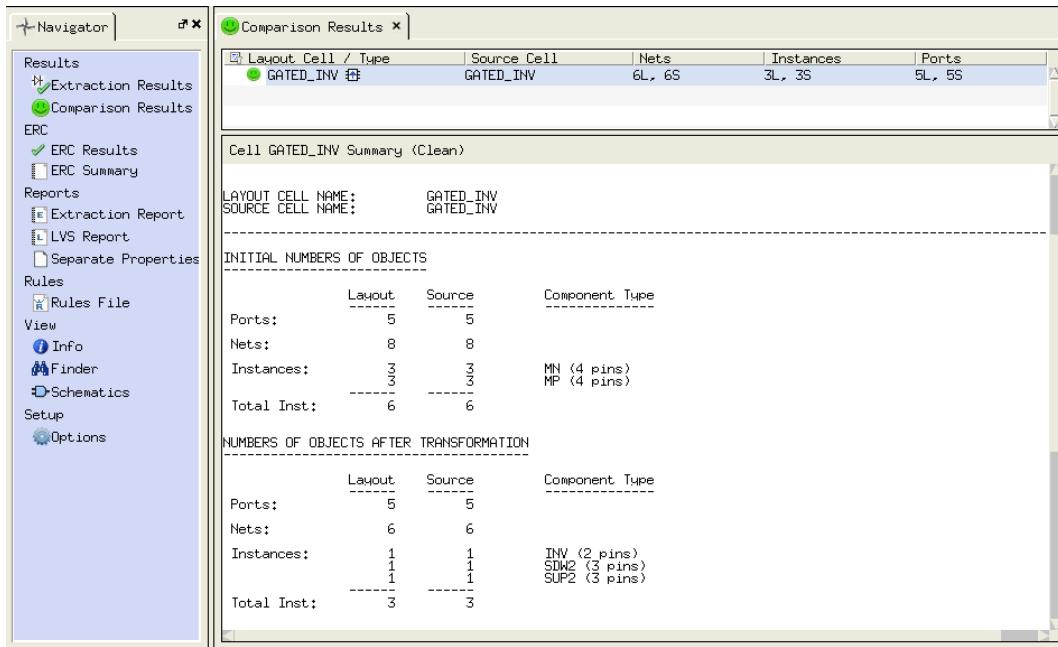


Figure 6.4: LVS of Gated Inverter

## 6.2 Latch

In this section, the schematic design, layout of Latch are presented. The latch is created using two inverters and two Gated inverters.

### 6.2.1 Schematic

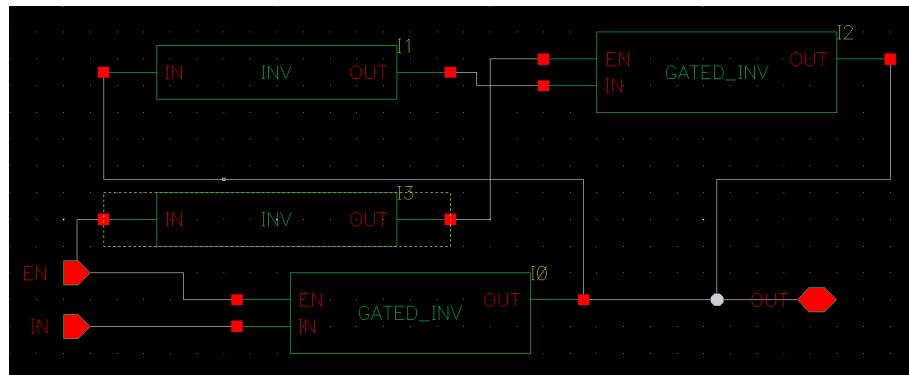


Figure 6.5: Schematic of Latch

Figure 6.5 shows the schematic of latch and Figure 6.6 shows the layout of latch.

### 6.2.2 Layout

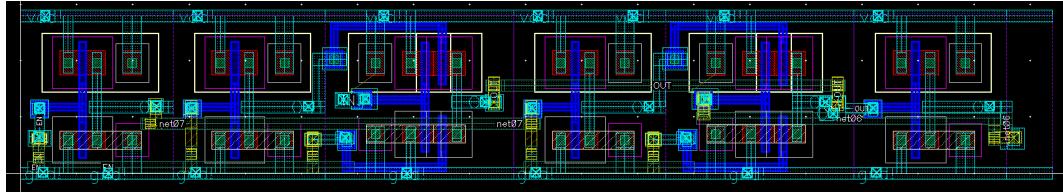


Figure 6.6: Layout of Latch

### 6.2.3 DRC

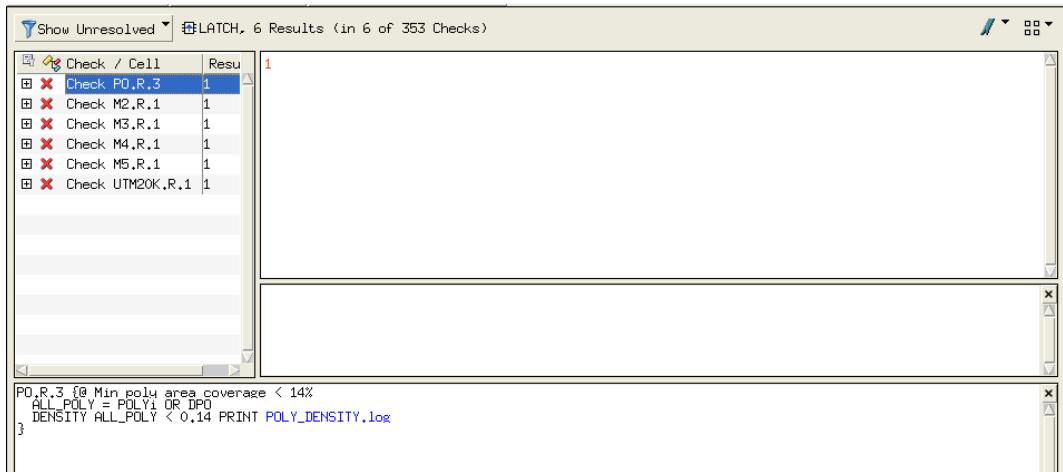


Figure 6.7: DRC of Latch

Figure 6.7 and 6.8 represents the drc and lvs results of the Latch layout.

### 6.2.4 LVS

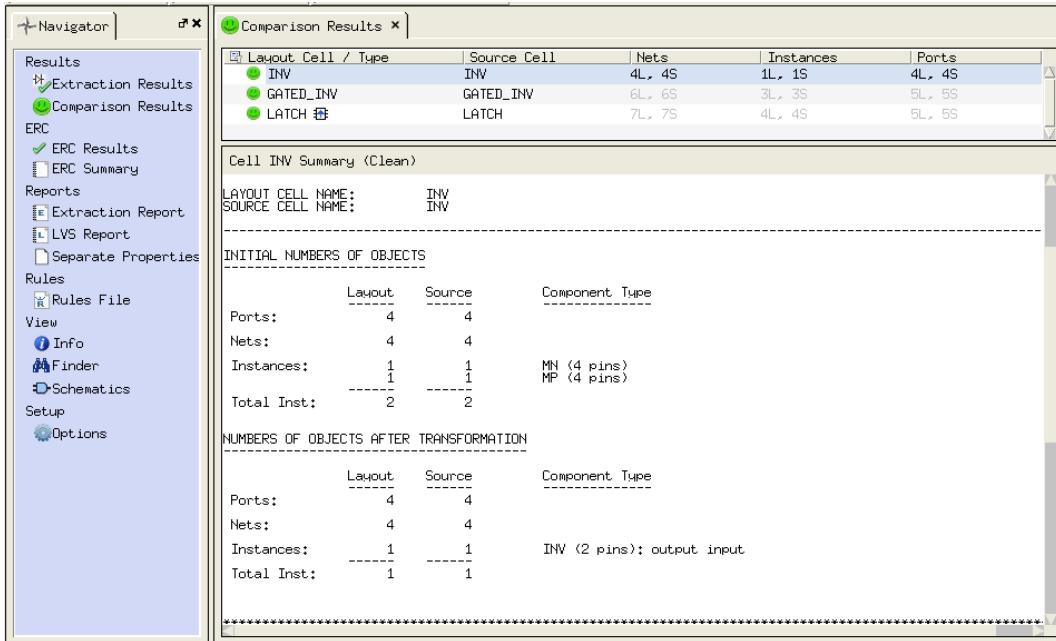


Figure 6.8: LVS of Latch

## 6.3 DFF

In this section, the schematic design, layout of DFF are presented. The DFF is created using two latches and one inverter.

### 6.3.1 Schematic

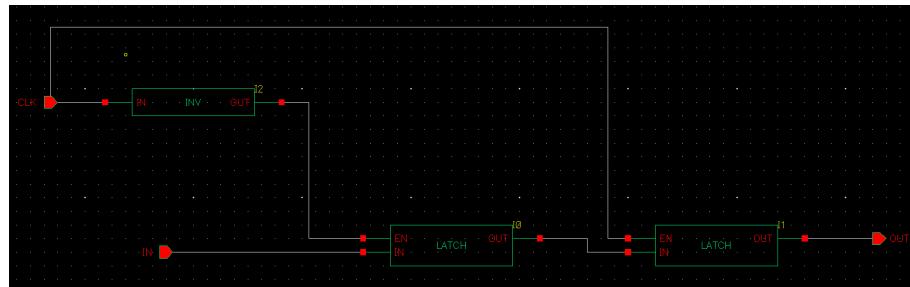


Figure 6.9: Schematic of DFF

Figure 6.9 shows the schematic of DFF and Figure 6.10 shows the layout of DFF.

### 6.3.2 Layout

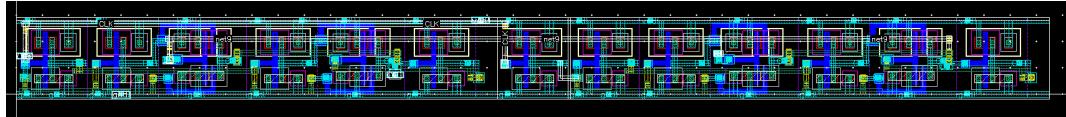


Figure 6.10: Layout of DFF

### 6.3.3 DRC



Figure 6.11: DRC of DFF

Figure 6.11 and 6.12 represents the drc and lvs results of the DFF layout.

### 6.3.4 LVS

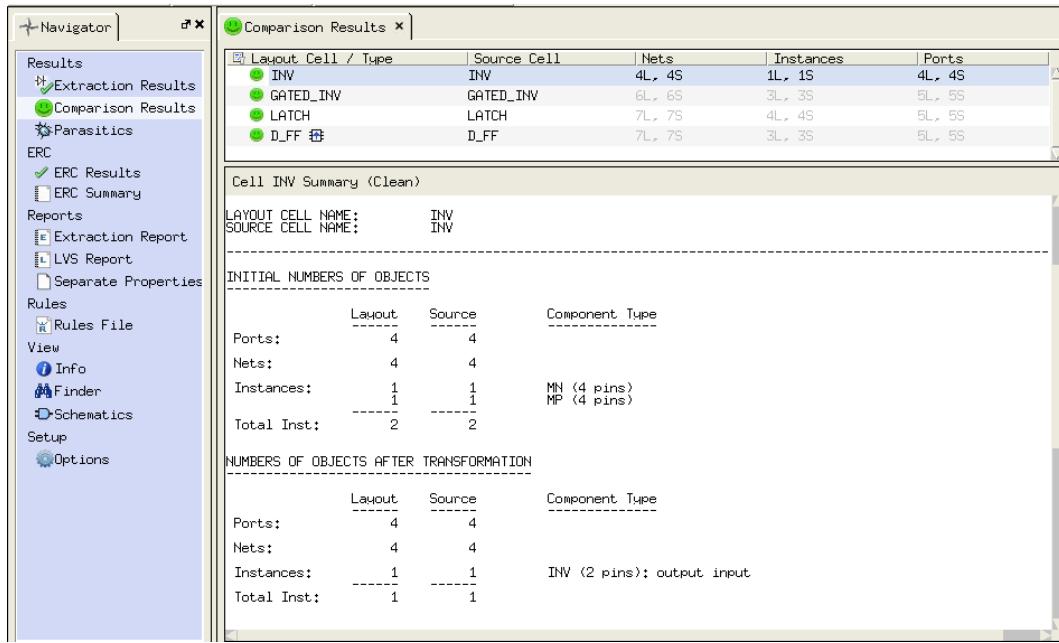


Figure 6.12: LVS of DFF

## 6.4 clock-to-q delay, setup and hold

The delay between the clock rise and the output is called clock-to-q delay.

Setup and hold times are defined relative to the clock rise. Setup time defines how long before the clock rise must the data arrive, hold time defines how long after the clock rise must the data not change. Setup and hold times are not delays but constraints. They are to be followed for proper storage of data in flip-flop.

## 6.5 Simulation results

The DFF doesn't work when clock edge and data arrives at the same time at DFF input as it violates the setup time constraint. Figure 6.13 shows the testbench for DFF.

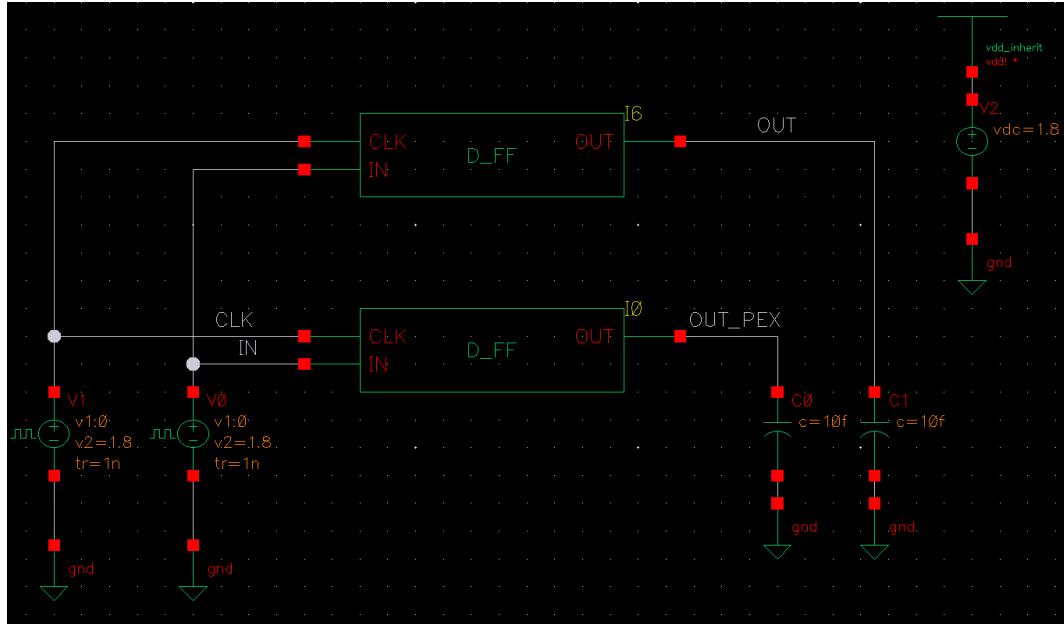


Figure 6.13: Testbench for DFF

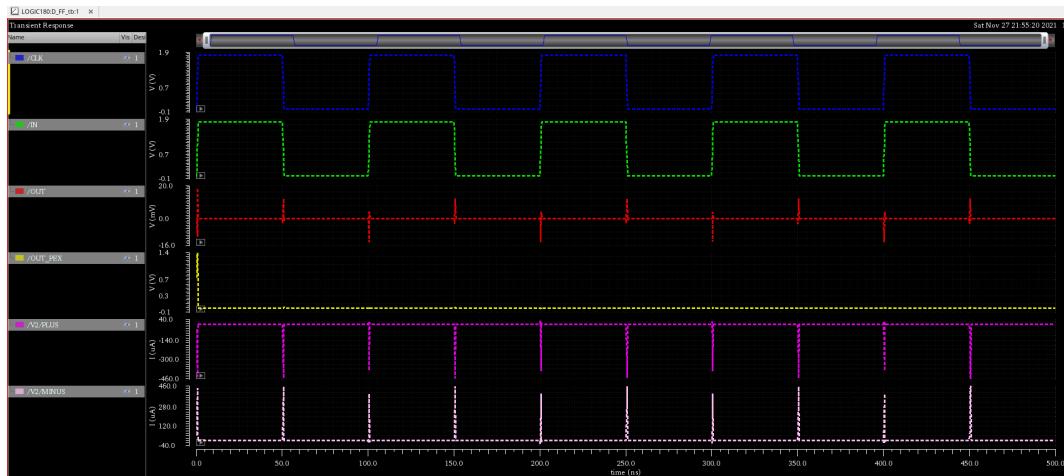


Figure 6.14: Simulation of DFF when data and clock edge arrives at same time

### 6.5.1 clk-to-Q delay

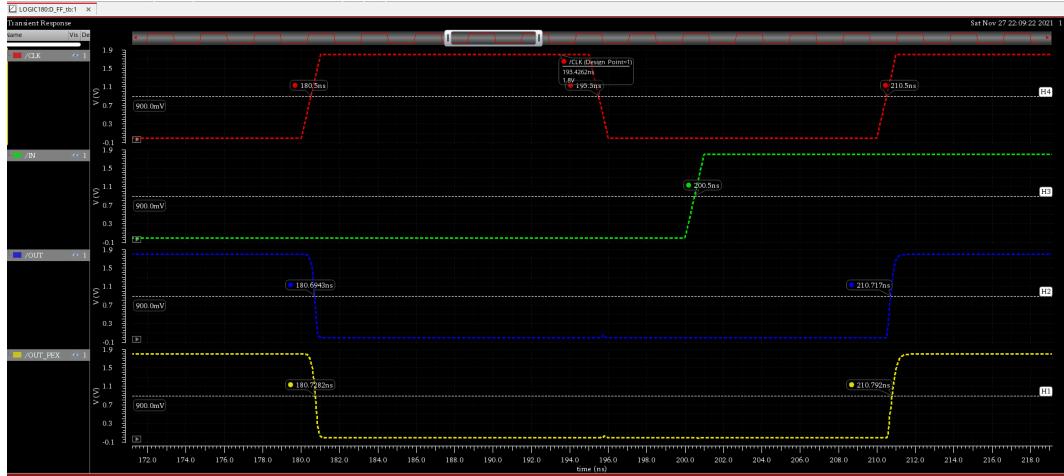


Figure 6.15: clk-to-Q delay

1. clk-to-q delay pre-layout:

$$(a) t_{cq} = 210.717\text{ns} - 210.5\text{ns} = 217 \text{ ps (low to high)}$$

2. clk-to-q delay post-layout:

$$(a) t_{cq} = 210.792\text{ns} - 210.5\text{ns} = 292 \text{ ps (low to high)}$$

The DFF works as data arrives before the clock edge and satisfies setup timing constraint.

# Chapter 7

## LAB 7

### 7.1 Setup time

We hold the clock stable and sweep the delay of the data to find a spot where the DFF gives wrong output. Figure 7.1 shows the setup time, any data not arriving before or at the setup time results in setup violation.

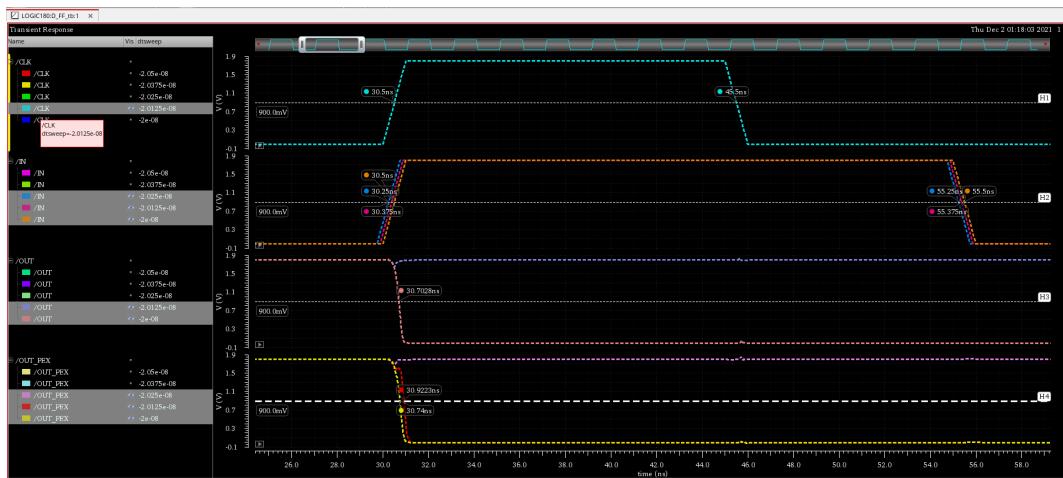


Figure 7.1: setup time simulation

$$\text{Setup time for pre-layout} = 30.5\text{ns} - 30.375\text{ns} = 125 \text{ ps.}$$

$$\text{Setup time for post-layout} = 30.5\text{ns} - 30.25\text{ns} = 250 \text{ ps.}$$

We hold the Data stable and sweep the delay of the clock to find a spot where the DFF gives wrong output. Figure 7.2 shows the hold time, any data not stable for hold time results in hold time violation.

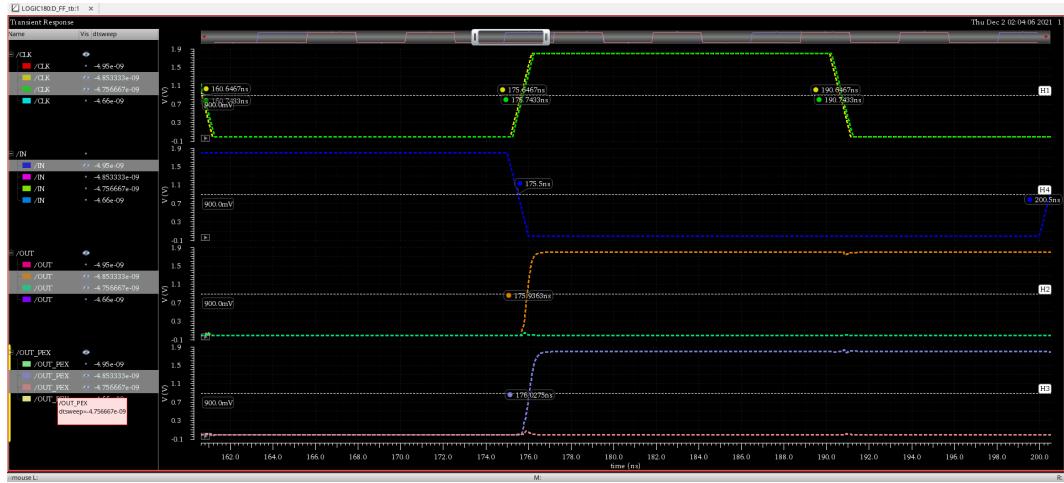


Figure 7.2: hold time simulation

Hold time for pre-layout = $175.5\text{ ns} - 175.6467\text{ ns} = -146.7\text{ ps}$ .

Hold time for post-layout =  $175.5\text{ ns} - 175.6467\text{ ns} = -146.7\text{ ps}$ .

# Chapter 8

## LAB 8

In this lab we perform corner analysis on NOR design and present the propagation delays between input and one output. Later we perform Monte Carlo Simulation on the same NOR design and plot a histogram of propagation delays between input and one output.

### 8.1 Corner Analysis

In this section the simulation results of both pre and post-layout measuring propagation delays between input and one output of NOR design are shown.

#### 8.1.1 Pre and Post-layout simulation of NOR

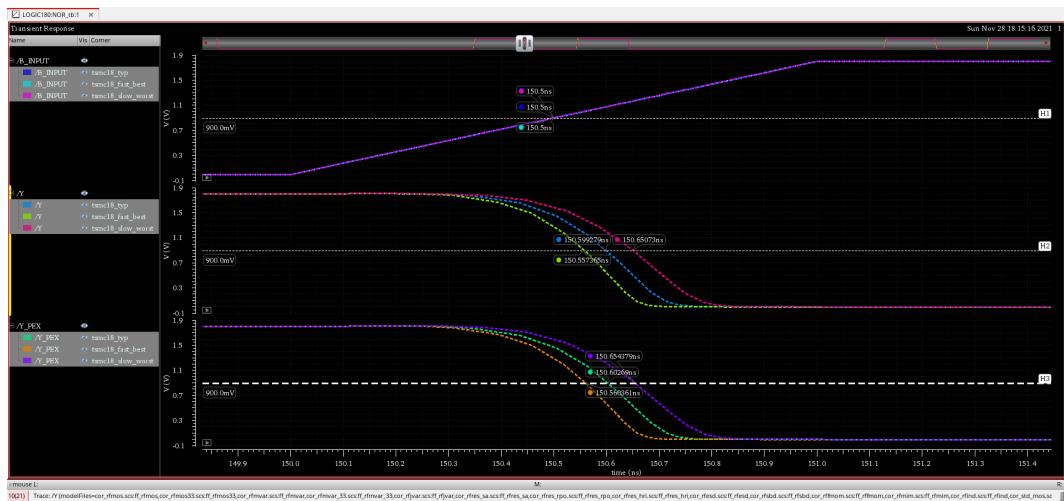


Figure 8.1: Output high to low transition

The propagation delays between B\_INPUT and OUTPUT high to low transition of both pre and post-layout are shown below.

#### Pre-layout:

- 1) typical  $t_{phl} = 150.599279 - 150.5 = 99.279 \text{ ps}$ .
- 2) fast  $t_{phl} = 150.557365 - 150.5 = 57.365 \text{ ps}$ .
- 3) slow  $t_{phl} = 150.650736 - 150.5 = 150.73 \text{ ps}$ .

**Post-layout:**

- 1) typical  $t_{phl} = 150.60269 - 150.5 = 102.69$  ps.
- 2) fast  $t_{phl} = 150.560361 - 150.5 = 60.361$  ps.
- 3) slow  $t_{phl} = 150.654379 - 150.5 = 154.379$  ps.

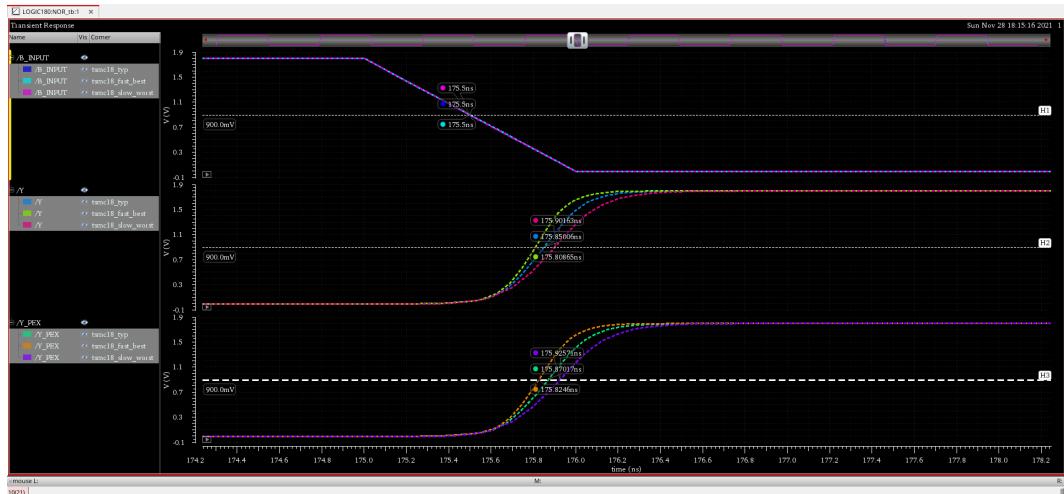


Figure 8.2: Output low to high transition

The propagation delays between B\_INPUT and OUTPUT low to high trasition of both pre and post-layout are shown below.

**Pre-layout:**

- 1) typical  $t_{plh} = 175.85006 - 175.5 = 350.06$  ps.
- 2) fast  $t_{plh} = 175.80865 - 175.5 = 308.65$  ps.
- 3) slow  $t_{plh} = 175.90163 - 175.5 = 401.63$  ps.

**Post-layout:**

- 1) typical  $t_{plh} = 175.87017 - 175.5 = 370.17$  ps.
- 2) fast  $t_{plh} = 175.8246 - 175.5 = 324.6$  ps.
- 3) slow  $t_{plh} = 175.92571 - 175.5 = 425.71$  ps.

## 8.2 Fast, slow, typical corners

Corners are used to study how the circuits works under the effects of process and environmental variations like temperature. Fast corners work at high frequency and slow corners work at low frequency.

## 8.3 Monte Carlo Simulation

### 8.3.1 Pre-layout

Fig 8.3 and 8.4 shows the histogram of propagation delays after running Monte Carlo Simulation of Pre-layout.

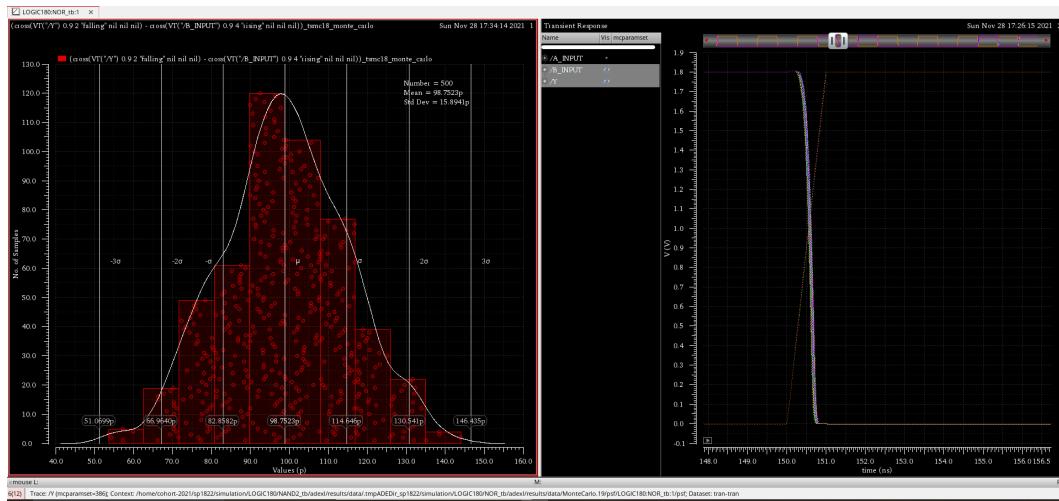


Figure 8.3: Histogram of propagation delay Pre-layout -1

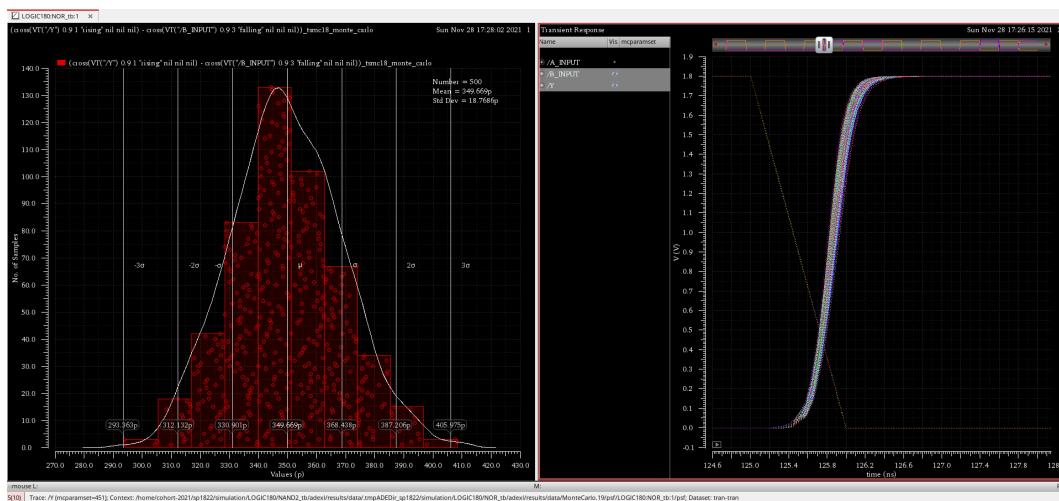


Figure 8.4: Histogram of propagation delay Pre-layout -2

### 8.3.2 Post-layout

Fig 8.5 and 8.6 shows the histogram of propagation delays after running Monte Carlo Simulation of Post-layout.

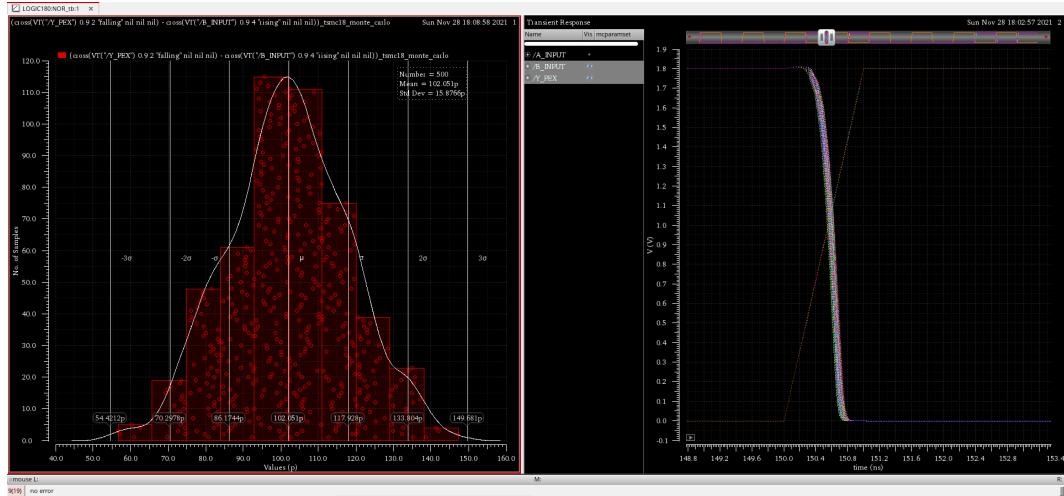


Figure 8.5: Histogram of propagation delay Post-layout -1

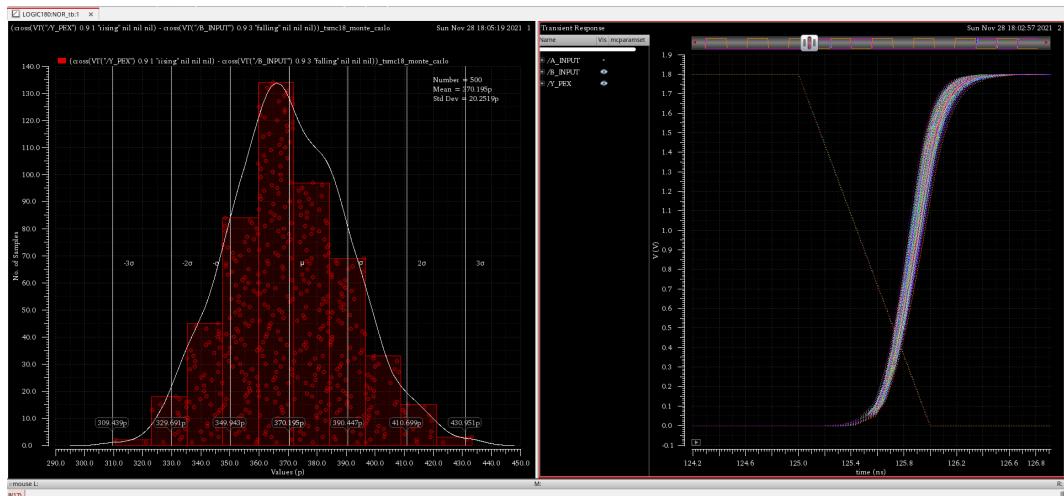


Figure 8.6: Histogram of propagation delay Post-layout -2

### 8.3.3 Monte Carlo Simulation

Monte carlo Simulation is used to find the effects of random variations of CMOS process parameters on a circuit and to effectively predict the performance of a circuit in various working conditions.