

# NGSPICE Simulation of CMOS Circuits

Wuqiong Zhao\*

June 6, 2023

NGSPICE is a powerful open-source SPICE simulation software in command line, which can efficiently simulate CMOS circuits. Basic logic gates, including NOT, NAND, AND, NOR, are implemented and analyzed. The delay parameters and response plots can help understand the circuit characteristics. As examples, the 8-input NAND gate with three distinct designs is investigated, and the clock controlled SR latch is also simulated to design appropriate MOS parameters.

## Contents

<b>1. Environments</b>	<b>2</b>
1.1. Preparation	2
1.2. Settings	2
1.3. Development	2
<b>2. Definitions</b>	<b>2</b>
<b>3. Sources</b>	<b>3</b>
3.1. FreePDK45	3
3.2. Inverter	3
3.3. NAND2	4
3.4. AND2	4
3.5. NOR2	5
3.6. AND8	6
3.6.1. Basic Components	6
3.6.2. AND8A (Symmetrical Design)	8
3.6.3. AND8B (NAND4A×2 + NOR2×1)	9
3.6.4. AND8C (AND4B×2 + AND2×1)	10
3.6.5. Analysis	11
3.7. Clock Controlled SR Latch	11
3.7.1. Schematic Design	11
3.7.2. MOS W/L Design	12
3.7.3. Simulation	12
<b>4. License</b>	<b>12</b>

---

\* This is the course project of FUNDAMENTALS OF VLSI DESIGN, Southeast University, 2023 Spring. This document is available online at [https://spice.tvj.one/report/NGSPICE\\_CMOS\\_Report.pdf](https://spice.tvj.one/report/NGSPICE_CMOS_Report.pdf).

The author is with the National Mobile Communications Research Laboratory, Southeast University, Nanjing 211189, China. (E-Mail: [me@wqzhao.org](mailto:me@wqzhao.org))

<b>References</b>	<b>12</b>
<b>A. Resources</b>	<b>13</b>
A.1. Links	13
A.2. Schematic Graph	13
<b>B. Source Code</b>	<b>13</b>
B.1. Inverter	13
B.2. NAND2	15
B.3. AND2	16
B.4. NOR2	18
B.5. NAND4	20
B.6. AND4	21
B.7. AND8	21
B.7.1. Test Circuit	21
B.7.2. AND8A	22
B.7.3. AND8B	24
B.7.4. AND8C	27
B.8. Clock Controlled SR Latch	30
B.9. MISC	32

## 1. Environments

### 1.1. Preparation

Install NGSPICE [1] CLI app. View the user manual [2] for more information.

### 1.2. Settings

NGSPICE is set to be compatible with HSPICE (see `.spiceinit`).

### 1.3. Development

My development environments:

- macOS 13 (Ventura) with M1 chip;
- NGSPICE 40 (Homebrew version).

There is *no* guarantee that the provided code can run on other platforms or other SPICE tools. Make changes if appropriate.

## 2. Definitions

Delay:

- $t_r$  ( $\tau_r$ ): rise time (from output crossing  $0.1V_{DD}$  to  $0.9V_{DD}$ );
- $t_f$  ( $\tau_f$ ): fall time (from output crossing  $0.9V_{DD}$  to  $0.1V_{DD}$ );
- $t_{pdr}$  ( $\tau_{pdr}$ ): rising propagation delay (from input to rising output crossing  $V_{DD}/2$ );
- $t_{pdf}$  ( $\tau_{pdf}$ ): falling propagation delay (from input to falling output crossing  $V_{DD}/2$ );
- $t_{pd}$  ( $\tau_{pd}$ ): average propagation delay ( $t_{pd} = (t_{pdr} + t_{pdf})/2$ ).

Operating corner:

- **SS**: slow-slow;
- **MOM**: nominal (average);
- **FF**: fast-fast.

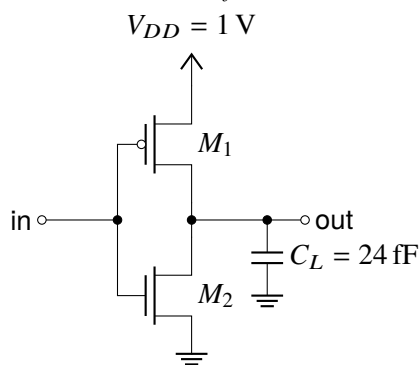
### 3. Sources

#### 3.1. FreePDK45

This is a 45 nm CMOS library. See [README](#) for more information. TNOM of FreePDK45 is 27C.

#### 3.2. Inverter

The schematic diagram of the CMOS inverter is shown in Figure 1, with 1 PMOS and 1 NMOS. The design requirement is  $t_r = t_f$  when  $C_L = 24$  fF. The designed MOS parameters are shown in Table 1.



MOS	W	L
PMOS	360 nm	45 nm
NMOS	225 nm	45 nm

Figure 1: CMOS inverter schematic.

Table 1: Designed inverter MOS parameters.

#### Inverter Subcircuit

```

1 .subckt INV gnd i o vdd
2 * src gate drain body type
3 M1 vdd i o vdd PMOS_VTL W=360nm L=45nm
4 M2 gnd i o gnd NMOS_VTL W=225nm L=45nm
5 .ends INV

```

Simulate with ngspice `inv.cir`, and the response of the inverter can be obtained as depicted in Figure 2. The delay parameters are  $t_r = 119$  ps,  $t_f = 120$  ps,  $t_{pdr} = 60$  ps,  $t_{pdf} = 64$  ps and  $t_{pd} = 62$  ps.

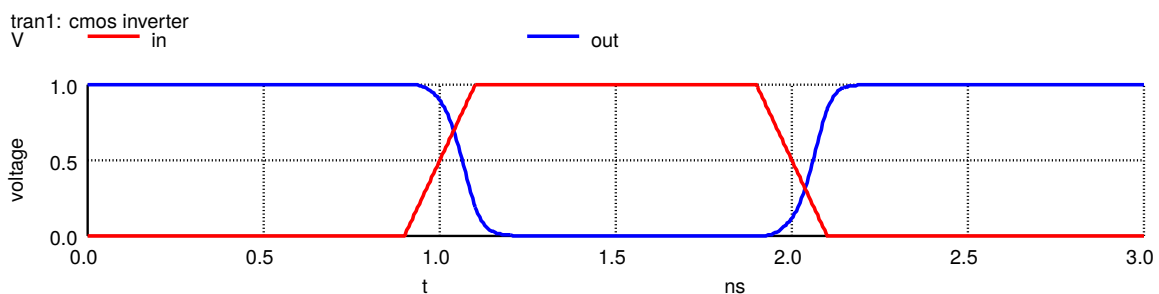
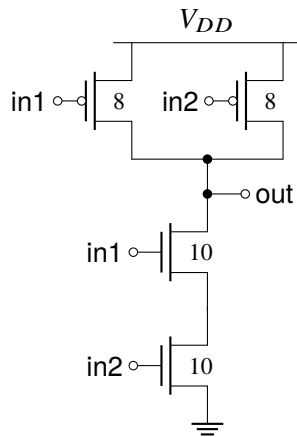


Figure 2: Response of the inverter.

### 3.3. NAND2

The CMOS NAND2 gate is symmetrically designed with parameters for the worst case. The schematic is shown in Figure 3, where the value of  $W/L$  is labeled next to each MOS. The load capacitor is omitted. The designed parameters are shown in Table 2.



MOS	Number	$W$	$L$
PMOS	2	360 nm	45 nm
NMOS	2	225 nm	45 nm

Figure 3: CMOS NAND2 schematic.

Table 2: Designed NAND2 MOS parameters.

#### NAND2 Subcircuit

```

1 .subckt NAND2 gnd i1 i2 o vdd
2 *   src  gate  drain  body  type
3 Mp1 vdd  i1   o      vdd   PMOS_VTL W=360nm L=45nm
4 Mp2 vdd  i2   o      vdd   PMOS_VTL W=360nm L=45nm
5 Mn1 gnd  i1   o      gnd   NMOS_VTL W=450nm L=45nm
6 Mn2 gnd  i2   t1     gnd   NMOS_VTL W=450nm L=45nm
7 .ends NAND2

```

The response simulated with ngspice `nand2.cir` is shown in Figure 4.

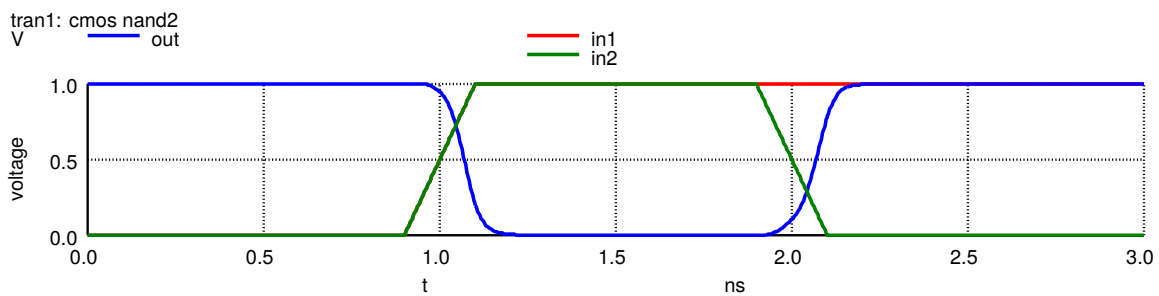


Figure 4: Response of the NAND2 gate.

### 3.4. AND2

AND2 is `NAND2 + INV`. The response of AND2 at the worst case is shown in Figure 5.

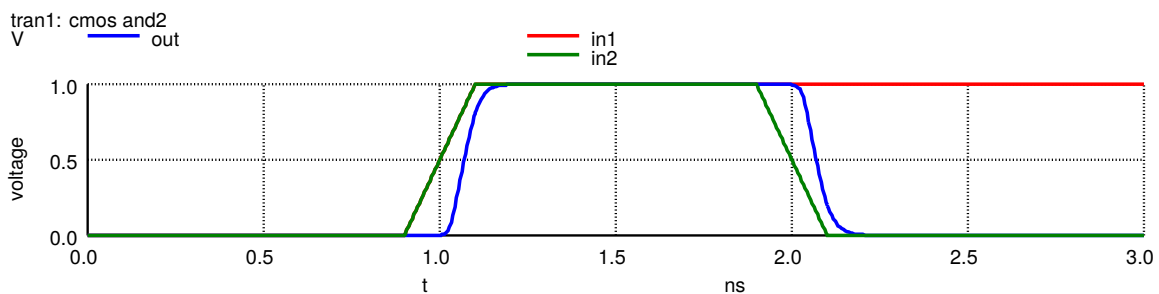


Figure 5: Response of the AND2 gate.

### 3.5. NOR2

The schematic of the NOR2 gate is shown in Figure 6.

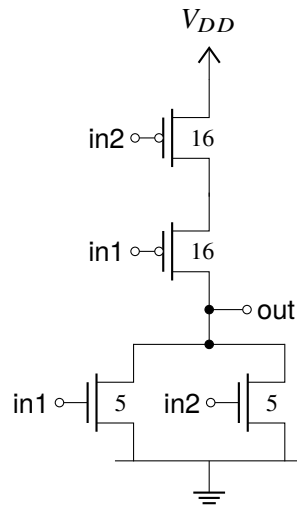


Figure 6: CMOS NOR2 schematic.

#### NOR2 Subcircuit

```

1 .subckt NOR2 gnd i1 i2 o vdd
2 *   src  gate drain body type
3 Mp1 t1  i1  o    vdd PMOS_VTL W=720nm L=45nm
4 Mp2 vdd i2  t1   vdd PMOS_VTL W=720nm L=45nm
5 Mn1 gnd i1  o    gnd NMOS_VTL W=225nm L=45nm
6 Mn2 gnd i2  o    gnd NMOS_VTL W=225nm L=45nm
7 .ends NOR2

```

The response of NOR2 gate is given in Figure 7 when simulating with ngspice nor2.cir.

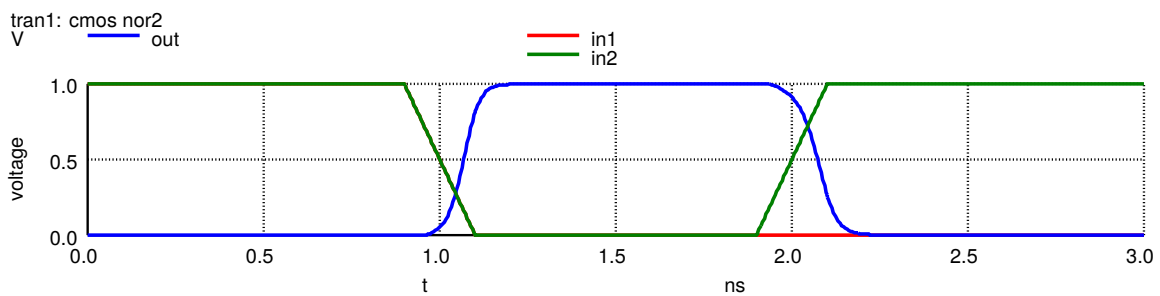


Figure 7: Response of the AND2 gate.

### 3.6. AND8

8-input AND gate. With a large fan-in, there can be several distinct designs. Here we want to investigate the performance of different designs.

The test circuit (defined in `add8_test_inv2.inc`) involves a 24 fF capacitor load at the output, and 8 sets of two stages of inverters for each input. For the inverter in the test circuit, NMOS has  $W = 0.75 \mu\text{m}$ ,  $L = 0.25 \mu\text{m}$ , and PMOS has  $W = 2.60 \mu\text{m}$ ,  $L = 0.25 \mu\text{m}$ .

The response simulation has the PVT condition of 1.0 V, FF, 25°C.

#### 3.6.1. Basic Components

**NAND4A** This design directly extends the structure of NAND2 into NAND4. Its schematic is shown in Figure 8.

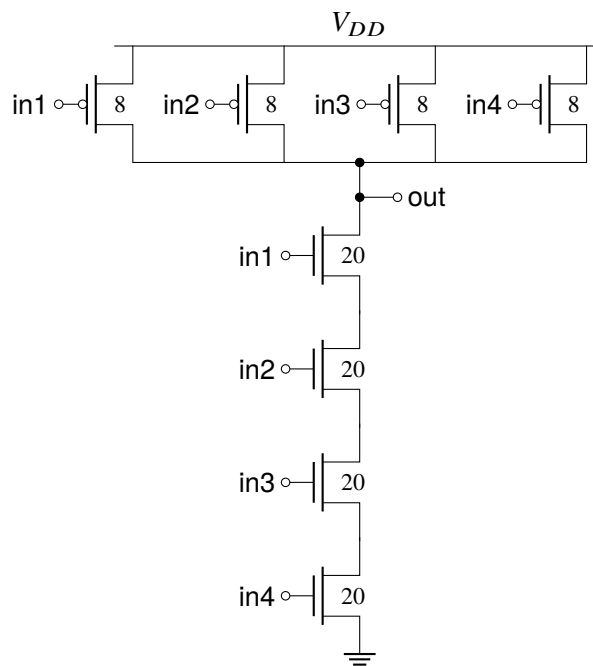


Figure 8: CMOS NAND4A schematic.

**NAND8A** This design directly extends the structure of NAND2 into NAND8. Its schematic is shown in Figure 9.

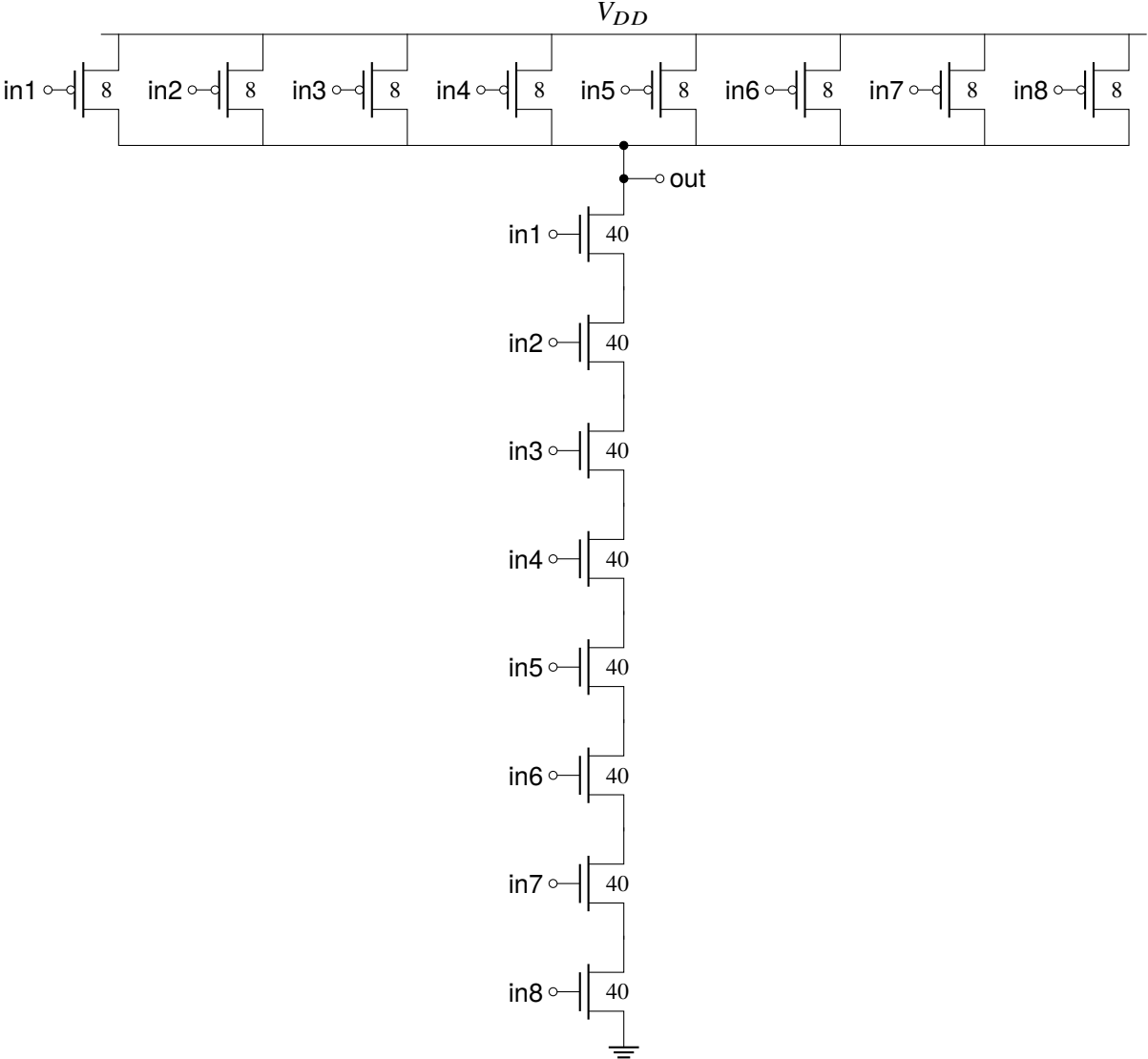


Figure 9: CMOS NAND8A schematic.

### 3.6.2. AND8A (Symmetrical Design)

This is the most basic case, extending 2-input NAND to 8-input NAND, before applying an inverter. The schematic of AND8A is shown in Figure 10. MOS parameters are specified in [NAND8A](#).

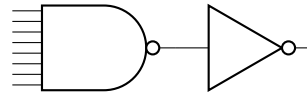


Figure 10: CMOS AND8A schematic.

The simulation result with ngspice `and8a.cir` is shown in Table 3 and Figure 11.

Table 3: AND8A gate simulation result.

PVT Condition	$t_r$ (ps)	$t_f$ (ps)	$t_{pdr}$ (ps)	$t_{pdf}$ (ps)	P static ( $\mu$ W)	P dynamic ( $\mu$ W)
0.9 V, SS, 70°C	132.6	136.9	137.3	159.9	0.076	2.366
1.35 V, SS, 70°C	110.6	124.4	102.4	125.9	1.023	5.591
1.0 V, NOM, 25°C	90.5	99.2	83.8	110.5	0.227	2.733
1.5 V, NOM, 25°C	79.8	95.0	69.4	92.4	6.566	9.669
1.1 V, FF, 0°C	72.5	84.6	62.7	89.4	0.955	5.321
1.65 V, FF, 0°C	69.1	83.6	54.2	75.3	51.582	1.121

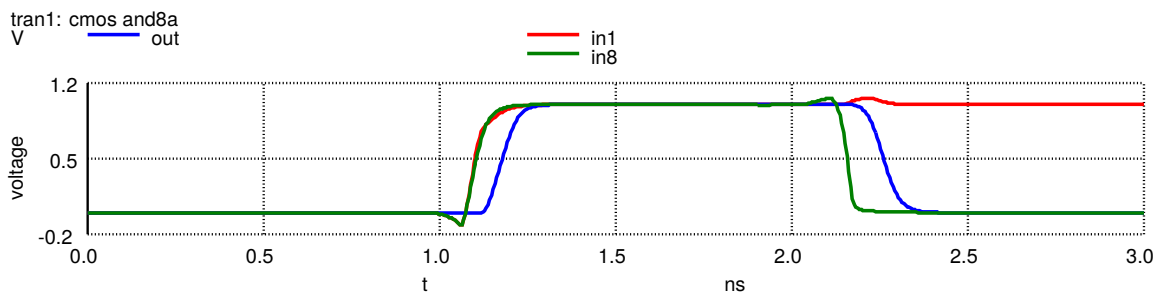


Figure 11: Response of the AND8A gate.



### 3.6.3. AND8B (NAND4A×2 + NOR2×1)

The schematic design using 2 NAND4A gates and 1 NOR gate is shown in Figure 12.

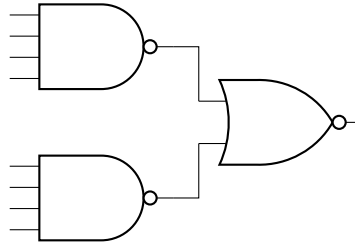


Figure 12: CMOS AND8B schematic.

The simulation result with ngspice and8b.cir is shown in Table 4 and Figure 13.

Table 4: AND8B gate simulation result.

PVT Condition	$t_r$ (ps)	$t_f$ (ps)	$t_{pdr}$ (ps)	$t_{pdf}$ (ps)	P static ( $\mu$ W)	P dynamic ( $\mu$ W)
0.9 V, SS, 70°C	118.8	131.7	102.8	106.2	0.030	2.233
1.35 V, SS, 70°C	96.2	113.1	78.3	83.3	0.641	6.204
1.0 V, NOM, 25°C	77.2	94.9	62.7	73.0	0.125	2.592
1.5 V, NOM, 25°C	65.7	85.5	52.0	61.0	4.895	12.301
1.1 V, FF, 0°C	59.4	79.9	46.2	58.8	0.510	5.289
1.65 V, FF, 0°C	53.3	74.3	30.5	50.3	43.767	11.248

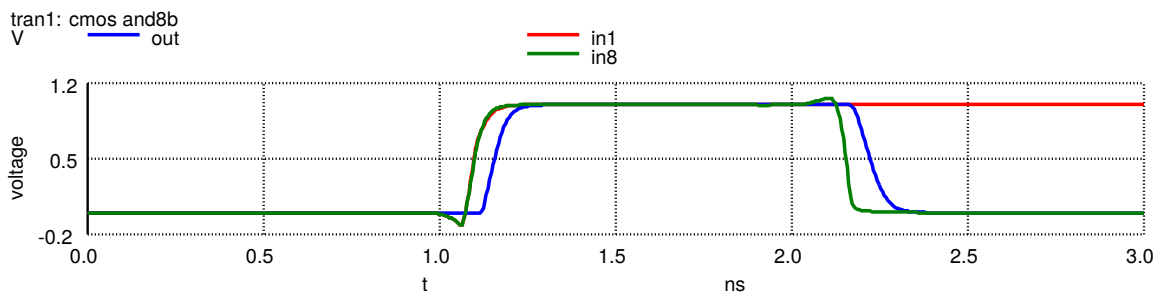


Figure 13: Response of the AND8B gate.

### 3.6.4. AND8C (AND4B $\times$ 2 + AND2 $\times$ 1)

An AND4B gate is composed of 2 NAND gates and 1 NOR gate. The schematic design of AND8C using 2 NAND4A gates and 1 NOR gate is shown in Figure 14.

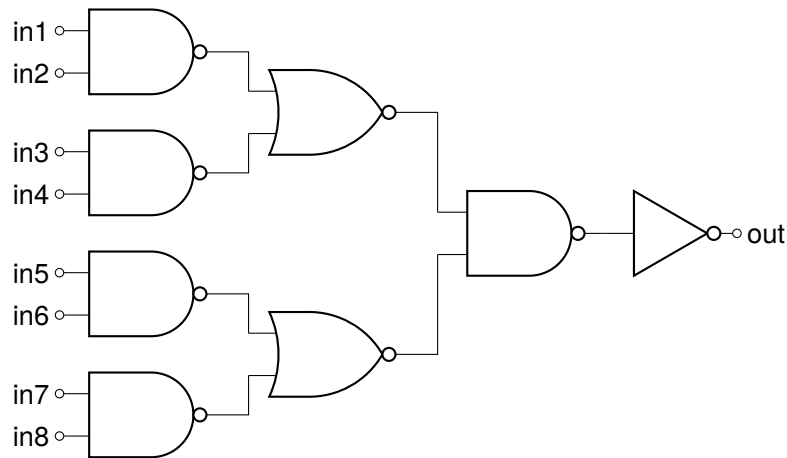


Figure 14: CMOS AND8C schematic.

The simulation result with ngspice `and8c.cir` is shown in Table 5 and Figure 15.

Table 5: AND8C gate simulation result.

PVT Condition	$t_r$ (ps)	$t_f$ (ps)	$t_{pdr}$ (ps)	$t_{pdf}$ (ps)	P static ( $\mu$ W)	P dynamic ( $\mu$ W)
0.9 V, SS, 70°C	118.8	120.3	103.3	105.0	0.111	3.067
1.35 V, SS, 70°C	96.7	102.6	79.8	83.5	1.253	9.458
1.0 V, NOM, 25°C	81.5	86.9	68.4	73.3	0.301	4.164
1.5 V, NOM, 25°C	69.1	77.8	56.6	62.4	8.471	16.348
1.1 V, FF, 0°C	65.0	73.0	53.6	60.0	1.175	6.652
1.65 V, FF, 0°C	47.9	67.8	46.9	52.2	73.357	20.798

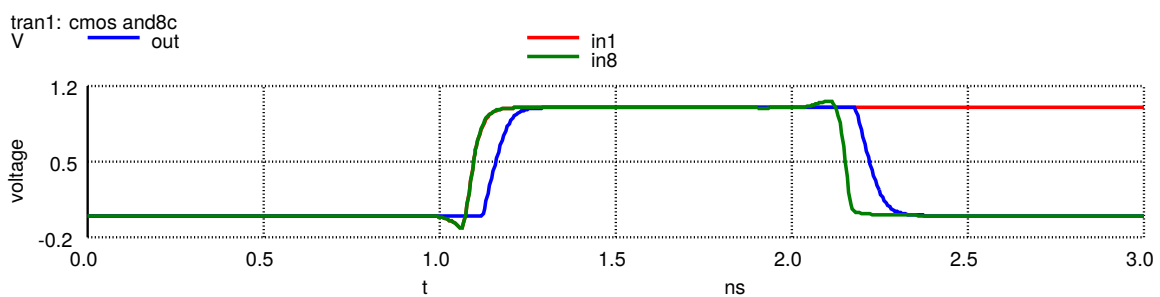


Figure 15: Response of the AND8C gate.

### 3.6.5. Analysis

Among the three designs ([AND8A](#), [AND8B](#) and [AND8C](#)), [AND8B](#) has the smallest latency, closely followed by [AND8C](#), and the largest latency is observed with [AND8A](#). Compared with [AND8A](#), the fan-in of [AND8B](#) is significantly reduced, resulting in lower latency. Though [AND8C](#) has an even smaller fan-in, the number of stages in the circuit is larger than that of [AND8B](#). Thus, [AND8B](#) achieves a reasonable tradeoff, and has the lowest latency.

There are also some other observations:

- A larger VDD can reduce the latency to some extent, but resulting in much larger power consumption.
- A faster operating corner (FF > NOM > SS) will help cut down latency, but also increases power.
- A higher temperature increases power in return for a reduced latency.

## 3.7. Clock Controlled SR Latch

### 3.7.1. Schematic Design

The schematic design of the clock controlled SR latch is shown in Figure 16, with 2 PMOS and 6 NMOS.

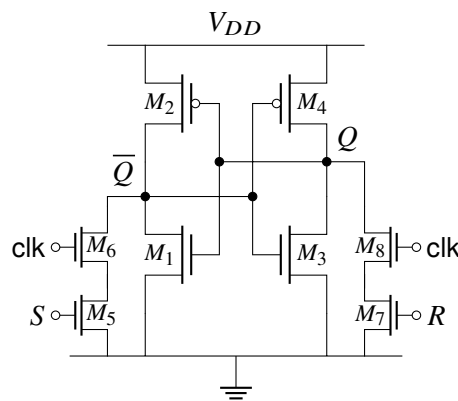


Figure 16: CMOS clock controlled SR latch schematic.

#### Clock Controlled SR Latch Subcircuit

```

1  * .param WL = 5
2  .subckt SR_LATCH_CLK gnd s r clk q qn vdd
3  *   src  gate  drain  body  type
4  M1 qn  q     gnd    gnd   NMOS_VTL W=    90nm L=45nm
5  M2 qn  q     vdd    vdd   PMOS_VTL W=   270nm L=45nm
6  M3 q   qn    gnd    gnd   NMOS_VTL W=    90nm L=45nm
7  M4 q   qn    vdd    vdd   PMOS_VTL W=   270nm L=45nm
8  M5 ts  s     gnd    gnd   NMOS_VTL W={WL*45nm} L=45nm
9  M6 qn  clk   ts     gnd   NMOS_VTL W={WL*45nm} L=45nm
10 M7 tr  r     gnd    gnd   NMOS_VTL W={WL*45nm} L=45nm
11 M8 q   clk   tr     gnd   NMOS_VTL W={WL*45nm} L=45nm
12 .ends SR_LATCH_CLK

```

You need to specify the parameter WL, for example `.param WL = 5`.

### 3.7.2. MOS W/L Design

We need to determine the appropriate  $W/L$  for  $M_5$  to  $M_8$ . Using a sweep, implemented by `alterparam` within a `foreach` loop, we can obtain Figure 17.

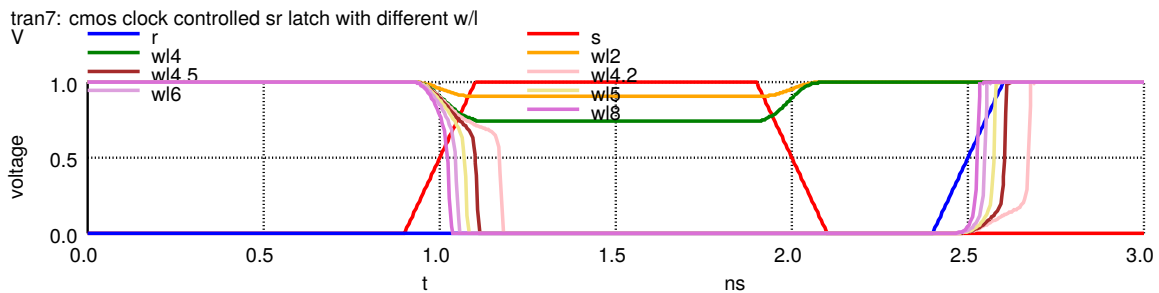


Figure 17: Clock controlled SR latch with different  $W/L$  values.

Clearly, we need  $W/L > 4.5$  (at least 4.2) for the latch to work properly. ( $M_1/M_3$  and  $M_2/M_4$  have  $W/L$  as 2 and 6, respectively.)

### 3.7.3. Simulation

The response of NOR2 gate is given in Figure 18 when simulating with `ngspice SR_latch_clk.cir`.

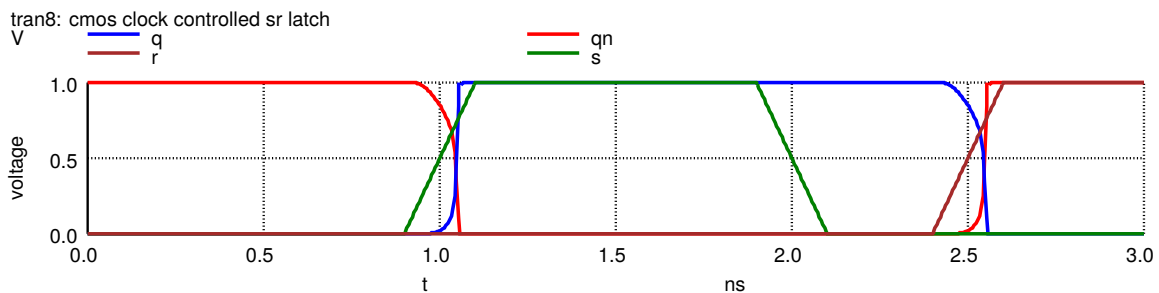


Figure 18: Response of the clock controlled SR latch when  $W/L = 6$ .

## 4. License

Copyright © 2023 Wuqiong Zhao ([me@wqzhao.org](mailto:me@wqzhao.org)). This project is distributed by an [MIT License](https://opensource.org/licenses/MIT).

## References

- [1] H. Vogt, A. Giles, P. Nenzi, and D. Warning, *NGSPICE 40 — open source spice simulator*, 2023. [Online]. Available: <https://ngspice.sourceforge.io/>.
- [2] H. Vogt, A. Giles, P. Nenzi, and D. Warning, *Ngspice user's manual version 40 (ngspice release version)*, Apr. 2023. [Online]. Available: <https://ngspice.sourceforge.io/docs/ngspice-40-manual.pdf>.

## A. Resources

### A.1. Links

- GitHub: [Teddy-van-Jerry/ngspice-cmos](#)
- Website: [spice.tvj.one](#)

### A.2. Schematic Graph

All schematic graphs in this document is drawn with TikZ using L<sup>A</sup>T<sub>E</sub>X, you can find the source code in the GitHub repository.

## B. Source Code

### B.1. Inverter

Listing 5: inv.inc

```

1 * =====
2 * Circuit      : CMOS Inverter
3 * Description: 1 PMOS + 1 NMOS
4 *
5 * Author       : Wuqiong Zhao (me@wqzhao.org)
6 * Date        : 2023-06-02
7 * License     : MIT
8 * =====
9
10 .subckt INV gnd i o vdd
11 * src gate drain body type
12 M1 vdd i o vdd PMOS_VTL W=360nm L=45nm
13 M2 gnd i o gnd NMOS_VTL W=225nm L=45nm
14 .ends INV
15
16 .subckt NOT gnd i o vdd
17 * src gate drain body type
18 M1 vdd i o vdd PMOS_VTL W=360nm L=45nm
19 M2 gnd i o gnd NMOS_VTL W=225nm L=45nm
20 .ends NOT

```

Listing 6: inv.cir

```

1 * =====
2 * Circuit      : CMOS Inverter with 1 PMOS + 1 NMOS
3 * Description: tr = tf when C_L = 0.024pF
4 *
5 * Author       : Wuqiong Zhao (me@wqzhao.org)
6 * Date        : 2023-06-01
7 * License     : MIT
8 * =====
9
10 * Reference:
11 * https://github.com/cornell-ece5745/ece5745-tut10-spice/blob/master/sim/inv-sim.sp
12
13 .title CMOS Inverter
14
15 * Parameters and Model
16 * -----
17 .param VDD='1.0V'
18 .temp 27
19 .inc ./FreePDK45/ff.inc
20
21 * Supply Voltage Source
22 * -----
23 Vdd vdd gnd VDD
24

```

```

25 * Inverter
26 * -----
27 .inc ./inv.inc
28 XInv gnd in out vdd INV
29
30 * Load Capacitor
31 * -----
32 CL out gnd 24fF
33
34 * Input Signals
35 * -----
36 Vin in gnd PWL
37 + (
38 + 0.0ns 0V
39 + 0.9ns 0V
40 + 1.1ns VDD
41 + 1.9ns VDD
42 + 2.1ns 0V
43 + 3.0ns 0V
44 + )
45
46 * Analysis
47 * -----
48 .ic V(out)=VDD
49 .tran 0.005ns 3ns
50
51 .control
52 run
53 * >>>> plot >>>>>
54 set xgridwidth = 2
55 set xbrushwidth = 3
56 * "svgwidth", "svgheight", "svgfont-size", "svgfont-width", "svguse-color",
57 ↵ "svgstroke-width", "svggrid-width",
58 set svg_intopts = ( 1024 256 16 0 1 2 0 )
59 * "svgbackground", "svgfont-family", "svgfont"
60 setcs svg_stropts = ( white Arial Arial )
61 set hcopydevtype = svg
62 set color1 = black
63 set color2 = blue
64 set color3 = red
65
66 hardcopy fig/plot_inv_t.svg
67 + out in
68 + title 'CMOS Inverter'
69 + xlabel 't'
70 + ylabel 'Voltage'
71 + ylimit 0 1
72
73 * for MS Windows, using Edge
74 if $oscompiled = 1 | $oscompiled = 8
75     shell Start fig/plot_inv_t.svg
76 else
77     if $oscompiled = 7
78         * macOS (using Safari, no need to install X11)
79         shell open -a safari fig/plot_inv_t.svg &
80     else
81         * for CYGWIN, Linux, using feh and X11
82         shell feh --magick-timeout 1 fig/plot_inv_t.svg &
83     end
84 end
85 * <<<<< plot <<<<<
86 .endc
87
88 * Measurement
89 * -----
90 .measure tran tr   trig V(out) val='VDD*0.1' rise=1 targ V(out) val='VDD*0.9' rise=1
91 .measure tran tf   trig V(out) val='VDD*0.9' fall=1 targ V(out) val='VDD*0.1' fall=1
92 .measure tran tpdr trig V(in)  val='VDD/2'   fall=1 targ V(out) val='VDD/2'   rise=1
93 .measure tran tpdf trig V(in)  val='VDD/2'   rise=1 targ V(out) val='VDD/2'   fall=1
94 .measure tran tpd  param='(tpdr+tpdf)/2'
95 .end

```

## B.2. NAND2

Listing 7: nand2.inc

```

1  * =====
2  * Circuit      : CMOS NAND2 Gate
3  * Description: 2 PMOS + 2 NMOS
4  *
5  * Author       : Wuqiong Zhao (me@wqzhao.org)
6  * Date        : 2023-06-01
7  * License     : MIT
8  * =====
9
10 .subckt NAND2 gnd i1 i2 o vdd
11  *   src  gate drain body type
12  Mp1 vdd i1  o    vdd PMOS_VTL W=360nm L=45nm
13  Mp2 vdd i2  o    vdd PMOS_VTL W=360nm L=45nm
14  Mn1 t1  i1  o    gnd NMOS_VTL W=450nm L=45nm
15  Mn2 gnd i2  t1   gnd NMOS_VTL W=450nm L=45nm
16 .ends NAND2
17
18 .subckt NAND gnd i1 i2 o vdd
19  *   src  gate drain body type
20  Mp1 vdd i1  o    vdd PMOS_VTL W=360nm L=45nm
21  Mp2 vdd i2  o    vdd PMOS_VTL W=360nm L=45nm
22  Mn1 t1  i1  o    gnd NMOS_VTL W=450nm L=45nm
23  Mn2 gnd i2  t1   gnd NMOS_VTL W=450nm L=45nm
24 .ends NAND

```

Listing 8: nand2.cir

```

1  * =====
2  * Circuit      : CMOS NAND2 Gate
3  * Description: 2 PMOS + 2 NMOS
4  *
5  * Author       : Wuqiong Zhao (me@wqzhao.org)
6  * Date        : 2023-06-01
7  * License     : MIT
8  * =====
9
10 .title CMOS NAND2
11
12 * Parameters and Model
13 * -----
14 .param VDD='1.0V'
15 .temp 27
16 .inc ./FreePDK45/ff.inc
17
18 * Supply Voltage Source
19 * -----
20 Vdd vdd gnd VDD
21
22 * NAND2
23 * -----
24 XNAND2 gnd in1 in2 out vdd NAND2
25 .inc ./nand2.inc
26
27 * Load Capacitor
28 * -----
29 CL out gnd 24fF
30
31 * Input Signals
32 * -----
33 Vin1 in1 gnd PWL
34 + (
35 +   0.0ns 0V
36 +   0.9ns 0V
37 +   1.1ns VDD
38 +   3.0ns VDD
39 + )
40

```

```

41 Vin2 in2 gnd PWL
42 + (
43 + 0.0ns 0V
44 + 0.9ns 0V
45 + 1.1ns VDD
46 + 1.9ns VDD
47 + 2.1ns 0V
48 + 3.0ns 0V
49 + )
50
51 * Analysis
52 * -----
53 .ic V(out)=VDD
54 .tran 0.005ns 3ns
55
56 .control
57 run
58 * >>>> plot >>>>>
59 set xgridwidth = 2
60 set xbrushwidth = 3
61 * "svgwidth", "svgheight", "svgfont-size", "svgfont-width", "svguse-color",
62 ↵ "svgstroke-width", "svggrid-width",
63 set svg_intopts = ( 1024 256 16 0 1 2 0 )
64 * "svgbackground", "svgfont-family", "svgfont"
65 setcs svg_stropts = ( white Arial Arial )
66 set hcopydevtype = svg
67 set color1 = black
68 set color2 = red
69 set color3 = blue
70 set color4 = green
71
72 hardcopy fig/plot_nand2_t.svg
73 + in1 out in2
74 + title 'CMOS NAND2'
75 + xlabel 't'
76 + ylabel 'Voltage'
77 + ylimit 0 1
78
79 * for MS Windows, using Edge
80 if $oscompiled = 1 | $oscompiled = 8
81 shell Start fig/plot_nand2_t.svg
82 else
83 if $oscompiled = 7
84 * macOS (using Safari, no need to install X11)
85 shell open -a safari fig/plot_nand2_t.svg &
86 else
87 * for CYGWIN, Linux, using feh and X11
88 shell feh --magick-timeout 1 fig/plot_nand2_t.svg &
89 end
90 end
91 * <<<<< plot <<<<<
92 .endc
93
94 * Measurement
95 * -----
96 .measure tran tr trig V(out) val='VDD*0.1' rise=1 targ V(out) val='VDD*0.9' rise=1
97 .measure tran tf trig V(out) val='VDD*0.9' fall=1 targ V(out) val='VDD*0.1' fall=1
98 .measure tran tpdr trig V(in2) val='VDD/2' fall=1 targ V(out) val='VDD/2' rise=1
99 .measure tran tpdf trig V(in1) val='VDD/2' rise=1 targ V(out) val='VDD/2' fall=1
100 .measure tran tpd param='(tpdr+tpdf)/2'
101 .end

```

### B.3. AND2

Listing 9: and2.inc

```

1 * =====
2 * Circuit      : CMOS AND2 Gate
3 * Description: NMOS2 + Inverter (3 PMOS + 3 NMOS)

```



```

4 *
5 * Author      : Wuqiong Zhao (me@wqzhao.org)
6 * Date       : 2023-06-02
7 * License    : MIT
8 * =====
9
10 .subckt AND2 gnd i1 i2 o vdd
11     XNAND gnd i1 i2 o1 vdd NAND2
12     XInv  gnd o1   o vdd INV
13 .ends AND2
14
15 .subckt AND gnd i1 i2 o vdd
16     XNAND gnd i1 i2 o1 vdd NAND2
17     XInv  gnd o1   o vdd INV
18 .ends AND
19
20 .inc ./nand2.inc
21 .inc ./inv.inc

```

Listing 10: and2.cir

```

1 * =====
2 * Circuit      : CMOS AND2 Gate
3 * Description: NMOS2 + Inverter (3 PMOS + 3 NMOS)
4 *
5 * Author      : Wuqiong Zhao (me@wqzhao.org)
6 * Date       : 2023-06-02
7 * License    : MIT
8 * =====
9
10 .title CMOS AND2
11
12 * Parameters and Model
13 * -----
14 .param VDD='1.0V'
15 .temp 27
16 .inc ./FreePDK45/ff.inc
17
18 * Supply Voltage Source
19 * -----
20 Vdd vdd gnd VDD
21
22 * AND2
23 * -----
24 XAND2 gnd in1 in2 out vdd AND2
25 .inc ./and2.inc
26
27 * Load Capacitor
28 * -----
29 CL out gnd 24fF
30
31 * Input Signals
32 * -----
33 Vin1 in1 gnd PWL
34 + (
35 + 0.0ns 0V
36 + 0.9ns 0V
37 + 1.1ns VDD
38 + 3.0ns VDD
39 + )
40
41 Vin2 in2 gnd PWL
42 + (
43 + 0.0ns 0V
44 + 0.9ns 0V
45 + 1.1ns VDD
46 + 1.9ns VDD
47 + 2.1ns 0V
48 + 3.0ns 0V
49 + )
50
51 * Analysis

```

```

52 * -----
53 .ic      V(out)=0
54 .tran   0.005ns 3ns
55
56 .control
57     run
58     * >>>> plot >>>>
59     set xgridwidth = 2
60     set xbrushwidth = 3
61     * "svgwidth", "svgheight", "svgfont-size", "svgfont-width", "svguse-color",
62     ↪ "svgstroke-width", "svggrid-width",
63     set svg_intopts = ( 1024 256 16 0 1 2 0 )
64     * "svgbackground", "svgfont-family", "svgfont"
65     setcs svg_stropts = ( white Arial Arial )
66     set hcopydevtype = svg
67     set color1      = black
68     set color2      = red
69     set color3      = blue
70     set color4      = green
71
72     hardcopy fig/plot_and2_t.svg
73     + in1 out in2
74     + title 'CMOS AND2'
75     + xlabel 't'
76     + ylabel 'Voltage'
77     + ylimit 0 1
78
79     * for MS Windows, using Edge
80     if $oscompiled = 1 | $oscompiled = 8
81         shell Start fig/plot_and2_t.svg
82     else
83         if $oscompiled = 7
84             * macOS (using Safari, no need to install X11)
85             shell open -a safari fig/plot_and2_t.svg &
86         else
87             * for CYGWIN, Linux, using feh and X11
88             shell feh --magick-timeout 1 fig/plot_and2_t.svg &
89         end
90     end
91     * <<<<< plot <<<<<
92 .endc
93
94 * Measurement
95 * -----
96 .measure tran tr   trig V(out) val='VDD*0.1' rise=1 targ V(out) val='VDD*0.9' rise=1
97 .measure tran tf   trig V(out) val='VDD*0.9' fall=1 targ V(out) val='VDD*0.1' fall=1
98 .measure tran tpdf trig V(in1) val='VDD/2'   rise=1 targ V(out) val='VDD/2'   rise=1
99 .measure tran tpdf trig V(in2) val='VDD/2'   fall=1 targ V(out) val='VDD/2'   fall=1
100 .measure tran tpd  param='(tpdr+tpdf)/2'
101 .end

```

## B.4. NOR2

Listing 11: nor2.inc

```

1 * =====
2 * Circuit      : CMOS NOR2 Gate
3 * Description: 2 PMOS + 2 NMOS
4 *
5 * Author       : Wuqiong Zhao (me@wqzhao.org)
6 * Date         : 2023-06-01
7 * License      : MIT
8 * =====
9
10 .subckt NOR2 gnd i1 i2 o vdd
11     * src gate drain body type
12     Mp1 t1 i1 o vdd PMOS_VTL W=720nm L=45nm
13     Mp2 vdd i2 t1 o vdd PMOS_VTL W=720nm L=45nm
14     Mn1 gnd i1 o gnd NMOS_VTL W=225nm L=45nm

```

```

15 Mn2 gnd i2 o gnd NMOS_VTL W=225nm L=45nm
16 .ends NOR2
17
18 .subckt NOR gnd i1 i2 o vdd
19 * src gate drain body type
20 Mp1 t1 i1 o vdd PMOS_VTL W=720nm L=45nm
21 Mp2 vdd i2 t1 vdd PMOS_VTL W=720nm L=45nm
22 Mn1 gnd i1 o gnd NMOS_VTL W=225nm L=45nm
23 Mn2 gnd i2 o gnd NMOS_VTL W=225nm L=45nm
24 .ends NOR

```

Listing 12: nor2.cir

```

1 * =====
2 * Circuit      : CMOS NOR2 Gate
3 * Description: 2 PMOS + 2 NMOS
4 *
5 * Author       : Wuqiong Zhao (me@wqzhao.org)
6 * Date        : 2023-06-02
7 * License     : MIT
8 * =====
9
10 .title CMOS NOR2
11
12 * Parameters and Model
13 * -----
14 .param VDD='1.0V'
15 .temp 27
16 .inc ./FreePDK45/ff.inc
17
18 * Supply Voltage Source
19 * -----
20 Vdd vdd gnd VDD
21
22 * NOR2
23 * -----
24 XNOR2 gnd in1 in2 out vdd NOR2
25 .inc ./nor2.inc
26
27 * Load Capacitor
28 * -----
29 CL out gnd 24fF
30
31 * Input Signals
32 * -----
33 Vin1 in1 gnd PWL
34 + (
35 + 0.0ns VDD
36 + 0.9ns VDD
37 + 1.1ns 0V
38 + 3.0ns 0V
39 + )
40
41 Vin2 in2 gnd PWL
42 + (
43 + 0.0ns VDD
44 + 0.9ns VDD
45 + 1.1ns 0V
46 + 1.9ns 0V
47 + 2.1ns VDD
48 + 3.0ns VDD
49 + )
50
51 * Analysis
52 * -----
53 .ic V(out)=0V
54 .tran 0.005ns 3ns
55
56 .control
57 run
58 * >>>> plot >>>>>
59 set xgridwidth = 2

```

```

60 set xbrushwidth = 3
61 * "svgwidth", "svgheight", "svgfont-size", "svgfont-width", "svguse-color",
62 ↪ "svgstroke-width", "svggrid-width",
63 set svg_intopts = ( 1024 256 16 0 1 2 0 )
64 * "svgbackground", "svgfont-family", "svgfont"
65 setcs svg_stropts = ( white Arial Arial )
66 set hcopydevtype = svg
67 set color1 = black
68 set color2 = red
69 set color3 = blue
70 set color4 = green
71
72 hardcopy fig/plot_nor2_t.svg
73 + in1 out in2
74 + title 'CMOS NOR2'
75 + xlabel 't'
76 + ylabel 'Voltage'
77 + ylimit 0 1
78
79 * for MS Windows, using Edge
80 if $oscompiled = 1 | $oscompiled = 8
81     shell Start fig/plot_nor2_t.svg
82 else
83     if $oscompiled = 7
84         * macOS (using Safari, no need to install X11)
85         shell open -a safari fig/plot_nor2_t.svg &
86     else
87         * for CYGWIN, Linux, using feh and X11
88         shell feh --magick-timeout 1 fig/plot_nor2_t.svg &
89     end
90 end
91 * <<<<< plot <<<<<
92 .endc
93
94 * Measurement
95 -----
96 .measure tran tr   trig V(out) val='VDD*0.1' rise=1 targ V(out) val='VDD*0.9' rise=1
97 .measure tran tf   trig V(out) val='VDD*0.9' fall=1 targ V(out) val='VDD*0.1' fall=1
98 .measure tran tpdr trig V(in2) val='VDD/2'   fall=1 targ V(out) val='VDD/2'   rise=1
99 .measure tran tpdf trig V(in2) val='VDD/2'   rise=1 targ V(out) val='VDD/2'   fall=1
100 .measure tran tpd  param='(tpdr+tpdf)/2'
101 .end

```

## B.5. NAND4

Listing 13: nand4a.inc

```

1 * =====
2 * Circuit      : CMOS NAND4 Gate Type
3 * Description: 4 PMOS + 4 NMOS (Symmetrical Design)
4 *
5 * Author       : Wuqiong Zhao (me@wqzhao.org)
6 * Date         : 2023-06-02
7 * License      : MIT
8 * =====
9
10 .subckt NAND4A gnd i1 i2 i3 i4 o vdd
11 *   src  gate drain body type
12 Mp1 vdd i1 o vdd PMOS_VTL W=360nm L=45nm
13 Mp2 vdd i2 o vdd PMOS_VTL W=360nm L=45nm
14 Mp3 vdd i3 o vdd PMOS_VTL W=360nm L=45nm
15 Mp4 vdd i4 o vdd PMOS_VTL W=360nm L=45nm
16 Mn1 t1 i1 o gnd NMOS_VTL W=900nm L=45nm
17 Mn2 t2 i2 t1 gnd NMOS_VTL W=900nm L=45nm
18 Mn3 t3 i3 t2 gnd NMOS_VTL W=900nm L=45nm
19 Mn4 gnd i4 t3 gnd NMOS_VTL W=900nm L=45nm
20 .ends NAND4A

```

## B.6. AND4

Listing 14: and4b.inc

```

1 * =====
2 * Circuit      : CMOS AND4 Gate Type B
3 * Description: NAND2 * 2 + NOR * 1
4 *
5 * Author       : Wuqiong Zhao (me@wqzhao.org)
6 * Date        : 2023-06-02
7 * License     : MIT
8 * =====
9
10 .subckt AND4B gnd i1 i2 i3 i4 o vdd
11 *   src gate drain body type
12   XNAND2_1 gnd i1 i2 t1 vdd NAND2
13   XNAND2_2 gnd i3 i4 t2 vdd NAND2
14   XNOR     gnd t1 t2 o  vdd NOR2
15 .ends AND4B
16
17 .inc ./nand2.inc
18 .inc ./nor2.inc

```

## B.7. AND8

### B.7.1. Test Circuit

Listing 15: and8\_test\_inv2.inc

```

1 * =====
2 * Circuit      : Test Circuit With Capicitor Load
3 * Description: 2 Inverters at the Input
4 *
5 * Author       : Wuqiong Zhao (me@wqzhao.org)
6 * Date        : 2023-06-02
7 * License     : MIT
8 * =====
9
10 .subckt INV2_TEST gnd
11 +           i1 i2 i3 i4 i5 i6 i7 i8
12 +           t1 t2 t3 t4 t5 t6 t7 t8
13 +           out vdd
14 *           gnd i  o  vdd
15   XInv_ss1 gnd i1 t1 vdd INV_SS
16   XInv_ss2 gnd i2 t2 vdd INV_SS
17   XInv_ss3 gnd i3 t3 vdd INV_SS
18   XInv_ss4 gnd i4 t4 vdd INV_SS
19   XInv_ss5 gnd i5 t5 vdd INV_SS
20   XInv_ss6 gnd i6 t6 vdd INV_SS
21   XInv_ss7 gnd i7 t7 vdd INV_SS
22   XInv_ss8 gnd i8 t8 vdd INV_SS
23 * Load
24   CL out gnd 24fF
25 .ends INV2_TEST
26
27 .subckt INV_SS gnd i o vdd
28   XInv_s1 gnd i t vdd INV_S
29   XInv_s2 gnd t o vdd INV_S
30 .ends
31
32 * Inverter used in the test circuit
33 .subckt INV_S gnd i o vdd
34 *   src gate drain body type
35   M1 vdd i  o  vdd PMOS_VTL W=0.75um L=0.25um
36   M2 gnd i  o  gnd NMOS_VTL W=2.60um L=0.25um
37 .ends INV_S

```

Listing 16: and8\_test\_pow.inc

```

1  * =====
2  * Script      : Measure AND8 Gate Static and Dynamic Power
3  * Description: Measure the power for device 'xand8'.
4  *
5  * Author      : Wuqiong Zhao (me@wqzhao.org)
6  * Date       : 2023-06-02
7  * License    : MIT
8  * =====
9
10 .probe P(XAND8)
11 .control
12  * Total power
13  let p_total = mean(xand8:power)
14  * Static power (average of high and low)
15  let p_static =
16  + (
17  + mean(xand8:power[length(xand8:power) / 6, length(xand8:power) / 3]) +
18  + mean(xand8:power[length(xand8:power) / 2, length(xand8:power) / 1.5])
19  + ) / 2
20  * Dynamic power (difference of total power and static power)
21  let p_dynamic = p_total - p_static
22  print p_total
23  print p_static
24  print p_dynamic
25 .endc

```

## B.7.2. AND8A

Listing 17: and8a.inc

```

1  * =====
2  * Circuit     : CMOS AND8 Gate Type A
3  * Description: 8 PMOS + 8 NMOS (Symmetrical Design) + 1 Inv
4  *
5  * Author     : Wuqiong Zhao (me@wqzhao.org)
6  * Date      : 2023-06-02
7  * License    : MIT
8  * =====
9
10 .subckt AND8A gnd i1 i2 i3 i4 i5 i6 i7 i8 o vdd
11  XNAND8a gnd i1 i2 i3 i4 i5 i6 i7 i8 o_inv vdd NAND8A
12  Xinv gnd o_inv o vdd INV
13 .ends AND8A
14
15 .inc ./nand8a.inc
16 .inc ./inv.inc

```

Listing 18: and8a.cir

```

1  * =====
2  * Circuit     : CMOS AND8 Gate Type A
3  * Description: 8 PMOS + 8 NMOS (Symmetrical Design) + 1 Inv
4  *
5  * Author     : Wuqiong Zhao (me@wqzhao.org)
6  * Date      : 2023-06-02
7  * License    : MIT
8  * =====
9
10 .title CMOS AND8A
11
12 * Parameters and Model
13 * -----
14 .param VDD='1.0V'
15 .temp 25
16 .inc ./FreePDK45/ff.inc
17
18 * Supply Voltage Source

```

```

19 * -----
20 Vdd vdd gnd VDD
21
22 * AND8A
23 * -----
24 XAND8 gnd in1 in2 in3 in4 in5 in6 in7 in8 out vdd AND8A
25 .inc ./and8a.inc
26
27 * Test Circuit
28 * -----
29 XTest gnd
30 +     ii1 ii2 ii3 ii4 ii5 ii6 ii7 ii8
31 +     in1 in2 in3 in4 in5 in6 in7 in8
32 +     out vdd             INV2_TEST
33 .inc ./and8_test_inv2.inc
34
35 * Input Signals
36 * -----
37 Vii1 ii1 gnd PWL
38 + (
39 +   0.0ns 0V
40 +   0.9ns 0V
41 +   1.1ns VDD
42 +   3.0ns VDD
43 + )
44
45 Vii2 ii2 gnd PWL
46 + (
47 +   0.0ns 0V
48 +   0.9ns 0V
49 +   1.1ns VDD
50 +   3.0ns VDD
51 + )
52
53 Vii3 ii3 gnd PWL
54 + (
55 +   0.0ns 0V
56 +   0.9ns 0V
57 +   1.1ns VDD
58 +   3.0ns VDD
59 + )
60
61 Vii4 ii4 gnd PWL
62 + (
63 +   0.0ns 0V
64 +   0.9ns 0V
65 +   1.1ns VDD
66 +   3.0ns VDD
67 + )
68
69 Vii5 ii5 gnd PWL
70 + (
71 +   0.0ns 0V
72 +   0.9ns 0V
73 +   1.1ns VDD
74 +   3.0ns VDD
75 + )
76
77 Vii6 ii6 gnd PWL
78 + (
79 +   0.0ns 0V
80 +   0.9ns 0V
81 +   1.1ns VDD
82 +   3.0ns VDD
83 + )
84
85 Vii7 ii7 gnd PWL
86 + (
87 +   0.0ns 0V
88 +   0.9ns 0V
89 +   1.1ns VDD
90 +   3.0ns VDD
91 + )

```

```

92
93 Vii8 ii8 gnd PWL
94 + (
95 +   0.0ns 0V
96 +   0.9ns 0V
97 +   1.1ns VDD
98 +   1.9ns VDD
99 +   2.1ns 0V
100 +   3.0ns 0V
101 + )
102
103 * Analysis
104 * -----
105 .ic      V(out)=0
106 .tran   0.005ns 3ns
107
108 .control
109   run
110   * >>>> plot >>>>>
111   set xgridwidth = 2
112   set xbrushwidth = 3
113   * "svgwidth", "svgheight", "svgfont-size", "svgfont-width", "svguse-color",
114     ↵ "svgstroke-width", "svggrid-width",
115   set svg_intopts = ( 1024 256 16 0 1 2 0 )
116   * "svgbackground", "svgfont-family", "svgfont"
117   setcs svg_stropts = ( white Arial Arial )
118   set hcopydevtype = svg
119   set color1      = black
120   set color2      = red
121   set color3      = blue
122   set color4      = green
123
124   hardcopy fig/plot_and8a.t.svg
125   + in1 out in8
126   + title 'CMOS AND8a'
127   + xlabel 't'
128   + ylabel 'Voltage'
129   + ylimit -0.2 1.2
130   + ydelta 0.5
131
132   * for MS Windows, using Edge
133   if $oscompiled = 1 | $oscompiled = 8
134     shell Start fig/plot_and8a.t.svg
135   else
136     if $oscompiled = 7
137       * macOS (using Safari, no need to install X11)
138       shell open -a safari fig/plot_and8a.t.svg &
139     else
140       * for CYGWIN, Linux, using feh and X11
141       shell feh --magick-timeout 1 fig/plot_and8a.t.svg &
142     end
143   end
144   * <<<<< plot <<<<<
145 .endc
146
147 * Measurement
148 * -----
149 .measure tran tr   trig V(out) val='VDD*0.1' rise=1 targ V(out) val='VDD*0.9' rise=1
150 .measure tran tf   trig V(out) val='VDD*0.9' fall=1 targ V(out) val='VDD*0.1' fall=1
151 .measure tran tpdr trig V(in1) val='VDD/2'   rise=1 targ V(out) val='VDD/2'   rise=1
152 .measure tran tpdf trig V(in8) val='VDD/2'   fall=1 targ V(out) val='VDD/2'   fall=1
153 .measure tran tpd  param='(tpdr+tpdf)/2'
154
155 * power dissipation of the AND8 gate
156 .inc ./and8_test_pow.inc
157 .end

```

### B.7.3. AND8B



Listing 19: and8b.inc

```

1  * =====
2  * Circuit      : CMOS AND8 Gate Type B
3  * Description: NAND4A * 2 + NOR2 * 1
4  *
5  * Author       : Wuqiong Zhao (me@wqzhao.org)
6  * Date        : 2023-06-02
7  * License     : MIT
8  * =====
9
10 .subckt AND8B gnd i1 i2 i3 i4 i5 i6 i7 i8 o vdd
11   *   src  gate drain body type
12   XNAND4A_1 gnd i1 i2 i3 i4 t1 vdd NAND4A
13   XNAND4A_2 gnd i5 i6 i7 i8 t2 vdd NAND4A
14   XNOR2     gnd t1 t2      o  vdd NOR2
15 .ends AND8B
16
17 .inc ./nand4a.inc
18 .inc ./nor2.inc

```

Listing 20: and8b.cir

```

1  * =====
2  * Circuit      : CMOS AND8 Gate Type B
3  * Description: NAND4A * 2 + NOR2 * 1
4  *
5  * Author       : Wuqiong Zhao (me@wqzhao.org)
6  * Date        : 2023-06-02
7  * License     : MIT
8  * =====
9
10 .title CMOS AND8B
11
12 * Parameters and Model
13 * -----
14 .param VDD='1.0V'
15 .temp 25
16 .inc ./FreePDK45/ff.inc
17
18 * Supply Voltage Source
19 * -----
20 Vdd vdd gnd VDD
21
22 * AND8B
23 * -----
24 XAND8 gnd in1 in2 in3 in4 in5 in6 in7 in8 out vdd AND8B
25 .inc ./and8b.inc
26
27 * Test Circuit
28 * -----
29 XTest gnd
30 +   ii1 ii2 ii3 ii4 ii5 ii6 ii7 ii8
31 +   in1 in2 in3 in4 in5 in6 in7 in8
32 +   out vdd                INV2_TEST
33 .inc ./and8_test_inv2.inc
34
35 * Input Signals
36 * -----
37 Vii1 ii1 gnd PWL
38 + (
39 +   0.0ns 0V
40 +   0.9ns 0V
41 +   1.1ns VDD
42 +   3.0ns VDD
43 + )
44
45 Vii2 ii2 gnd PWL
46 + (
47 +   0.0ns 0V
48 +   0.9ns 0V
49 +   1.1ns VDD
50 +   3.0ns VDD

```

```

51 + )
52
53 Vii3 ii3 gnd PWL
54 + (
55 + 0.0ns 0V
56 + 0.9ns 0V
57 + 1.1ns VDD
58 + 3.0ns VDD
59 + )
60
61 Vii4 ii4 gnd PWL
62 + (
63 + 0.0ns 0V
64 + 0.9ns 0V
65 + 1.1ns VDD
66 + 3.0ns VDD
67 + )
68
69 Vii5 ii5 gnd PWL
70 + (
71 + 0.0ns 0V
72 + 0.9ns 0V
73 + 1.1ns VDD
74 + 3.0ns VDD
75 + )
76
77 Vii6 ii6 gnd PWL
78 + (
79 + 0.0ns 0V
80 + 0.9ns 0V
81 + 1.1ns VDD
82 + 3.0ns VDD
83 + )
84
85 Vii7 ii7 gnd PWL
86 + (
87 + 0.0ns 0V
88 + 0.9ns 0V
89 + 1.1ns VDD
90 + 3.0ns VDD
91 + )
92
93 Vii8 ii8 gnd PWL
94 + (
95 + 0.0ns 0V
96 + 0.9ns 0V
97 + 1.1ns VDD
98 + 1.9ns VDD
99 + 2.1ns 0V
100 + 3.0ns 0V
101 + )
102
103 * Analysis
104 * -----
105 .ic V(out)=0
106 .tran 0.005ns 3ns
107
108 .control
109 run
110 * >>>> plot >>>>>
111 set xgridwidth = 2
112 set xbrushwidth = 3
113 * "svgwidth", "svgheight", "svgfont-size", "svgfont-width", "svguse-color",
114 ↪ "svgstroke-width", "svggrid-width",
115 set svg_intopts = ( 1024 256 16 0 1 2 0 )
116 * "svgbackground", "svgfont-family", "svgfont"
117 setcs svg_stropts = ( white Arial Arial )
118 set hcopydevtype = svg
119 set color1 = black
120 set color2 = red
121 set color3 = blue
122 set color4 = green

```

```

123  hardcopy fig/plot_and8b_t.svg
124  + in1 out in8
125  + title 'CMOS AND8b'
126  + xlabel 't'
127  + ylabel 'Voltage'
128  + ylimit -0.2 1.2
129  + ydelta 0.5
130
131  * for MS Windows, using Edge
132  if $oscompiled = 1 | $oscompiled = 8
133    shell Start fig/plot_and8b_t.svg
134  else
135    if $oscompiled = 7
136      * macOS (using Safari, no need to install X11)
137      shell open -a safari fig/plot_and8b_t.svg &
138    else
139      * for CYGWIN, Linux, using feh and X11
140      shell feh --magick-timeout 1 fig/plot_and8b_t.svg &
141    end
142  end
143  * <<<<< plot <<<<<
144 .endc
145
146 * Measurement
147 * -----
148 .measure tran tr   trig V(out) val='VDD*0.1' rise=1 targ V(out) val='VDD*0.9' rise=1
149 .measure tran tf   trig V(out) val='VDD*0.9' fall=1 targ V(out) val='VDD*0.1' fall=1
150 .measure tran tpdr trig V(in1) val='VDD/2'   rise=1 targ V(out) val='VDD/2'   rise=1
151 .measure tran tpdf trig V(in8) val='VDD/2'   fall=1 targ V(out) val='VDD/2'   fall=1
152 .measure tran tpd  param='(tpdr+tpdf)/2'
153
154 * power dissipation of the AND8 gate
155 .inc ./and8_test_pow.inc
156
157 .end

```

## B.7.4. AND8C

Listing 21: and8c.inc

```

1  * =====
2  * Circuit      : CMOS AND8 Gate Type C
3  * Description: AND4B * 2 + AND2 * 1
4  *
5  * Author       : Wuqiong Zhao (me@wqzhao.org)
6  * Date        : 2023-06-02
7  * License     : MIT
8  * =====
9
10 * The following warning may be raised:
11 *
12 *   Warning: redefinition of .subckt nand2, ignored
13 *   Warning: redefinition of .subckt nand, ignored
14 *
15 * This is due to recursive .inc commands load the same .subckt.
16 * You can safely ignore these 2 warnings.
17
18 .subckt AND8C gnd i1 i2 i3 i4 i5 i6 i7 i8 o vdd
19   *   src gate drain body type
20   XAND4B_1 gnd i1 i2 i3 i4 t1 vdd AND4B
21   XAND4B_2 gnd i5 i6 i7 i8 t2 vdd AND4B
22   XAND2   gnd t1 t2      o vdd AND2
23 .ends AND8C
24
25 .inc ./and4b.inc
26 .inc ./and2.inc

```

Listing 22: and8c.cir

```

1  * =====
2  * Circuit      : CMOS AND8 Gate Type C
3  * Description: AND4B * 2 + AND2 * 1
4  *
5  * Author       : Wuqiong Zhao (me@wqzhao.org)
6  * Date         : 2023-06-02
7  * License      : MIT
8  * =====
9
10 .title CMOS AND8C
11
12 * Parameters and Model
13 * -----
14 .param VDD='1.0V'
15 .temp 25
16 .inc ./FreePDK45/ff.inc
17
18 * Supply Voltage Source
19 * -----
20 Vdd vdd gnd VDD
21
22 * AND8C
23 * -----
24 XAND8 gnd in1 in2 in3 in4 in5 in6 in7 in8 out vdd AND8C
25 .inc ./and8c.inc
26
27 * Test Circuit
28 * -----
29 XTest gnd
30 +   ii1 ii2 ii3 ii4 ii5 ii6 ii7 ii8
31 +   in1 in2 in3 in4 in5 in6 in7 in8
32 +   out vdd                INV2_TEST
33 .inc ./and8_test_inv2.inc
34
35 * Input Signals
36 * -----
37 Vii1 ii1 gnd PWL
38 + (
39 +   0.0ns 0V
40 +   0.9ns 0V
41 +   1.1ns VDD
42 +   3.0ns VDD
43 + )
44
45 Vii2 ii2 gnd PWL
46 + (
47 +   0.0ns 0V
48 +   0.9ns 0V
49 +   1.1ns VDD
50 +   3.0ns VDD
51 + )
52
53 Vii3 ii3 gnd PWL
54 + (
55 +   0.0ns 0V
56 +   0.9ns 0V
57 +   1.1ns VDD
58 +   3.0ns VDD
59 + )
60
61 Vii4 ii4 gnd PWL
62 + (
63 +   0.0ns 0V
64 +   0.9ns 0V
65 +   1.1ns VDD
66 +   3.0ns VDD
67 + )
68
69 Vii5 ii5 gnd PWL
70 + (
71 +   0.0ns 0V
72 +   0.9ns 0V
73 +   1.1ns VDD

```

```

74 + 3.0ns VDD
75 + )
76
77 Vii6 ii6 gnd PWL
78 + (
79 + 0.0ns 0V
80 + 0.9ns 0V
81 + 1.1ns VDD
82 + 3.0ns VDD
83 + )
84
85 Vii7 ii7 gnd PWL
86 + (
87 + 0.0ns 0V
88 + 0.9ns 0V
89 + 1.1ns VDD
90 + 3.0ns VDD
91 + )
92
93 Vii8 ii8 gnd PWL
94 + (
95 + 0.0ns 0V
96 + 0.9ns 0V
97 + 1.1ns VDD
98 + 1.9ns VDD
99 + 2.1ns 0V
100 + 3.0ns 0V
101 + )
102
103 * Analysis
104 * -----
105 .ic V(out)=0
106 .tran 0.005ns 3ns
107
108 .control
109 run
110 * >>>> plot >>>>>
111 set xgridwidth = 2
112 set xbrushwidth = 3
113 * "svgwidth", "svgheight", "svgfont-size", "svgfont-width", "svguse-color",
↔ "svgstroke-width", "svggrid-width",
114 set svg_intopts = ( 1024 256 16 0 1 2 0 )
115 * "svgbackground", "svgfont-family", "svgfont"
116 setcs svg_stropts = ( white Arial Arial )
117 set hcopydevtype = svg
118 set color1 = black
119 set color2 = red
120 set color3 = blue
121 set color4 = green
122
123 hardcopy fig/plot_and8c_t.svg
124 + in1 out in8
125 + title 'CMOS AND8c'
126 + xlabel 't'
127 + ylabel 'Voltage'
128 + ylimit -0.2 1.2
129 + ydelta 0.5
130
131 * for MS Windows, using Edge
132 if $oscompiled = 1 | $oscompiled = 8
133 shell Start fig/plot_and8c_t.svg
134 else
135 if $oscompiled = 7
136 * macOS (using Safari, no need to install X11)
137 shell open -a safari fig/plot_and8c_t.svg &
138 else
139 * for CYGWIN, Linux, using feh and X11
140 shell feh --magick-timeout 1 fig/plot_and8c_t.svg &
141 end
142 end
143 * <<<<< plot <<<<<
144 .endc
145

```

```

146 * Measurement
147 * -----
148 .measure tran tr   trig V(out) val='VDD*0.1' rise=1 targ V(out) val='VDD*0.9' rise=1
149 .measure tran tf   trig V(out) val='VDD*0.9' fall=1 targ V(out) val='VDD*0.1' fall=1
150 .measure tran tpdr trig V(in1) val='VDD/2'   rise=1 targ V(out) val='VDD/2'   rise=1
151 .measure tran tpdf trig V(in8) val='VDD/2'   fall=1 targ V(out) val='VDD/2'   fall=1
152 .measure tran tpd  param='(tpdr+tpdf)/2'
153
154 * power dissipation of the AND8 gate
155 .inc ./and8_test_pow.inc
156
157 .end

```

## B.8. Clock Controlled SR Latch

Listing 23: SR\_latch\_clk.inc

```

1 * =====
2 * Circuit      : Clock Controlled SR Latch
3 * Description: 2 PMOS + 6 NMOS
4 *
5 * Author       : Wuqiong Zhao (me@wqzhao.org)
6 * Date         : 2023-06-03
7 * License      : MIT
8 * =====
9
10 * .param WL = 5
11 .subckt SR_LATCH_CLK gnd s r clk q qn vdd
12 *  src  gate drain body type
13 M1 qn  q    gnd  gnd  NMOS_VTL W=    90nm L=45nm
14 M2 qn  q    vdd  vdd  PMOS_VTL W=   270nm L=45nm
15 M3 q   qn   gnd  gnd  NMOS_VTL W=    90nm L=45nm
16 M4 q   qn   vdd  vdd  PMOS_VTL W=   270nm L=45nm
17 M5 ts  s    gnd  gnd  NMOS_VTL W={WL*45nm} L=45nm
18 M6 qn  clk  ts   gnd  NMOS_VTL W={WL*45nm} L=45nm
19 M7 tr  r    gnd  gnd  NMOS_VTL W={WL*45nm} L=45nm
20 M8 q   clk  tr   gnd  NMOS_VTL W={WL*45nm} L=45nm
21 .ends SR_LATCH_CLK

```

Listing 24: SR\_latch\_clk.cir

```

1 * =====
2 * Circuit      : Clock Controlled SR Latch
3 * Description: 2 PMOS + 6 NMOS
4 *
5 * Author       : Wuqiong Zhao (me@wqzhao.org)
6 * Date         : 2023-06-03
7 * License      : MIT
8 * =====
9
10 .title CMOS Clock Controlled SR Latch
11
12 * Parameters and Model
13 * -----
14 .param VDD='1.0V'
15 .temp 27
16 .inc ./FreePDK45/ff.inc
17
18 * Supply Voltage Source
19 * -----
20 Vdd vdd gnd VDD
21
22 * Clock Controlled SR Latch
23 * -----
24 .inc ./SR_latch_clk.inc
25 XLatch gnd s r clk q qn vdd SR_LATCH_CLK
26
27 * Input Signals
28 * -----

```

```

29 Vs s gnd PWL
30 + (
31 + 0.0ns 0V
32 + 0.9ns 0V
33 + 1.1ns VDD
34 + 1.9ns VDD
35 + 2.1ns 0V
36 + 3.0ns 0V
37 + )
38
39 Vr r gnd PWL
40 + (
41 + 0.0ns 0V
42 + 2.4ns 0V
43 + 2.6ns VDD
44 + 3.5ns VDD
45 + )
46
47 Vclk clk gnd PWL
48 + (
49 + 0.0ns VDD
50 + 3.0ns VDD
51 + )
52
53 * Analysis
54 * -----
55 .ic V(qn)=VDD
56 .ic V(q) =0V
57 .tran 0.005ns 3ns
58
59 .param WL = 10
60 .probe V(qn)
61
62 .control
63 set xgridwidth = 2
64 set xbrushwidth = 3
65 * "svgwidth", "svgheight", "svgfont-size", "svgfont-width", "svguse-color",
66 ↔ "svgstroke-width", "svggrid-width",
67 set svg_intopts = ( 1024 256 16 0 1 2 0 )
68 * "svgbackground", "svgfont-family", "svgfont"
69 setcs svg_stropts = ( white Arial Arial )
70 set hcopydevtype = svg
71 set color1 = black
72
73 * Sweep Parameters
74 foreach x 2 4 4.2 4.5 5 6 8
75     alterparam WL = $x
76     reset
77     run
78 end
79
80 let WL2 = tran1.V(qn)
81 let WL4 = tran2.V(qn)
82 let WL4.2 = tran3.V(qn)
83 let WL4.5 = tran4.V(qn)
84 let WL5 = tran5.V(qn)
85 let WL6 = tran6.V(qn)
86 let WL8 = tran7.V(qn)
87
88 hardcopy fig/plot_sr_latch_wl_t.svg
89 + s r WL2 WL4 WL4.2 WL4.5 WL5 WL6 WL8
90 + title 'CMOS Clock Controlled SR Latch With Different W/L'
91 + xlabel 't'
92 + ylabel 'Voltage'
93 + ylimit 0 1
94
95 * for MS Windows, using Edge
96 if $oscompiled = 1 | $oscompiled = 8
97     shell Start fig/plot_sr_latch_wl_t.svg
98 else
99     if $oscompiled = 7
100         * macOS (using Safari, no need to install X11)
101         shell open -a safari fig/plot_sr_latch_wl_t.svg &

```

```

101     else
102         * for CYGWIN, Linux, using feh and X11
103         shell feh --magick-timeout 1 fig/plot_sr_latch_wl_t.svg &
104     end
105 end
106
107 * One example that This SR Latch will work.
108 alterparam WL = 6
109 reset
110 run
111 set color2      = red
112 set color3      = blue
113 set color4      = green
114 set color5      = brown
115
116 hardcopy fig/plot_sr_latch_t.svg
117 + qn q s r
118 + title 'CMOS Clock Controlled SR Latch'
119 + xlabel 't'
120 + ylabel 'Voltage'
121 + ylimit 0 1
122
123 * for MS Windows, using Edge
124 if $oscompiled = 1 | $oscompiled = 8
125     shell Start fig/plot_sr_latch_t.svg
126 else
127     if $oscompiled = 7
128         * macOS (using Safari, no need to install X11)
129         shell open -a safari fig/plot_sr_latch_t.svg &
130     else
131         * for CYGWIN, Linux, using feh and X11
132         shell feh --magick-timeout 1 fig/plot_sr_latch_t.svg &
133     end
134 end
135 .endc
136
137 .end

```

## B.9. MISC

Listing 25: .spiceinit

```

1 * Compatibility with HSPICE
2 set ngbehavior=hs

```