NGSPICE Simulation of CMOS Circuits

Wuqiong Zhao*

June 5, 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

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

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 (`tr`): rise time (from output crossing $0.1V_{DD}$ to $0.9V_{DD}$);
- t_f (`tf`): fall time (from output crossing $0.9V_{DD}$ to $0.1V_{DD}$);
- t_{pdr} (`tpdr`): rising propagation delay (from input to rising output crossing $V_{DD}/2$);
- t_{pdf} (`tpdf`): falling propagation delay (from input to falling output crossing $V_{DD}/2$);
- t_{pd} (`tpd`): 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 $C_L = 24\,\text{fF}$. The designed MOS parameters are shown in Table 1.

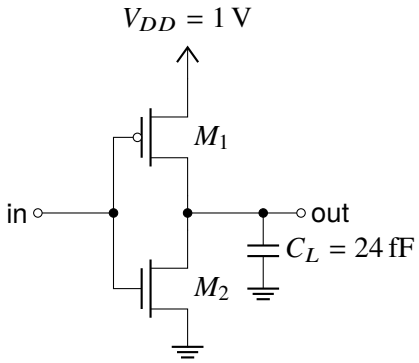

Figure 1: CMOS inverter schematic.

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

Table 1: Designed inverter MOS parameters.

Inverter Subcircuit

```
.subckt INV gnd i o vdd
  *   src   gate drain body type
  M1 vdd   i     o      vdd   PMOS_VTL W=360nm L=45nm
  M2 gnd   i     o      gnd   NMOS_VTL W=225nm L=45nm
.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 $t_r = 119\,\text{ps}$, $t_f = 120\,\text{ps}$, $t_{pdr} = 60\,\text{ps}$, $t_{pdf} = 64\,\text{ps}$ and $t_{pd} = 62\,\text{ps}$.

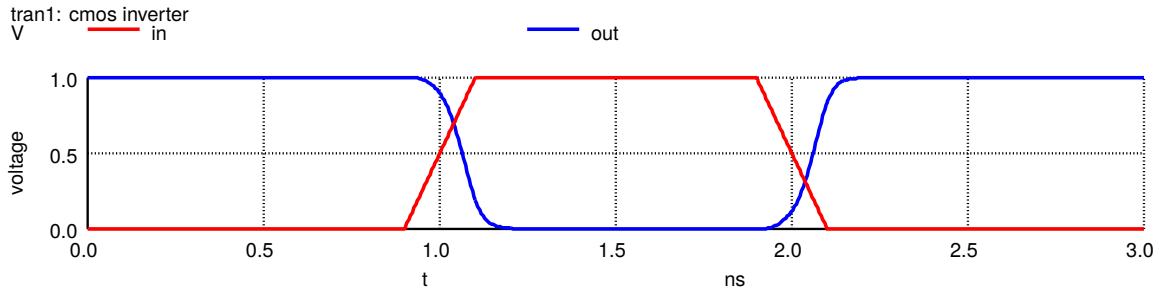

Figure 2: Response of the inverter.

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.

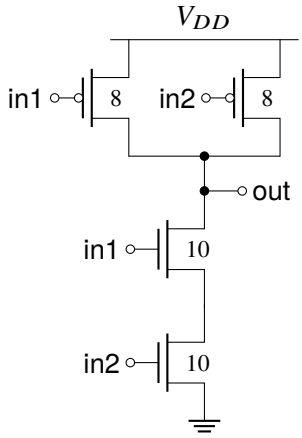

Figure 3: CMOS NAND2 schematic.

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

Table 2: Designed NAND2 MOS parameters.

NAND2 Subcircuit

```
.subckt NAND2 gnd i1 i2 o vdd
  *    src  gate drain body type
  Mp1 vdd  i1   o     vdd  PMOS_VTL W=360nm L=45nm
  Mp2 vdd  i2   o     vdd  PMOS_VTL W=360nm L=45nm
  Mn1 t1   i1   o     gnd  NMOS_VTL W=450nm L=45nm
  Mn2 gnd  i2   t1    gnd  NMOS_VTL W=450nm L=45nm
.ends NAND2
```

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

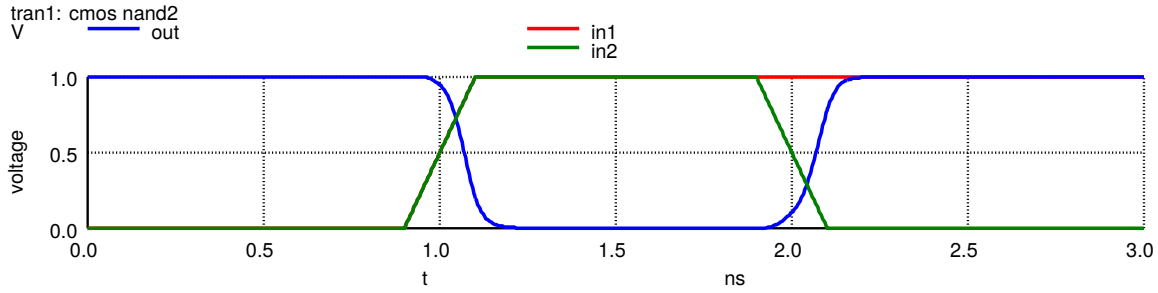

Figure 4: Response of the NAND2 gate.

3.4. AND2

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

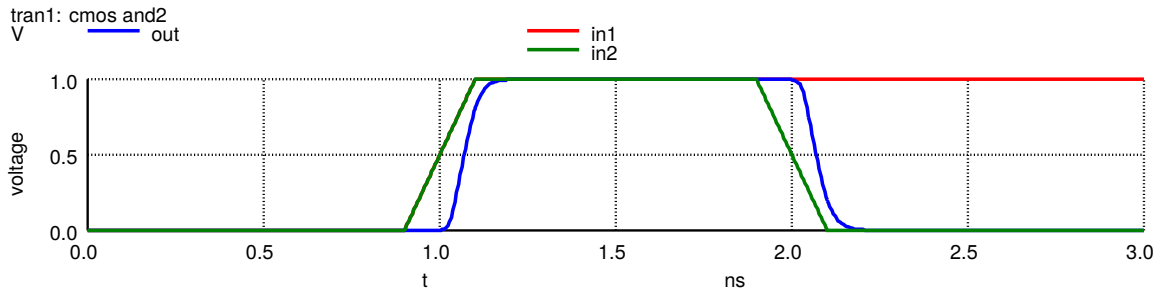

Figure 5: Response of the AND2 gate.

3.5. NOR2

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

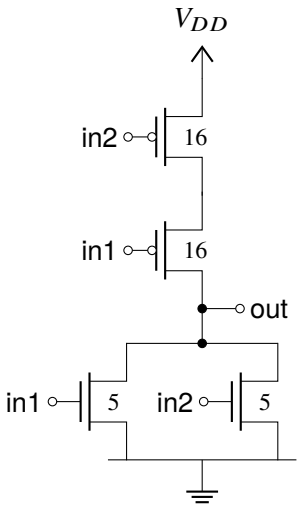

Figure 6: CMOS NOR2 schematic.

NOR2 Subcircuit

```
.subckt NOR2 gnd i1 i2 o vdd
  *    src  gate drain body type
  Mp1 t1   i1   o     vdd  PMOS_VTL W=720nm L=45nm
  Mp2 vdd  i2   t1    vdd  PMOS_VTL W=720nm L=45nm
  Mn1 gnd  i1   o     gnd  NMOS_VTL W=225nm L=45nm
  Mn2 gnd  i2   o     gnd  NMOS_VTL W=225nm L=45nm
.ends NOR2
```

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

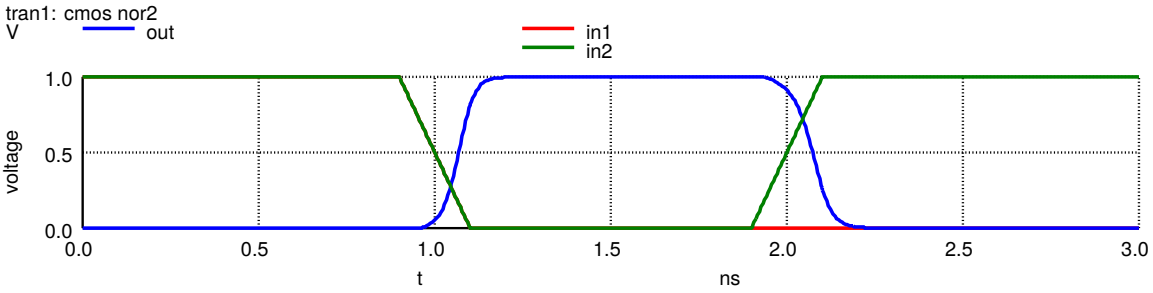

Figure 7: Response of the AND2 gate.

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 $W = 0.75\,\mu m$, $L = 0.25\,\mu m$, and PMOS has $W = 2.60\,\mu m$, $L = 0.25\,\mu m$.

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.

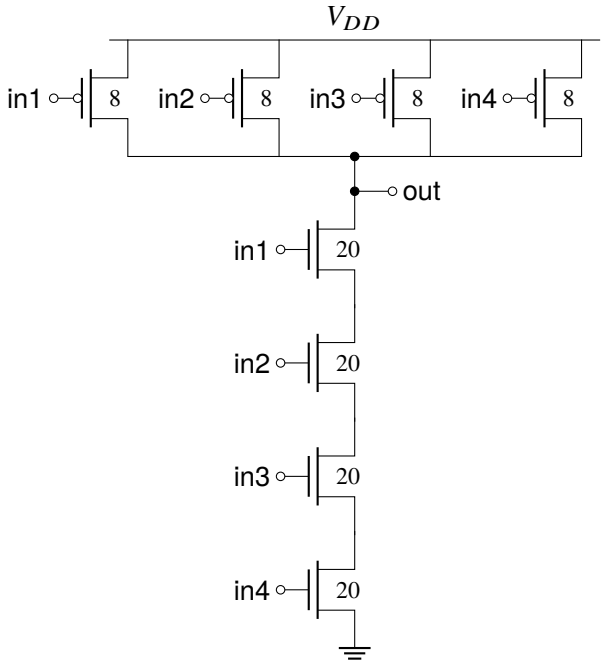

Figure 8: CMOS NAND4A schematic.

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

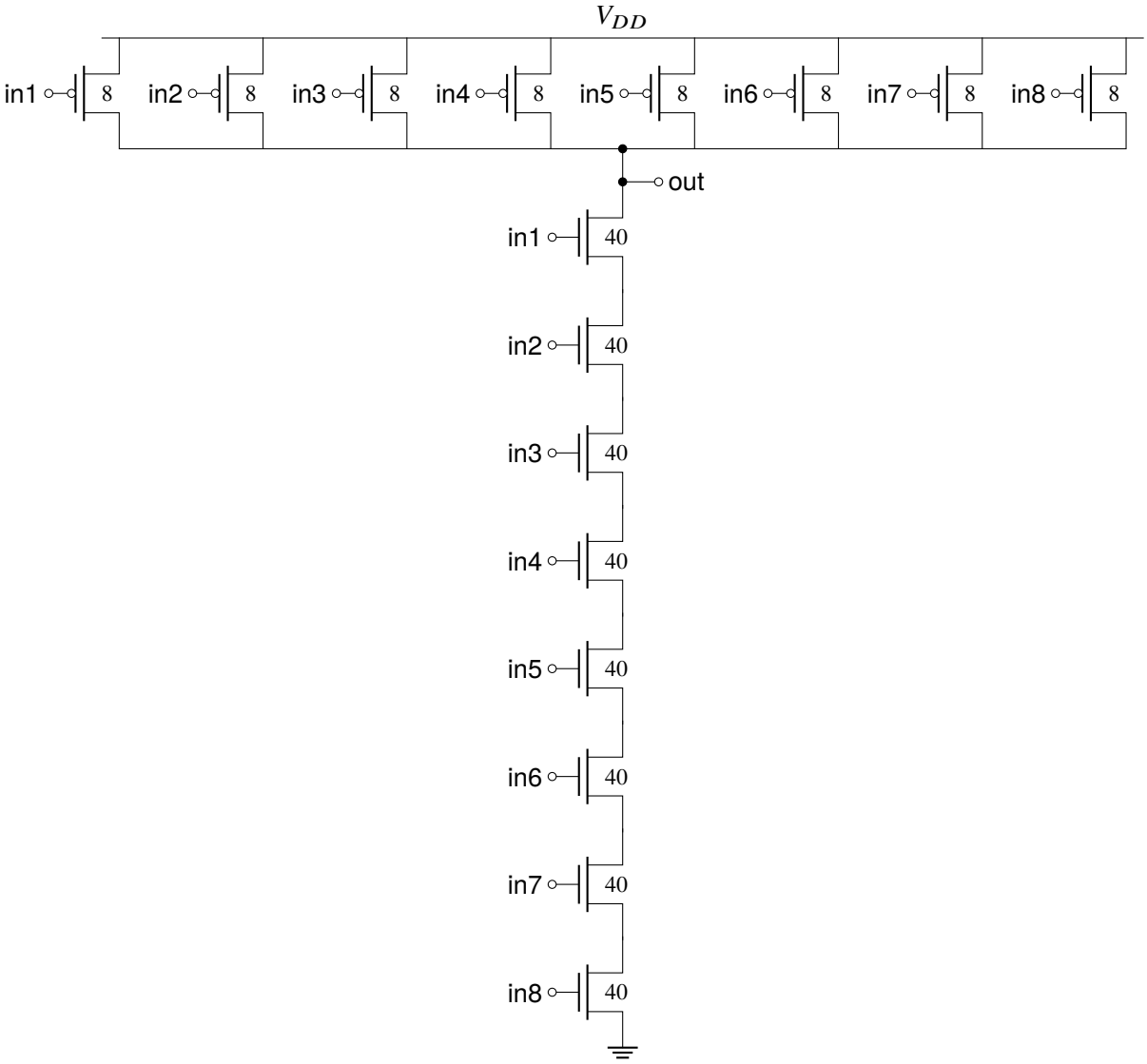

Figure 9: CMOS NAND8A schematic.

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.

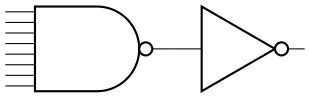

Figure 10: CMOS AND8A schematic.

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 (µW)	P dynamic (µW)
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

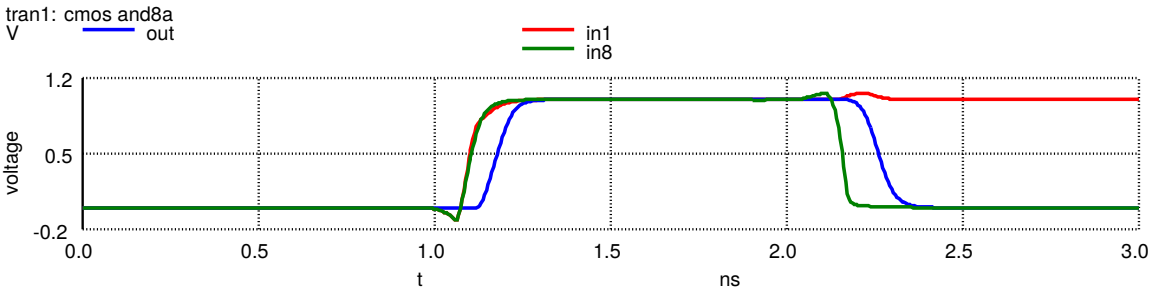

Figure 11: Response of the AND8A gate.

3.6.3. AND8B (NAND4A×2 + NOR2×1)

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

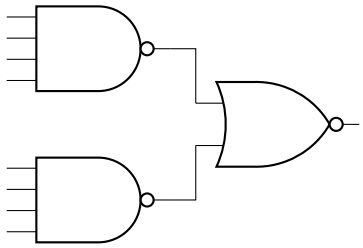

Figure 12: CMOS AND8B schematic.

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 (µW)	P dynamic (µW)
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

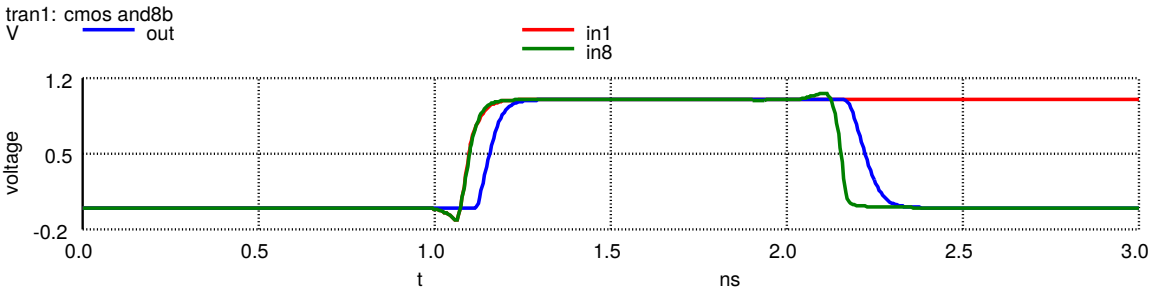

Figure 13: Response of the AND8B gate.

3.6.4. AND8C (AND4B×2 + AND2×1)

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.

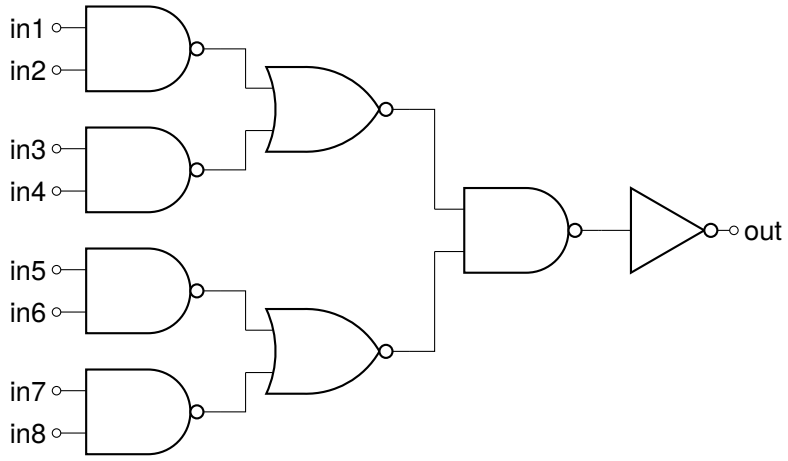

Figure 14: CMOS AND8C schematic.

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 (µW)	P dynamic (µW)
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

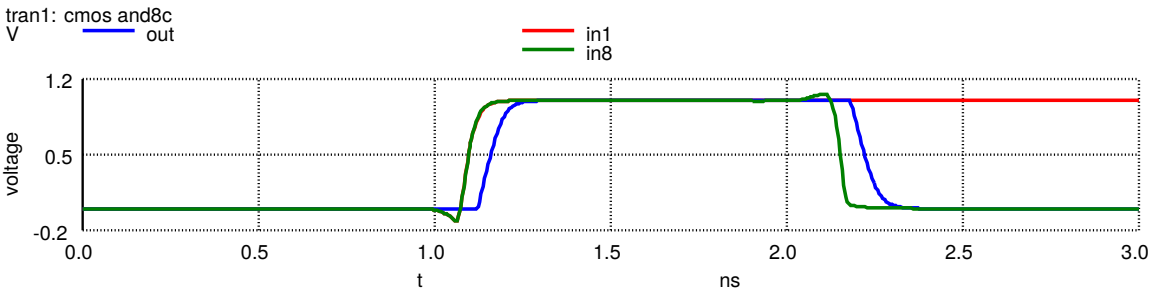

Figure 15: Response of the AND8C gate.

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 AND8B, 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.

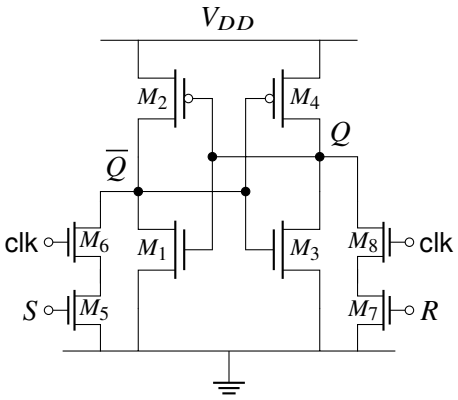

Figure 16: CMOS clock controlled SR latch schematic.

Clock Controlled SR Latch Subcircuit

```
* .param WL = 5
.subckt SR_LATCH_CLK gnd s r clk q qn vdd
  *   src   gate  drain  body  type
  M1 qn    q     gnd    gnd   NMOS_VTL W=      90nm L=45nm
  M2 qn    q     vdd    vdd   PMOS_VTL W=     270nm L=45nm
  M3 q     qn    gnd    gnd   NMOS_VTL W=      90nm L=45nm
  M4 q     qn    vdd    vdd   PMOS_VTL W=     270nm L=45nm
  M5 ts    s     gnd    gnd   NMOS_VTL W={WL*45nm} L=45nm
  M6 qn    clk   ts     gnd   NMOS_VTL W={WL*45nm} L=45nm
  M7 tr    r     gnd    gnd   NMOS_VTL W={WL*45nm} L=45nm
  M8 q     clk   tr     gnd   NMOS_VTL W={WL*45nm} L=45nm
.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.

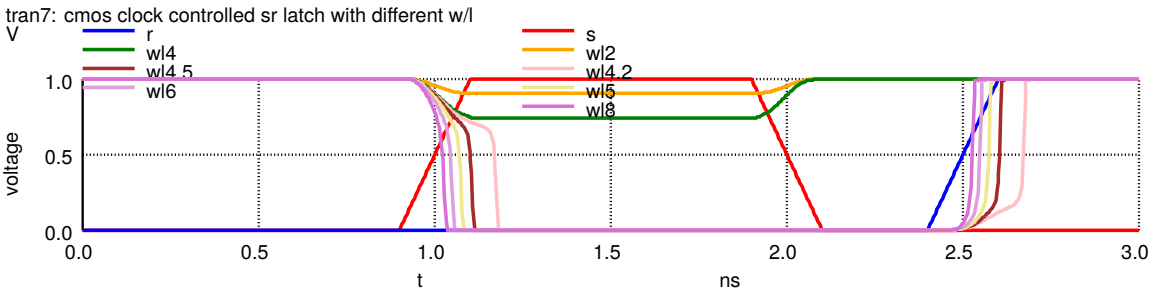

Figure 17: Clock controlled SR latch with different W/L values.

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`.

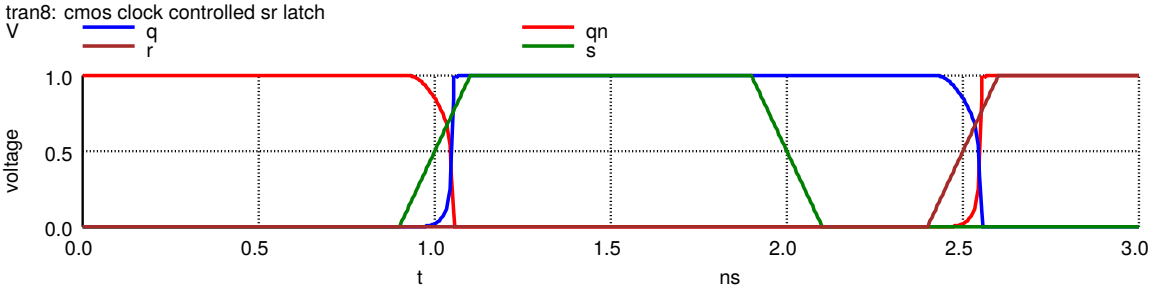

Figure 18: Response of the clock controlled SR latch when $W/L = 6$.

4. License

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 Ti*k*Z using LaTeX, you can find the source code in the GitHub repository.

B. Source Code

B.1. Inverter

Listing 5: `inv.inc`

```
* ================================================================================
* Circuit     : CMOS Inverter
* Description: 1 PMOS + 1 NMOS
*
* Author      : Wuqiong Zhao (me@wqzhao.org)
* Date        : 2023-06-02
* License     : MIT
* ================================================================================

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

.subckt NOT gnd i o vdd
  *  src  gate drain body type
  M1 vdd  i    o     vdd  PMOS_VTL W=360nm L=45nm
  M2 gnd  i    o     gnd  NMOS_VTL W=225nm L=45nm
.ends NOT
```

Listing 6: `inv.cir`

```
* ================================================================================
* Circuit     : CMOS Inverter with 1 PMOS + 1 NMOS
* Description: tr = tf when C_L = 0.024pF
*
* Author      : Wuqiong Zhao (me@wqzhao.org)
* Date        : 2023-06-01
* License     : MIT
* ================================================================================

* Reference:
* https://github.com/cornell-ece5745/ece5745-tut10-spice/blob/master/sim/inv-sim.sp

.title CMOS Inverter

* Parameters and Model
* --------------------------------------------------------------------------------
.param VDD='1.0V'
.temp  27
.inc ./FreePDK45/ff.inc

* Supply Voltage Source
* --------------------------------------------------------------------------------
Vdd vdd gnd VDD

```

```
* Inverter
* --------------------------------------------------------------------------------
.inc ./inv.inc
XInv gnd in out vdd INV

* Load Capacitor
* --------------------------------------------------------------------------------
CL out gnd 24fF

* Input Signals
* --------------------------------------------------------------------------------
Vin in gnd PWL
+ (
+   0.0ns  0V
+   0.9ns  0V
+   1.1ns VDD
+   1.9ns VDD
+   2.1ns  0V
+   3.0ns  0V
+ )

* Analysis
* --------------------------------------------------------------------------------
.ic    V(out)=VDD
.tran  0.005ns 3ns

.control
  run
  * >>>>> plot >>>>>>
  set xgridwidth  = 2
  set xbrushwidth = 3
  * "svgwidth", "svgheight",  "svgfont-size", "svgfont-width", "svguse-color",
   ↪ "svgstroke-width", "svggrid-width",
  set svg_intopts = ( 1024 256 16 0 1 2 0 )
  * "svgbackground", "svgfont-family", "svgfont"
  setcs svg_stropts = ( white Arial Arial )
  set hcopydevtype = svg
  set color1       = black
  set color2       = blue
  set color3       = red

  hardcopy fig/plot_inv_t.svg
  + out in
  + title 'CMOS Inverter'
  + xlabel 't'
  + ylabel 'Voltage'
  + ylimit 0 1

  * for MS Windows, using Edge
  if $oscompiled = 1 | $oscompiled = 8
    shell Start fig/plot_inv_t.svg
  else
    if $oscompiled = 7
      * macOS (using Safari, no need to install X11)
      shell open -a safari fig/plot_inv_t.svg &
    else
      * for CYGWIN, Linux, using feh and X11
      shell feh --magick-timeout 1 fig/plot_inv_t.svg &
    end
  end
  * <<<<< plot <<<<<
.endc

* Measurement
* --------------------------------------------------------------------------------
.measure tran tr   trig V(out) val='VDD*0.1' rise=1 targ V(out) val='VDD*0.9' rise=1
.measure tran tf   trig V(out) val='VDD*0.9' fall=1 targ V(out) val='VDD*0.1' fall=1
.measure tran tpdr trig V(in)  val='VDD/2'   fall=1 targ V(out) val='VDD/2'   rise=1
.measure tran tpdf trig V(in)  val='VDD/2'   rise=1 targ V(out) val='VDD/2'   fall=1
.measure tran tpd  param='(tpdr+tpdf)/2'

.end
```

B.2. NAND2

Listing 7: nand2.inc

```
* ================================================================================
* Circuit     : CMOS NAND2 Gate
* Description: 2 PMOS + 2 NMOS
*
* Author      : Wuqiong Zhao (me@wqzhao.org)
* Date        : 2023-06-01
* License     : MIT
* ================================================================================

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

.subckt NAND gnd i1 i2 o vdd
  *    src  gate drain body type
  Mp1 vdd  i1   o     vdd  PMOS_VTL W=360nm L=45nm
  Mp2 vdd  i2   o     vdd  PMOS_VTL W=360nm L=45nm
  Mn1 t1   i1   o     gnd  NMOS_VTL W=450nm L=45nm
  Mn2 gnd  i2   t1    gnd  NMOS_VTL W=450nm L=45nm
.ends NAND
```

Listing 8: nand2.cir

```
* ================================================================================
* Circuit     : CMOS NAND2 Gate
* Description: 2 PMOS + 2 NMOS
*
* Author      : Wuqiong Zhao (me@wqzhao.org)
* Date        : 2023-06-01
* License     : MIT
* ================================================================================

.title CMOS NAND2

* Parameters and Model
* --------------------------------------------------------------------------------
.param VDD='1.0V'
.temp  27
.inc ./FreePDK45/ff.inc

* Supply Voltage Source
* --------------------------------------------------------------------------------
Vdd vdd gnd VDD

* NAND2
* --------------------------------------------------------------------------------
XNAND2 gnd in1 in2 out vdd NAND2
.inc ./nand2.inc

* Load Capacitor
* --------------------------------------------------------------------------------
CL out gnd 24fF

* Input Signals
* --------------------------------------------------------------------------------
Vin1 in1 gnd PWL
+ (
+   0.0ns  0V
+   0.9ns  0V
+   1.1ns VDD
+   3.0ns VDD
+ )

```

```
Vin2 in2 gnd PWL
+ (
+   0.0ns  0V
+   0.9ns  0V
+   1.1ns VDD
+   1.9ns VDD
+   2.1ns  0V
+   3.0ns  0V
+ )

* Analysis
* --------------------------------------------------------------------------
.ic    V(out)=VDD
.tran  0.005ns 3ns

.control
  run
  * >>>>> plot >>>>>>
  set xgridwidth  = 2
  set xbrushwidth = 3
  * "svgwidth", "svgheight",  "svgfont-size", "svgfont-width", "svguse-color",
   ↪ "svgstroke-width", "svggrid-width",
  set svg_intopts = ( 1024 256 16 0 1 2 0 )
  * "svgbackground", "svgfont-family", "svgfont"
  setcs svg_stropts = ( white Arial Arial )
  set hcopydevtype = svg
  set color1       = black
  set color2       = red
  set color3       = blue
  set color4       = green

  hardcopy fig/plot_nand2_t.svg
  + in1 out in2
  + title 'CMOS NAND2'
  + xlabel 't'
  + ylabel 'Voltage'
  + ylimit 0 1

  * for MS Windows, using Edge
  if $oscompiled = 1 | $oscompiled = 8
    shell Start fig/plot_nand2_t.svg
  else
    if $oscompiled = 7
      * macOS (using Safari, no need to install X11)
      shell open -a safari fig/plot_nand2_t.svg &
    else
      * for CYGWIN, Linux, using feh and X11
      shell feh --magick-timeout 1 fig/plot_nand2_t.svg &
    end
  end
  * <<<<< plot <<<<<
.endc

* Measurement
* --------------------------------------------------------------------------
.measure tran tr   trig V(out) val='VDD*0.1' rise=1 targ V(out) val='VDD*0.9' rise=1
.measure tran tf   trig V(out) val='VDD*0.9' fall=1 targ V(out) val='VDD*0.1' fall=1
.measure tran tpdr trig V(in2) val='VDD/2'   fall=1 targ V(out) val='VDD/2'   rise=1
.measure tran tpdf trig V(in1) val='VDD/2'   rise=1 targ V(out) val='VDD/2'   fall=1
.measure tran tpd  param='(tpdr+tpdf)/2'

.end
```

B.3. AND2

Listing 9: and2.inc

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

```
*
* Author      : Wuqiong Zhao (me@wqzhao.org)
* Date        : 2023-06-02
* License     : MIT
* ================================================================================

.subckt AND2 gnd i1 i2 o vdd
  XNAND gnd i1 i2 o1 vdd NAND2
  XInv  gnd o1     o vdd INV
.ends AND2

.subckt AND gnd i1 i2 o vdd
  XNAND gnd i1 i2 o1 vdd NAND2
  XInv  gnd o1     o vdd INV
.ends AND

.inc ./nand2.inc
.inc ./inv.inc
```

Listing 10: and2.cir

```
* ================================================================================
* Circuit     : CMOS AND2 Gate
* Description: NMOS2 + Inverter (3 PMOS + 3 NMOS)
*
* Author      : Wuqiong Zhao (me@wqzhao.org)
* Date        : 2023-06-02
* License     : MIT
* ================================================================================

.title CMOS AND2

* Parameters and Model
* --------------------------------------------------------------------------------
.param VDD='1.0V'
.temp  27
.inc ./FreePDK45/ff.inc

* Supply Voltage Source
* --------------------------------------------------------------------------------
Vdd vdd gnd VDD

* AND2
* --------------------------------------------------------------------------------
XAND2 gnd in1 in2 out vdd AND2
.inc ./and2.inc

* Load Capacitor
* --------------------------------------------------------------------------------
CL out gnd 24fF

* Input Signals
* --------------------------------------------------------------------------------
Vin1 in1 gnd PWL
+ (
+   0.0ns  0V
+   0.9ns  0V
+   1.1ns VDD
+   3.0ns VDD
+ )

Vin2 in2 gnd PWL
+ (
+   0.0ns  0V
+   0.9ns  0V
+   1.1ns VDD
+   1.9ns VDD
+   2.1ns  0V
+   3.0ns  0V
+ )

* Analysis
```

```
* --------------------------------------------------------------------------------
.ic    V(out)=0
.tran  0.005ns 3ns

.control
  run
  * >>>>> plot >>>>>>
  set xgridwidth  = 2
  set xbrushwidth = 3
  * "svgwidth", "svgheight",  "svgfont-size", "svgfont-width", "svguse-color",
   ↪ "svgstroke-width", "svggrid-width",
  set svg_intopts = ( 1024 256 16 0 1 2 0 )
  * "svgbackground", "svgfont-family", "svgfont"
  setcs svg_stropts = ( white Arial Arial )
  set hcopydevtype = svg
  set color1       = black
  set color2       = red
  set color3       = blue
  set color4       = green

  hardcopy fig/plot_and2_t.svg
  + in1 out in2
  + title 'CMOS AND2'
  + xlabel 't'
  + ylabel 'Voltage'
  + ylimit 0 1

  * for MS Windows, using Edge
  if $oscompiled = 1 | $oscompiled = 8
    shell Start fig/plot_and2_t.svg
  else
    if $oscompiled = 7
      * macOS (using Safari, no need to install X11)
      shell open -a safari fig/plot_and2_t.svg &
    else
      * for CYGWIN, Linux, using feh and X11
      shell feh --magick-timeout 1 fig/plot_and2_t.svg &
    end
  end
  * <<<<< plot <<<<<
.endc

* Measurement
* --------------------------------------------------------------------------------
.measure tran tr   trig V(out) val='VDD*0.1' rise=1 targ V(out) val='VDD*0.9' rise=1
.measure tran tf   trig V(out) val='VDD*0.9' fall=1 targ V(out) val='VDD*0.1' fall=1
.measure tran tpdr trig V(in1) val='VDD/2'   rise=1 targ V(out) val='VDD/2'   rise=1
.measure tran tpdf trig V(in2) val='VDD/2'   fall=1 targ V(out) val='VDD/2'   fall=1
.measure tran tpd  param='(tpdr+tpdf)/2'

.end
```

B.4. NOR2

Listing 11: nor2.inc

```
* ================================================================================
* Circuit     : CMOS NOR2 Gate
* Description: 2 PMOS + 2 NMOS
*
* Author      : Wuqiong Zhao (me@wqzhao.org)
* Date        : 2023-06-01
* License     : MIT
* ================================================================================

.subckt NOR2 gnd i1 i2 o vdd
  *   src  gate drain body type
  Mp1 t1   i1   o     vdd  PMOS_VTL W=720nm L=45nm
  Mp2 vdd  i2   t1    vdd  PMOS_VTL W=720nm L=45nm
  Mn1 gnd  i1   o     gnd  NMOS_VTL W=225nm L=45nm
```

```
  Mn2 gnd  i2   o     gnd  NMOS_VTL W=225nm L=45nm
.ends NOR2

.subckt NOR gnd i1 i2 o vdd
  *   src  gate drain body type
  Mp1 t1   i1   o     vdd  PMOS_VTL W=720nm L=45nm
  Mp2 vdd  i2   t1    vdd  PMOS_VTL W=720nm L=45nm
  Mn1 gnd  i1   o     gnd  NMOS_VTL W=225nm L=45nm
  Mn2 gnd  i2   o     gnd  NMOS_VTL W=225nm L=45nm
.ends NOR
```

Listing 12: nor2.cir

```
* ================================================================================
* Circuit    : CMOS NOR2 Gate
* Description: 2 PMOS + 2 NMOS
*
* Author     : Wuqiong Zhao (me@wqzhao.org)
* Date       : 2023-06-02
* License    : MIT
* ================================================================================

.title CMOS NOR2

* Parameters and Model
* --------------------------------------------------------------------------------
.param VDD='1.0V'
.temp  27
.inc ./FreePDK45/ff.inc

* Supply Voltage Source
* --------------------------------------------------------------------------------
Vdd vdd gnd VDD

* NOR2
* --------------------------------------------------------------------------------
XNOR2 gnd in1 in2 out vdd NOR2
.inc ./nor2.inc

* Load Capacitor
* --------------------------------------------------------------------------------
CL out gnd 24fF

* Input Signals
* --------------------------------------------------------------------------------
Vin1 in1 gnd PWL
+ (
+   0.0ns VDD
+   0.9ns VDD
+   1.1ns  0V
+   3.0ns  0V
+ )

Vin2 in2 gnd PWL
+ (
+   0.0ns VDD
+   0.9ns VDD
+   1.1ns  0V
+   1.9ns  0V
+   2.1ns VDD
+   3.0ns VDD
+ )

* Analysis
* --------------------------------------------------------------------------------
.ic    V(out)=0V
.tran  0.005ns 3ns

.control
  run
  * >>>>> plot >>>>>>
  set xgridwidth  = 2
```

```
  set xbrushwidth = 3
  * "svgwidth", "svgheight",  "svgfont-size", "svgfont-width", "svguse-color",
   ↪ "svgstroke-width", "svggrid-width",
  set svg_intopts = ( 1024 256 16 0 1 2 0 )
  * "svgbackground", "svgfont-family", "svgfont"
  setcs svg_stropts = ( white Arial Arial )
  set hcopydevtype = svg
  set color1       = black
  set color2       = red
  set color3       = blue
  set color4       = green

  hardcopy fig/plot_nor2_t.svg
  + in1 out in2
  + title 'CMOS NOR2'
  + xlabel 't'
  + ylabel 'Voltage'
  + ylimit 0 1

  * for MS Windows, using Edge
  if $oscompiled = 1 | $oscompiled = 8
    shell Start fig/plot_nor2_t.svg
  else
    if $oscompiled = 7
      * macOS (using Safari, no need to install X11)
      shell open -a safari fig/plot_nor2_t.svg &
    else
      * for CYGWIN, Linux, using feh and X11
      shell feh --magick-timeout 1 fig/plot_nor2_t.svg &
    end
  end
  * <<<<< plot <<<<<
.endc

* Measurement
* --------------------------------------------------------------------------
.measure tran tr   trig V(out) val='VDD*0.1' rise=1 targ V(out) val='VDD*0.9' rise=1
.measure tran tf   trig V(out) val='VDD*0.9' fall=1 targ V(out) val='VDD*0.1' fall=1
.measure tran tpdr trig V(in2) val='VDD/2'   fall=1 targ V(out) val='VDD/2'   rise=1
.measure tran tpdf trig V(in2) val='VDD/2'   rise=1 targ V(out) val='VDD/2'   fall=1
.measure tran tpd  param='(tpdr+tpdf)/2'

.end
```

B.5. NAND4

Listing 13: nand4a.inc

```
* ==========================================================================
* Circuit     : CMOS NAND4 Gate Type
* Description: 4 PMOS + 4 NMOS (Symmetrical Design)
*
* Author      : Wuqiong Zhao (me@wqzhao.org)
* Date        : 2023-06-02
* License     : MIT
* ==========================================================================

.subckt NAND4A gnd i1 i2 i3 i4 o vdd
  *    src  gate drain body type
  Mp1 vdd  i1   o     vdd  PMOS_VTL W=360nm L=45nm
  Mp2 vdd  i2   o     vdd  PMOS_VTL W=360nm L=45nm
  Mp3 vdd  i3   o     vdd  PMOS_VTL W=360nm L=45nm
  Mp4 vdd  i4   o     vdd  PMOS_VTL W=360nm L=45nm
  Mn1 t1   i1   o     gnd  NMOS_VTL W=900nm L=45nm
  Mn2 t2   i2   t1    gnd  NMOS_VTL W=900nm L=45nm
  Mn3 t3   i3   t2    gnd  NMOS_VTL W=900nm L=45nm
  Mn4 gnd  i4   t3    gnd  NMOS_VTL W=900nm L=45nm
.ends NAND4A
```

B.6. AND4

Listing 14: and4b.inc

```
* ================================================================================
* Circuit     : CMOS AND4 Gate Type B
* Description: NAND2 * 2 + NOR * 1
*
* Author      : Wuqiong Zhao (me@wqzhao.org)
* Date        : 2023-06-02
* License     : MIT
* ================================================================================

.subckt AND4B gnd i1 i2 i3 i4 o vdd
  *    src  gate drain body type
  XNAND2_1 gnd i1 i2 t1 vdd NAND2
  XNAND2_2 gnd i3 i4 t2 vdd NAND2
  XNOR     gnd t1 t2 o  vdd NOR2
.ends AND4B

.inc ./nand2.inc
.inc ./nor2.inc
```

B.7. AND8

B.7.1. Test Circuit

Listing 15: and8_test_inv2.inc

```
* ================================================================================
* Circuit     : Test Circuit With Capicitor Load
* Description: 2 Inverters at the Input
*
* Author      : Wuqiong Zhao (me@wqzhao.org)
* Date        : 2023-06-02
* License     : MIT
* ================================================================================

.subckt INV2_TEST gnd
+                 i1 i2 i3 i4 i5 i6 i7 i8
+                 t1 t2 t3 t4 t5 t6 t7 t8
+                 out vdd
  *         gnd i  o  vdd
  XInv_ss1 gnd i1 t1 vdd INV_SS
  XInv_ss2 gnd i2 t2 vdd INV_SS
  XInv_ss3 gnd i3 t3 vdd INV_SS
  XInv_ss4 gnd i4 t4 vdd INV_SS
  XInv_ss5 gnd i5 t5 vdd INV_SS
  XInv_ss6 gnd i6 t6 vdd INV_SS
  XInv_ss7 gnd i7 t7 vdd INV_SS
  XInv_ss8 gnd i8 t8 vdd INV_SS
  * Load
  CL out gnd 24fF
.ends INV2_TEST

.subckt INV_SS gnd i o vdd
  XInv_s1 gnd i t vdd INV_S
  XInv_s2 gnd t o vdd INV_S
.ends

* Inverter used in the test circuit
.subckt INV_S gnd i o vdd
  *  src  gate drain body type
  M1 vdd  i    o     vdd  PMOS_VTL W=0.75um L=0.25um
  M2 gnd  i    o     gnd  NMOS_VTL W=2.60um L=0.25um
.ends INV_S
```

Listing 16: and8_test_pow.inc

```
* ================================================================================
* Script      : Measure AND8 Gate Static and Dynamic Power
* Description: Measure the power for device 'xand8'.
*
* Author      : Wuqiong Zhao (me@wqzhao.org)
* Date        : 2023-06-02
* License     : MIT
* ================================================================================

.probe P(XAND8)
.control
  * Total power
  let p_total   = mean(xand8:power)
  * Static power (average of high and low)
  let p_static  =
  + (
  +   mean(xand8:power[length(xand8:power) / 6, length(xand8:power) / 3]) +
  +   mean(xand8:power[length(xand8:power) / 2, length(xand8:power) / 1.5])
  + ) / 2
  * Dynamic power (difference of total power and static power)
  let p_dynamic = p_total - p_static
  print p_total
  print p_static
  print p_dynamic
.endc
```

B.7.2. AND8A

Listing 17: and8a.inc

```
* ================================================================================
* Circuit     : CMOS AND8 Gate Type A
* Description: 8 PMOS + 8 NMOS (Symmetrical Design) + 1 Inv
*
* Author      : Wuqiong Zhao (me@wqzhao.org)
* Date        : 2023-06-02
* License     : MIT
* ================================================================================

.subckt AND8A gnd i1 i2 i3 i4 i5 i6 i7 i8 o vdd
  XNAND8a gnd i1 i2 i3 i4 i5 i6 i7 i8 o_inv vdd NAND8A
  Xinv gnd o_inv o vdd INV
.ends AND8A

.inc ./nand8a.inc
.inc ./inv.inc
```

Listing 18: and8a.cir

```
* ================================================================================
* Circuit     : CMOS AND8 Gate Type A
* Description: 8 PMOS + 8 NMOS (Symmetrical Design) + 1 Inv
*
* Author      : Wuqiong Zhao (me@wqzhao.org)
* Date        : 2023-06-02
* License     : MIT
* ================================================================================

.title CMOS AND8A

* Parameters and Model
* --------------------------------------------------------------------------------
.param VDD='1.0V'
.temp  25
.inc ./FreePDK45/ff.inc

* Supply Voltage Source
```

```
* --------------------------------------------------------------------------------
Vdd vdd gnd VDD

* AND8A
* --------------------------------------------------------------------------------
XAND8 gnd in1 in2 in3 in4 in5 in6 in7 in8 out vdd AND8A
.inc ./and8a.inc

* Test Circuit
* --------------------------------------------------------------------------------
XTest gnd
+     ii1 ii2 ii3 ii4 ii5 ii6 ii7 ii8
+     in1 in2 in3 in4 in5 in6 in7 in8
+     out vdd               INV2_TEST
.inc ./and8_test_inv2.inc

* Input Signals
* --------------------------------------------------------------------------------
Vii1 ii1 gnd PWL
+ (
+   0.0ns  0V
+   0.9ns  0V
+   1.1ns VDD
+   3.0ns VDD
+ )

Vii2 ii2 gnd PWL
+ (
+   0.0ns  0V
+   0.9ns  0V
+   1.1ns VDD
+   3.0ns VDD
+ )

Vii3 ii3 gnd PWL
+ (
+   0.0ns  0V
+   0.9ns  0V
+   1.1ns VDD
+   3.0ns VDD
+ )

Vii4 ii4 gnd PWL
+ (
+   0.0ns  0V
+   0.9ns  0V
+   1.1ns VDD
+   3.0ns VDD
+ )

Vii5 ii5 gnd PWL
+ (
+   0.0ns  0V
+   0.9ns  0V
+   1.1ns VDD
+   3.0ns VDD
+ )

Vii6 ii6 gnd PWL
+ (
+   0.0ns  0V
+   0.9ns  0V
+   1.1ns VDD
+   3.0ns VDD
+ )

Vii7 ii7 gnd PWL
+ (
+   0.0ns  0V
+   0.9ns  0V
+   1.1ns VDD
+   3.0ns VDD
+ )
```

```
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_and8a_t.svg
124   + in1 out in8
125   + title 'CMOS AND8a'
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_and8a_t.svg
134   else
135     if $oscompiled = 7
136       * macOS (using Safari, no need to install X11)
137       shell open -a safari fig/plot_and8a_t.svg &
138     else
139       * for CYGWIN, Linux, using feh and X11
140       shell feh --magick-timeout 1 fig/plot_and8a_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.7.3. AND8B

Listing 19: and8b.inc

```
* ================================================================================
* Circuit     : CMOS AND8 Gate Type B
* Description: NAND4A * 2 + NOR2 * 1
*
* Author      : Wuqiong Zhao (me@wqzhao.org)
* Date        : 2023-06-02
* License     : MIT
* ================================================================================

.subckt AND8B gnd i1 i2 i3 i4 i5 i6 i7 i8 o vdd
  *    src  gate drain body type
  XNAND4A_1 gnd i1 i2 i3 i4 t1 vdd NAND4A
  XNAND4A_2 gnd i5 i6 i7 i8 t2 vdd NAND4A
  XNOR2     gnd t1 t2       o  vdd NOR2
.ends AND8B

.inc ./nand4a.inc
.inc ./nor2.inc
```

Listing 20: and8b.cir

```
* ================================================================================
* Circuit     : CMOS AND8 Gate Type B
* Description: NAND4A * 2 + NOR2 * 1
*
* Author      : Wuqiong Zhao (me@wqzhao.org)
* Date        : 2023-06-02
* License     : MIT
* ================================================================================

.title CMOS AND8B

* Parameters and Model
* --------------------------------------------------------------------------------
.param VDD='1.0V'
.temp  25
.inc ./FreePDK45/ff.inc

* Supply Voltage Source
* --------------------------------------------------------------------------------
Vdd vdd gnd VDD

* AND8B
* --------------------------------------------------------------------------------
XAND8 gnd in1 in2 in3 in4 in5 in6 in7 in8 out vdd AND8B
.inc ./and8b.inc

* Test Circuit
* --------------------------------------------------------------------------------
XTest gnd
+     ii1 ii2 ii3 ii4 ii5 ii6 ii7 ii8
+     in1 in2 in3 in4 in5 in6 in7 in8
+     out vdd               INV2_TEST
.inc ./and8_test_inv2.inc

* Input Signals
* --------------------------------------------------------------------------------
Vii1 ii1 gnd PWL
+ (
+   0.0ns  0V
+   0.9ns  0V
+   1.1ns VDD
+   3.0ns VDD
+ )

Vii2 ii2 gnd PWL
+ (
+   0.0ns  0V
+   0.9ns  0V
+   1.1ns VDD
+   3.0ns VDD
```

```
+ )

Vii3 ii3 gnd PWL
+ (
+   0.0ns  0V
+   0.9ns  0V
+   1.1ns VDD
+   3.0ns VDD
+ )

Vii4 ii4 gnd PWL
+ (
+   0.0ns  0V
+   0.9ns  0V
+   1.1ns VDD
+   3.0ns VDD
+ )

Vii5 ii5 gnd PWL
+ (
+   0.0ns  0V
+   0.9ns  0V
+   1.1ns VDD
+   3.0ns VDD
+ )

Vii6 ii6 gnd PWL
+ (
+   0.0ns  0V
+   0.9ns  0V
+   1.1ns VDD
+   3.0ns VDD
+ )

Vii7 ii7 gnd PWL
+ (
+   0.0ns  0V
+   0.9ns  0V
+   1.1ns VDD
+   3.0ns VDD
+ )

Vii8 ii8 gnd PWL
+ (
+   0.0ns  0V
+   0.9ns  0V
+   1.1ns VDD
+   1.9ns VDD
+   2.1ns  0V
+   3.0ns  0V
+ )

* Analysis
* ------------------------------------------------------------------------------
.ic    V(out)=0
.tran  0.005ns 3ns

.control
  run
  * >>>>> plot >>>>>>
  set xgridwidth  = 2
  set xbrushwidth = 3
  * "svgwidth", "svgheight",  "svgfont-size", "svgfont-width", "svguse-color",
   ↪ "svgstroke-width", "svggrid-width",
  set svg_intopts = ( 1024 256 16 0 1 2 0 )
  * "svgbackground", "svgfont-family", "svgfont"
  setcs svg_stropts = ( white Arial Arial )
  set hcopydevtype = svg
  set color1       = black
  set color2       = red
  set color3       = blue
  set color4       = green

```

```
  hardcopy fig/plot_and8b_t.svg
  + in1 out in8
  + title 'CMOS AND8b'
  + xlabel 't'
  + ylabel 'Voltage'
  + ylimit -0.2 1.2
  + ydelta 0.5

  * for MS Windows, using Edge
  if $oscompiled = 1 | $oscompiled = 8
    shell Start fig/plot_and8b_t.svg
  else
    if $oscompiled = 7
      * macOS (using Safari, no need to install X11)
      shell open -a safari fig/plot_and8b_t.svg &
    else
      * for CYGWIN, Linux, using feh and X11
      shell feh --magick-timeout 1 fig/plot_and8b_t.svg &
    end
  end
  * <<<<< plot <<<<<
.endc

* Measurement
* --------------------------------------------------------------------------------
.measure tran tr   trig V(out) val='VDD*0.1' rise=1 targ V(out) val='VDD*0.9' rise=1
.measure tran tf   trig V(out) val='VDD*0.9' fall=1 targ V(out) val='VDD*0.1' fall=1
.measure tran tpdr trig V(in1) val='VDD/2'   rise=1 targ V(out) val='VDD/2'   rise=1
.measure tran tpdf trig V(in8) val='VDD/2'   fall=1 targ V(out) val='VDD/2'   fall=1
.measure tran tpd  param='(tpdr+tpdf)/2'

* power dissipation of the AND8 gate
.inc ./and8_test_pow.inc

.end
```

B.7.4. AND8C

Listing 21: and8c.inc

```
* ================================================================================
* Circuit     : CMOS AND8 Gate Type C
* Description: AND4B * 2 + AND2 * 1
*
* Author      : Wuqiong Zhao (me@wqzhao.org)
* Date        : 2023-06-02
* License     : MIT
* ================================================================================

* The following warning may be raised:
*
*   Warning: redefinition of .subckt nand2, ignored
*   Warning: redefinition of .subckt nand, ignored
*
* This is due to recursive .inc commands load the same .subckt.
* You can safely ignore these 2 warnings.

.subckt AND8C gnd i1 i2 i3 i4 i5 i6 i7 i8 o vdd
  *    src  gate drain body type
  XAND4B_1 gnd i1 i2 i3 i4 t1 vdd AND4B
  XAND4B_2 gnd i5 i6 i7 i8 t2 vdd AND4B
  XAND2    gnd t1 t2       o  vdd AND2
.ends AND8C

.inc ./and4b.inc
.inc ./and2.inc
```

Listing 22: and8c.cir

```
* ================================================================================
* Circuit     : CMOS AND8 Gate Type C
* Description: AND4B * 2 + AND2 * 1
*
* Author      : Wuqiong Zhao (me@wqzhao.org)
* Date        : 2023-06-02
* License     : MIT
* ================================================================================

.title CMOS AND8C

* Parameters and Model
* --------------------------------------------------------------------------------
.param VDD='1.0V'
.temp  25
.inc ./FreePDK45/ff.inc

* Supply Voltage Source
* --------------------------------------------------------------------------------
Vdd vdd gnd VDD

* AND8C
* --------------------------------------------------------------------------------
XAND8 gnd in1 in2 in3 in4 in5 in6 in7 in8 out vdd AND8C
.inc ./and8c.inc

* Test Circuit
* --------------------------------------------------------------------------------
XTest gnd
+     ii1 ii2 ii3 ii4 ii5 ii6 ii7 ii8
+     in1 in2 in3 in4 in5 in6 in7 in8
+     out vdd               INV2_TEST
.inc ./and8_test_inv2.inc

* Input Signals
* --------------------------------------------------------------------------------
Vii1 ii1 gnd PWL
+ (
+   0.0ns  0V
+   0.9ns  0V
+   1.1ns VDD
+   3.0ns VDD
+ )

Vii2 ii2 gnd PWL
+ (
+   0.0ns  0V
+   0.9ns  0V
+   1.1ns VDD
+   3.0ns VDD
+ )

Vii3 ii3 gnd PWL
+ (
+   0.0ns  0V
+   0.9ns  0V
+   1.1ns VDD
+   3.0ns VDD
+ )

Vii4 ii4 gnd PWL
+ (
+   0.0ns  0V
+   0.9ns  0V
+   1.1ns VDD
+   3.0ns VDD
+ )

Vii5 ii5 gnd PWL
+ (
+   0.0ns  0V
+   0.9ns  0V
+   1.1ns VDD
```

```
+   3.0ns VDD
+ )

Vii6 ii6 gnd PWL
+ (
+   0.0ns  0V
+   0.9ns  0V
+   1.1ns VDD
+   3.0ns VDD
+ )

Vii7 ii7 gnd PWL
+ (
+   0.0ns  0V
+   0.9ns  0V
+   1.1ns VDD
+   3.0ns VDD
+ )

Vii8 ii8 gnd PWL
+ (
+   0.0ns  0V
+   0.9ns  0V
+   1.1ns VDD
+   1.9ns VDD
+   2.1ns  0V
+   3.0ns  0V
+ )

* Analysis
* ------------------------------------------------------------------------------
.ic    V(out)=0
.tran  0.005ns 3ns

.control
  run
  * >>>>> plot >>>>>>
  set xgridwidth  = 2
  set xbrushwidth = 3
  * "svgwidth", "svgheight",  "svgfont-size", "svgfont-width", "svguse-color",
   ↪ "svgstroke-width", "svggrid-width",
  set svg_intopts = ( 1024 256 16 0 1 2 0 )
  * "svgbackground", "svgfont-family", "svgfont"
  setcs svg_stropts = ( white Arial Arial )
  set hcopydevtype = svg
  set color1       = black
  set color2       = red
  set color3       = blue
  set color4       = green

  hardcopy fig/plot_and8c_t.svg
  + in1 out in8
  + title 'CMOS AND8c'
  + xlabel 't'
  + ylabel 'Voltage'
  + ylimit -0.2 1.2
  + ydelta 0.5

  * for MS Windows, using Edge
  if $oscompiled = 1 | $oscompiled = 8
    shell Start fig/plot_and8c_t.svg
  else
    if $oscompiled = 7
      * macOS (using Safari, no need to install X11)
      shell open -a safari fig/plot_and8c_t.svg &
    else
      * for CYGWIN, Linux, using feh and X11
      shell feh --magick-timeout 1 fig/plot_and8c_t.svg &
    end
  end
  * <<<<< plot <<<<<
.endc

```

```
* Measurement
* --------------------------------------------------------------------------
.measure tran tr   trig V(out) val='VDD*0.1' rise=1 targ V(out) val='VDD*0.9' rise=1
.measure tran tf   trig V(out) val='VDD*0.9' fall=1 targ V(out) val='VDD*0.1' fall=1
.measure tran tpdr trig V(in1) val='VDD/2'   rise=1 targ V(out) val='VDD/2'   rise=1
.measure tran tpdf trig V(in8) val='VDD/2'   fall=1 targ V(out) val='VDD/2'   fall=1
.measure tran tpd  param='(tpdr+tpdf)/2'

* power dissipation of the AND8 gate
.inc ./and8_test_pow.inc

.end
```

B.8. Clock Controlled SR Latch

Listing 23: SR_latch_clk.inc

```
* ============================================================================
* Circuit     : Clock Controlled SR Latch
* Description: 2 PMOS + 6 NMOS
*
* Author      : Wuqiong Zhao (me@wqzhao.org)
* Date        : 2023-06-03
* License     : MIT
* ============================================================================

* .param WL = 5
.subckt SR_LATCH_CLK gnd s r clk q qn vdd
  *  src  gate drain body type
  M1 qn   q    gnd   gnd  NMOS_VTL W=     90nm L=45nm
  M2 qn   q    vdd   vdd  PMOS_VTL W=    270nm L=45nm
  M3 q    qn   gnd   gnd  NMOS_VTL W=     90nm L=45nm
  M4 q    qn   vdd   vdd  PMOS_VTL W=    270nm L=45nm
  M5 ts   s    gnd   gnd  NMOS_VTL W={WL*45nm} L=45nm
  M6 qn   clk  ts    gnd  NMOS_VTL W={WL*45nm} L=45nm
  M7 tr   r    gnd   gnd  NMOS_VTL W={WL*45nm} L=45nm
  M8 q    clk  tr    gnd  NMOS_VTL W={WL*45nm} L=45nm
.ends SR_LATCH_CLK
```

Listing 24: SR_latch_clk.cir

```
* ============================================================================
* Circuit     : Clock Controlled SR Latch
* Description: 2 PMOS + 6 NMOS
*
* Author      : Wuqiong Zhao (me@wqzhao.org)
* Date        : 2023-06-03
* License     : MIT
* ============================================================================

.title CMOS Clock Controlled SR Latch

* Parameters and Model
* --------------------------------------------------------------------------
.param VDD='1.0V'
.temp  27
.inc ./FreePDK45/ff.inc

* Supply Voltage Source
* --------------------------------------------------------------------------
Vdd vdd gnd VDD

* Clock Controlled SR Latch
* --------------------------------------------------------------------------
.inc ./SR_latch_clk.inc
XLatch gnd s r clk q qn vdd SR_LATCH_CLK

* Input Signals
* --------------------------------------------------------------------------
```

```
Vs s gnd PWL
+ (
+   0.0ns  0V
+   0.9ns  0V
+   1.1ns VDD
+   1.9ns VDD
+   2.1ns  0V
+   3.0ns  0V
+ )

Vr r gnd PWL
+ (
+   0.0ns  0V
+   2.4ns  0V
+   2.6ns VDD
+   3.5ns VDD
+ )

Vclk clk gnd PWL
+ (
+   0.0ns VDD
+   3.0ns VDD
+ )

* Analysis
* --------------------------------------------------------------------------------
.ic    V(qn)=VDD
.ic    V(q) =0V
.tran  0.005ns 3ns

.param WL = 10
.probe V(qn)

.control
  set xgridwidth  = 2
  set xbrushwidth = 3
  * "svgwidth", "svgheight",  "svgfont-size", "svgfont-width", "svguse-color",
   ↪ "svgstroke-width", "svggrid-width",
  set svg_intopts = ( 1024 256 16 0 1 2 0 )
  * "svgbackground", "svgfont-family", "svgfont"
  setcs svg_stropts = ( white Arial Arial )
  set hcopydevtype = svg
  set color1       = black

  * Sweep Parameters
  foreach x 2 4 4.2 4.5 5 6 8
    alterparam WL = $x
    reset
    run
  end

  let WL2   = tran1.V(qn)
  let WL4   = tran2.V(qn)
  let WL4.2 = tran3.V(qn)
  let WL4.5 = tran4.V(qn)
  let WL5   = tran5.V(qn)
  let WL6   = tran6.V(qn)
  let WL8   = tran7.V(qn)

  hardcopy fig/plot_sr_latch_wl_t.svg
  + s r WL2 WL4 WL4.2 WL4.5 WL5 WL6 WL8
  + title 'CMOS Clock Controlled SR Latch With Different W/L'
  + xlabel 't'
  + ylabel 'Voltage'
  + ylimit 0 1

  * for MS Windows, using Edge
  if $oscompiled = 1 | $oscompiled = 8
    shell Start fig/plot_sr_latch_wl_t.svg
  else
    if $oscompiled = 7
      * macOS (using Safari, no need to install X11)
      shell open -a safari fig/plot_sr_latch_wl_t.svg &
```

```
    else
      * for CYGWIN, Linux, using feh and X11
      shell feh --magick-timeout 1 fig/plot_sr_latch_wl_t.svg &
    end
  end

  * One example that This SR Latch will work.
  alterparam WL = 6
  reset
  run
  set color2       = red
  set color3       = blue
  set color4       = green
  set color5       = brown

  hardcopy fig/plot_sr_latch_t.svg
  + qn q s r
  + title 'CMOS Clock Controlled SR Latch'
  + xlabel 't'
  + ylabel 'Voltage'
  + ylimit 0 1

  * for MS Windows, using Edge
  if $oscompiled = 1 | $oscompiled = 8
    shell Start fig/plot_sr_latch_t.svg
  else
    if $oscompiled = 7
      * macOS (using Safari, no need to install X11)
      shell open -a safari fig/plot_sr_latch_t.svg &
    else
      * for CYGWIN, Linux, using feh and X11
      shell feh --magick-timeout 1 fig/plot_sr_latch_t.svg &
    end
  end
.endc

.end
```

B.9. MISC

Listing 25: `.spiceinit`

```
* Compatibility with HSPICE
set ngbehavior=hs
```