Safe Box FSM Place and Route

Eric Wu, 75286914

University of British Columbia

1. Cell library layout

(1) Description of FSM function

The Safe Box design is separated into two modules. The number button capturer and the locking FSM.

The number button capturer assumes number buttons are logic 0 when unpressed and logic 1 when pressed. It not only detects the edge of the button press, but also keeps the value of the number.

The locking FSM can be separated into three parts:

- Setting up a security code (for hotel owners)
- Setting up a customer code (for customer use)
- Unlocking/Resetting

The state diagram below shows the FSM flow:

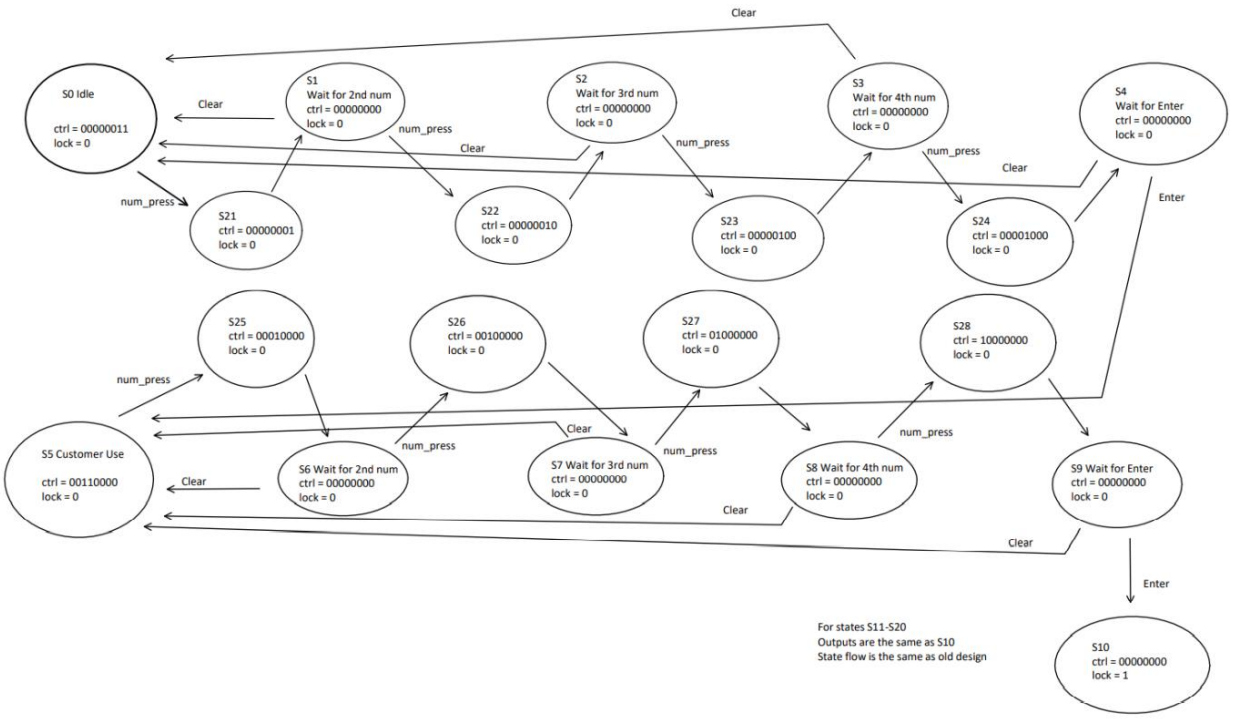


Figure 1. FSM states for setting up security code (~S5) and customer code (~S10)

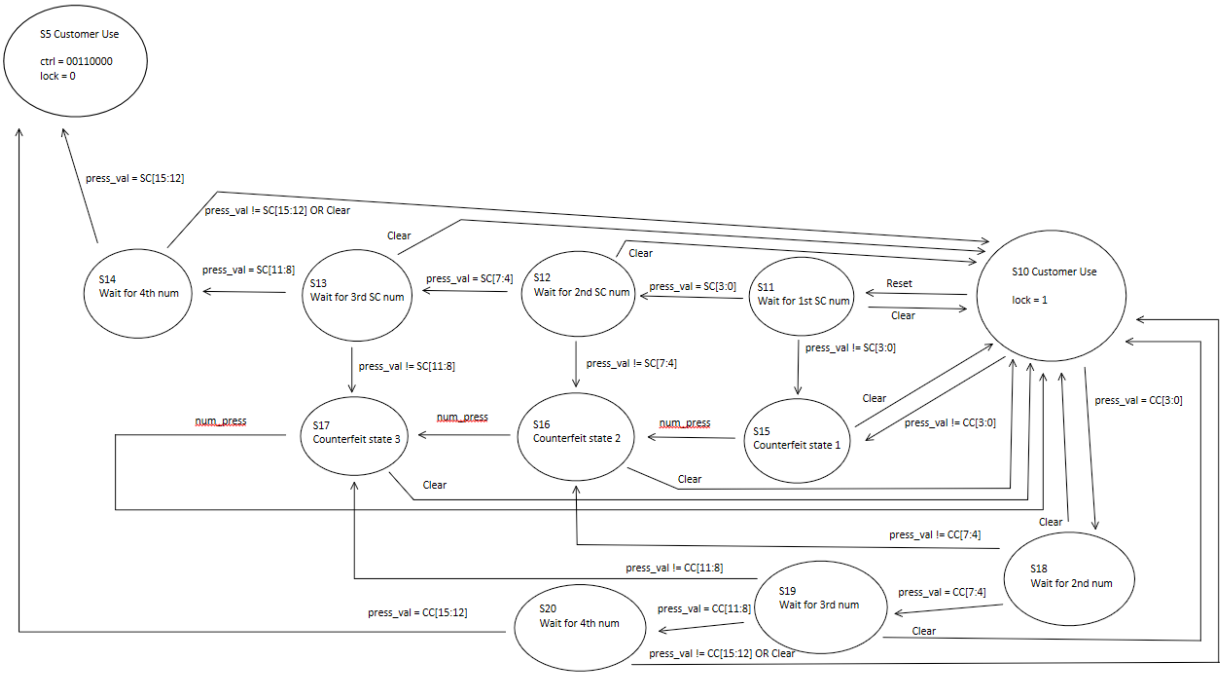


Figure 2. FSM states for unlocking using customer code (S10 number press path) and security code (S10 reset path)

(2) Inputs/Outputs description

User interface (Figure 3):

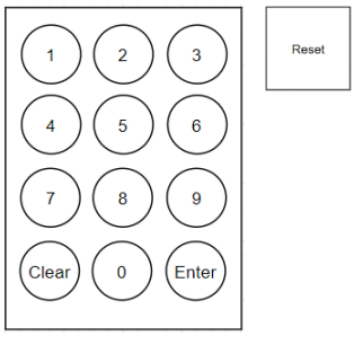


Figure 3. User interface (inputs) for Safe Box FSM

The user interface contains all inputs for our FSM.

Number buttons are for locking and unlocking purposes, clear button is for clearing up numbers entered so far, enter button is for confirming security code/customer code (to enter next stage) and

reset button is for when we want to unlock the safe box using the security code instead of the customer code.

There are two output signals, one is the lock signal, which is logic 0 when unlocked and 1 when locked. The second one is press_val[3:0] which shows the value of the number button we pressed, this can be used for HEX display (not implemented for this FSM) .

The block diagram below shows the connection between the two modules and input/outputs (Figure 4):

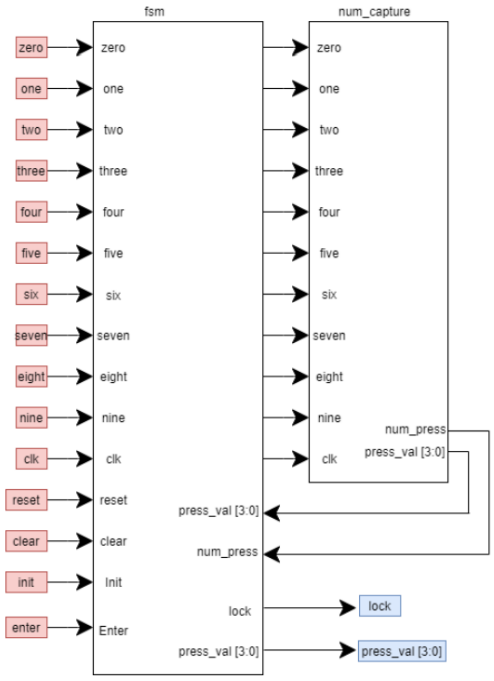


Figure 4. Block diagram for FSM (inputs are in red, outputs are in blue)

(3) Testing procedure

The original testbench for synthesis tests all possible paths.

However, simulation in virtuoso for our layout symbol takes too much time if we use the same testbench. We simplified the testbench and separated it into two tests:

- Function test: go through SC/CC setup states and both unlock paths using security code and customer code.
- Clear test: go through the locking process with the addition of “clear” input, which should be enough to verify the clear function since we have the same clear function implementation throughout our whole design.

(4) Layout of FSM

Using innovus, we did the layout for our synthesized verilog using 45nm gates (Figure 5):

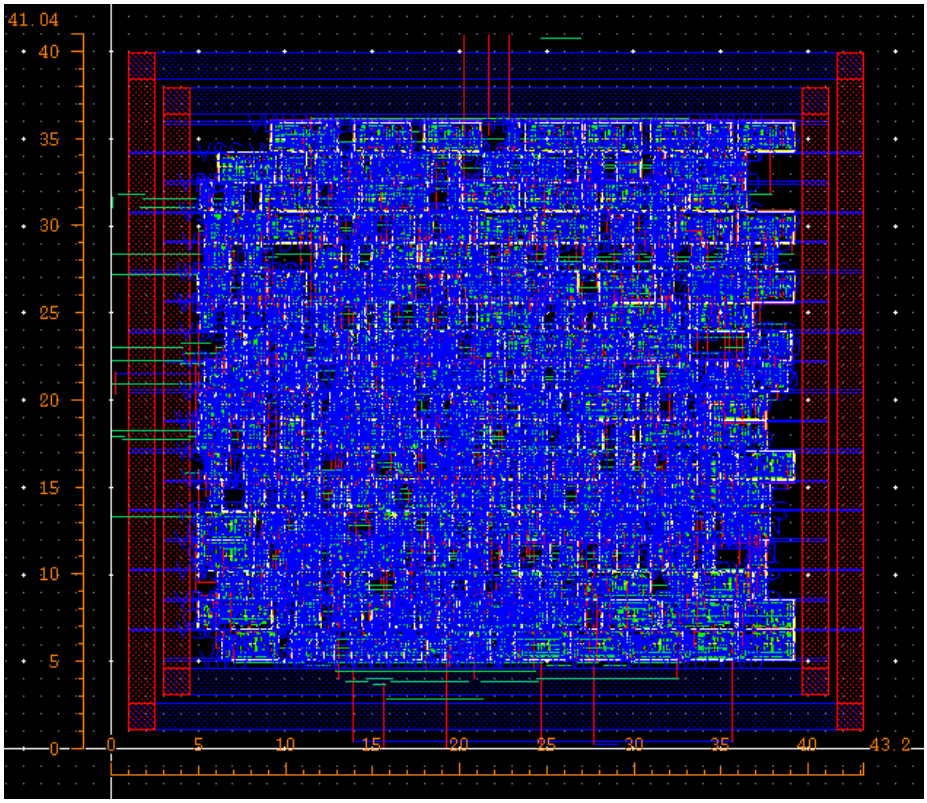


Figure 5. Layout of FSM using 45nm gates

(5) Test files and simulation results

We made a symbol using our layout and created a test circuit for simulation (Figure 6)

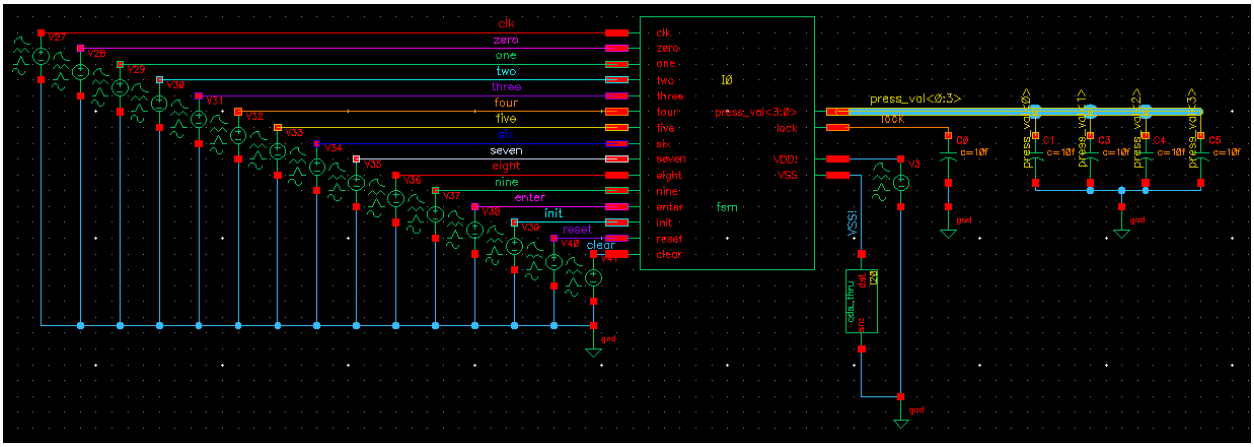


Figure 6. Test circuit for FSM verification

Input vectors were implemented by editing each input voltage source, the properties shown below is our input vector for input “zero” in our function test (Figure 7).

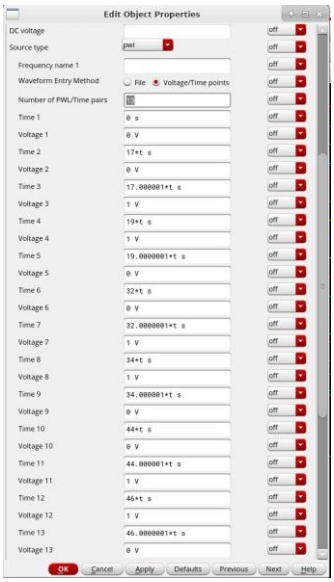
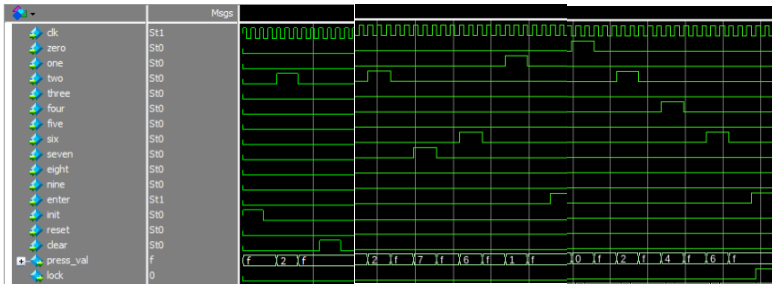


Figure 7. Showcase of input vector (input “zero”)

Function test:

Figure 8. Comparison of synthesis simulation to layout simulation (Function test)

Clear test:



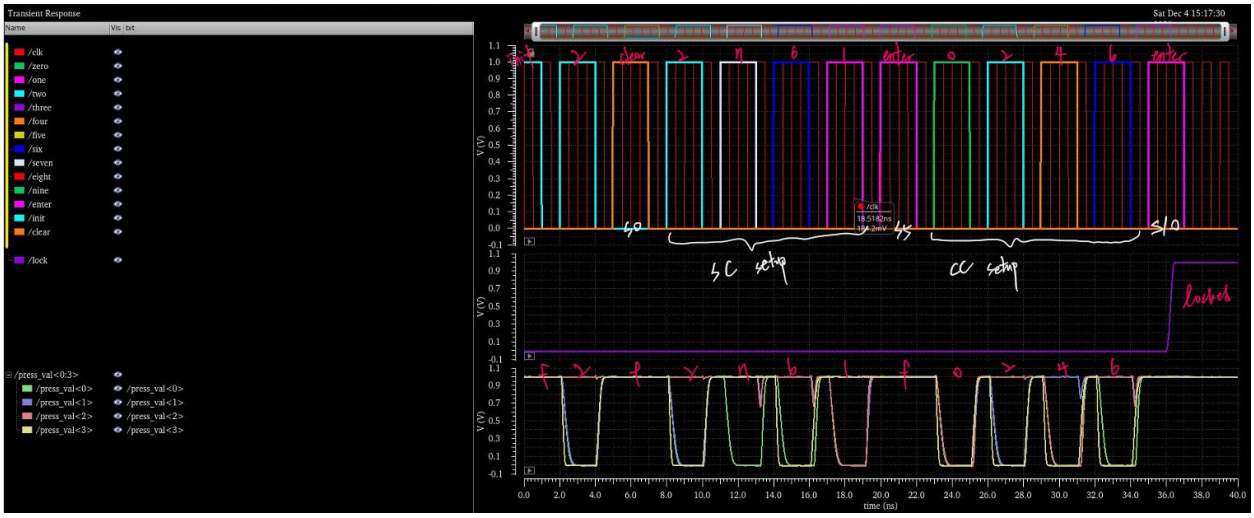


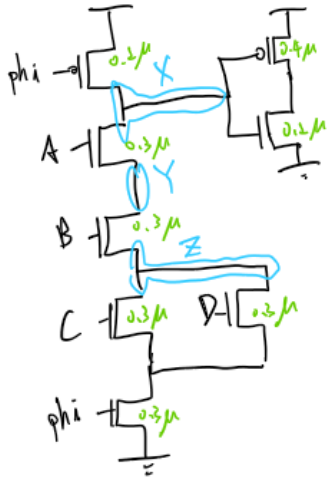
Figure 9. Comparison of synthesis simulation to layout simulation (Clear test)

2. Domino logic

1 (a).

$OUT = AB(C+D)$

2(a).



Worst case : (phi = 1)

A B C D

1. previous cycle evaluation phase : 0 1 0 1

⇒ Node Y, Z discharged

2. current cycle precharge (phi = 0) phase : A B C D = 0 1 0 1

⇒ keeps node Y, Z from charging

3. current cycle evaluation (phi = 1) phase : A B C D = 1 1 0 0

⇒ WC charge sharing between X and Y, Z

Assume A, D are still on after charge sharing.

$C_X = C_{cell} \times (0.2+0.3) + C_g \times (0.4+0.2) = 1.7\,fF$

$C_Y = C_{cell} \times 0.3 = 0.3\,fF$

$C_Z = C_{cell} \times (0.3+0.3) = 0.6\,fF$

$$V_X = V_Y = V_Z = \frac{1.7 \times 1}{2.6} = 0.654\ V$$

Assuming $V_{DD} - 0.654 > V_{TH}$, the worst case voltage drop $= V_{DD} - 0.654$

3. TG delay

Node capacitances:

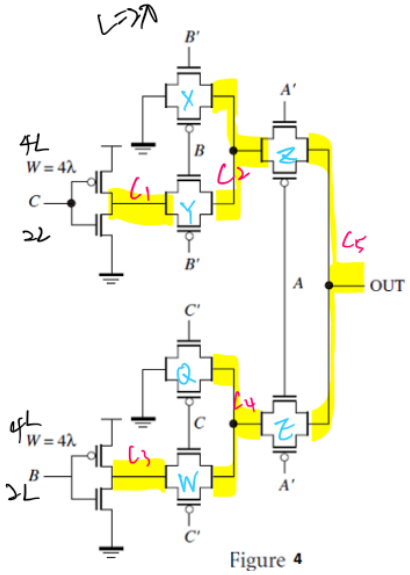


Figure 4

C_1: $6C_{cell}\cdot 2\lambda + (2C_{cell}\cdot 2\lambda + \alpha_Y C_g \cdot 2\lambda)$ (on $\alpha=1$, off $\alpha=0$)

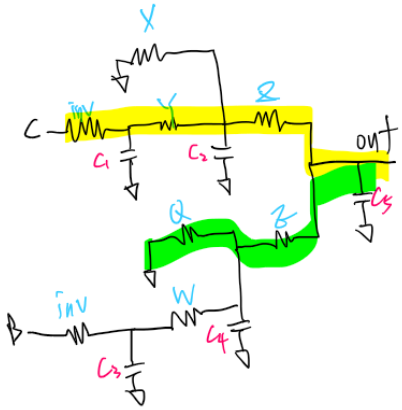
C_2: $(2C_{cell}\cdot 2\lambda + \alpha_X C_g\cdot 2\lambda) + (2C_{cell}\cdot 2\lambda + \alpha_Y C_g\cdot 2\lambda) + (2C_{cell}\cdot 2\lambda + \alpha_Z C_g\cdot 2\lambda)$

C_3: $6C_{cell}\cdot 2\lambda + (2C_{cell}\cdot 2\lambda + \alpha_W C_g\cdot 2\lambda)$

C_4: $(2C_{cell}\cdot 2\lambda + \alpha_Q C_g\cdot 2\lambda) + (2C_{cell}\cdot 2\lambda + \alpha_W C_g\cdot 2\lambda) + (2C_{cell}\cdot 2\lambda + \alpha_Z C_g\cdot 2\lambda)$

C_5: $(2C_{cell}\cdot 2\lambda + \alpha_Z C_g\cdot 2\lambda) + (2C_{cell}\cdot 2\lambda + \alpha_Z C_g\cdot 2\lambda)$

Max/Min delay:



Max delay (yellow path)

(RA)

Elmore delay = $R_{inv}C_1 + (R_{inv}+R_Y)C_2 + (R_{inv}+R_Y+R_Z)C_5$

with $\alpha_Y=1$, $\alpha_Z=1$, $\alpha_X=0$

Min delay (green path)

(1A)

Elmore delay: $R_Q C_4 + (R_Q+R_Z)C_5$

with $\alpha_Q=1$, $\alpha_W=0$, $\alpha_Z=1$

4. Power consumption

(1) Inverter power analysis

(a) Power calculation

Left circuit:

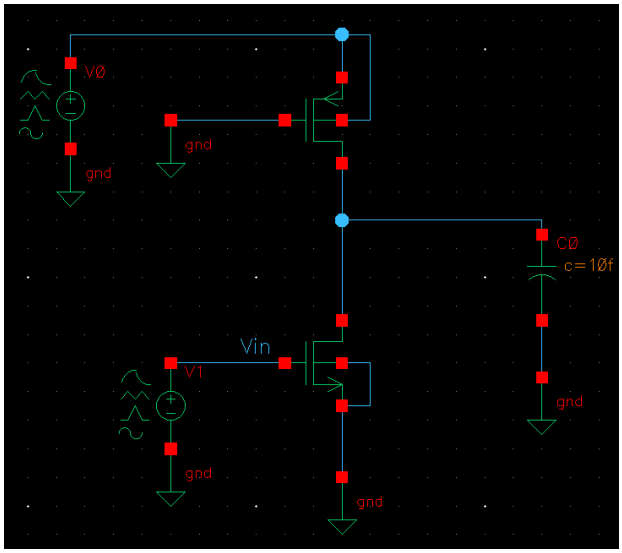


Figure 10. Schematic for left circuit

Static power can be calculated by multiplying (|V) V_{DD} and the current going through the PMOS.

Figure 11. PMOS current under static condition

$\Rightarrow P_{static} = V_{DD} \times I_p = 20.5\ nW$

Dynamic power can be calculated by Vdd*I(pmos) with a 100MHz input:

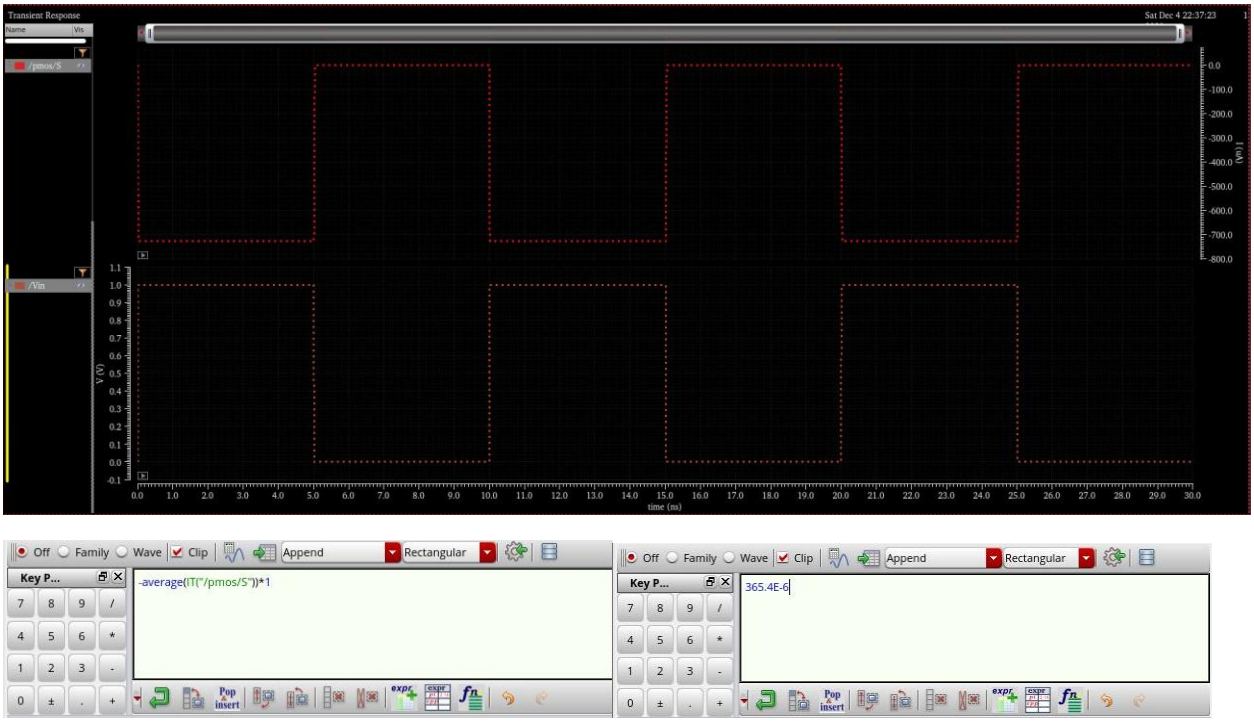


Figure 12. Dynamic power simulation for left circuit

Using the calculator, we found average dynamic power = 365.4e-6 W.

Right circuit:

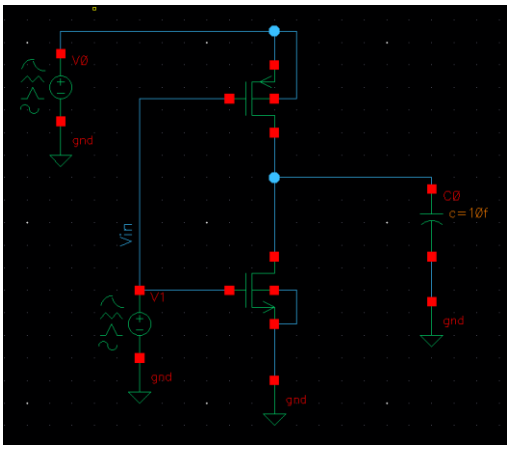


Figure 13. Schematic for right circuit

We estimated dynamic power with switching power only:

$$P_{switch} = [(8 \times 0.015\,\mu m \times 1\,fF/\mu m) + (4 \times 0.015\,\mu m \times 1\,fF/\mu m) + 10\,fF] \times 1^2 \times 10^8$$
$$= 1.01 \times 10^{-6}\ W$$

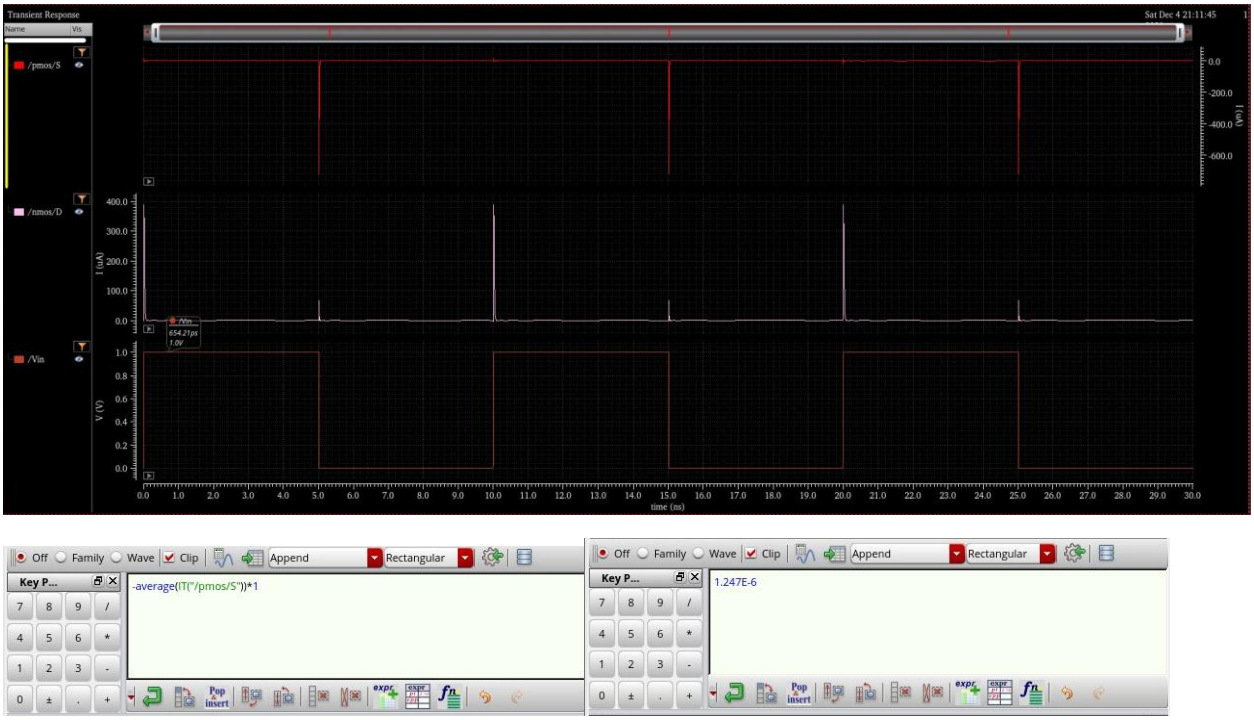


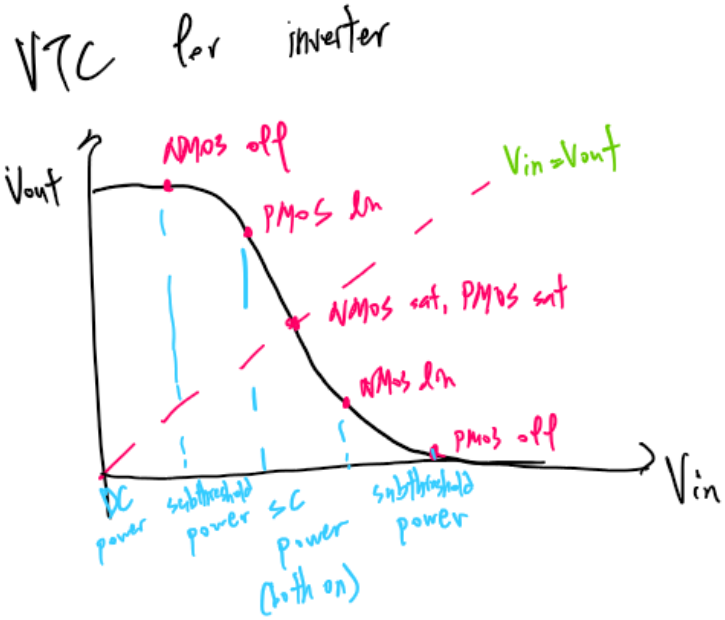
Figure 14. Dynamic power simulation for right circuit

We can clearly see the large pulses for switching power and small pulses for short circuit power.

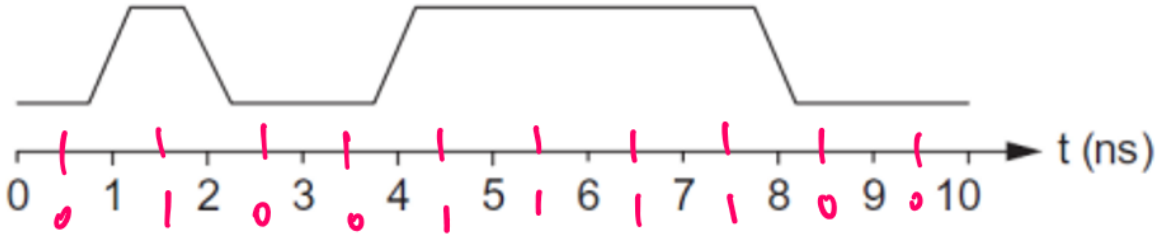
Since simulation considers short circuit power, the simulation power was expected to be higher.

We got 1.247e-6 W from our simulation which is reasonable compared to our calculated value 1.01*10e-6 W.

(c) VTC



(2) Activity factor



$P_i = 0.5$, activity factor = 0.25

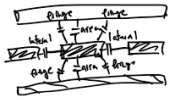
6. Interconnects

(a)

Sheet resistance $R_{\square} = \frac{1.7\times10^{-8}}{0.8\times10^{-6}} = 0.02125\ \Omega$

Total resistance $R = 0.02125 \times \frac{18\times10^{-3}}{0.4\times10^{-6}} = 4.5\ k\Omega$

R per unit length $R_{unit} = 4500 \times \frac{1}{18000} = 0.25\ \Omega/\mu m$



$C_{unit} = (2\,C_{area} + 2\,C_{lateral} + 4\,C_{fringe})$

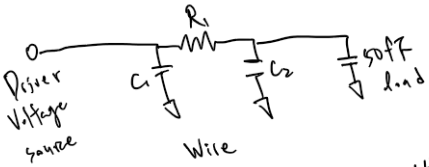
$= \left[\frac{2\varepsilon_{ox}W}{H} + \frac{2\varepsilon_{ox}T}{S} + 4\varepsilon_{ox}\ln\left(1+\frac{T}{H}\right)\right]$

$= (0.04248 + 0.02124 + 0.1015)\ fF/\mu m$

$= 0.165\ fF/\mu m$

(b)

Eq circuit - π model



$R_1 = 4500\ \Omega$, $C_1 = \frac{1}{2}\times0.165\times18000 = 1485\ fF$

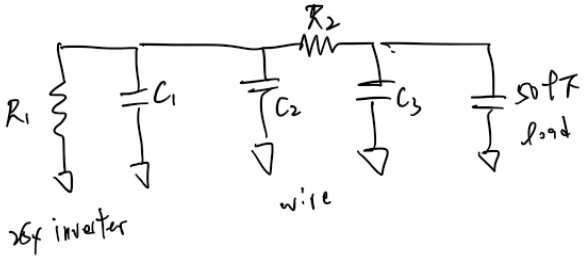
$C_2 = \frac{1}{2}\times0.165\times18000 = 1485\ fF$

Elmore delay : $0\times C_1 + (0+R_1)\times(C_2+50f)$

$= 6.91\ ns$

(c)

Eq circuit - π model



$R_1 = 34\ k\Omega/\mu m \cdot (75\times0.04\ \mu m)$

$= 34\ k\Omega$

$C_1 = 75\times0.04\ \mu m \times Cell$

$= 3\ fF$

Elmore delay : $R_1(C_1+C_3) + (R_1+R_2)(C_3+50)$

$= 109.6\ ns$