Experiment – 2

Schematic & Layout of CMOS Inverter with Equal Rise & Fall Resistance

Theory:

Complementary metal-oxide-semiconductor (CMOS) is a type of metal-oxide-semiconductor field-effect transistor (MOSFET) fabrication process that uses complementary and symmetrical pairs of p-type and n-type MOSFETs for logic functions.

A CMOS inverter contains a pMOS and a nMOS transistor connected at the drain and gate terminals, a supply voltage VDD at the pMOS source terminal, and a ground connected at the nMOS source terminal, where input is applied to the shorted gate terminals and output is measured from the shorted drain terminals.

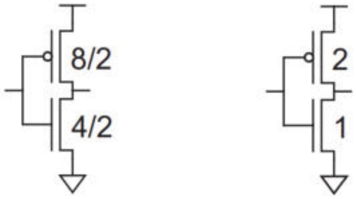


Figure A

Fig. A shows the schematic for a CMOS inverter (NOT gate) using one nMOS transistor and one pMOS transistor. Fig. A (left) shows a schematic for the inverter annotated with Width/Length for each transistor, Fig. A (right) shows a shorthand we will often use, specifying multiples of unit width and assuming minimum length. The bar at the top indicates VDD and the triangle at the bottom indicates GND (ground). When the input A is 0, the nMOS transistor is OFF and the pMOS transistor is ON. Thus, the output Y is pulled up to 1 because it is connected to VDD but not to GND. Conversely, when A is 1, the nMOS is ON, the pMOS is OFF, and Y is pulled down to 0. This is summarized in Table 1.

Table 1

A	pMOS	nMOS	Y
0	ON	OFF	**1**
1	OFF	ON	**0**

Layout design rules describe how small features can be and how closely they can be reliably packed in a particular manufacturing process. Mead and Conway popularized scalable design rules based on a single parameter, λ, that characterizes the resolution of the process. λ is generally half of the minimum drawn transistor channel length. This length is the distance between the source and drain of a transistor and is set by the minimum width of a polysilicon wire. For example, a 180 nm process has a minimum polysilicon width (and hence transistor length) of 180 nm and uses design rules with λ = 90 nm. Lambda-based rules are necessarily conservative because they round up dimensions to an integer multiple of λ. Designers often describe a process by its feature size. Feature size refers to minimum transistor length, so λ is half the feature size. Unfortunately, below 180 nm, design rules have become so complex and process specific that scalable design rules are difficult to apply.

Transistor dimensions are often specified by their Width/Length (W/L) ratio. For example, the nMOS transistor in Fig. B formed where polysilicon crosses n-diffusion has a W/L of 4λ/2λ. In a 0.6 µm process, this corresponds to an actual width of 1.2 µm and a length of 0.6 µm. Such a minimum-width contacted transistor is often called a unit transistor.

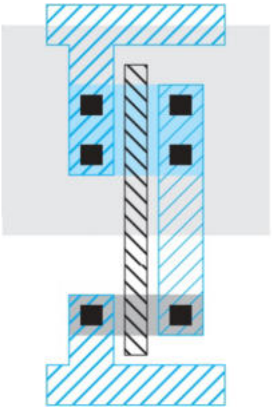


Figure B

pMOS transistors are often wider than nMOS transistors, because the mobility of holes in silicon is typically lower than that of electrons. This means that pMOS transistors provide less current than nMOS transistors of comparable size and hence are slower. The symbols μ_n and μ_p are used to distinguish mobility of electrons and of holes in nMOS and pMOS transistors, respectively. The mobility ratio μ_n/μ_p is typically 2–3; we will use 2 in this lab. That's why the pMOS in Fig. A has W/L of 8/2. Thus, the unit inverters are composed from an nMOS transistor of unit size and a pMOS transistor of twice unit width to achieve equal rise and fall resistance.

Rise time (t_r) is the time, during transition, when output switches from 10% to 90% of the maximum value. Fall time (t_f) is the time, during transition, when output switches from 90% to 10% of the maximum value.

Tools:

- DSCH2: for drawing the schematic and simulating
- Microwind2: for drawing the layout and simulating

Schematic:

After opening DSCH2, you will see an empty window, as shown in Fig. 1.

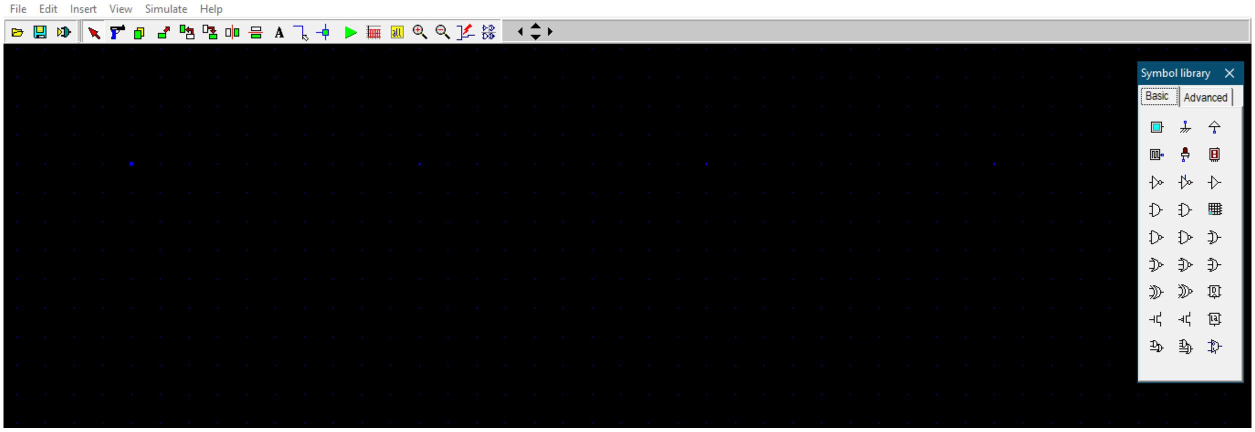
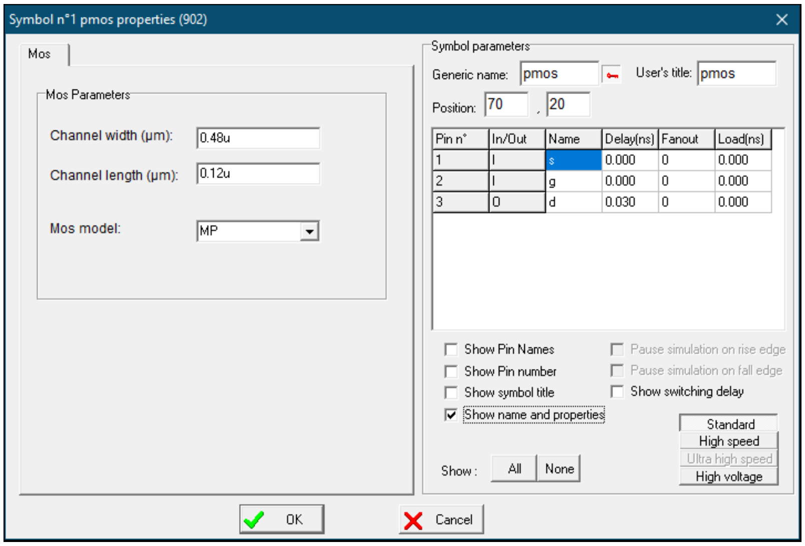


Figure 1

On the right-side of the window, there is a “Symbol library” panel. Drag and drop the following components to make the complete schematic of an inverter:

- n-channel MOS
- p-channel MOS
- Supply
- Ground
- Clock
- Light

Double click on any component to edit its properties. An example of pMOS properties is shown in Fig. 2 (after double clicking):



Symbol n°1 pmos properties (902)

Mos

Mos Parameters

Channel width (µm): 0.48u

Channel length (µm): 0.12u

Mos model: MP

Symbol parameters

Generic name: pmos User's title: pmos

Position: 70 , 20

Pin n°	In/Out	Name	Delay(ns)	Fanout	Load(ns)
1	I	s	0.000	0	0.000
2	I	g	0.000	0	0.000
3	O	d	0.030	0	0.000

☐ Show Pin Names
☐ Pause simulation on rise edge
☐ Show Pin number
☐ Pause simulation on fall edge
☐ Show symbol title
☐ Show switching delay
☑ Show name and properties

Figure 2

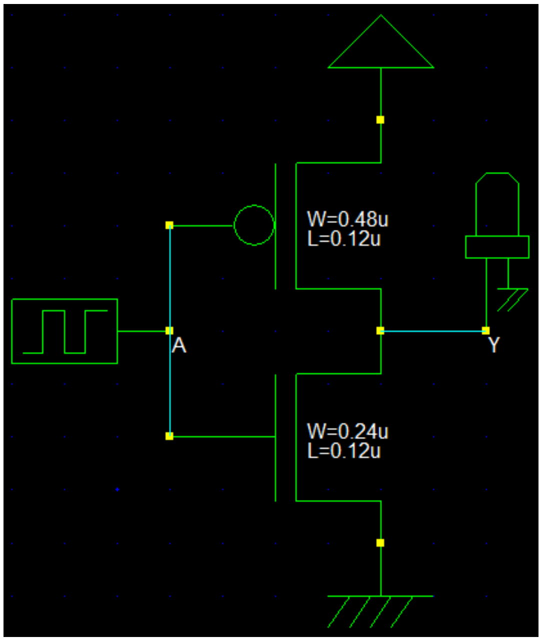


Figure 3

In Fig. 3, the full schematic of a CMOS inverter is shown.

Now click the "Run Simulation" button from top panel; and after several seconds, click the "Timing Diagram" button. You will see a waveform like Fig. 4.

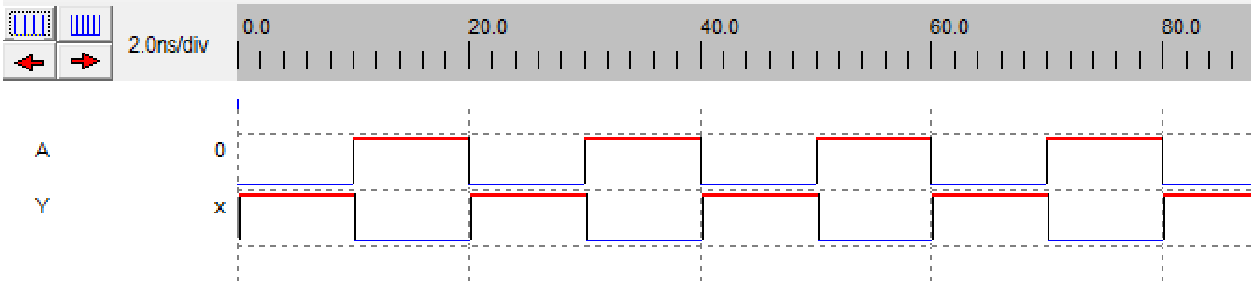


Figure 4

The waveform is generated as expected. The output (Y) is inverted from the input (A). When input is 0, output is 1. When input is 1, output is 0. But we can't calculate any delay from this plot.

Layout:

After opening Microwind2, you will see an empty window, as shown in Fig. 5.

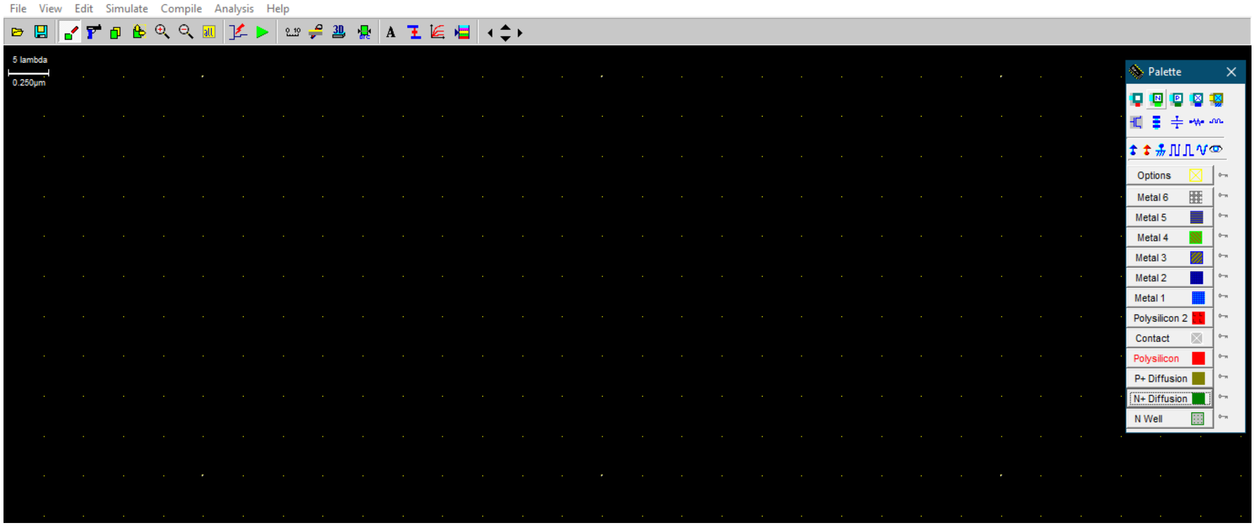


Figure 5

On the right-side of the opened window, there is a "Palette" panel, showing all the necessary layers for designing the layout of our CMOS inverter.

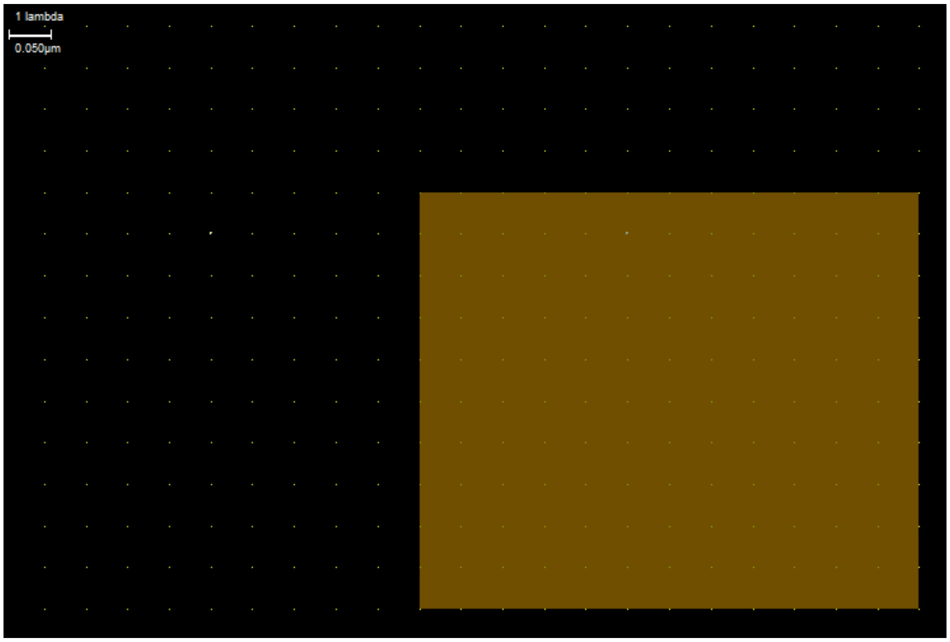


Figure 6

Firstly, we want to draw the Pull-Up Network (PUN). It consists of a single pMOS transistor. Use "Zoom In" button from the top panel until the minor-grid spacing becomes 1λ. Select "P+ Diffusion" from the "Palette" panel, then draw a box with length of 12λ and width of 10λ. It is shown in Fig. 6. In this figure, λ = 50 nm is used. This diffusion box width is chosen 10λ instead of

8λ, because we want to place two contacts later on each Metal1-Diffusion crossing region without violating any design rule (‘default.rul’ file) of Microwind2. If the minimum contact spacing was stated 2λ in that ‘rul’ file, we could have drawn the diffusion box width as 8λ (which would be same as the pMOS width stated in “Theory” section of this experiment).

Now select “Polysilicon” from the “Palette” panel. Draw a box of polysilicon of 2λ length and 16λ width, as shown in Fig. 7. Don’t forget to run DRC by clicking the “Design Rule Checker” button from the top panel.

The polysilicon is dividing the “P+ Diffusion” into two parts: Source and Drain. The left-side of the polysilicon is assumed “Source” here, and the right-side is “Drain”.

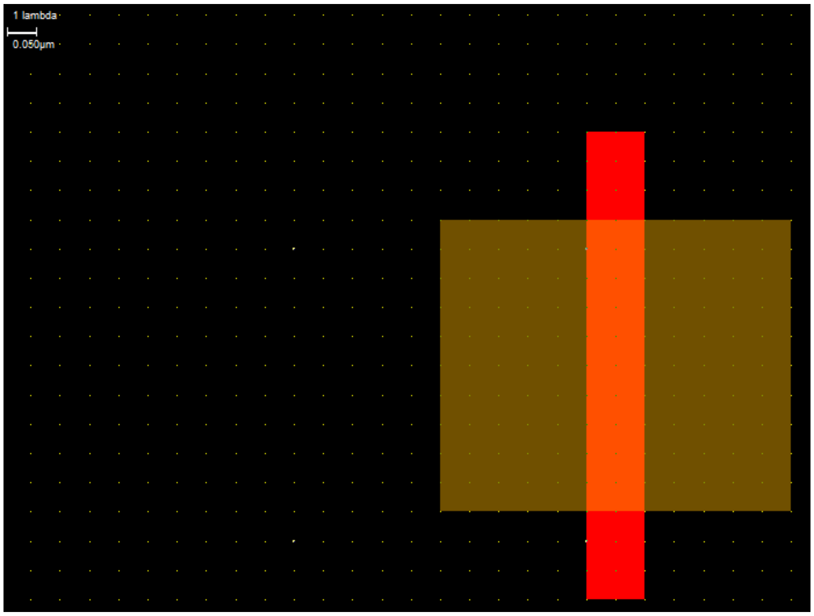


Figure 7

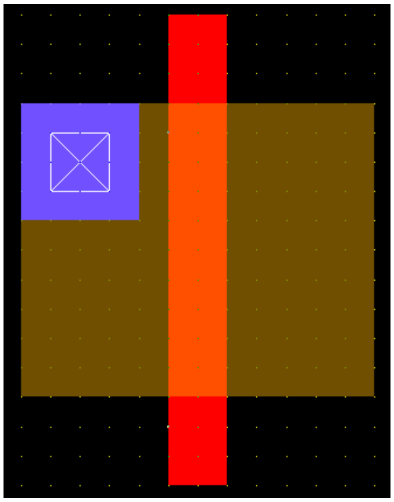


Figure 8

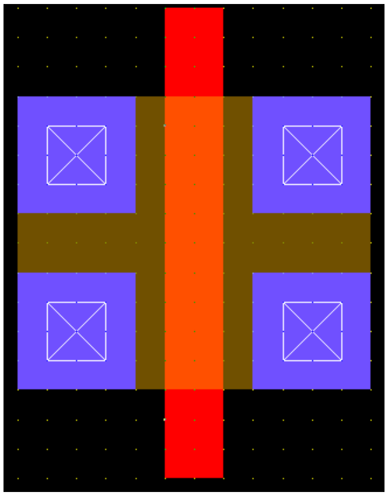
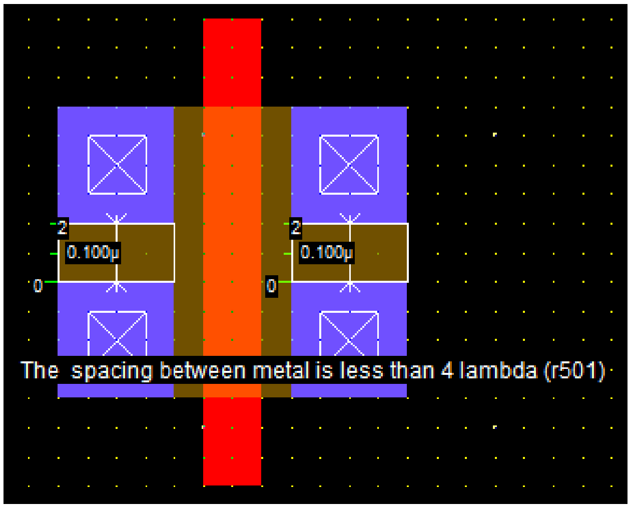


Figure 9

Now we need to place contacts. Select “Contact P+ diff/Metal1” from the “Palette” panel, then place it on the Source, as shown in Fig. 8. Again, run DRC.

Subsequently, place three more contacts by selecting “Contact P+ diff/Metal1” from the “Palette” panel, as shown in Fig. 9. Now, run DRC again. After running DRC, error will appear, as shown in Fig. 10.



The spacing between metal is less than 4 lambda (r501)

Figure 10

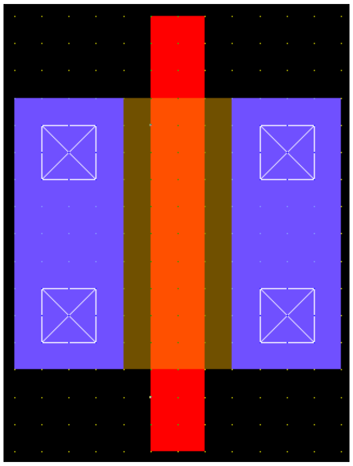


Figure 11

To solve the error in Fig. 10, join the Metal1 lines using "Stretch, Move" button, as shown in Fig. 11. Now the error will not show if you run DRC.

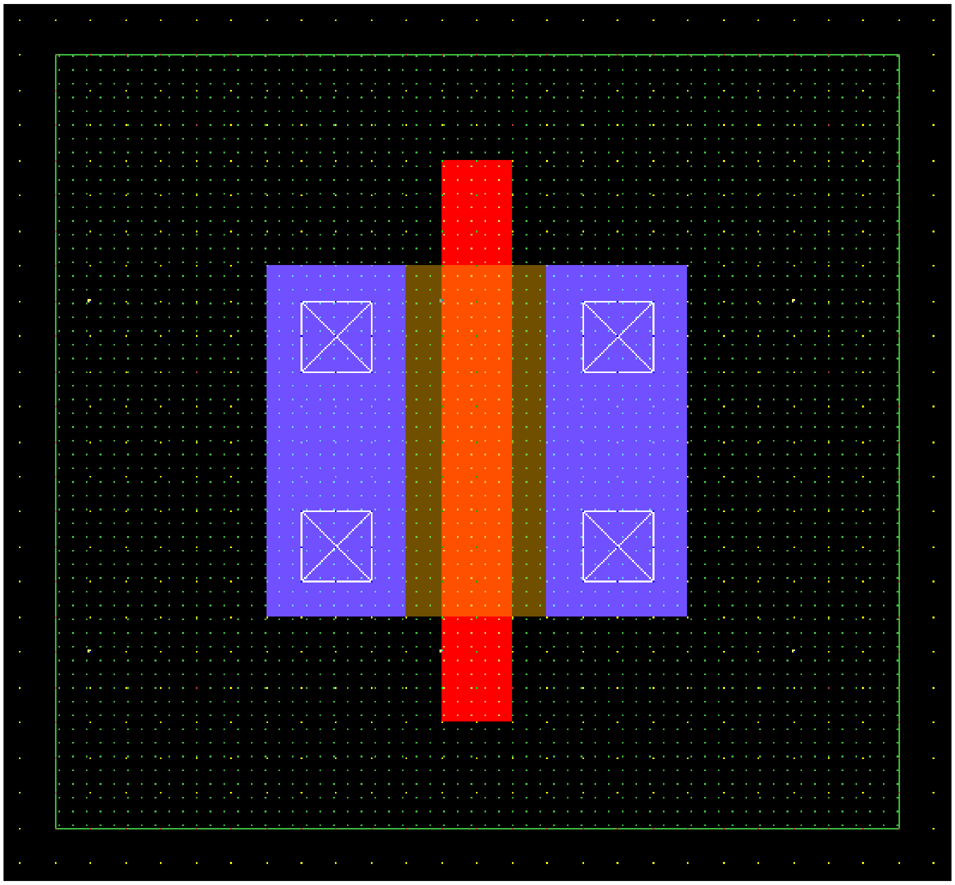


Figure 12

Now, draw a box of "N Well" by selecting it from the "Palette" panel. It is depicted in Fig. 12. Run DRC. If any error shows, use "Stretch, Move" button to clear the error.

The pMOS transistor is now completed. We now need to draw the Pull-Down Network (PDN) of the inverter, which consists of a single nMOS.

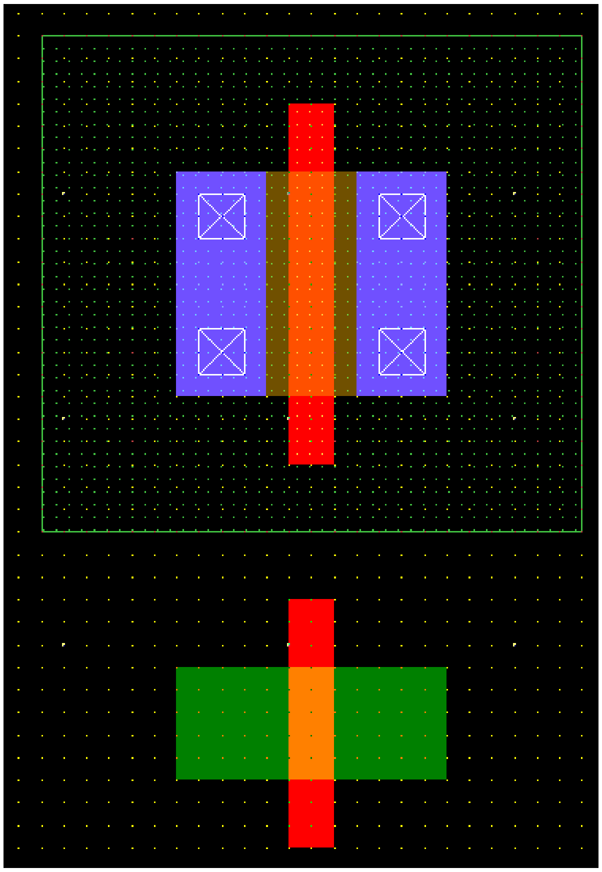


Figure 13

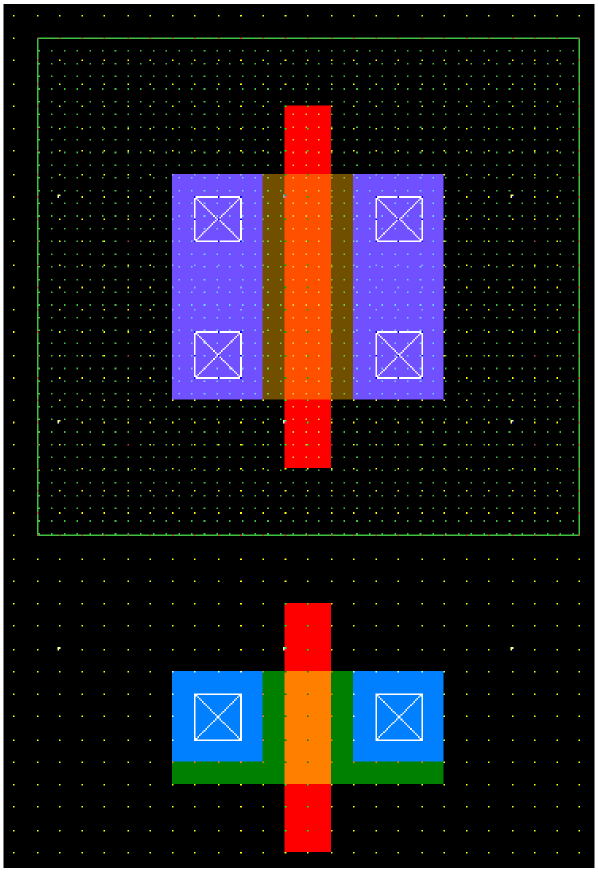


Figure 14

Draw a box of “N+ Diffusion” with length of 12λ and width of 5λ. Then draw polysilicon of 2λ length and 11λ width. It is depicted in Fig. 13. The polysilicon is dividing the “N+ Diffusion” into two parts: Source and Drain. The left-side of the polysilicon is assumed “Source” here, and the right-side is “Drain”.

Now we will place two contacts by selecting “Contact N+ diff/Metal1” from the “Palette” panel. This is depicted in Fig. 14. Now the nMOS transistor is also completed. Don’t forget to run DRC.

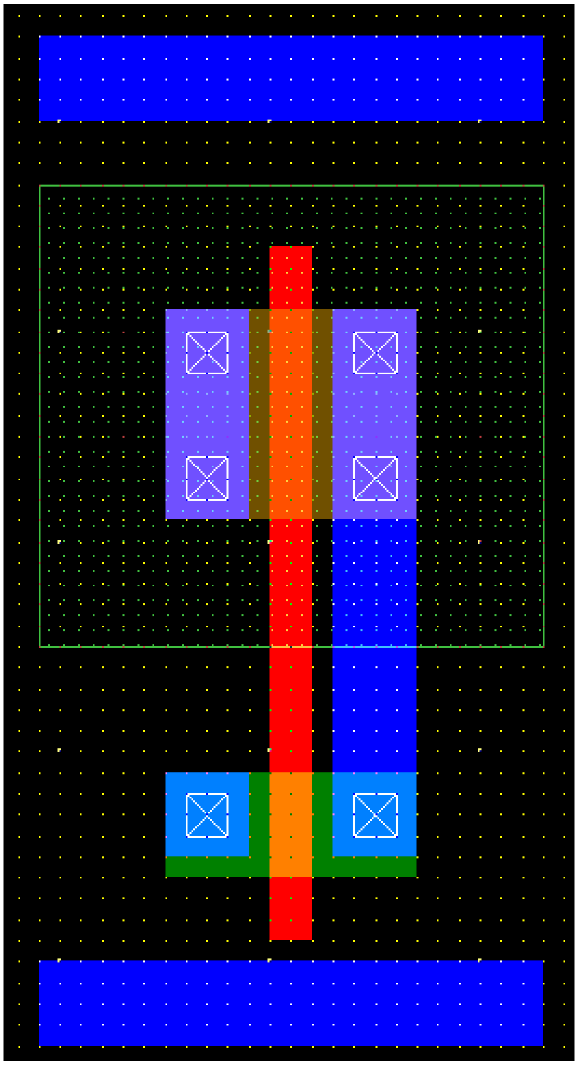


Figure 15

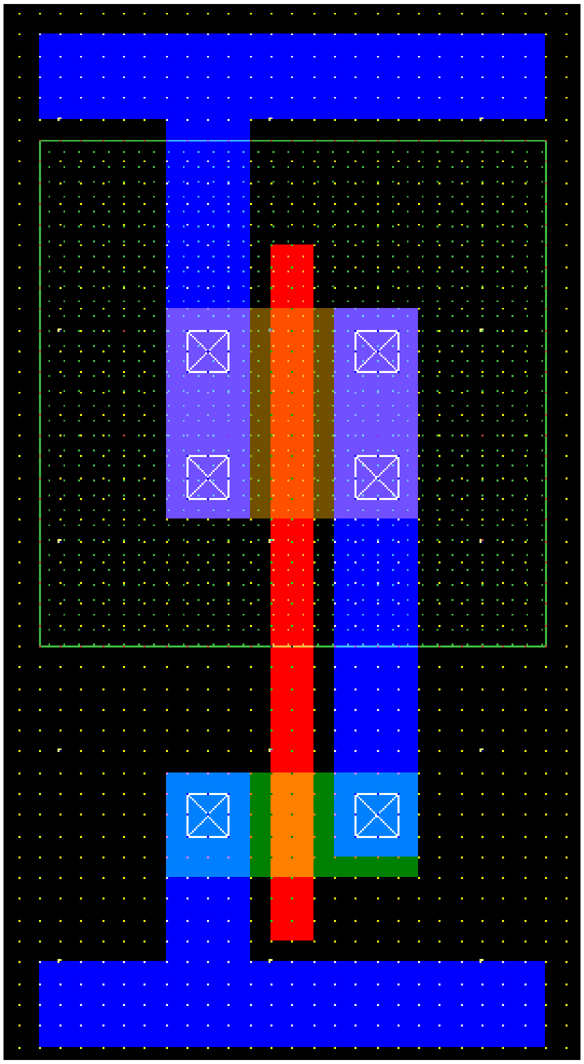


Figure 16

Now we need to connect the drain terminal of pMOS and the drain terminal of nMOS. Use Metal1 line to short them. You can also use “Stretch, Move” button to extend one Metal1 line to another. For CMOS inverter, both transistors’ gates are shorted. So, we short both polysilicon boxes. Then we need Supply rails (VDD and GND). We will use Metal1 line as these rails. It is depicted in Fig. 15. The top Metal1 rail is VDD, and the bottom Metal1 rail is GND. Again, run DRC.

Now we need to connect the source terminal of pMOS to VDD, and source terminal of nMOS to GND. We also need to add Well-Tap and Substrate-Tap in next step. For placing the Well-Tap, increase the width of “N Well” using “Stretch, Move” button. This is shown in Fig. 16. You can also increase its length instead of width, depending on your Well-Tap location. Then run DRC.

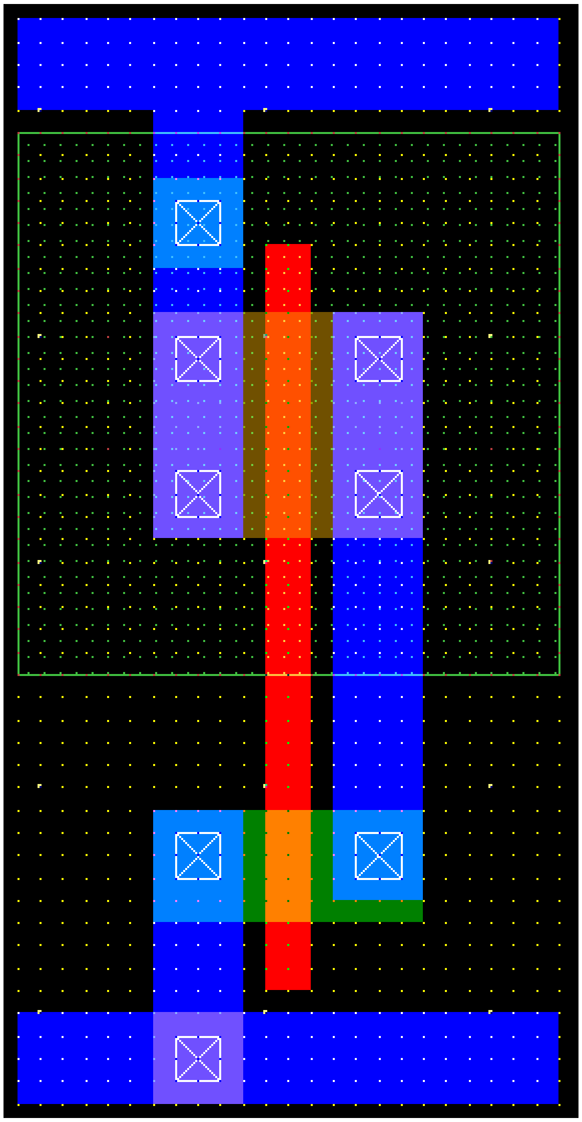


Figure 17

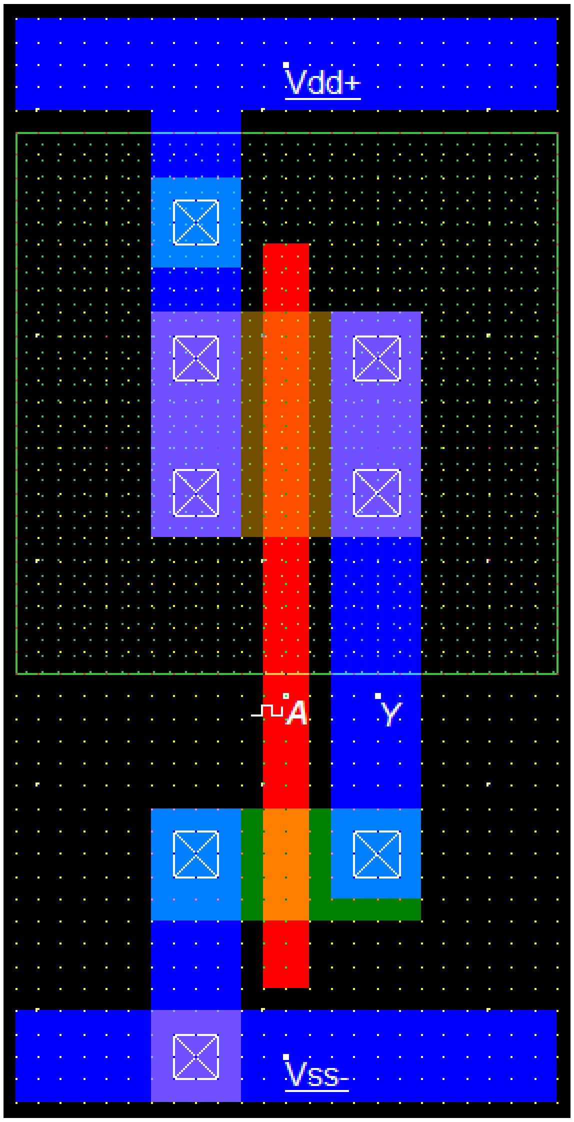
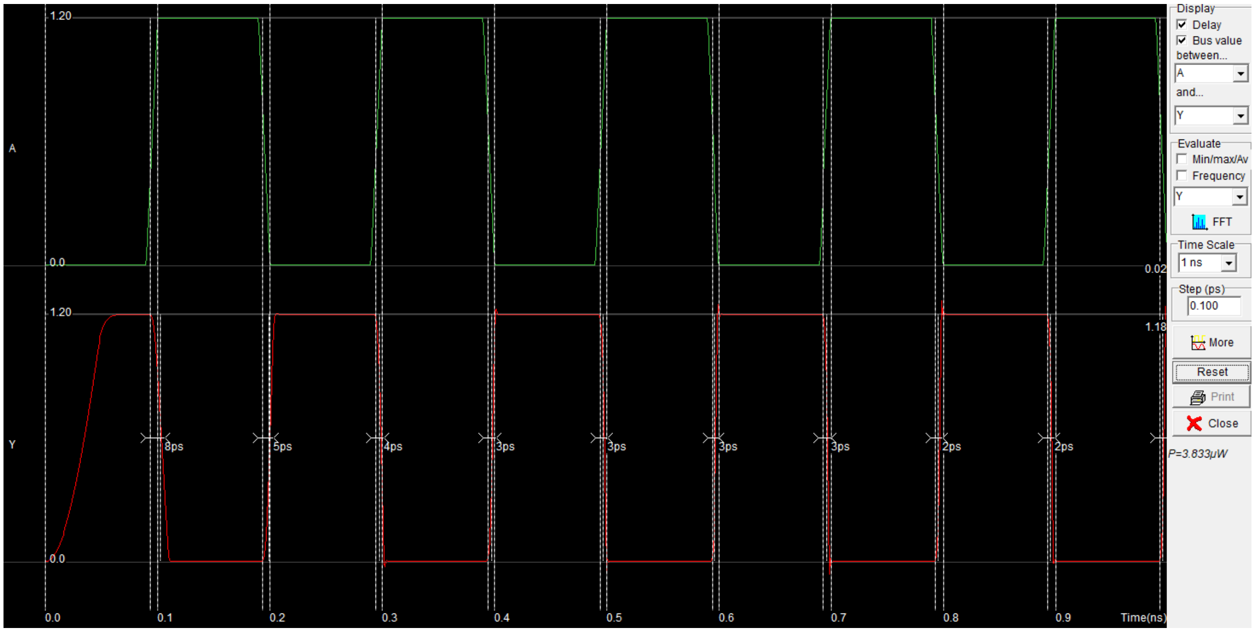


Figure 18

The Well-Tap is made by placing a “Contact N+ diff/Metal1”, and the Substrate-Tap is made by placing a “Contact P+ diff/Metal1”. It is shown in Fig. 17. Don’t forget to run DRC.

Now we apply signals to the appropriate terminals. From the “Palette” panel, select “Vdd Supply” and place it on the top Metal1 rail. Similarly, place “Ground” on the bottom Metal1 rail. It will show as “VDD+” and “VSS-”, respectively. We also need to mark input and output terminals. From the “Palette” panel, select “Add a clock” and place it on the polysilicon. You can change clock properties according to your wish. Label it as ‘A’. Similarly, select “Visible node” and place it on the shorted drain terminal Metal1. Label it as ‘Y’. The final layout is depicted in Fig. 18.



☑ Delay
☑ Bus value
between... A
and... Y
□ Min/max/Av
□ Frequency
Y
Time Scale 1 ns
Step (ps) 0.100
P=3.833µW

Figure 19

Now we need to see the waveform. Click the “Run Simulation” button. A waveform will appear, as shown in Fig. 19. It is clear from the plot that, the output (Y) is inverted from the input (A).

You can change the time-scale by choosing from “Time Scale” dropdown menu and then clicking “Reset”. Rise time (t_r) and fall time (t_f) can be easily obtained from this waveform. If you increase time-scale, you will notice that both t_r & t_f are almost equal after some cycles since we have chosen the transistors’ width carefully to maintain equal rise & fall resistance.

EEE 330: VLSI 1 Lab

Experiment – 3

Schematic & Layout of CMOS NAND2 & NOR2 with Equal Rise & Fall Resistance

Theory:

A NAND gate (NOT-AND) is a logic gate which produces an output which is false only if all its inputs are true; thus, its output is complement to that of an AND operator. The NOR gate (NOT-OR) is a logic gate that which produces an output which is false if any of its inputs is true; thus, its output is complement to that of an OR operator.

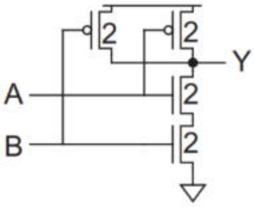


Figure A

Fig. A shows a 2-input CMOS NAND gate. It consists of two series nMOS transistors between Y and GND and two parallel pMOS transistors between Y and VDD. If either input A or B is 0, at least one of the nMOS transistors will be OFF, breaking the path from Y to GND. But at least one of the pMOS transistors will be ON, creating a path from Y to VDD. Hence, the output Y will be 1.

Table 1

A	B	Pull-Up Network	Pull-Down Network	Y
0	**0**	ON	OFF	**1**
0	**1**	ON	OFF	**1**
1	**0**	ON	OFF	**1**
1	**1**	OFF	ON	**0**

If both inputs are 1, both of the nMOS transistors will be ON and both of the pMOS transistors will be OFF. Hence, the output will be 0. The truth table is given in Table 1 above.

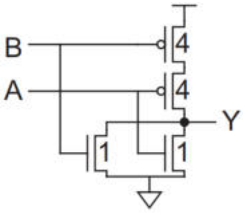


Figure B

A 2-input NOR gate is shown in Fig. B. The nMOS transistors are in parallel to pull the output low when either input is high. The pMOS transistors are in series to pull the output high when both inputs are low, as indicated in Table 2 below.

Table 2

A	B	Pull-Up Network	Pull-Down Network	Y
0	**0**	ON	OFF	**1**
0	**1**	OFF	ON	**0**
1	**0**	OFF	ON	**0**
1	**1**	OFF	ON	**0**

Any Boolean function can be implemented by using a combination of either only NAND gates or only NOR gates. This property is called "functional completeness".

Both Fig. A and Fig. B are annotated with shorthand width for each transistor, specifying multiples of unit width and assuming minimum length. The widths are chosen appropriately to achieve effective rise & fall resistance equal to that of a unit inverter.

Tools:

- DSCH2: for drawing the schematic and simulating
- Microwind2: for drawing the layout and simulating

Schematic:

At first, we want to draw the schematic of CMOS NAND2 gate. After opening DSCH2, you will see an empty window. On the right-side of the window, there is a "Symbol library" panel. Drag and drop the following components to make the complete schematic of CMOS NAND2:

- n-channel MOS
- p-channel MOS
- Supply
- Ground
- Clock
- Light

In Fig. 1, the full schematic of a CMOS NAND2 is shown.

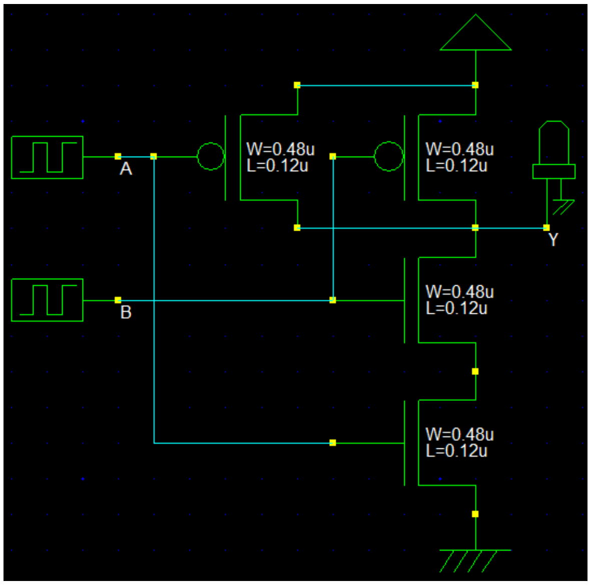


Figure 1

Now click the "Run Simulation" button from top panel; and after several seconds, click the "Timing Diagram" button. You will see a waveform like Fig. 2.

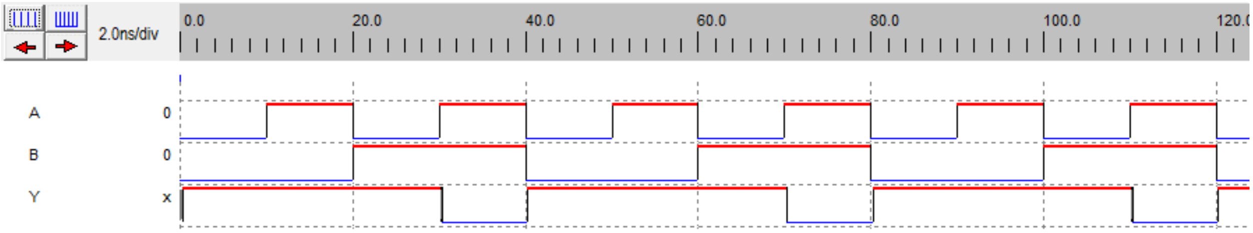


Figure 2

The waveform is generated as expected (NAND2 operator). When both inputs are 1, output is 0. On the contrary, when any of the inputs is 0, output is 1. But we can't calculate any delay from this plot.

Now we will draw a CMOS NOR2 gate. In Fig. 3, the full schematic is shown.

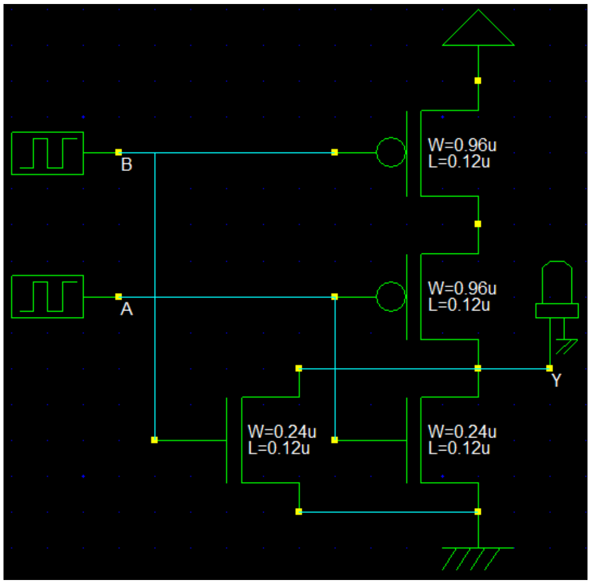


Figure 3

Now click the “Run Simulation” button from top panel; and after several seconds, click the “Timing Diagram” button. You will see a waveform like Fig. 4.

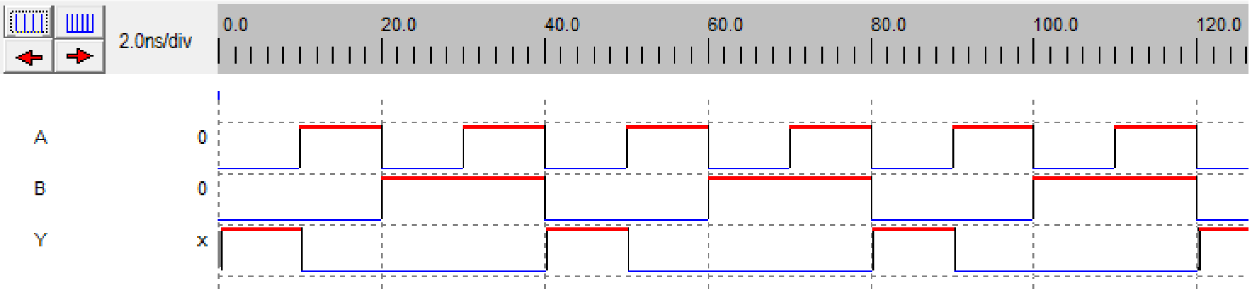
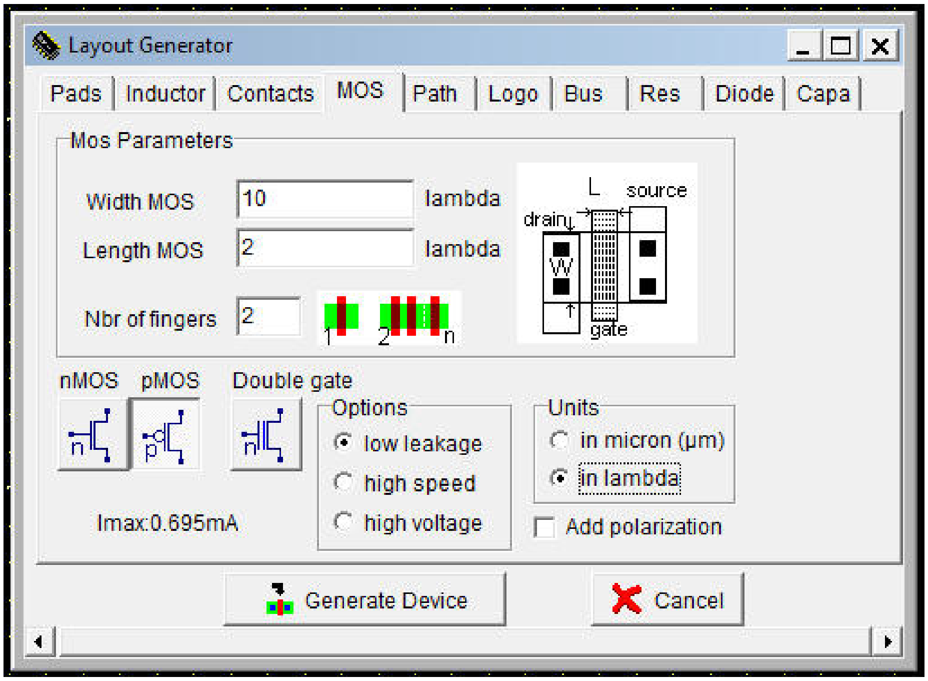


Figure 4

The waveform is generated as expected (NOR2 operator). When both inputs are 0, output is 1. On the contrary, when any of the inputs is 1, output is 0. But we can’t calculate any delay from this plot.

Layout:

At first, we want to draw the layout of CMOS NAND2 gate. After opening Microwind2, you will see an empty window. On the right-side of the opened window, there is a "Palette" panel. Click on the "MOS Generator". A window will appear. Select the "MOS" tab from there. Choose the options like Fig. 5 below to complete the pull-up network (PUN).



Layout Generator

Mos Parameters

Width MOS 10 lambda

Length MOS 2 lambda

Nbr of fingers 2

nMOS pMOS Double gate

Options
● low leakage
○ high speed
○ high voltage

Units
○ in micron (µm)
● in lambda

□ Add polarization

Imax:0.695mA

Figure 5

Select "pMOS", Units "in lambda", Width '10' lambda, Length '2' lambda, Number of fingers '2'. Width is chosen 10λ instead of 8λ, because we want to have two contacts on each Metal1-Diffusion crossing region of PUN without violating any design rule of Microwind2. Then click "Generate Device" and click on anywhere on the screen. The PUN will appear like Fig. 6.

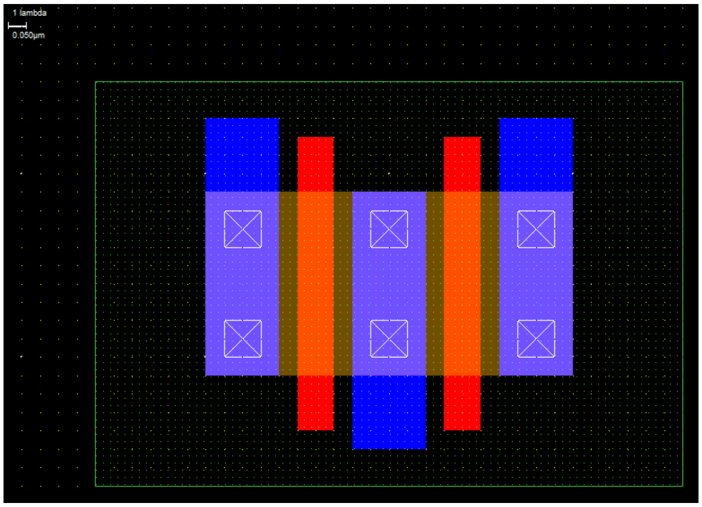


Figure 6

Again, click on the “MOS Generator”. Then select “nMOS”, Units “in lambda”, Width ‘10’ lambda, Length ‘2’ lambda, Number of fingers ‘2’. Then click “Generate Device” and click on anywhere on the screen below the PUN. The pull-down network (PDN) will appear like Fig. 7.

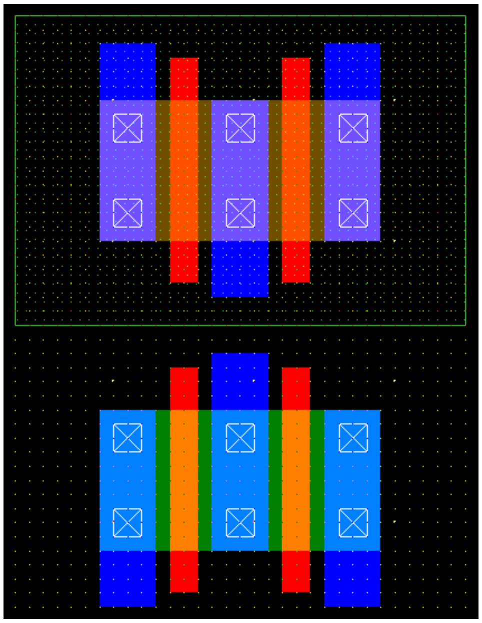


Figure 7

Align them using “Stretch, Move” button, if necessary. Run DRC. Then join the polysilicon of PUN and PDN. The shorted polysilicon fingers will be the inputs (A, B). Now remove metal & contact from the middle of PDN, since both nMOS of PDN are in series. Then form the output (Y) terminal using Metal1 by joining drain terminal of upper nMOS of PDN (that is closer to output) with shorted drain of both pMOS of PUN. The layout will look like Fig. 8 below.

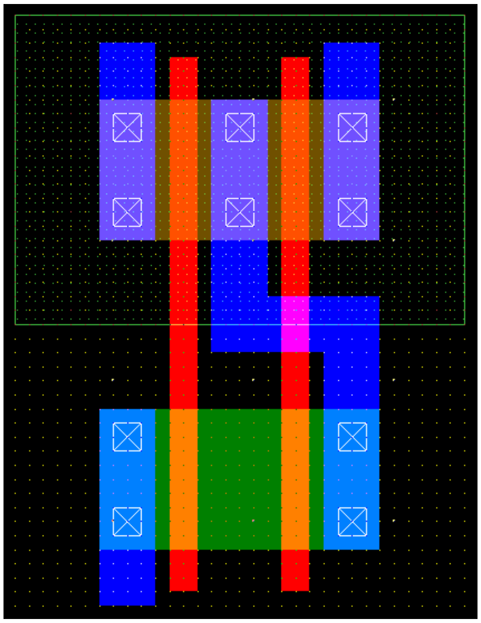


Figure 8

Now draw the VDD rail and GND rail. Connect the source of both pMOS of PUN with VDD rail using Metal1. Similarly, connect the source of bottom nMOS of PDN (that is farther from output) with GND rail. Place Well-Tap using a “Contact N+ diff/Metal1”, and the Substrate-Tap using a “Contact P+ diff/Metal1”. Don’t forget to run DRC. Extend N-Well using the “Stretch, Move” button, if necessary.

Then place appropriate input (clock) signals on polysilicon fingers, and VDD and GND on rails. Lastly, place a “Visible node” on output (Y) terminal. The final layout of CMOS NAND2 is depicted in Fig. 9.

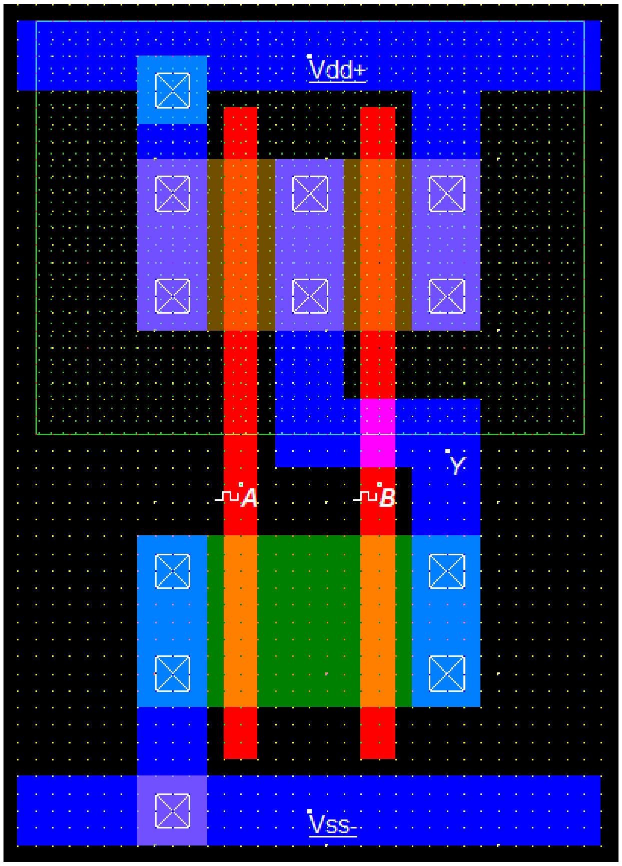


Figure 9

Now we need to see the waveform. Click the “Run Simulation” button. A waveform will appear, as shown in Fig. 10. It is clear from the plot that, when both inputs are 1, output is 0. On the contrary, when any of the inputs is 0, output is 1. Thus, it is acting like a 2-input NAND operator.

You can change the time-scale by choosing from “Time Scale” dropdown menu and then clicking “Reset”. Rise time (t_r) and fall time (t_f) can be easily obtained using dropdown menu on the top-right corner of the waveform window. If you increase time-scale, you will notice that both t_r & t_f are almost equal after some cycles since we have chosen the transistors’ width carefully to maintain effective rise & fall resistance equal to that of a unit inverter.

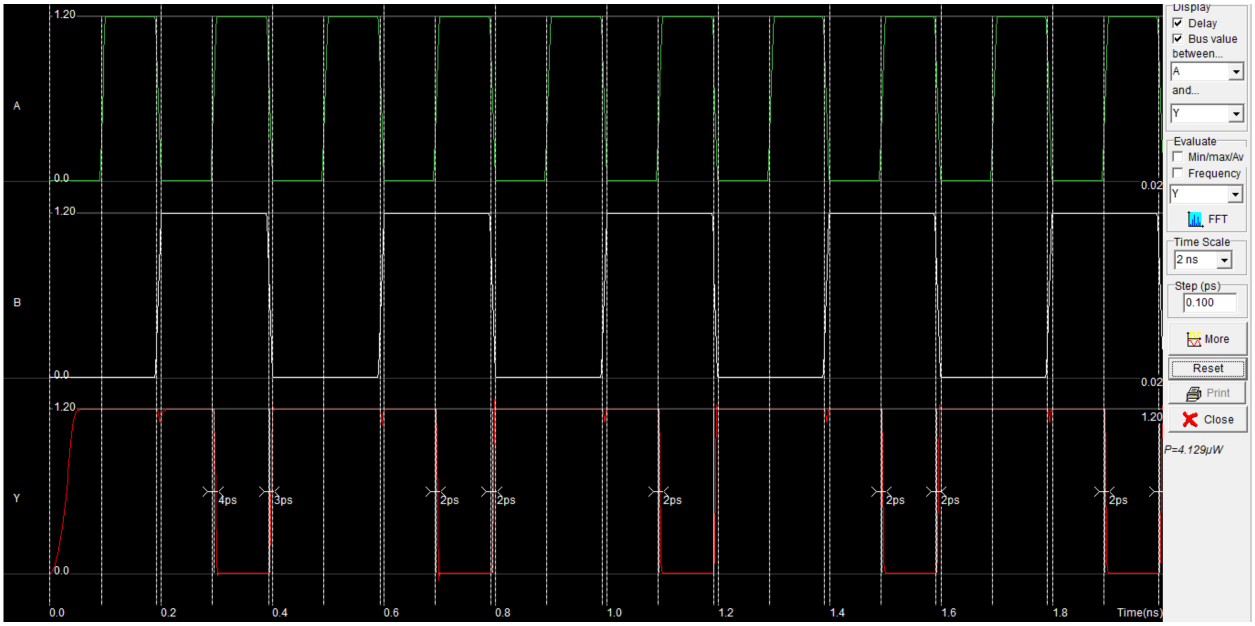


Figure 10

Now we want to draw the layout of CMOS NOR2 gate. Open a new window. Click on the “MOS Generator”. A window will appear. Select the “MOS” tab from there. Select “pMOS”, Units “in lambda”, Width ‘22’ lambda, Length ‘2’ lambda, Number of fingers ‘2’. Width is chosen 22λ instead of 16λ, because we want to have four contacts on each Metal1-Diffusion crossing region of PUN without violating any design rule of Microwind2. Then click “Generate Device” and click on anywhere on the screen. The PUN will appear.

Again, click on the “MOS Generator”. Then select “nMOS”, Units “in lambda”, Width ‘5’ lambda, Length ‘2’ lambda, Number of fingers ‘2’. Then click “Generate Device” and click on anywhere on the screen below the PUN. The PDN will appear. Align the PDN using the “Stretch, Move” button. Run DRC. It will look like Fig. 11 below.

Then join the polysilicon of PUN and PDN. The shorted polysilicon fingers will be the inputs (A, B). Now remove metal & contact from the middle of PUN, since both pMOS of PUN are in series. Then form the output (Y) terminal using Metal1 by joining drain terminal of bottom pMOS of PUN (that is closer to output) with shorted drain of both nMOS of PDN.

Now draw the VDD rail and GND rail. Connect the source of upper pMOS of PUN (that is farther from output) with VDD rail using Metal1. Similarly, connect the source of both nMOS of PDN with GND rail. Place Well-Tap using a “Contact N+ diff/Metal1”, and the Substrate-Tap using a “Contact P+ diff/Metal1”. Don’t forget to run DRC. Extend N-Well using the “Stretch, Move” button, if necessary.

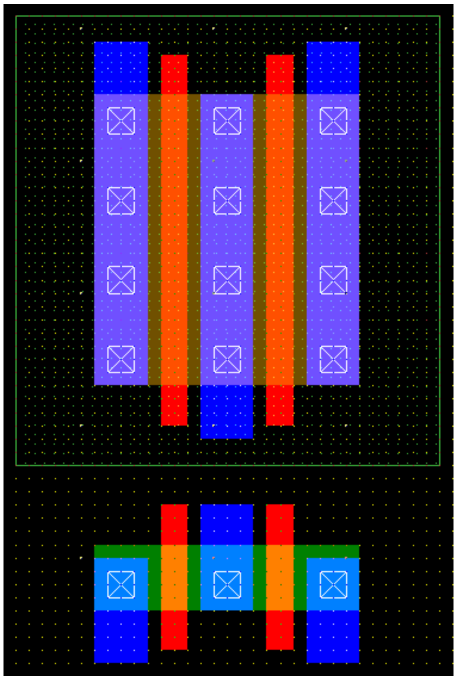


Figure 11

Then place appropriate input (clock) signals on polysilicon fingers, and VDD and GND on rails. Lastly, place a "Visible node" on output (Y) terminal. The final layout of CMOS NOR2 is depicted in Fig. 12.

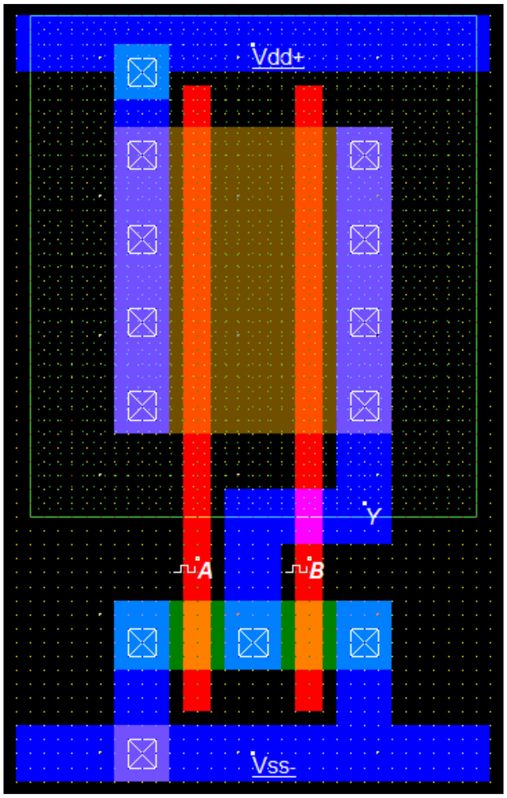
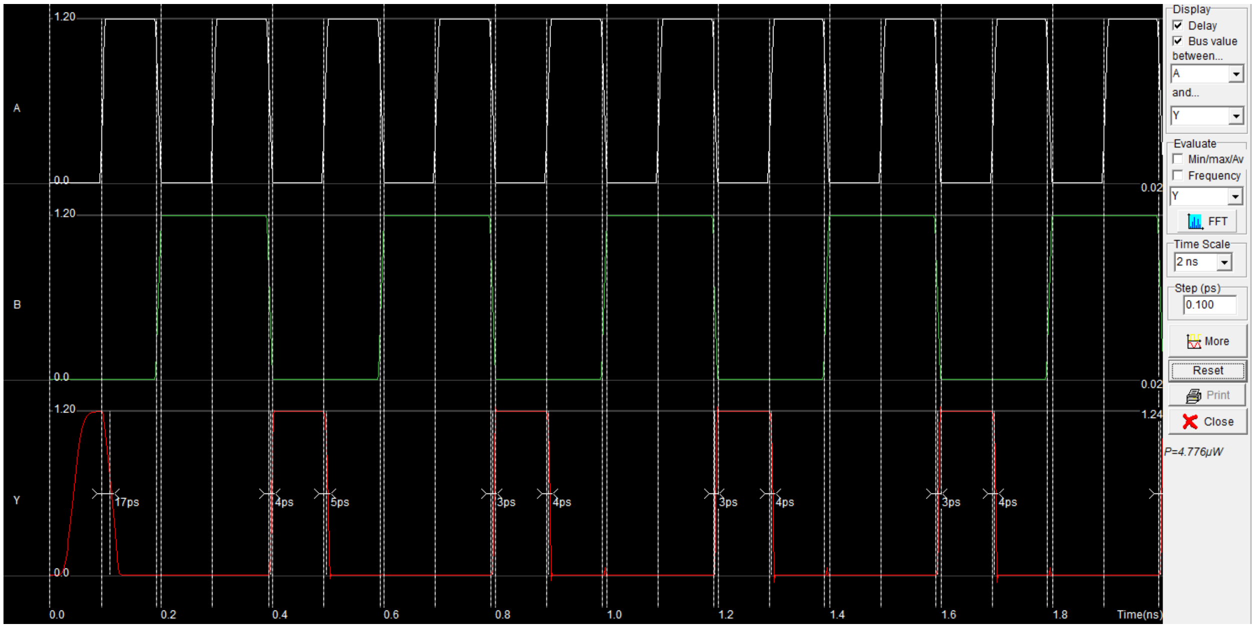


Figure 12

Now we need to see the waveform. Click the “Run Simulation” button. A waveform will appear, as shown in Fig. 13. It is clear from the plot that, when both inputs are 0, output is 1. On the contrary, when any of the inputs is 1, output is 0. Thus, it is acting like a 2-input NOR operator.



☑ Delay
☑ Bus value between... A and... Y
□ Min/max/Av
□ Frequency
Y
Time Scale: 2 ns
Step (ps): 0.100
P=4.776µW

Figure 13

You can change the time-scale by choosing from “Time Scale” dropdown menu and then clicking “Reset”. Rise time (t_r) and fall time (t_f) can be easily obtained using dropdown menu on the top-right corner of the waveform window. If you increase time-scale, you will notice that both t_r & t_f are almost equal after some cycles since we have chosen the transistors’ width carefully to maintain effective rise & fall resistance equal to that of a unit inverter.

Experiment – 4

Schematic & Layout of CMOS NAND3 & NOR3 with Equal Rise & Fall Resistance

Theory:

A NAND gate (NOT-AND) is a logic gate which produces an output which is false only if all its inputs are true; thus, its output is complement to that of an AND operator. The NOR gate (NOT-OR) is a logic gate that which produces an output which is false if any of its inputs is true; thus, its output is complement to that of an OR operator.

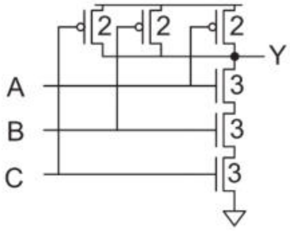


Figure A

Fig. A shows a 3-input CMOS NAND gate. It consists of three series nMOS transistors between Y and GND and three parallel pMOS transistors between Y and VDD. If either input A, B or C is 0, at least one of the nMOS transistors will be OFF, breaking the path from Y to GND. But at least one of the pMOS transistors will be ON, creating a path from Y to VDD. Hence, the output Y will be 1.

Table 1

A	B	C	Pull-Up Network	Pull-Down Network	Y
0	**0**	**0**	ON	OFF	**1**
0	**0**	**1**	ON	OFF	**1**
0	**1**	**0**	ON	OFF	**1**
0	**1**	**1**	ON	OFF	**1**
1	**0**	**0**	ON	OFF	**1**
1	**0**	**1**	ON	OFF	**1**
1	**1**	**0**	ON	OFF	**1**
1	**1**	**1**	OFF	ON	**0**

If all inputs are 1, all of the nMOS transistors will be ON and all of the pMOS transistors will be OFF. Hence, the output will be 0. The truth table is given in Table 1 above.

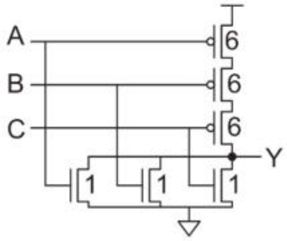


Figure B

A 3-input NOR gate is shown in Fig. B. The nMOS transistors are in parallel to pull the output low when any of the inputs is high. The pMOS transistors are in series to pull the output high when all the inputs are low, as indicated in Table 2 below.

Table 2

A	B	C	Pull-Up Network	Pull-Down Network	Y
0	**0**	**0**	ON	OFF	**1**
0	**0**	**1**	OFF	ON	**0**
0	**1**	**0**	OFF	ON	**0**
0	**1**	**1**	OFF	ON	**0**
1	**0**	**0**	OFF	ON	**0**
1	**0**	**1**	OFF	ON	**0**
1	**1**	**0**	OFF	ON	**0**
1	**1**	**1**	OFF	ON	**0**

Any Boolean function can be implemented by using a combination of either only NAND gates or only NOR gates. This property is called “functional completeness”.

Both Fig. A and Fig. B are annotated with shorthand width for each transistor, specifying multiples of unit width and assuming minimum length. The widths are chosen appropriately to achieve effective rise & fall resistance equal to that of a unit inverter.

Tools:

- DSCH2: for drawing the schematic and simulating
- Microwind2: for drawing the layout and simulating

Schematic:

At first, we want to draw the schematic of CMOS NAND3 gate. After opening DSCH2, you will see an empty window. On the right-side of the window, there is a “Symbol library” panel. Drag and drop the following components to make the complete schematic of CMOS NAND3:

- n-channel MOS
- p-channel MOS
- Supply
- Ground
- Clock
- Light

In Fig. 1, the full schematic of a CMOS NAND3 is shown.

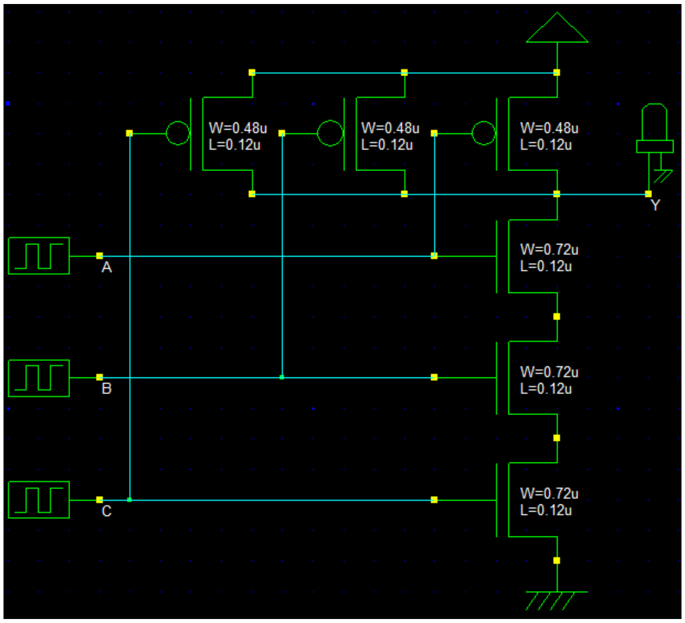


Figure 1

Now click the “Run Simulation” button from top panel; and after several seconds, click the “Timing Diagram” button. You will see a waveform like Fig. 2.

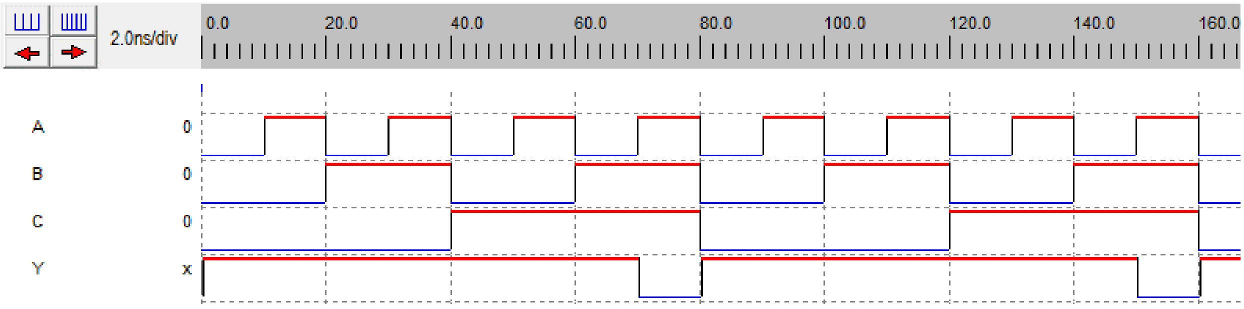


Figure 2

The waveform is generated as expected (NAND3 operator). When all three inputs are 1, output is 0. On the contrary, when any of the inputs is 0, output is 1. But we can't calculate any delay from this plot.

Now we will draw a CMOS NOR3 gate. In Fig. 3, the full schematic is shown.

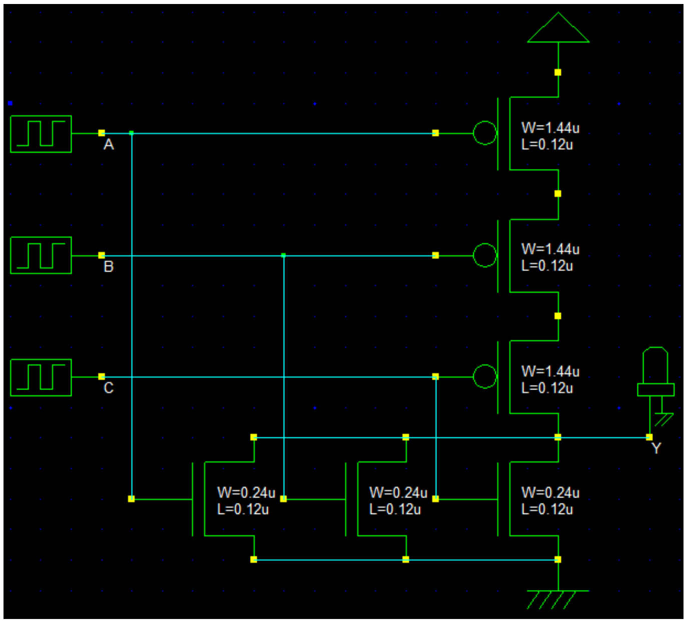


Figure 3

Now click the "Run Simulation" button from top panel; and after several seconds, click the "Timing Diagram" button. You will see a waveform like Fig. 4.

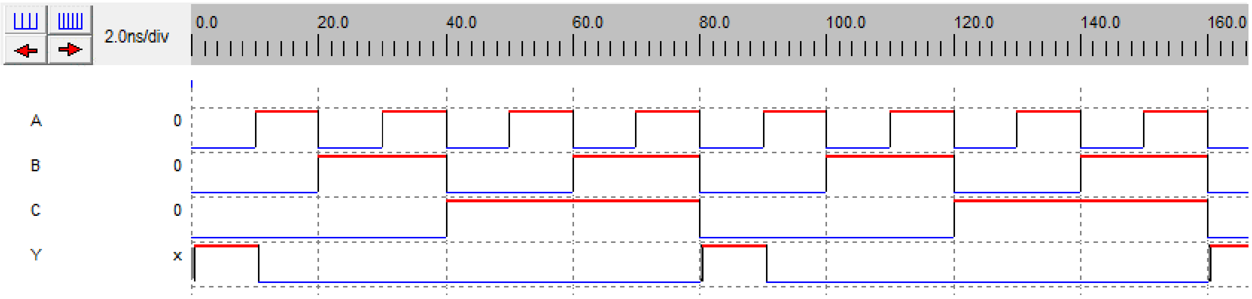


Figure 4

The waveform is generated as expected (NOR3 operator). When all three inputs are 0, output is 1. On the contrary, when any of the inputs is 1, output is 0. But we can't calculate any delay from this plot.

Layout:

At first, we want to draw the layout of CMOS NAND3 gate. After opening Microwind2, you will see an empty window. On the right-side of the opened window, there is a "Palette" panel. Click on the "MOS Generator". A window will appear. Select the "MOS" tab from there.

Select "pMOS", Units "in lambda", Width '10' lambda, Length '2' lambda, Number of fingers '3'. Width is chosen 10λ instead of 8λ, because we want to have two contacts on each Metal1-Diffusion crossing region of pull-up network (PUN) without violating any design rule of Microwind2. Then click "Generate Device" and click on anywhere on the screen. The PUN will appear.

Again, click on the "MOS Generator". Then select "nMOS", Units "in lambda", Width '16' lambda, Length '2' lambda, Number of fingers '3'. Width is chosen 16λ instead of 12λ, because we want to have three contacts on each Metal1-Diffusion crossing region without violating any design rule of Microwind2. Then click "Generate Device" and click on anywhere on the screen below the PUN. The pull-down network (PDN) will appear like Fig. 5.

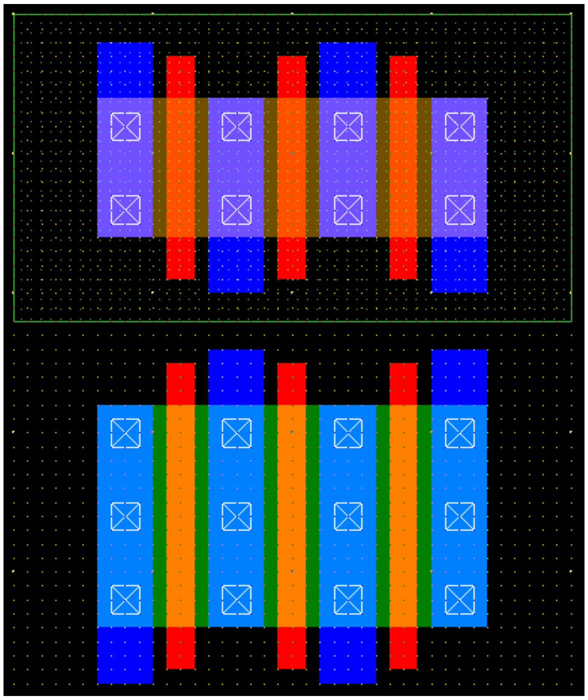


Figure 5

Align them using "Stretch, Move" button, if necessary. Run DRC. Then join the polysilicon of PUN and PDN. The shorted polysilicon fingers will be the inputs (A, B, C). Now remove metal & contact from the middle of PDN, since all three nMOS of PDN are in series. Then form the output (Y) terminal using Metal1 by joining drain terminal of upper nMOS of PDN (that is closer to output) with shorted drain of all three pMOS of PUN. The layout will look like Fig. 6 below.

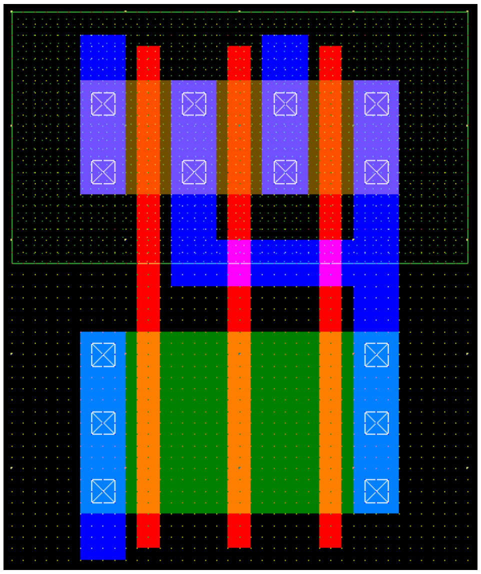


Figure 6

Now draw the VDD rail and GND rail. Connect the source of all three pMOS of PUN with VDD rail using Metal1. Similarly, connect the source of bottom nMOS of PDN (that is farther from output) with GND rail. Place Well-Tap using a “Contact N+ diff/Metal1”, and the Substrate-Tap using a “Contact P+ diff/Metal1”. Don’t forget to run DRC. Extend N-Well using the “Stretch, Move” button, if necessary.

Then place appropriate input (clock) signals on polysilicon fingers, and VDD and GND on rails. Lastly, place a “Visible node” on output (Y) terminal. The final layout of CMOS NAND3 is depicted in Fig. 7.

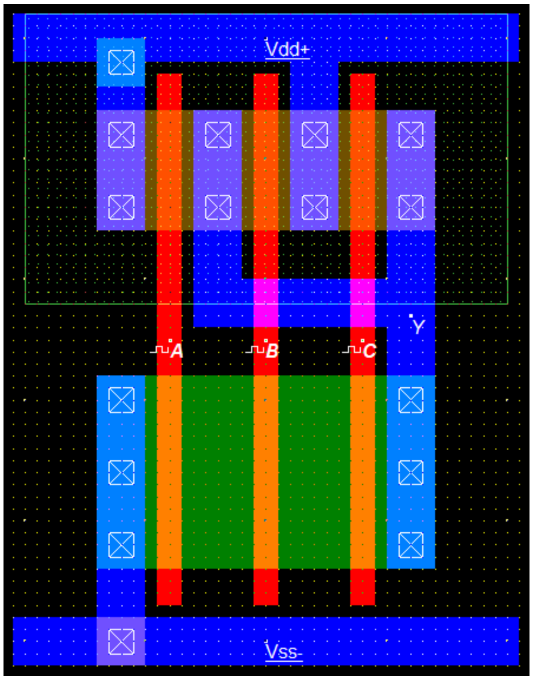
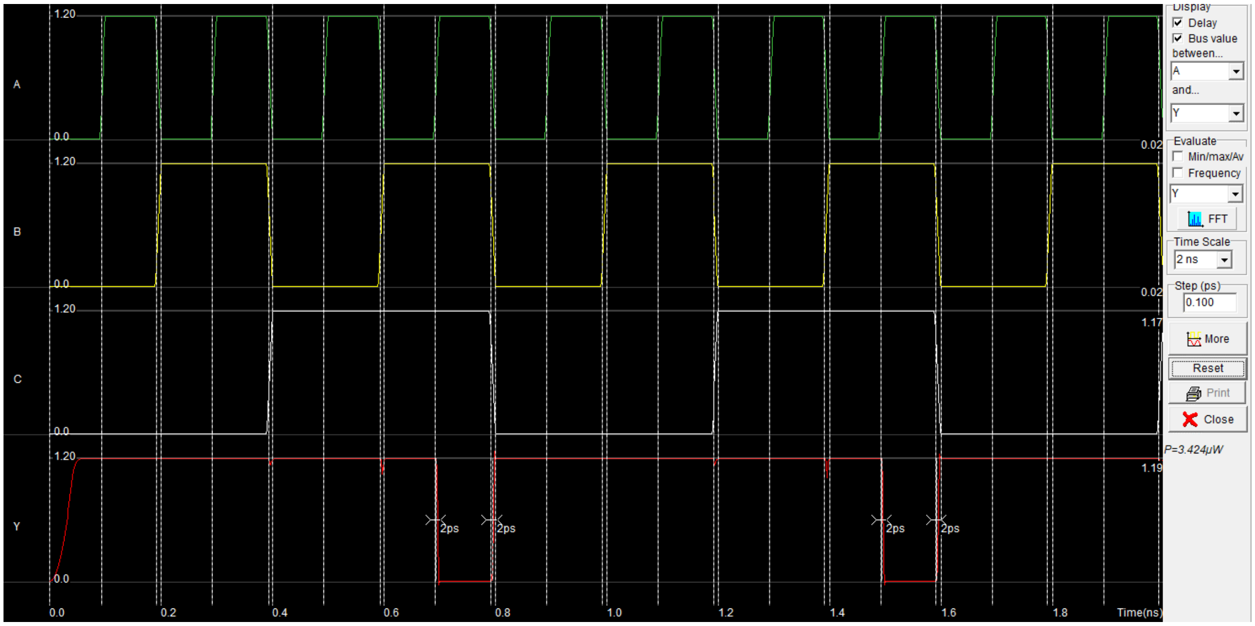


Figure 7

Now we need to see the waveform. Click the “Run Simulation” button. A waveform will appear, as shown in Fig. 8. It is clear from the plot that, when all three inputs are 1, output is 0. On the contrary, when any of the inputs is 0, output is 1. Thus, it is acting like a 3-input NAND operator.

You can change the time-scale by choosing from “Time Scale” dropdown menu and then clicking “Reset”. Rise time (t_r) and fall time (t_f) can be easily obtained using dropdown menu on the top-right corner of the waveform window. If you increase time-scale, you will notice that both t_r & t_f are almost equal after some cycles since we have chosen the transistors’ width carefully to maintain effective rise & fall resistance equal to that of a unit inverter.



☑ Delay
☑ Bus value
between... A
and... Y
□ Min/max/Av
□ Frequency
Y
Time Scale 2 ns
Step (ps) 0.100
P=3.424µW

Figure 8

Now we want to draw the layout of CMOS NOR3 gate. Open a new window. Click on the “MOS Generator”. A window will appear. Select the “MOS” tab from there. Select “pMOS”, Units “in lambda”, Width ‘34’ lambda, Length ‘2’ lambda, Number of fingers ‘3’. Width is chosen 34λ instead of 24λ, because we want to have six contacts on each Metal1-Diffusion crossing region of PUN without violating any design rule of Microwind2. Then click “Generate Device” and click on anywhere on the screen. The PUN will appear.

Again, click on the “MOS Generator”. Then select “nMOS”, Units “in lambda”, Width ‘5’ lambda, Length ‘2’ lambda, Number of fingers ‘3’. Then click “Generate Device” and click on anywhere on the screen below the PUN. The PDN will appear. Align the PDN using the “Stretch, Move” button. Now run DRC. It will look like Fig. 9 below.

Then join the polysilicon of PUN and PDN. The shorted polysilicon fingers will be the inputs (A, B, C). Now remove metal & contact from the middle of PUN, since all three pMOS of PUN are in series. Then form the output (Y) terminal using Metal1 by joining drain terminal of bottom pMOS of PUN (that is closer to output) with shorted drain of all three nMOS of PDN.

Now draw the VDD rail and GND rail. Connect the source of upper pMOS of PUN (that is farther from output) with VDD rail using Metal1. Similarly, connect the source of all three nMOS of PDN with GND rail. Place Well-Tap using a "Contact N+ diff/Metal1", and the Substrate-Tap using a "Contact P+ diff/Metal1". Don't forget to run DRC. Extend N-Well using the "Stretch, Move" button, if necessary.

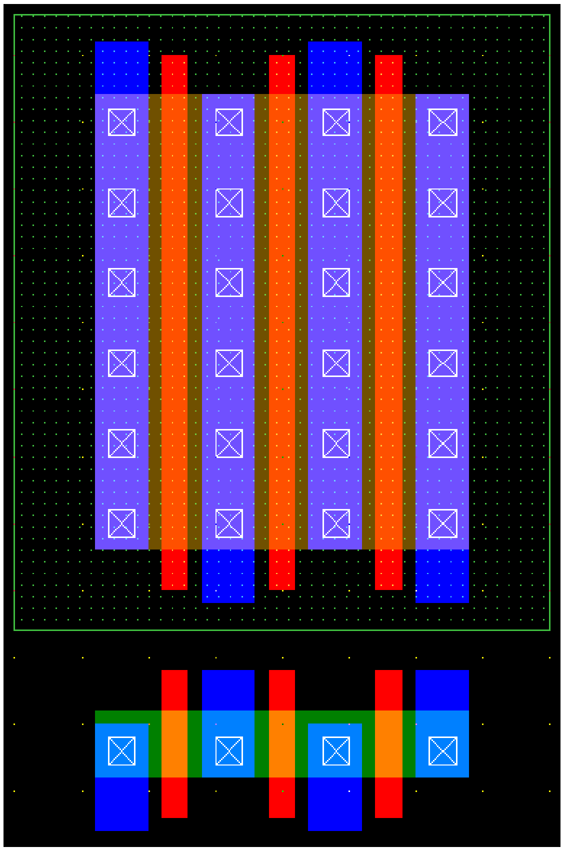


Figure 9

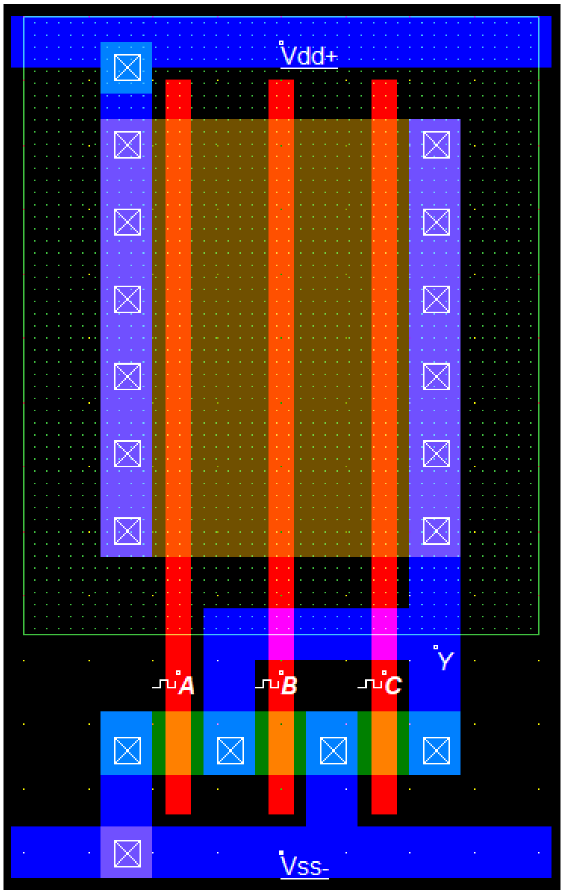
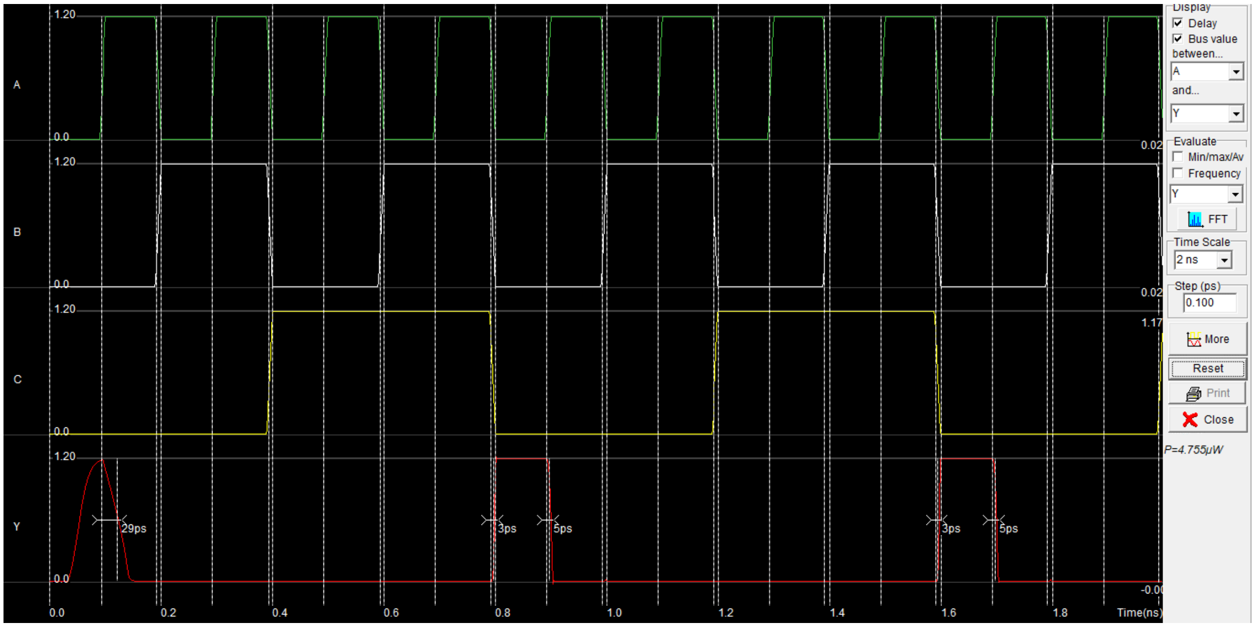


Figure 10

Then place appropriate input (clock) signals on polysilicon fingers, and VDD and GND on rails. Lastly, place a "Visible node" on output (Y) terminal. The final layout of CMOS NOR3 is depicted in Fig. 10.

Now we need to see the waveform. Click the "Run Simulation" button. A waveform will appear, as shown in Fig. 11. It is clear from the plot that, when all three inputs are 0, output is 1. On the contrary, when any of the inputs is 1, output is 0. Thus, it is acting like a 3-input NOR operator.



☑ Delay
☑ Bus value between... A and... Y

□ Min/max/Av
□ Frequency
Y

Time Scale: 2 ns

Step (ps): 0.100

P=4.755µW

Figure 11

You can change the time-scale by choosing from “Time Scale” dropdown menu and then clicking “Reset”. Rise time (t_r) and fall time (t_f) can be easily obtained using dropdown menu on the top-right corner of the waveform window. If you increase time-scale, you will notice that both t_r & t_f are almost equal after some cycles since we have chosen the transistors’ width carefully to maintain effective rise & fall resistance equal to that of a unit inverter.

Experiment – 5

Schematic & Layout of AOI-22

Theory:

A compound gate performing a more complex logic function in a single stage of logic is formed by using a combination of series and parallel switch structures. The derivation of the circuit for the function $Y = \overline{(A \cdot B) + (C \cdot D)}$ is shown in Fig. A. This function is called AND-OR-INVERT-22, or simply AOI22 because it performs the NOR of a pair of 2-input ANDs.

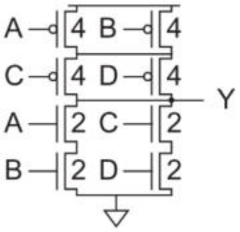


Figure A

For the nMOS pull-down network, take the noninverted expression $((A \cdot B) + (C \cdot D))$ indicating when the output should be pulled to '0.' The AND expressions $(A \cdot B)$ and $(C \cdot D)$ may be implemented by series connections of switches, as shown in Fig. B(a). Now ORing the result requires the parallel connection of these two structures, which is shown in Fig. B(b). For the pMOS pull-up network, we must compute the complementary expression using switches that turn on with inverted polarity.

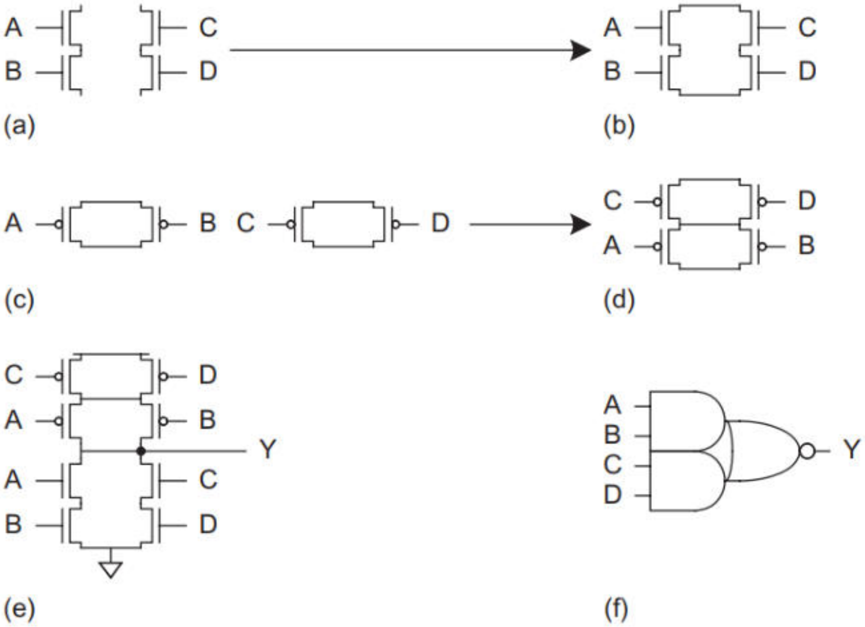


Figure B

By De Morgan's Law, this is equivalent to interchanging AND & OR operations. Hence, transistors that appear in series in the pull-down network must appear in parallel in the pull-up network, and vice versa. This principle is called conduction complements and has already been used in the design of the NAND and NOR gates. In the pull-up network, the parallel combination of A & B is placed in series with the parallel combination of C & D. This progression is evident in Fig. B(c) and Fig. B(d). Putting the networks together yields the full schematic (Fig. B(e)). The symbol is shown in Fig. B(f). The truth table of AOI22 is shown in Table 1 below.

Table 1

A	B	C	D	Pull-Up Network	Pull-Down Network	Y
0	**0**	**0**	**0**	ON	OFF	**1**
0	**0**	**0**	**1**	ON	OFF	**1**
0	**0**	**1**	**0**	ON	OFF	**1**
0	**0**	**1**	**1**	OFF	OFF	**0**
0	**1**	**0**	**0**	ON	OFF	**1**
0	**1**	**0**	**1**	ON	OFF	**1**
0	**1**	**1**	**0**	ON	OFF	**1**
0	**1**	**1**	**1**	OFF	ON	**0**
1	**0**	**0**	**0**	ON	OFF	**1**
1	**0**	**0**	**1**	ON	OFF	**1**
1	**0**	**1**	**0**	ON	OFF	**1**
1	**0**	**1**	**1**	OFF	ON	**0**
1	**1**	**0**	**0**	OFF	ON	**0**
1	**1**	**0**	**1**	OFF	ON	**0**
1	**1**	**1**	**0**	OFF	ON	**0**
1	**1**	**1**	**1**	OFF	ON	**0**

AOI gates are particularly advantaged in that the total number of transistors is less than if the AND, NOT, and OR functions were implemented separately. This results in increased speed, reduced power, smaller area, and potentially lower fabrication cost.

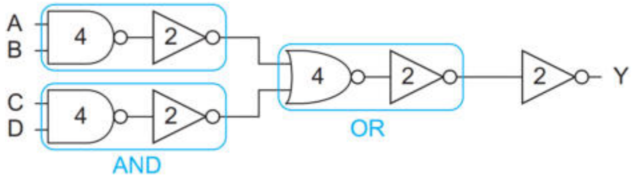


Figure C

For example, AOI22 could be built with two AND gates, an OR gate, and an inverter. The AND & OR gates in turn could be constructed from NAND/NOR gates and inverters, as shown in Fig. C, using a total of 20 transistors, as compared to 8 transistors in Fig. B(e).

Tools:

- DSCH2: for drawing the schematic and simulating
- Microwind2: for drawing the layout and simulating

Schematic:

After opening DSCH2, you will see an empty window. On the right-side of the window, there is a "Symbol library" panel. Drag and drop the following components to make the complete schematic of CMOS AOI22:

- n-channel MOS
- p-channel MOS
- Supply
- Ground
- Clock
- Light

In Fig. 1, the full schematic of a CMOS AOI22 is shown.

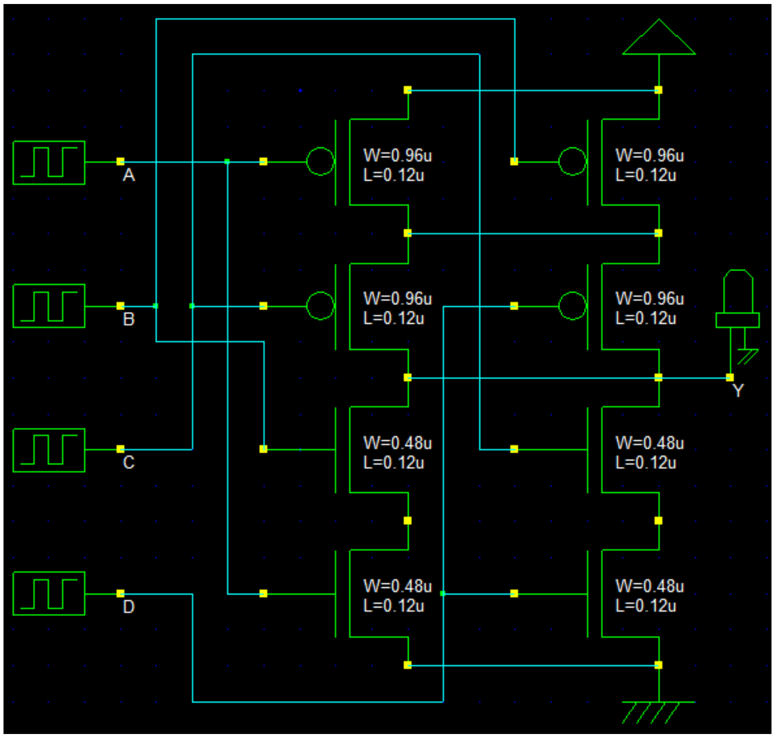


Figure 1

Now click the "Run Simulation" button from top panel; and after several seconds, click the "Timing Diagram" button. You will see a waveform like Fig. 2.

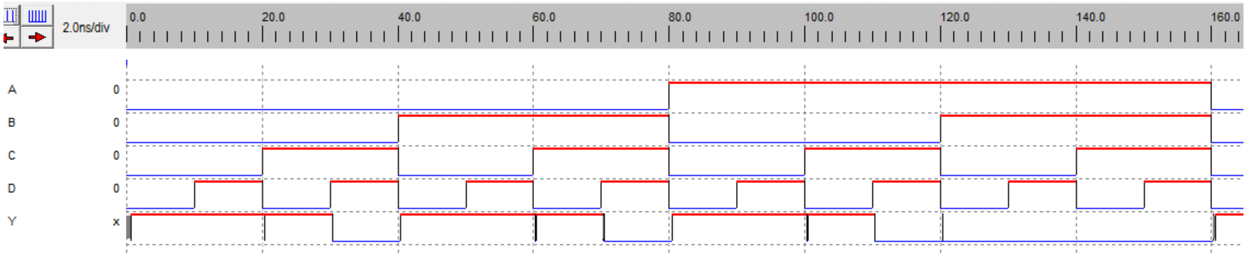


Figure 2

The waveform is generated as expected (AOI22). The waveform matches the truth table from "Theory" section (Table 1).

Layout:

After opening Microwind2, you will see an empty window. On the right-side of the opened window, there is a “Palette” panel. Click on the “MOS Generator”. A window will appear. Select the “MOS” tab from there.

Select “pMOS”, Units “in lambda”, Width ‘22’ lambda, Length ‘2’ lambda, Number of fingers ‘4’. Width is chosen 22λ instead of 16λ, because we want to have four contacts on each Metal1-Diffusion crossing region of pull-up network (PUN) without violating any design rule of Microwind2. Then click “Generate Device” and click on anywhere on the screen. The PUN will appear.

Again, click on the “MOS Generator”. Then select “nMOS”, Units “in lambda”, Width ‘10’ lambda, Length ‘2’ lambda, Number of fingers ‘4’. Width is chosen 10λ instead of 8λ, because we want to have two contacts on each Metal1-Diffusion crossing region of pull-down network (PDN) without violating any design rule of Microwind2. Then click “Generate Device” and click on anywhere on the screen below the PUN. The PDN will appear like Fig. 3.

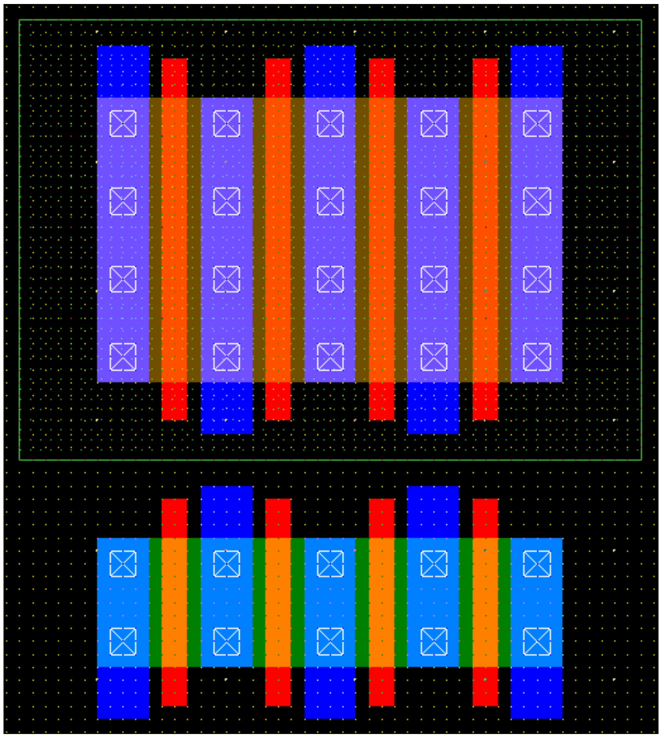


Figure 3

Align them using “Stretch, Move” button, if necessary. Run DRC. Then join the polysilicon of PUN and PDN. The shorted polysilicon fingers will be the inputs (A, B, C, D). Now remove metal & contact from the middle of A & B nMOS, since A & B nMOS of PDN are in series. Similarly, remove metal & contact from the middle of C & D nMOS, since they are in series too. Then form the output (Y) terminal using Metal1 by joining shorted drain of B & C nMOS (of PDN) with shorted drain of C & D pMOS (of PUN). The layout will look like Fig. 4 below.

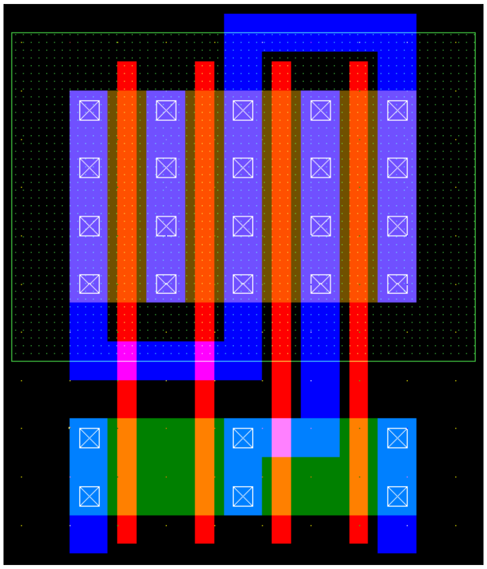


Figure 4

Now draw the VDD rail and GND rail. Connect the shorted source of A & B pMOS with VDD rail using Metal1. Similarly, connect the sources of A nMOS and D nMOS with GND rail. Place Well-Tap using a “Contact N+ diff/Metal1”, and the Substrate-Tap using a “Contact P+ diff/Metal1”. Don’t forget to run DRC. Extend N-Well using the “Stretch, Move” button, if necessary.

Then place appropriate input (clock) signals on polysilicon fingers, and VDD and GND on rails. Lastly, place a “Visible node” on output (Y) terminal. The final layout of CMOS AOI22 is depicted in Fig. 5 below.

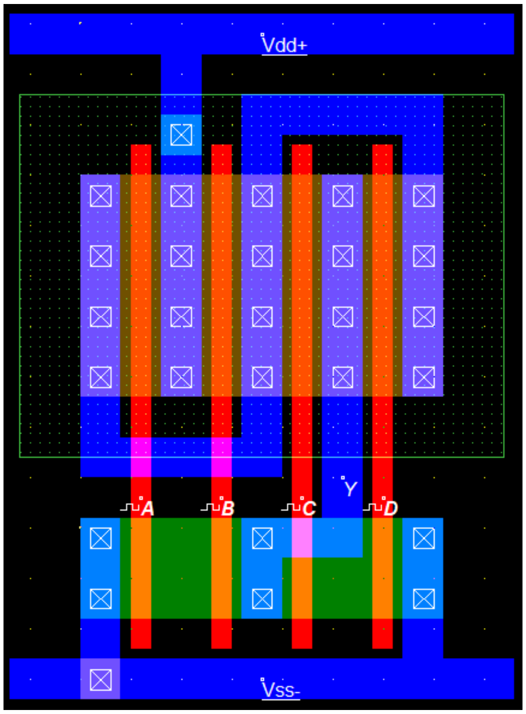
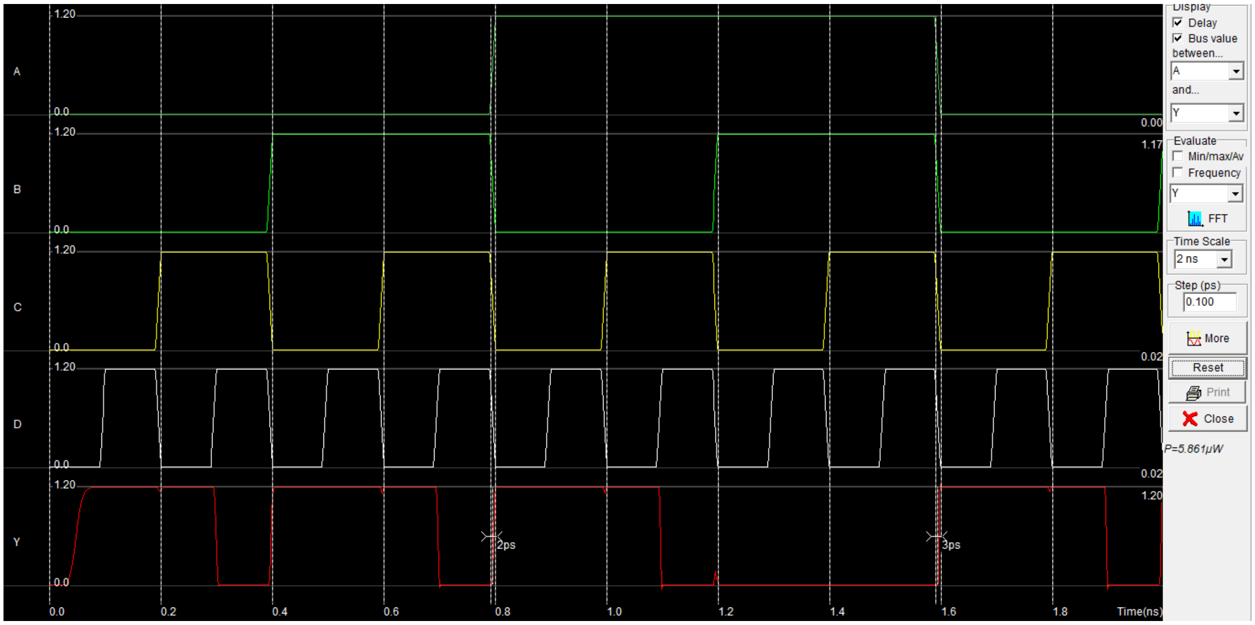


Figure 5

Now we need to see the waveform. Click the “Run Simulation” button. A waveform will appear, as shown in Fig. 6. It matches the waveform generated using DSCH2 from “Schematic” section (Fig. 2), and the truth table from “Theory” section (Table 1).



Display

☑ Delay

☑ Bus value

between... A

and... Y

Evaluate

□ Min/max/Av

□ Frequency

Y

Time Scale: 2 ns

Step (ps): 0.100

P=5.861µW

Figure 6

You can change the time-scale by choosing from “Time Scale” dropdown menu and then clicking “Reset”. Rise time (t_r) and fall time (t_f) can be easily obtained using dropdown menu on the top-right corner of the waveform window.

Experiment – 6

Schematic & Layout of OAI-31 & OAI-22

Theory:

A compound gate performing a more complex logic function in a single stage of logic is formed by using a combination of series and parallel switch structures. The derivation of the circuit for the function $Y = \overline{(A + B + C) \cdot D}$ is shown in Fig. A, with truth table shown in Table 1. This function is called OR-AND-INVERT-31, or simply OAI31.

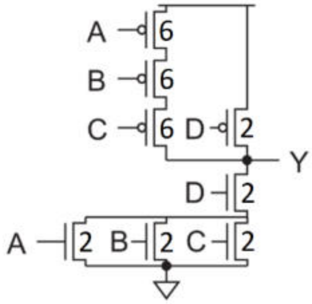


Figure A

The nMOS pull-down network pulls the output low if D is 1 and either A or B or C are 1, so D is in series with the parallel combination of A, B, C. The pMOS pull-up network is the conduction complement, so D must be in parallel with the series combination of A, B, C.

Table 1

A	B	C	D	Pull-Up Network	Pull-Down Network	Y
0	**0**	**0**	**0**	ON	OFF	**1**
0	**0**	**0**	**1**	ON	OFF	**1**
0	**0**	**1**	**0**	ON	OFF	**1**
0	**0**	**1**	**1**	OFF	ON	**0**
0	**1**	**0**	**0**	ON	OFF	**1**
0	**1**	**0**	**1**	OFF	ON	**0**
0	**1**	**1**	**0**	ON	OFF	**1**
0	**1**	**1**	**1**	OFF	ON	**0**
1	**0**	**0**	**0**	ON	OFF	**1**
1	**0**	**0**	**1**	OFF	ON	**0**
1	**0**	**1**	**0**	ON	OFF	**1**
1	**0**	**1**	**1**	OFF	ON	**0**
1	**1**	**0**	**0**	ON	OFF	**1**
1	**1**	**0**	**1**	OFF	ON	**0**
1	**1**	**1**	**0**	ON	OFF	**1**
1	**1**	**1**	**1**	OFF	ON	**0**

The derivation of the circuit for the function $Y = \overline{(A+B)\cdot(C+D)}$ is shown in Fig. B. This function is called OR-AND-INVERT-22, or simply OAI22. It performs the NAND of a pair of 2-input ORs.

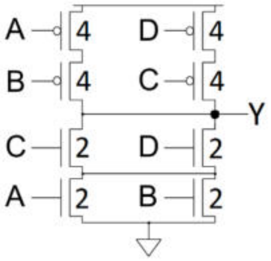


Figure B

In the pull-down network, the parallel combination of A & B is placed in series with the parallel combination of C & D. The pMOS pull-up network is the conduction complement, so the series combination of A & B must be in parallel with the series combination of C & D. The truth table of OAI22 is depicted in Table 2 below.

Table 2

A	B	C	D	Pull-Up Network	Pull-Down Network	Y
0	**0**	**0**	**0**	ON	OFF	**1**
0	**0**	**0**	**1**	ON	OFF	**1**
0	**0**	**1**	**0**	ON	OFF	**1**
0	**0**	**1**	**1**	ON	OFF	**1**
0	**1**	**0**	**0**	ON	OFF	**1**
0	**1**	**0**	**1**	OFF	ON	**0**
0	**1**	**1**	**0**	OFF	ON	**0**
0	**1**	**1**	**1**	OFF	ON	**0**
1	**0**	**0**	**0**	ON	OFF	**1**
1	**0**	**0**	**1**	OFF	ON	**0**
1	**0**	**1**	**0**	OFF	ON	**0**
1	**0**	**1**	**1**	OFF	ON	**0**
1	**1**	**0**	**0**	ON	OFF	**1**
1	**1**	**0**	**1**	OFF	ON	**0**
1	**1**	**1**	**0**	OFF	ON	**0**
1	**1**	**1**	**1**	OFF	ON	**0**

Tools:

- DSCH2: for drawing the schematic and simulating
- Microwind2: for drawing the layout and simulating

Schematic:

At first, we want to draw the schematic of CMOS OAI31. After opening DSCH2, you will see an empty window. On the right-side of the window, there is a "Symbol library" panel. Drag and drop the following components to make the complete schematic of CMOS OAI31:

- n-channel MOS
- p-channel MOS
- Supply
- Ground
- Clock
- Light

In Fig. 1, the full schematic of a CMOS OAI31 is shown.

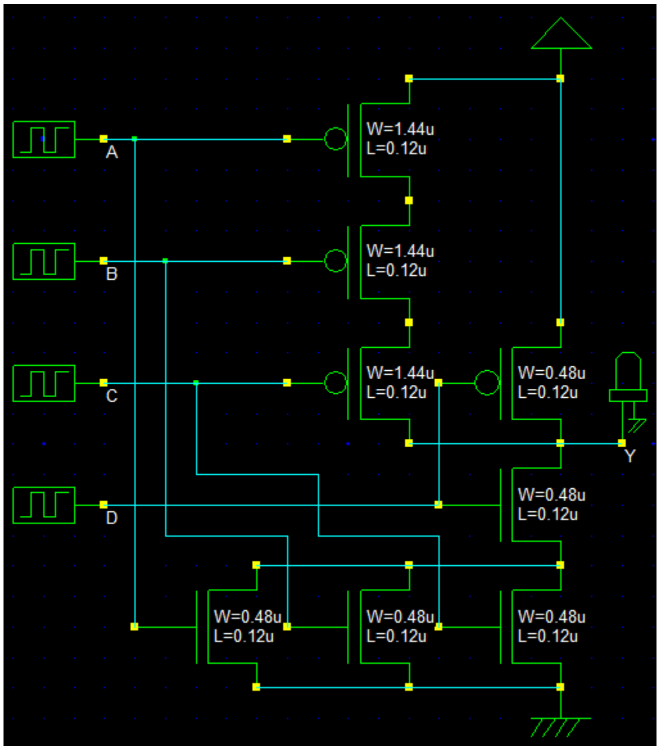


Figure 1

Now click the "Run Simulation" button from top panel; and after several seconds, click the "Timing Diagram" button. You will see a waveform like Fig. 2.

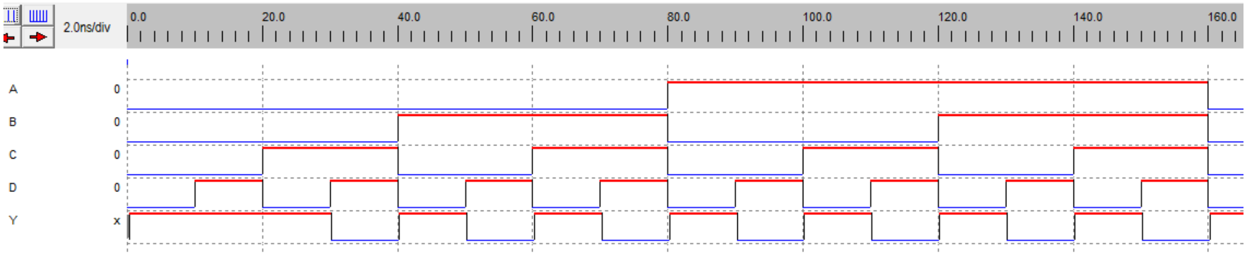


Figure 2

The waveform is generated as expected (OAI31). The waveform matches the truth table from "Theory" section (Table 1).

Now we want to draw the schematic of CMOS OAI22. The procedure is same as before. We need to open a new window in DSCH2, then drag and drop the necessary components. In Fig. 3, the full schematic of a CMOS OAI22 is shown.

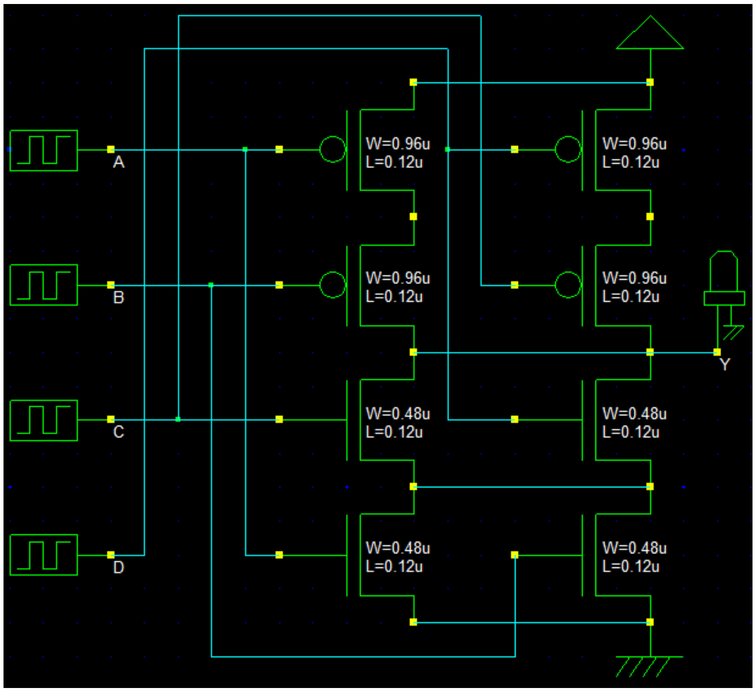


Figure 3

Now click the "Run Simulation" button from top panel; and after several seconds, click the "Timing Diagram" button. You will see a waveform like Fig. 4.

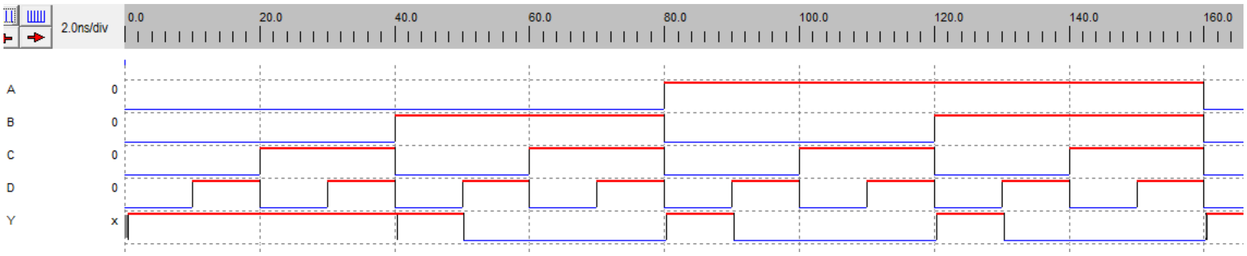


Figure 4

The waveform is generated as expected (OAI22). The waveform matches the truth table from "Theory" section (Table 2).

Layout:

At first, we want to draw the layout of CMOS OAI31. After opening Microwind2, you will see an empty window. On the right-side of the opened window, there is a "Palette" panel. Click on the "MOS Generator". A window will appear. Select the "MOS" tab from there.

Select "pMOS", Units "in lambda", Width '34' lambda, Length '2' lambda, Number of fingers '4'. Width is chosen 34λ instead of 24λ, because we want to have six contacts on each Metal1-Diffusion crossing region of pull-up network (PUN) without violating any design rule of Microwind2. Then click "Generate Device" and click on anywhere on the screen. The PUN will appear. Though the D pMOS of PUN should have less width (8λ), we are taking equal width for all transistors of PUN for the sake of simplicity.

Again, click on the "MOS Generator". Then select "nMOS", Units "in lambda", Width '10' lambda, Length '2' lambda, Number of fingers '4'. Width is chosen 10λ instead of 8λ, because we want to have two contacts on each Metal1-Diffusion crossing region of pull-down network (PDN) without violating any design rule of Microwind2. Then click "Generate Device" and click on anywhere on the screen below the PUN. The PDN will appear like Fig. 5.

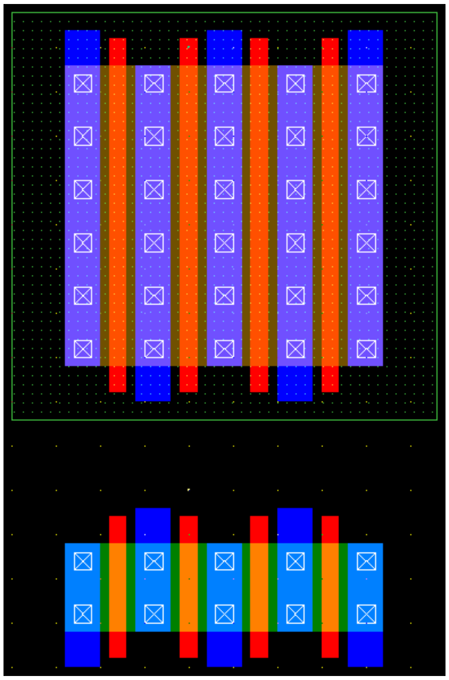


Figure 5

Align them using "Stretch, Move" button, if necessary. Run DRC. Then join the polysilicon of PUN and PDN. The shorted polysilicon fingers will be the inputs (A, B, C, D). Now remove metal & contact from the middle of A & B pMOS, since A & B pMOS of PUN are in series. Similarly, remove metal & contact from the middle of B & C pMOS, since they are in series too. Then form the output (Y) terminal using Metal1 by joining drain of D nMOS (of PDN) with shorted drain of C & D pMOS (of PUN). The layout will look like Fig. 6 below.

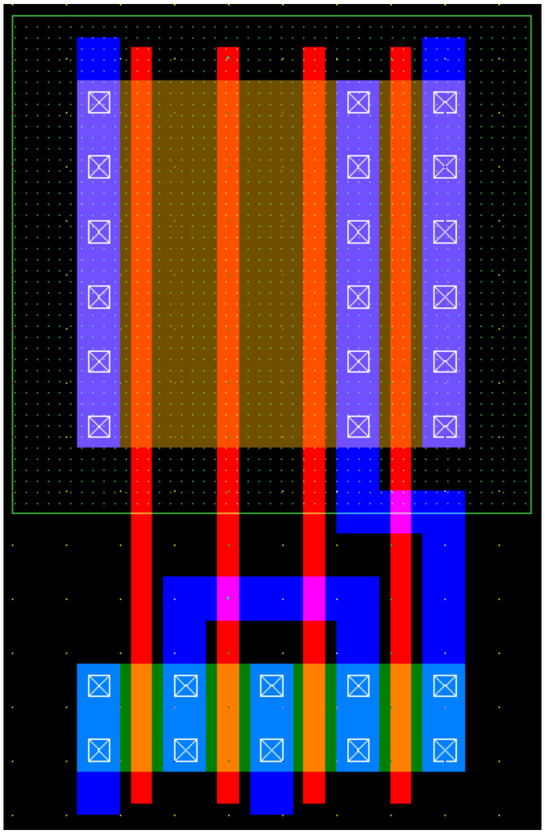


Figure 6

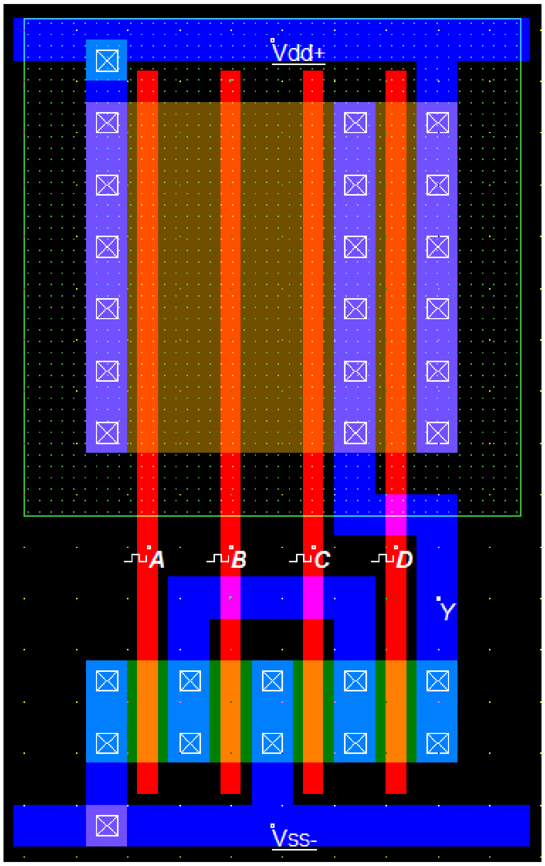


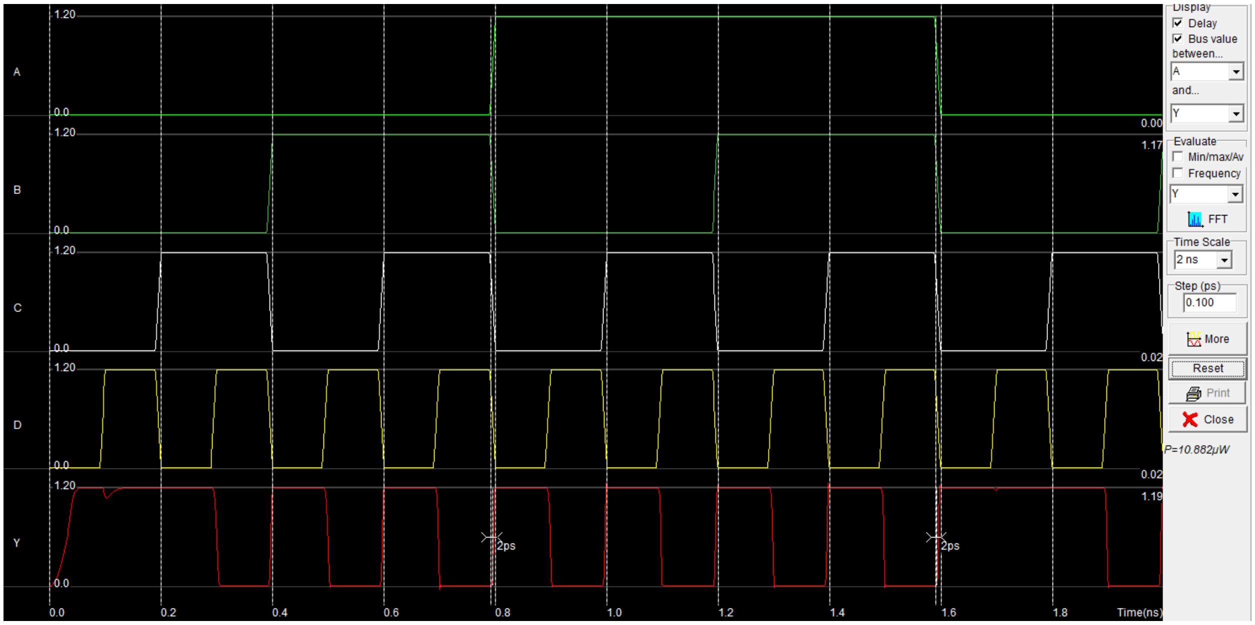
Figure 7

Now draw the VDD rail and GND rail. Connect the sources of A pMOS and D pMOS with VDD rail using Metal1. Similarly, connect the source of A nMOS and shorted source of B & C nMOS with GND rail. Place Well-Tap using a “Contact N+ diff/Metal1”, and the Substrate-Tap using a “Contact P+ diff/Metal1”. Don’t forget to run DRC. Extend N-Well using the “Stretch, Move” button, if necessary.

Then place appropriate input (clock) signals on polysilicon fingers, and VDD and GND on rails. Lastly, place a “Visible node” on output (Y) terminal. The final layout of CMOS OAI31 is depicted in Fig. 7.

Now we need to see the waveform. Click the “Run Simulation” button. A waveform will appear, as shown in Fig. 8. It matches the waveform generated using DSCH2 from “Schematic” section (Fig. 2), and the truth table from “Theory” section (Table 1).

You can change the time-scale by choosing from “Time Scale” dropdown menu and then clicking “Reset”. Rise time (t_r) and fall time (t_f) can be easily obtained using dropdown menu on the top-right corner of the waveform window.



☑ Delay
☑ Bus value between... A and... Y

□ Min/max/Av
□ Frequency
Y

Time Scale: 2 ns

Step (ps): 0.100

P=10.882μW

Figure 8

Now we want to draw the layout of CMOS OAI22. Open a new window. Click on the "MOS Generator". A window will appear. Select the "MOS" tab from there. Select "pMOS", Units "in lambda", Width '22' lambda, Length '2' lambda, Number of fingers '4'. Width is chosen 22λ instead of 16λ, because we want to have four contacts on each Metal1-Diffusion crossing region of PUN without violating any design rule of Microwind2. Then click "Generate Device" and click on anywhere on the screen. The PUN will appear.

Again, click on the "MOS Generator". Then select "nMOS", Units "in lambda", Width '10' lambda, Length '2' lambda, Number of fingers '4'. Width is chosen 10λ instead of 8λ, because we want to have two contacts on each Metal1-Diffusion crossing region of PDN without violating any design rule of Microwind2. Then click "Generate Device" and click on anywhere on the screen below the PUN. The PDN will appear. Align the PDN using the "Stretch, Move" button. Now run DRC. It will look like Fig. 9 below.

Then join the polysilicon of PUN and PDN. The shorted polysilicon fingers will be the inputs (A, B, C, D). Now remove metal & contact from the middle of A & B pMOS, since A & B pMOS of PUN are in series. Similarly, remove metal & contact from the middle of C & D pMOS, since they are in series too. Then form the output (Y) terminal using Metal1 by joining shorted drain of B & C pMOS (of PUN) with shorted drain of C & D nMOS (of PDN).

Now draw the VDD rail and GND rail. Connect the sources of A pMOS and D pMOS with VDD rail using Metal1. Similarly, connect the shorted source of A & B nMOS with GND rail. Place Well-Tap using a "Contact N+ diff/Metal1", and the Substrate-Tap using a "Contact P+ diff/Metal1". Don't forget to run DRC. Extend N-Well using the "Stretch, Move" button, if necessary.

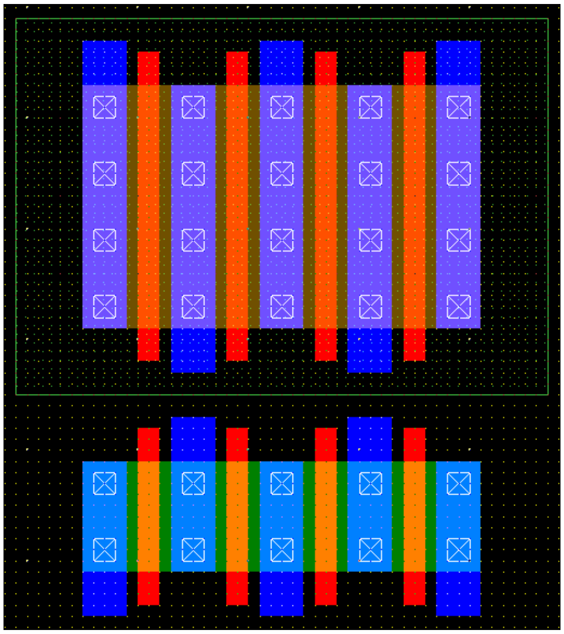


Figure 9

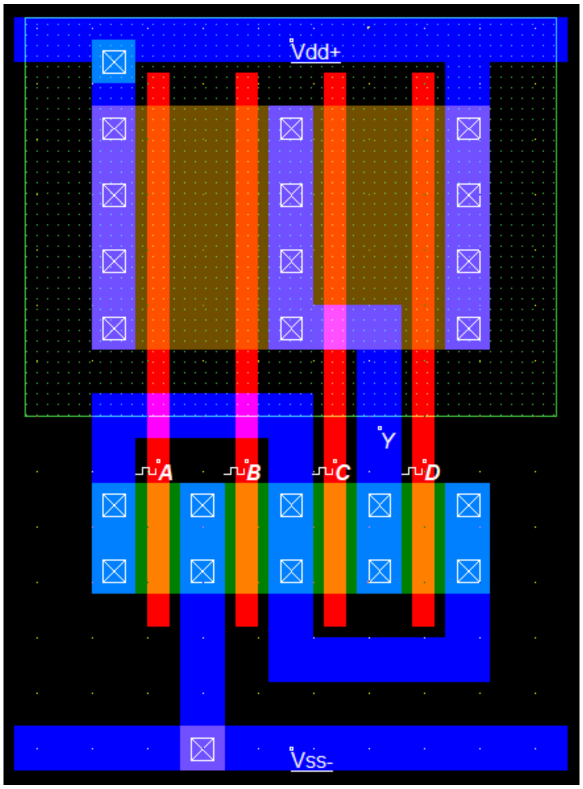
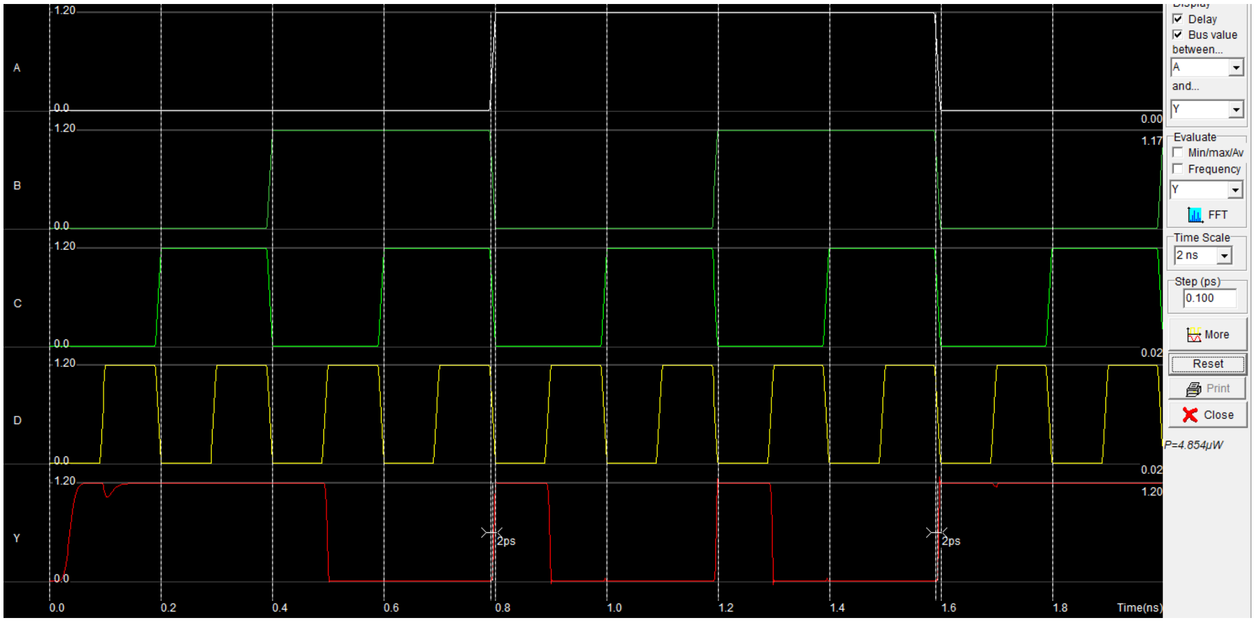


Figure 10

Then place appropriate input (clock) signals on polysilicon fingers, and VDD and GND on rails. Lastly, place a “Visible node” on output (Y) terminal. The final layout of CMOS OAI22 is depicted in Fig. 10.

Now we need to see the waveform. Click the “Run Simulation” button. A waveform will appear, as shown in Fig. 11. It matches the waveform generated using DSCH2 from “Schematic” section (Fig. 4), and the truth table from “Theory” section (Table 2).



between... A

and... Y

Time Scale 2 ns

Step (ps) 0.100

P=4.854µW

Figure 11

You can change the time-scale by choosing from “Time Scale” dropdown menu and then clicking “Reset”. Rise time (t_r) and fall time (t_f) can be easily obtained using dropdown menu on the top-right corner of the waveform window.