4:1 MUX using CMOS and Transmission Gate Logic

VLSI Design ECE 6346

—

Parshi Srinidhi
1778179

Sai Krishna Dirisala
1586546

1. Introduction

Multiplexing is described as a property of combination of one or more signals transmitted on a single channel. The device used for achieving this phenomenon is called a multiplexer. In general terminology a multiplexer called as MUX is also defined as a combinational logic circuit that picks one among the several analog or digital input signals and transmits the chosen signal to the available single output channel. For instance if a multiplexer has 2^n inputs, there will be **n** (resulting in 2^n combinations of 0's and 1's) selection lines which help to pick the desired signal to be transmitted. A general 2^n**:1** multiplexer with inputs $I_0, I_1, I_2 \ldots I_{2^n-1}$ and an Enable **E** having selection lines $S_0, S_1, S_2 \ldots\ldots S_n$ with output **Y** will be as shown in the following figure.

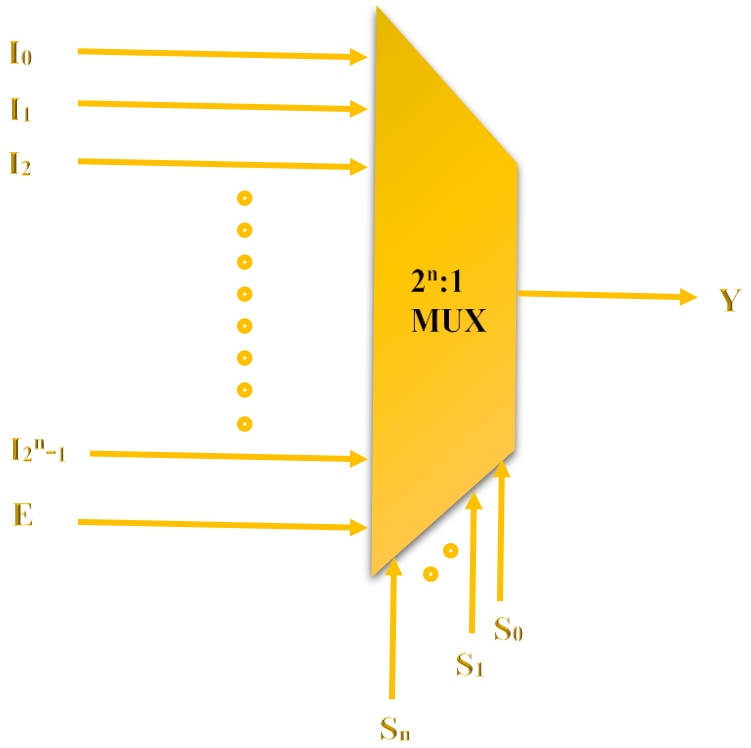


Fig 1. A generalized representation of a Multiplexer

A Multiplexer is also called as a switch or data selector. It allows the binary information from the available set of input lines and also depending on the given selection lines, the input line is driven

only to the one output line. The generalized idea can be represented as shown in the following diagram, where the several sources are sent to the single output line when the enable signal is switched to ON state.

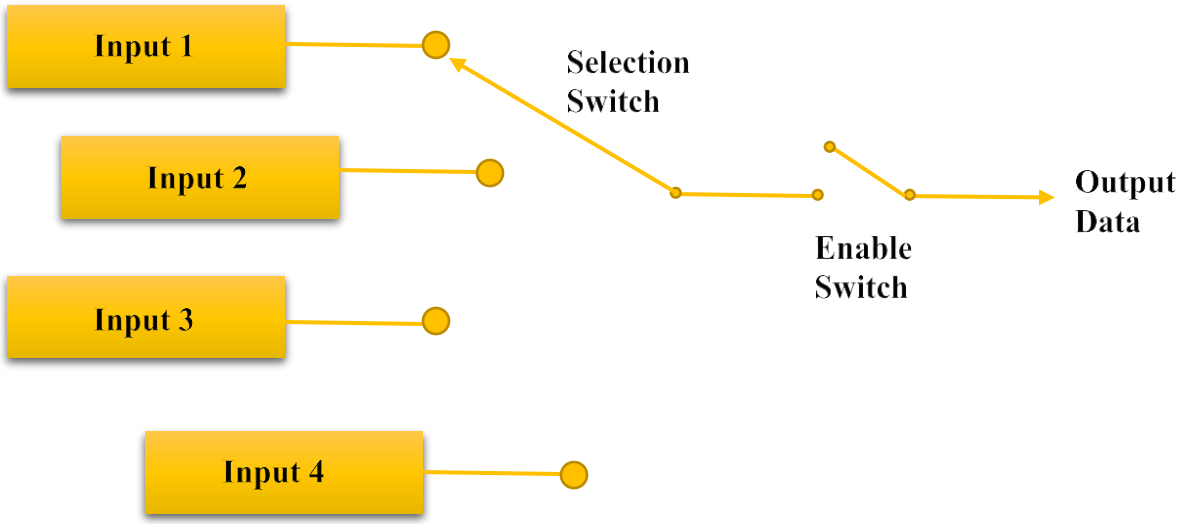


Fig 2. Basic ideology of a Multiplexer

These multiplexers are normally used in Communication network and Transmission system to increase the efficiency of audio and video from various mediums with the help of single lines or cables. They are also useful in Computer Memory and Telephone network.

In this project we are going to implement a 4:1 MUX using both CMOS and Transmission gate Logic. We use Cadence for simulations and to start with we first implement the 3 - input AND & 4 - input OR gates using individual CMOS and Transmission gates and insert them into the gate level logic of 4:1 MUX. We then compare both the results and finally tabulate those results.

2. A 4:1 Multiplexer

A four to one multiplexer comprises of four input data lines I_0, I_1, I_2, I_3 with two select lines S_0, S_1 and one output line Y. Both of the selection lines are useful for selecting one of the above input lines to transfer to the available output line.

The following figure displays the block diagram of a 4:1 multiplexer with all the input, selection and output lines

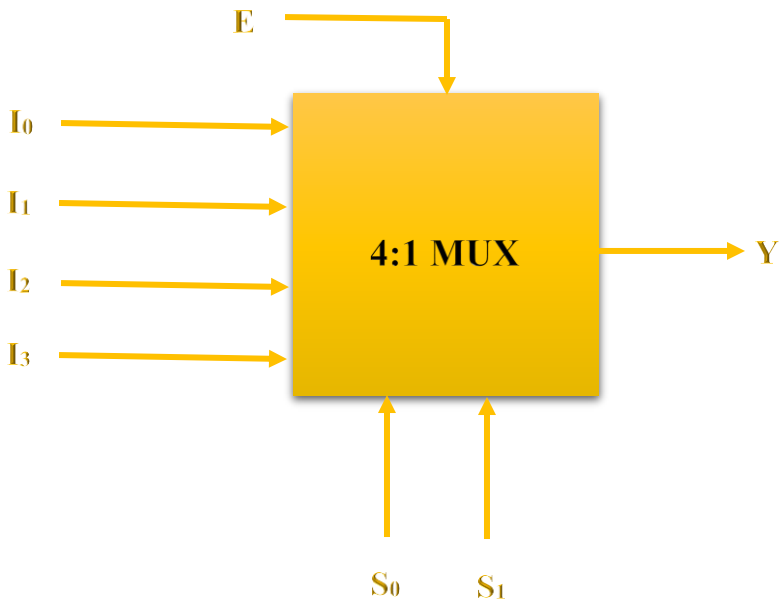


Fig 3. Block diagram of a 4:1 Multiplexer

The truth table of the above 4:1 MUX is as follows in which the 4 input combinations on the selection lines S_0 and S_1 respectively switches the inputs I_0, I_1, I_2 and I_3 to the output. This means that when $S_1 = 0$ & $S_0 = 0$, the output Y yields I_0, in similar ways it is I_1 if $S_1 = 0$ & $S_0 = 1$ so on.

Select Data Inputs		Outputs
S_1	S_0	Y
0	0	I_0
0	1	I_1
1	0	I_2
1	1	I_3

Table 1. Truth table for a 4:1 MUX

From the truth table the output expressions can be written as

$S_1 = 0$ & $S_0 = 0 \rightarrow Y = I_0\overline{S_1}\,\overline{S_0}$

$S_1 = 0$ & $S_0 = 1 \rightarrow Y = I_1\overline{S_1}\,S_0$

$S_1 = 1$ & $S_0 = 0 \rightarrow Y = I_2\,S_1\overline{S_0}$

$S_1 = 1$ & $S_0 = 1 \rightarrow Y = I_3\,S_1S_0$

The final Boolean expression of this multiplexer can be obtained by summing all the above individual expressions and it is as shown below.

$$Y = I_0\overline{S_1}\,\overline{S_0} + I_1\overline{S_1}\,S_0 + I_2\,S_1\overline{S_0} + I_3\,S_1S_0$$

From the above expression 4 to 1 Multiplexer can be implemented using basic logic gates AND & OR. The following figure depicts this implementation with the help of 3 input AND & 4 input OR gates.

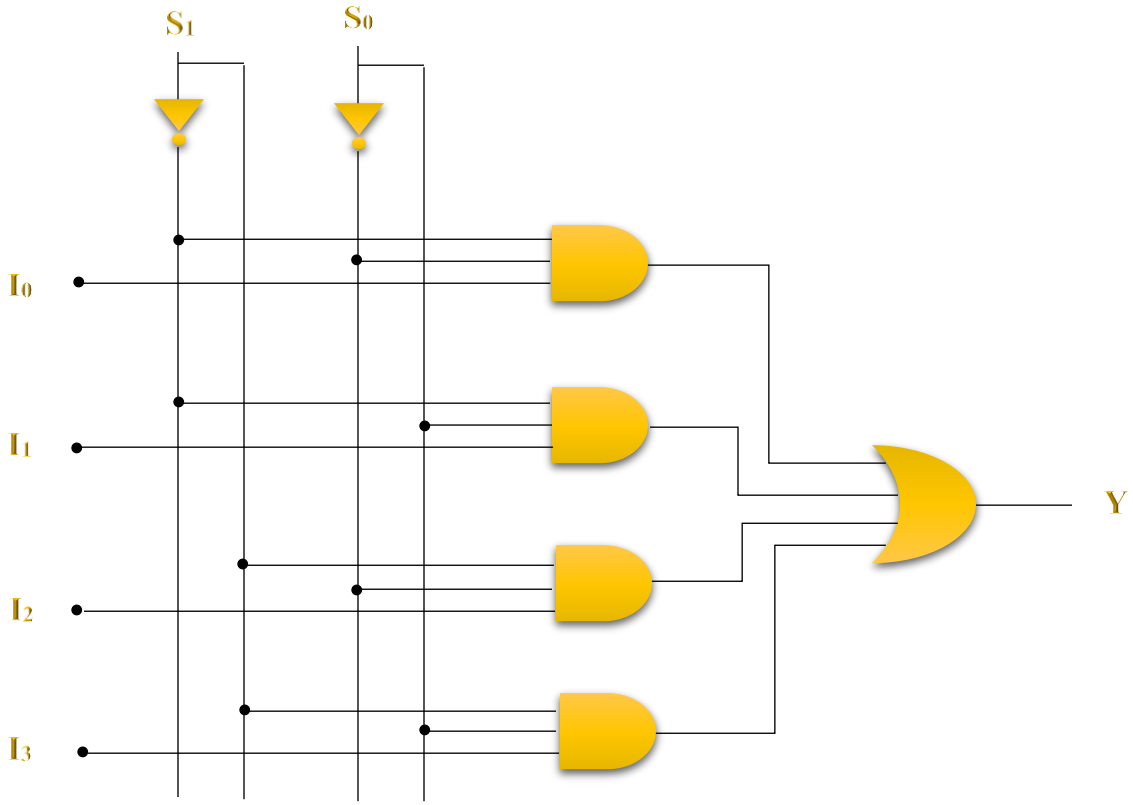


Fig 4. Logic diagram of a 4:1 MUX

3. Applications of 4:1 MUX

Multiplexer is a pivotal combinational circuit used in a number of different applications. They can select the input information as per the requirement of the user, and hence they are frequently termed as "Data Selectors" for this reason. These find applications in various digital as well as analog electronics.

Some of the applications are discussed below.

i. **Parallel Data Transmission:**
In general we need to transmit "n" parallel data bits we need **n** number of parallel channels, which is complex process. Whereas if use the technique of Multiplexing the process can be made really simple and easy to execute. Since a multiplexer can transmit the parallel data using single wire at transmission side and while at receiver it can be again converted back to parallel from series transmission using another concept called de-multiplexing. This makes the transmission economical and less complex, besides the ability to switch digital signals can be extended to switch video signals and analogue signals.

ii. **Data Routing:**
Multiplexers are a very good source of alternatives for data routing in the selected and desired path. The control over the flow of input data is done in a specified path with the help of selection lines that are available for the concerned multiplexer.

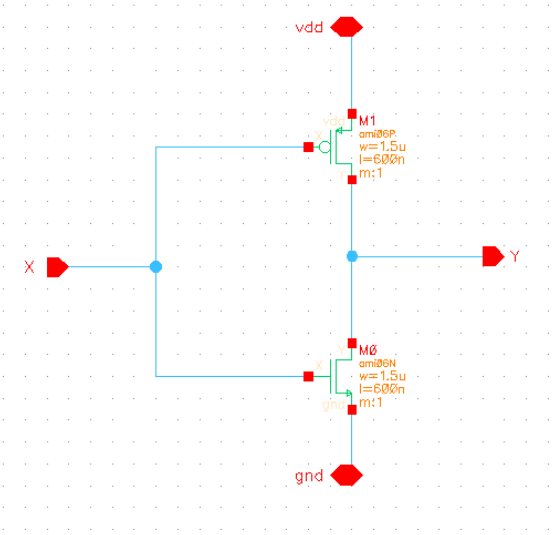
4. Design Procedure

- Cadence which is one among the most powerful tools available for building schematics, layouts and simulations is being used throughout this project.
- The basic idea of design is to start with universal gates, NAND and NOR and then implement the remaining using them.
- Schematics for NAND, NOT and NOR are drawn using both CMOS and Transmission Gates separately.
- Using the principles AND = NAND + NOT; OR = NOR + NOT we now design the 2 input AND & OR gates.
- As we require a 3 input AND & 4 input OR, we now use available 2 input gates to further develop them.
- Each of these AND & OR circuits obtained from CMOS and Transmission logic are placed into the logic diagram of a 4 to 1 multiplexer which is shown in **Fig 4**.
- We now obtain the simulations in both CMOS and Transmission gate and compare them.
- Finally Layouts for are drawn for them and post layout simulations obtained are compared.

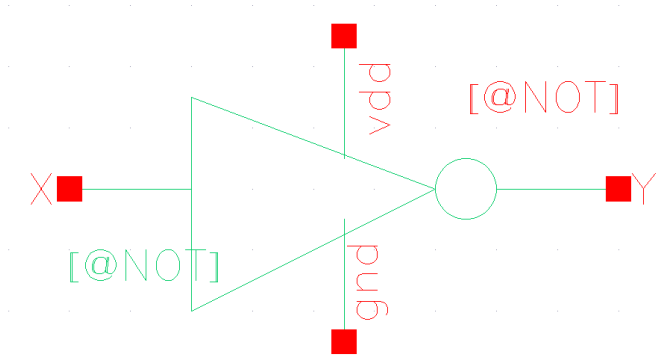
5. Schematic – Symbol and Truth Table

i. NOT Gate

The schematic for INVERTER using both PMOS and NMOS will be as shown in the following figure



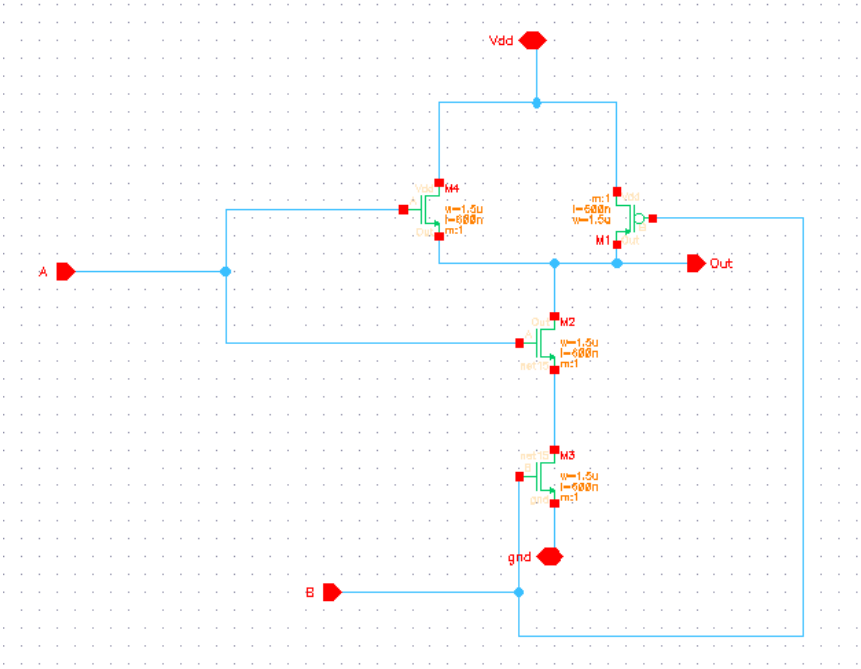
Corresponding Symbol and Truth table for NOT gate with input X and output Y would be as follows



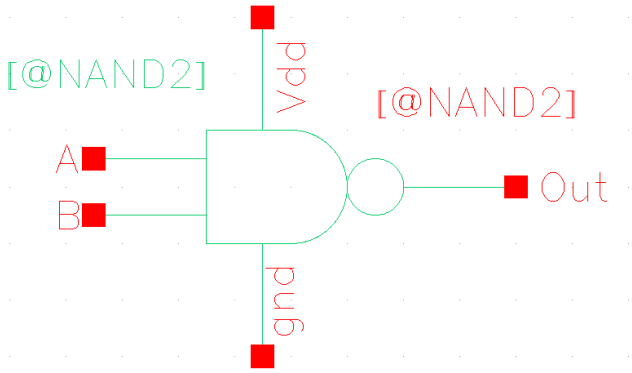
X	Y
0	1
1	0

ii. NAND Gate {CMOS Logic}

The schematic for NAND gate using PMOS and NMOS is shown in the following figure



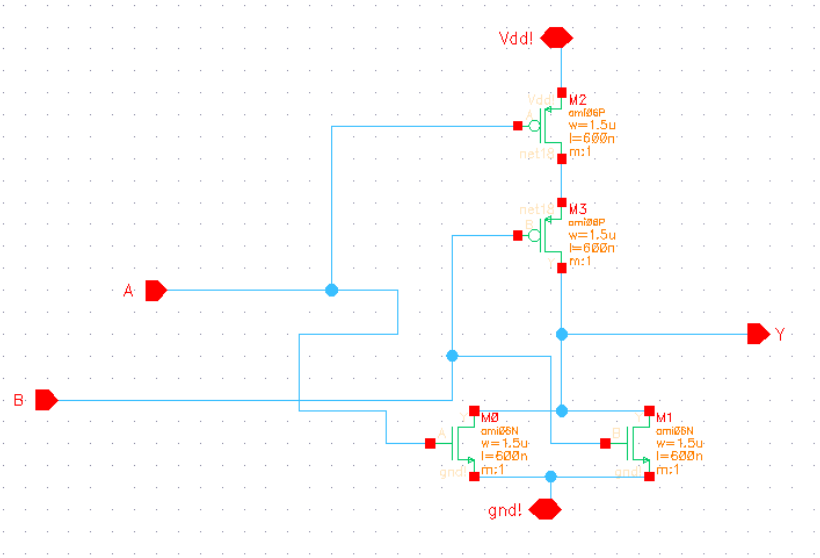
The symbol and truth table for a NAND with two inputs A, B and output “Out” is as shown below.



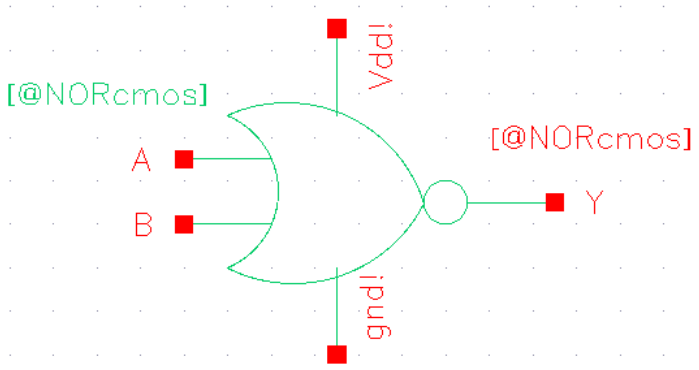
A	B	Out
0	0	1
0	1	1
1	0	1
1	1	0

iii. NOR Gate {CMOS Logic}

The NOR gate schematic implemented with CMOS logic will be as shown below.



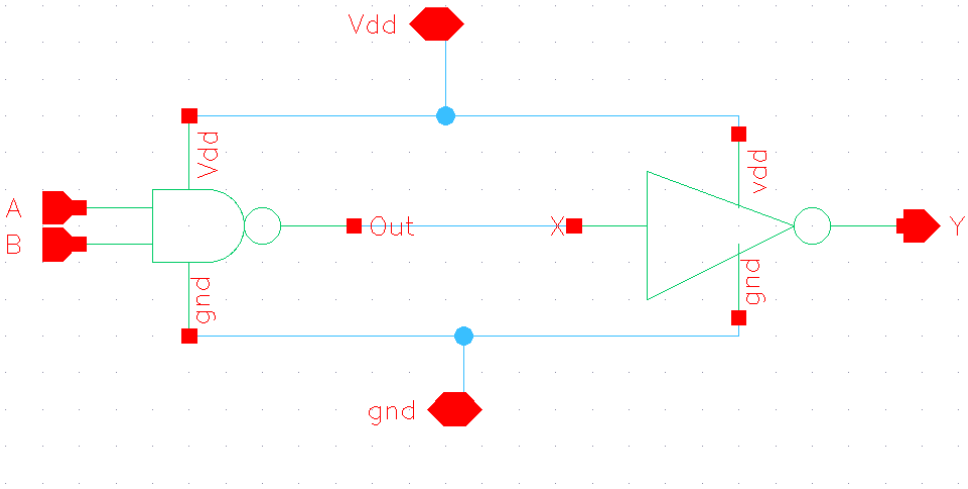
The symbol and truth table for a NOR gate with inputs A, B and output Y will be as shown below



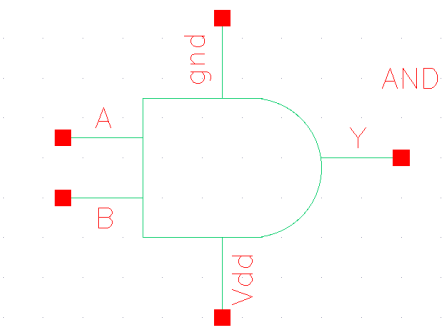
A	B	Y
0	0	1
0	1	0
1	0	0
1	1	0

iv. 2 input AND Gate {CMOS Logic}

The schematic of AND gate obtained by combining a NAND and NOT will be as shown below.



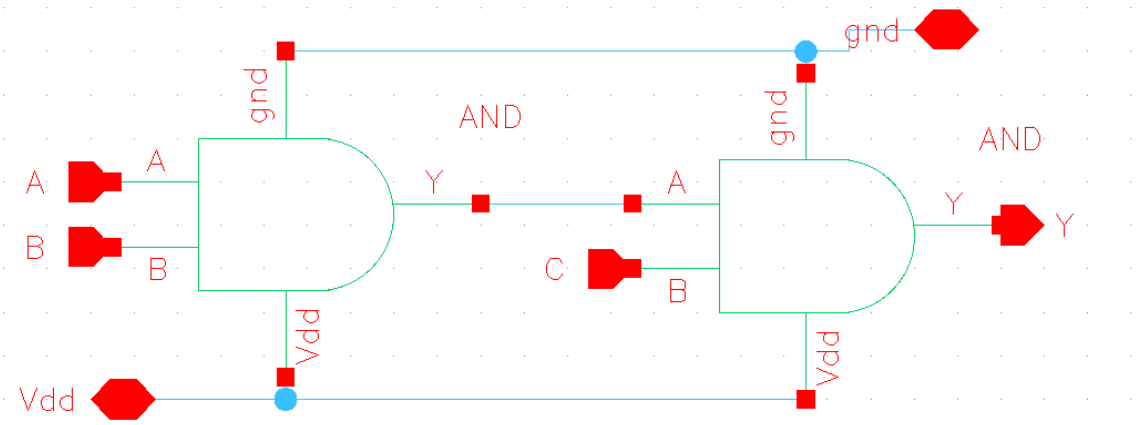
Symbol and truth table for AND gate with two inputs A, B and output Y will be as shown below.



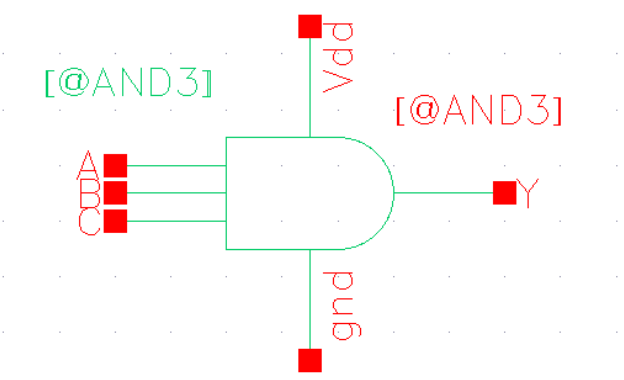
A	B	Y
0	0	0
0	1	0
1	0	0
1	1	1

v. 3 input AND Gate {CMOS Logic}

The schematic for a 3 input AND gate will be a combination of 2 input AND connected to another 2 input AND as shown in the following figure.



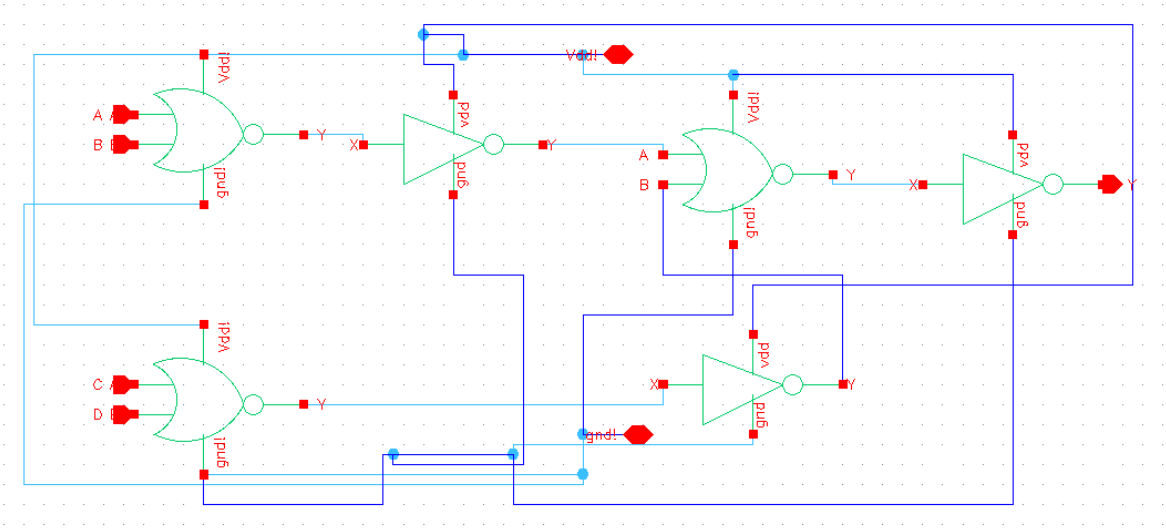
Symbol and truth table for an AND gate with three inputs A, B, C and output Y will be as shown below.



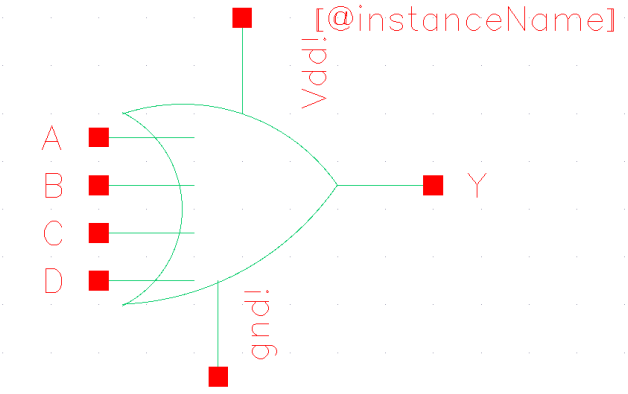
A	B	C	Y
0	0	0	0
0	0	1	0
0	1	0	0
0	1	1	0
1	0	0	0
1	0	1	0
1	1	0	0
1	1	1	1

vi. 4 input OR Gate {CMOS Logic}

The schematic for a 4 input OR gate using three 2 input NOR gates and 3 NOT gates will be as shown in the following figure.



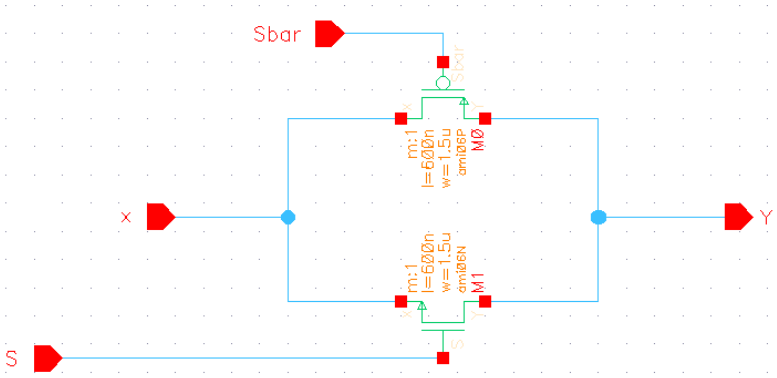
Symbol and truth table for a 4 input OR gate with inputs as A, B, C, D and output Y will be as shown below.



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

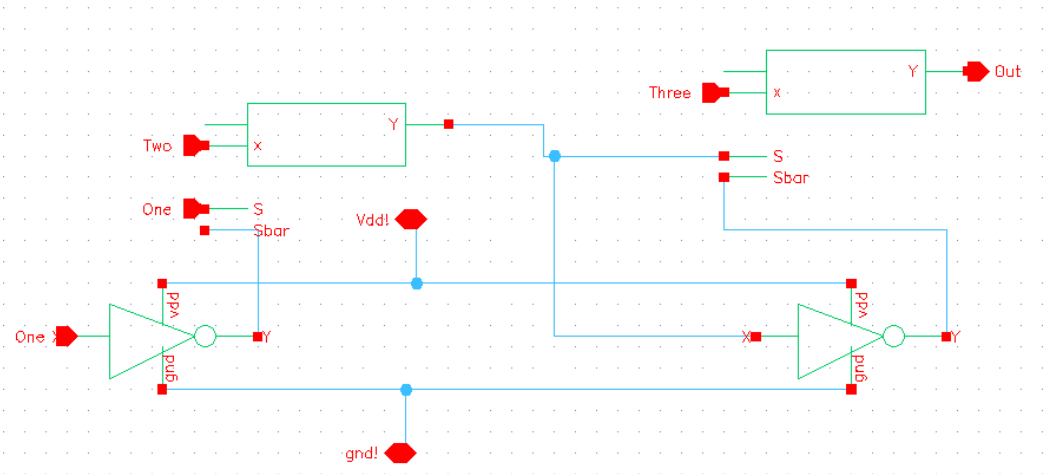
vii. Basic Transmission Gate

The schematic for the basic transmission gate will be as shown in the following figure

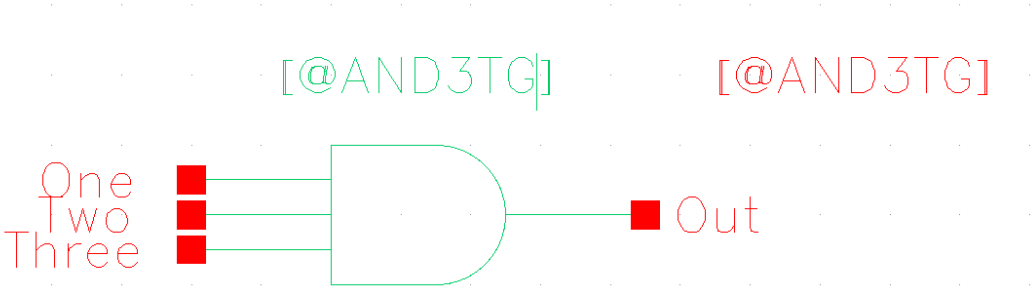


viii. 3 input AND Gate {Transmission Logic}

The schematic for a three input AND gate using transmission gates having inputs "One", "Two", "Three" and output "Out" will be as shown in the following figure.

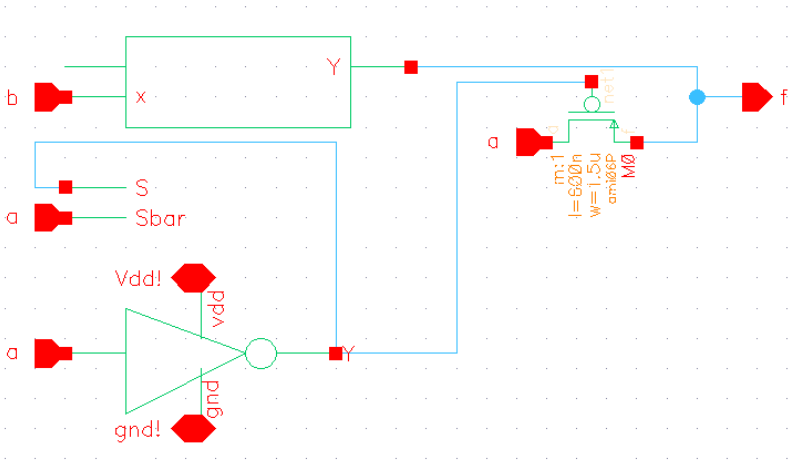


The symbol for the above AND gate would be as follows

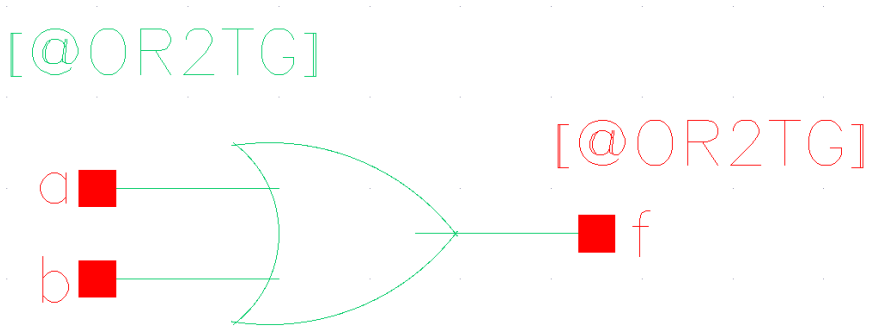


ix. 2 input OR Gate {Transmission Logic}

The schematic showing an OR gate with two inputs ‘a’ and ‘b’ and output ‘f’ will be as shown in the following figure.

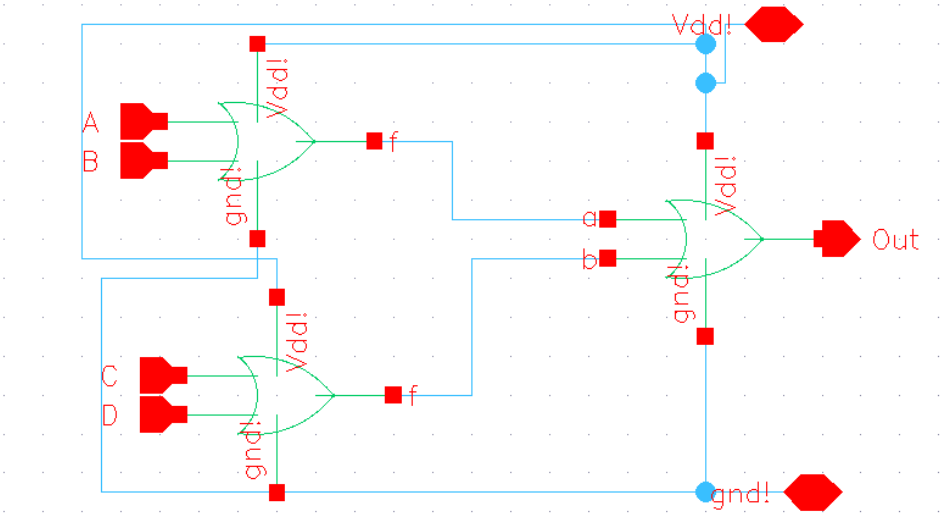


Symbol for above schematic is as shown below.

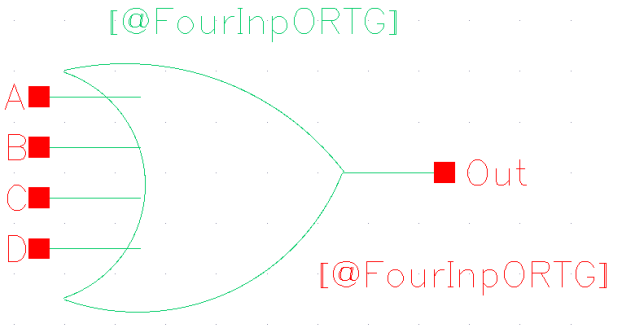


x. 4 input OR Gate {Transmission Logic}

The schematic for a four input OR gate with inputs A, B, C, D and output "Out" is as shown in the following figure.

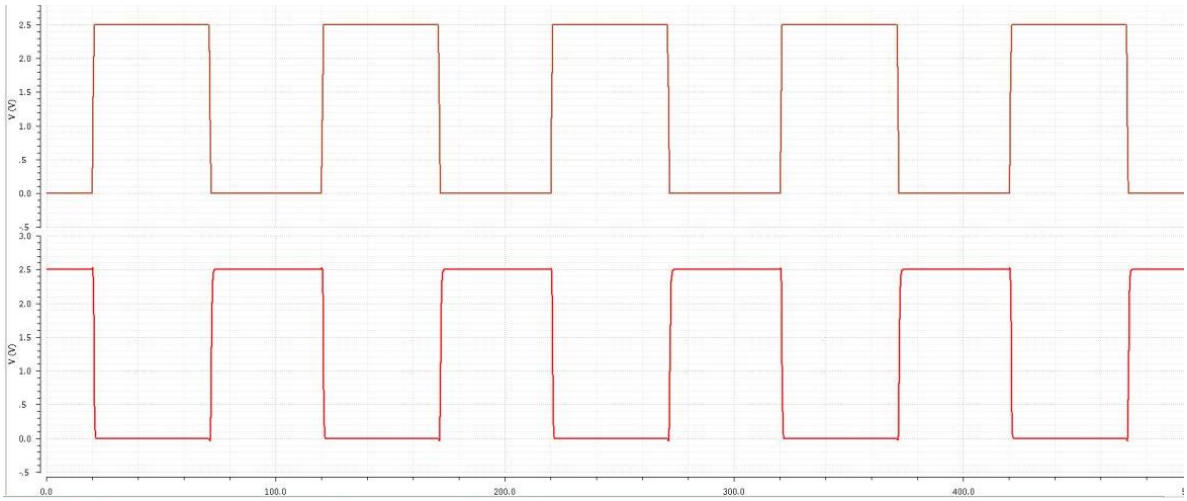


The symbol would be as follows

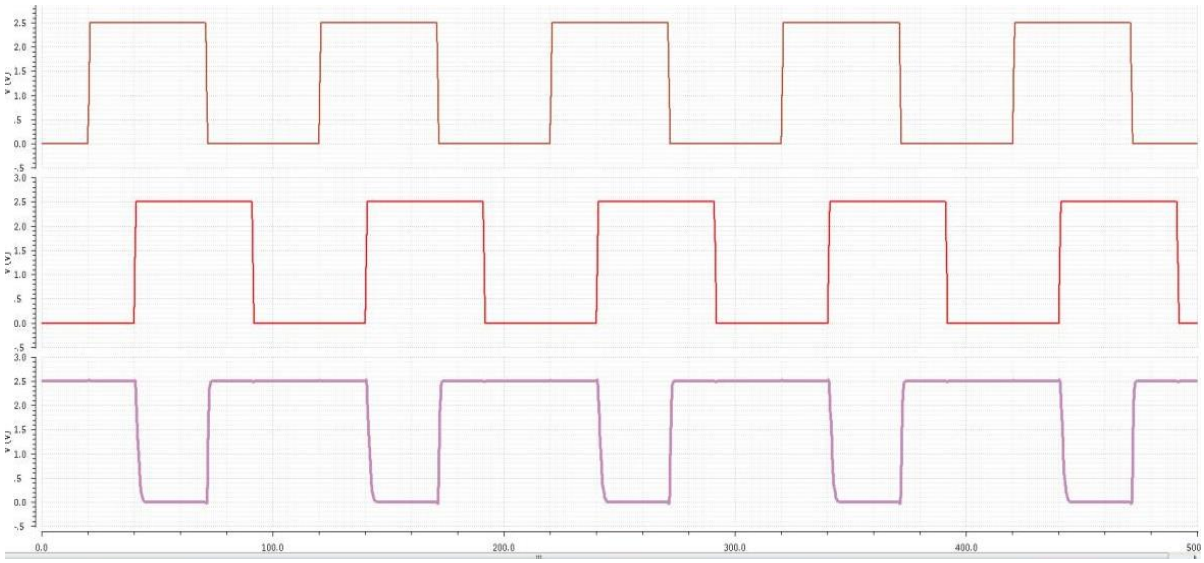


6. Simulations

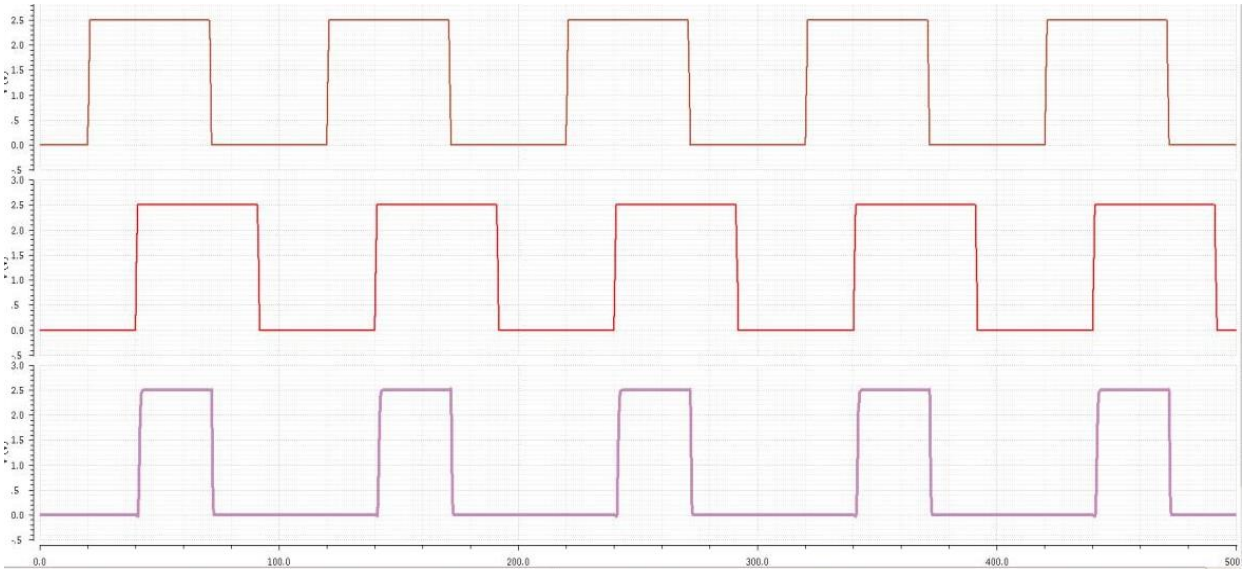
NOT Gate



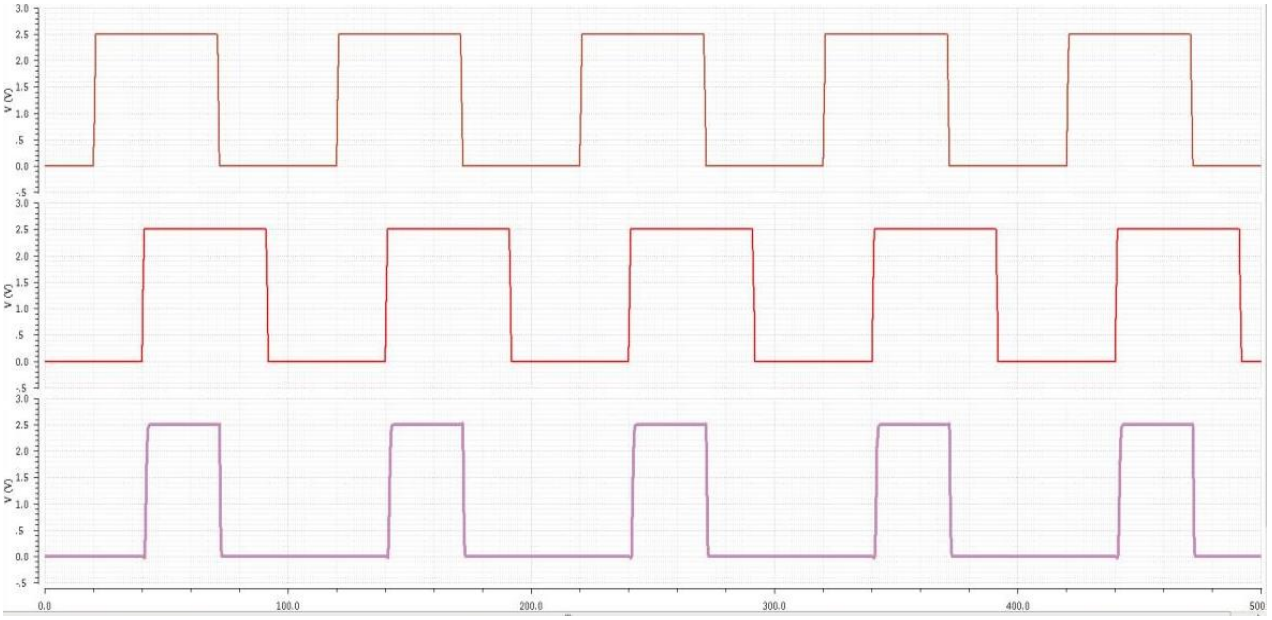
NAND Gate {CMOS Logic}



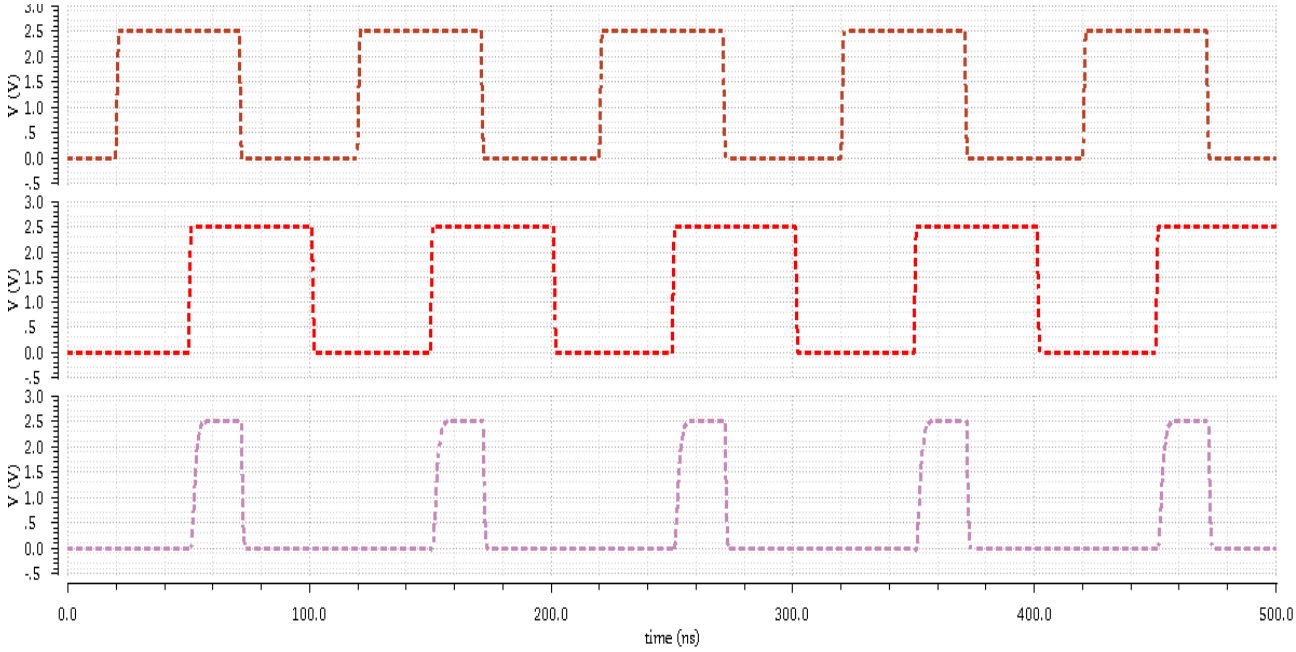
AND Gate {CMOS Logic}



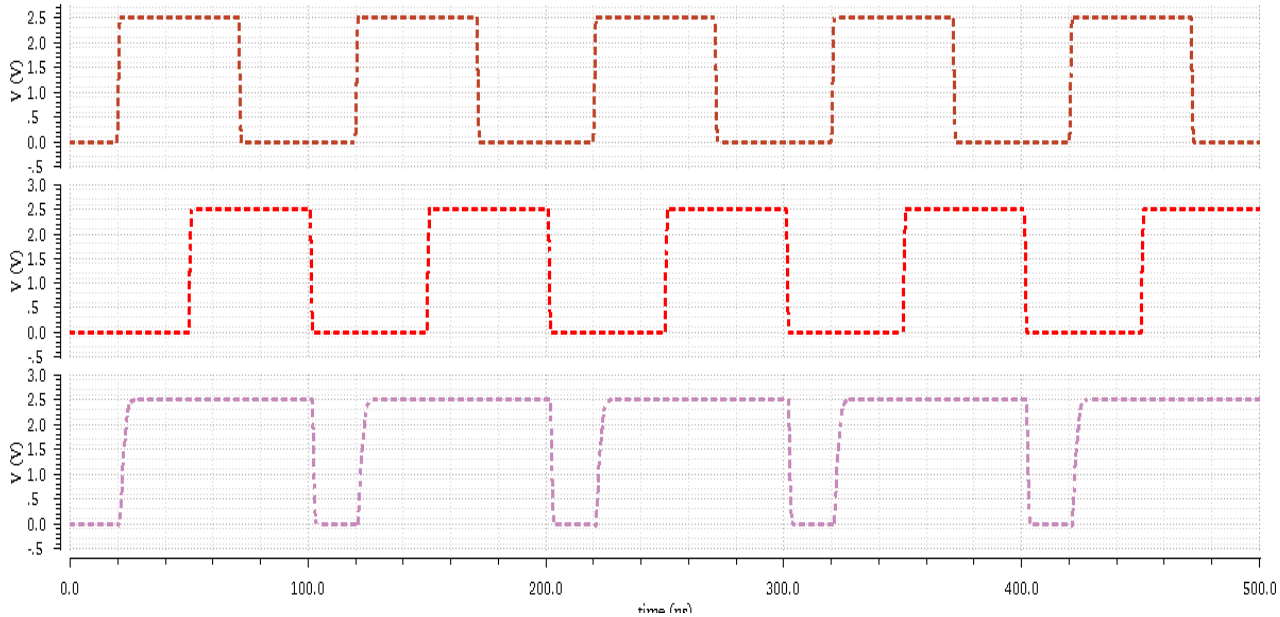
OR Gate {CMOS Logic}



AND Gate {Transmission Logic}

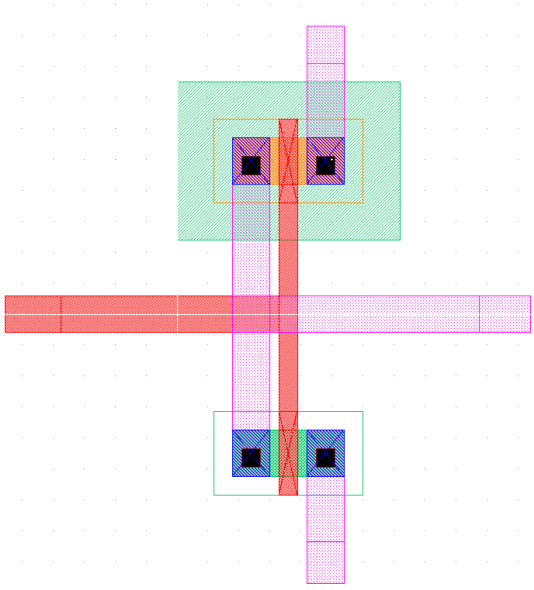


OR Gate {Transmission Logic}

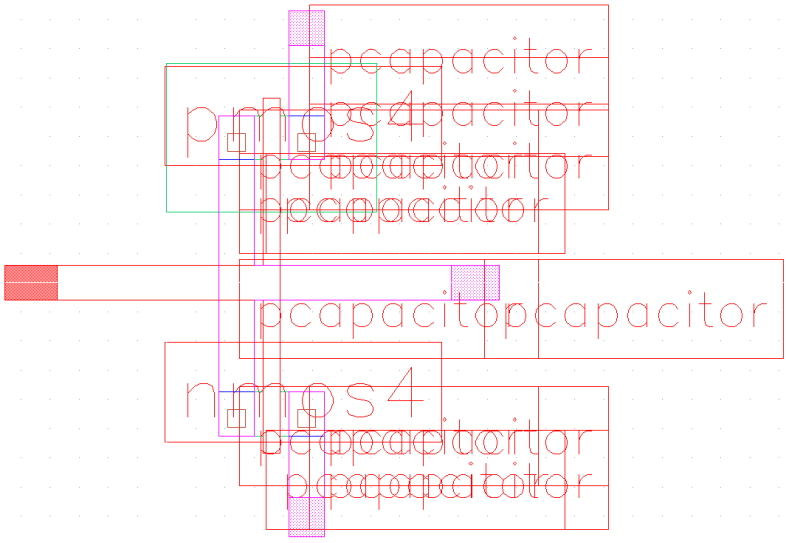


7. Layouts

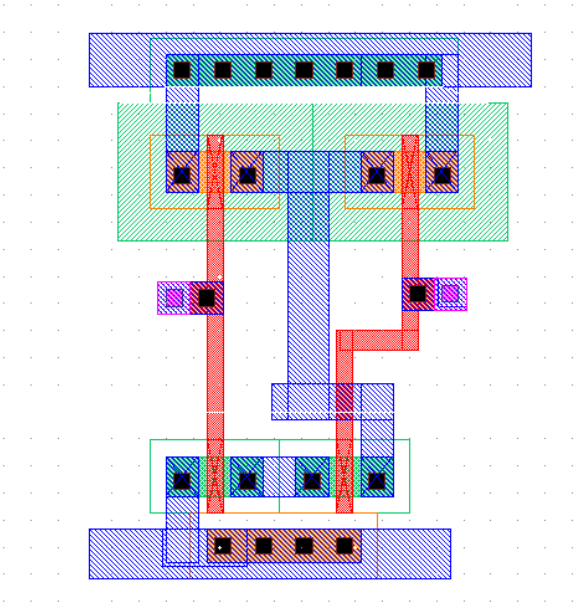
NOT gate



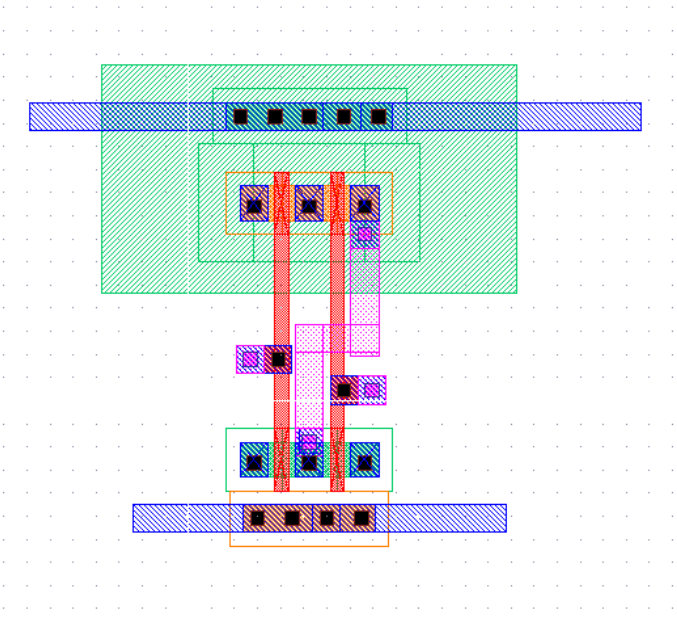
NOT extraction



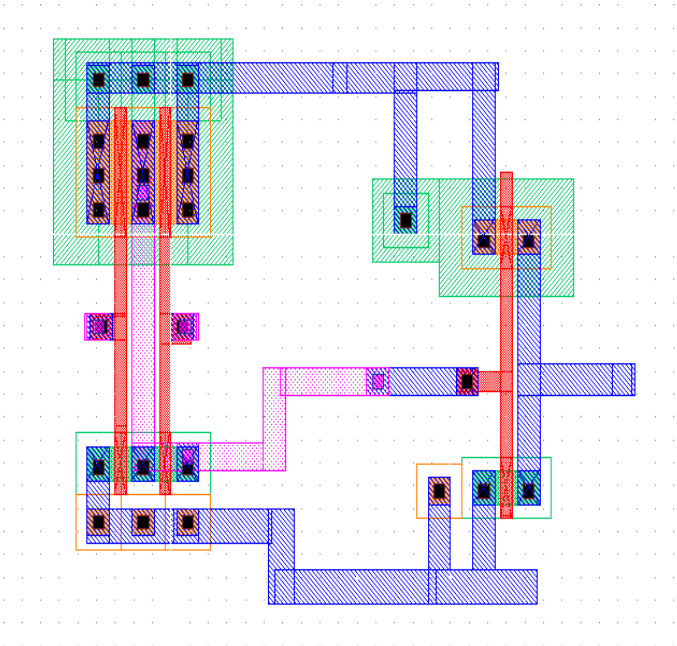
NAND {CMOS Logic}



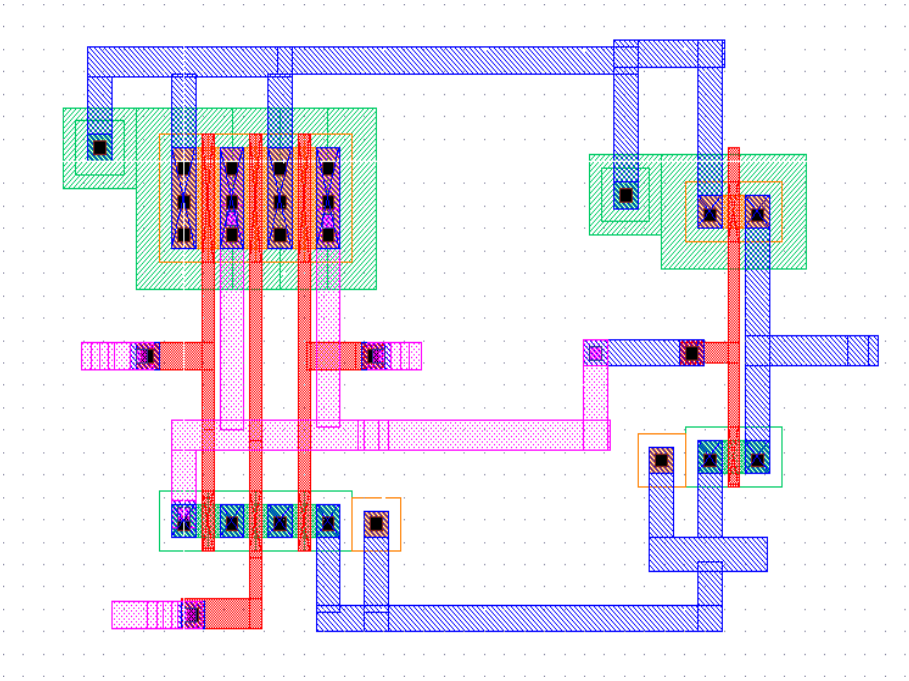
NOR {CMOS Logic}



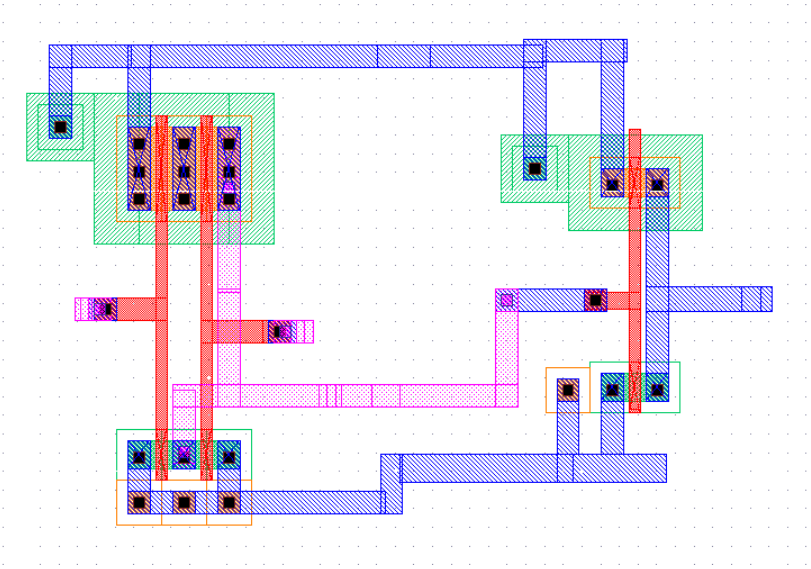
2 input AND gate {CMOS Logic}



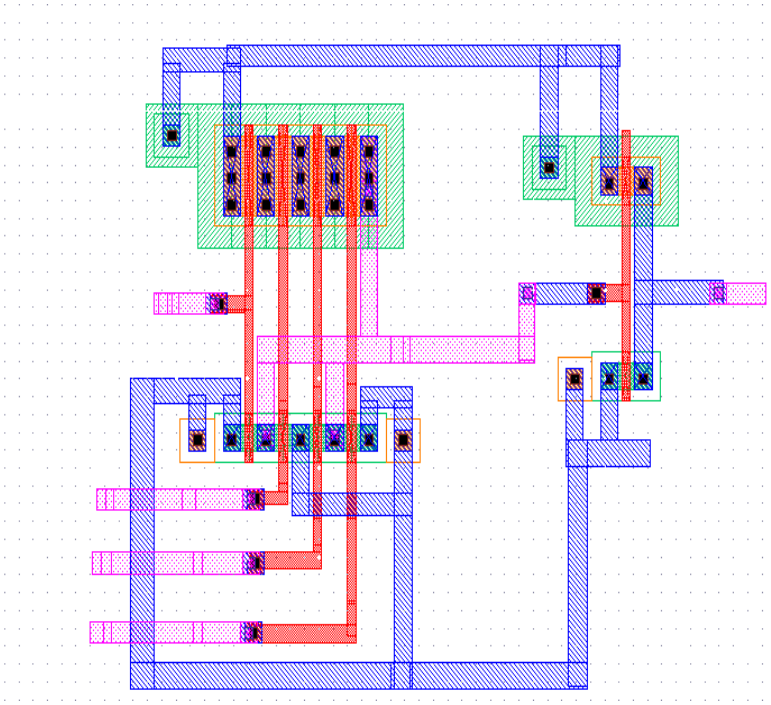
3 input AND gate {CMOS Logic}



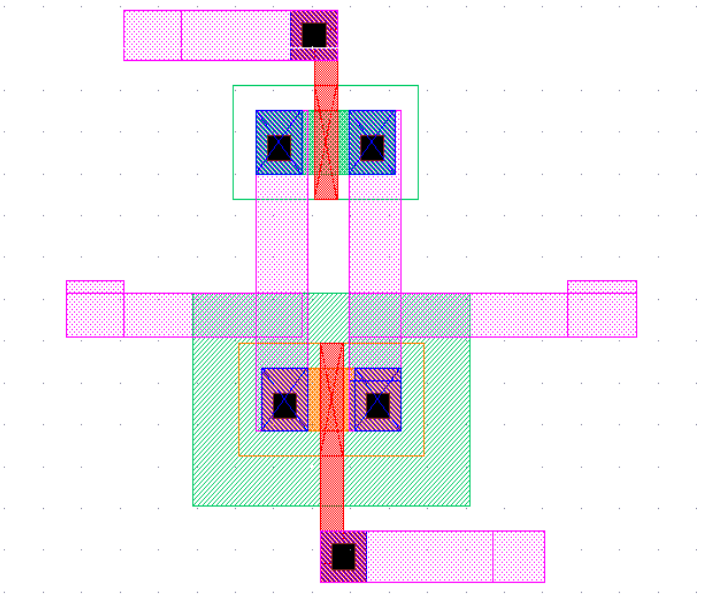
2 input OR gate {CMOS Logic}



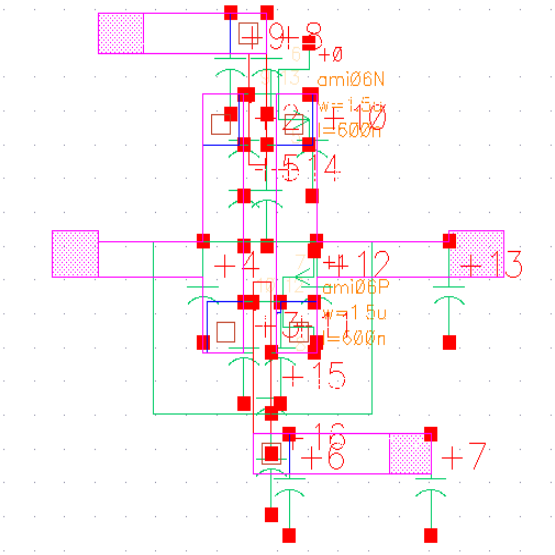
4 input OR gate {CMOS Logic}



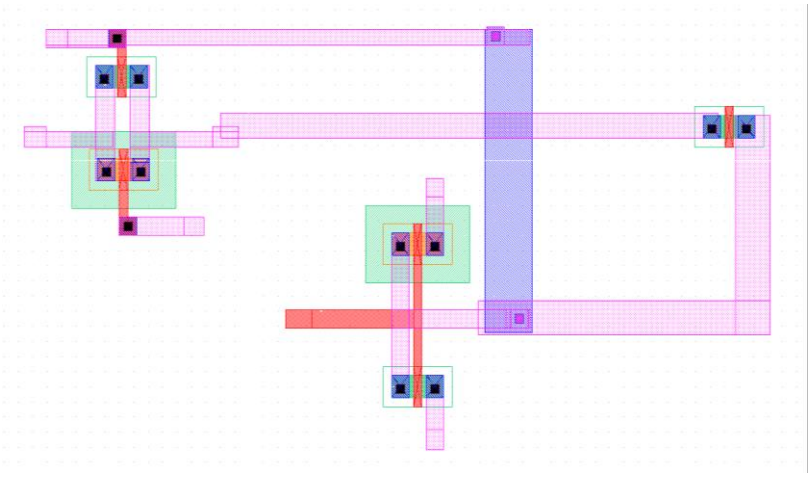
Basic Transmission Gate



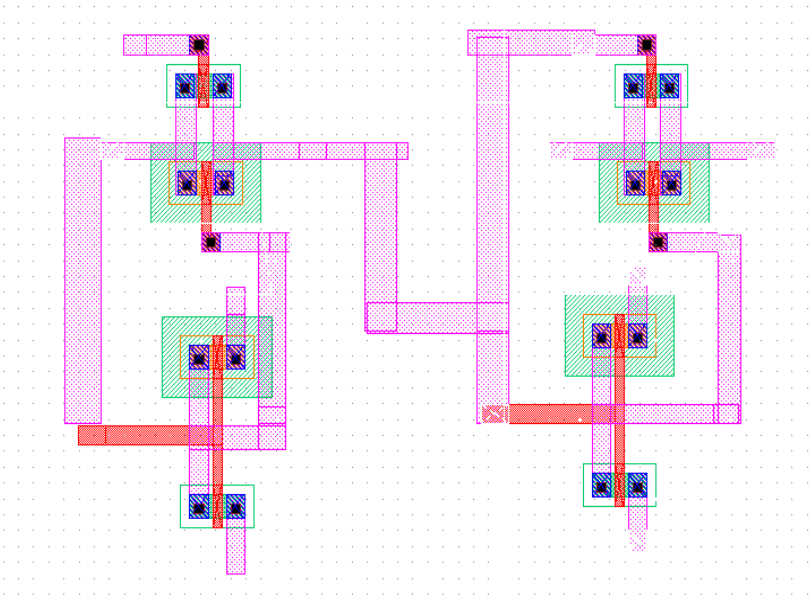
Transmission gate extraction



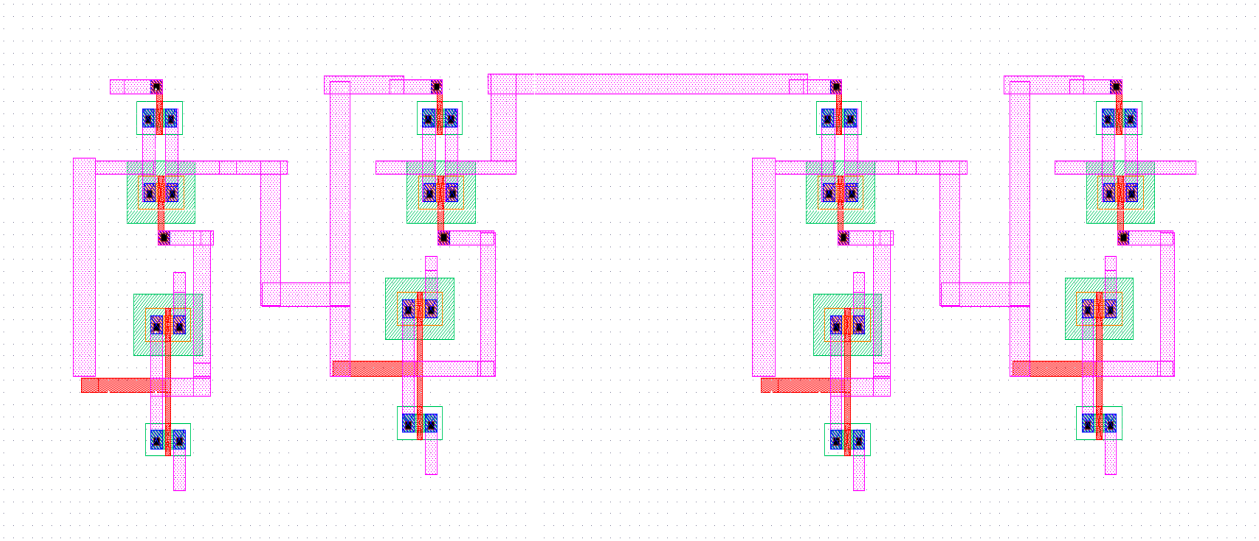
OR gate {Transmission Logic}



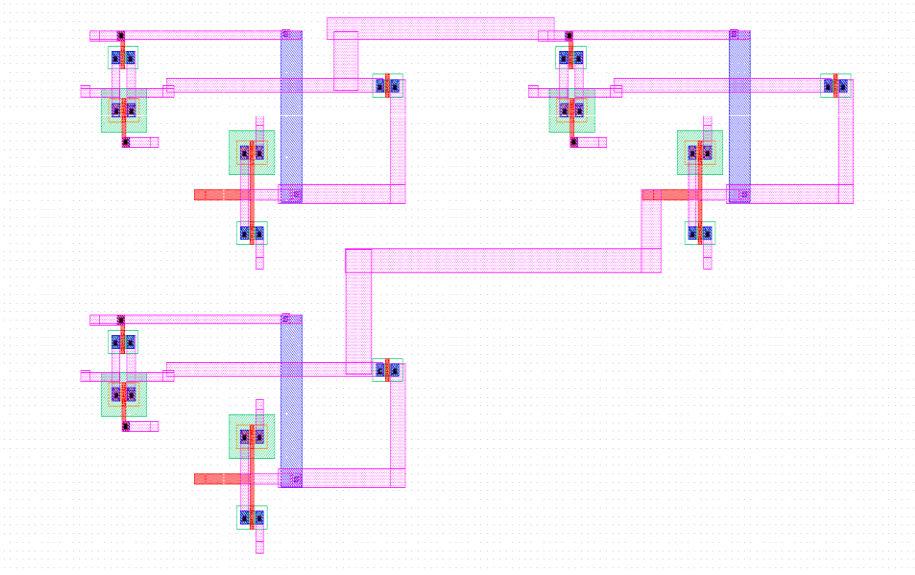
2 input AND {Transmission Logic}



3 input AND gate {Transmission Logic}

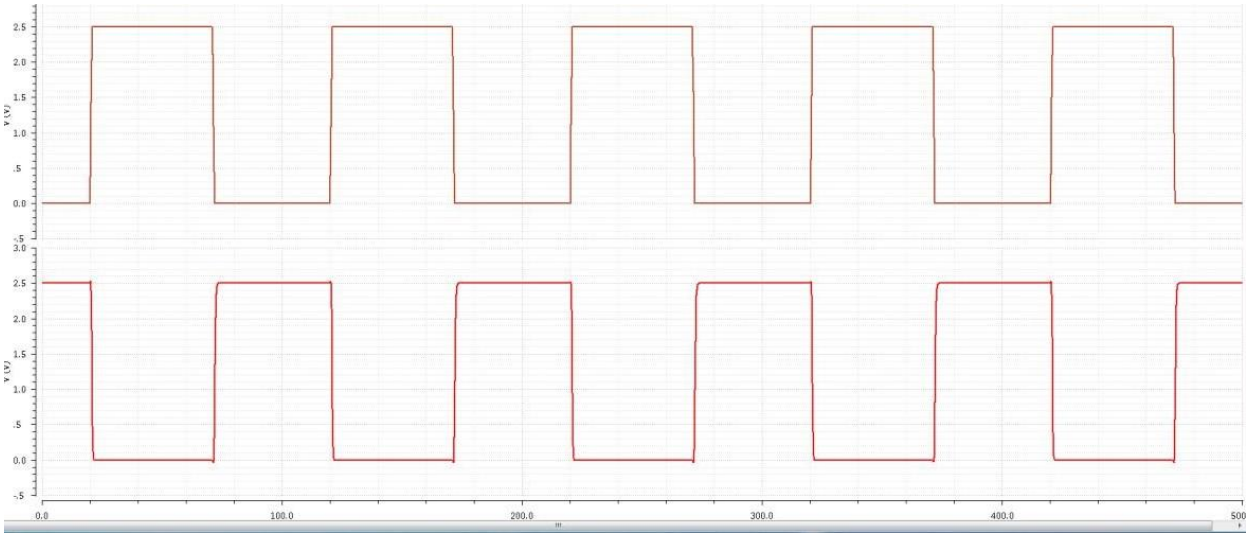


4 input OR gate {Transmission Logic}

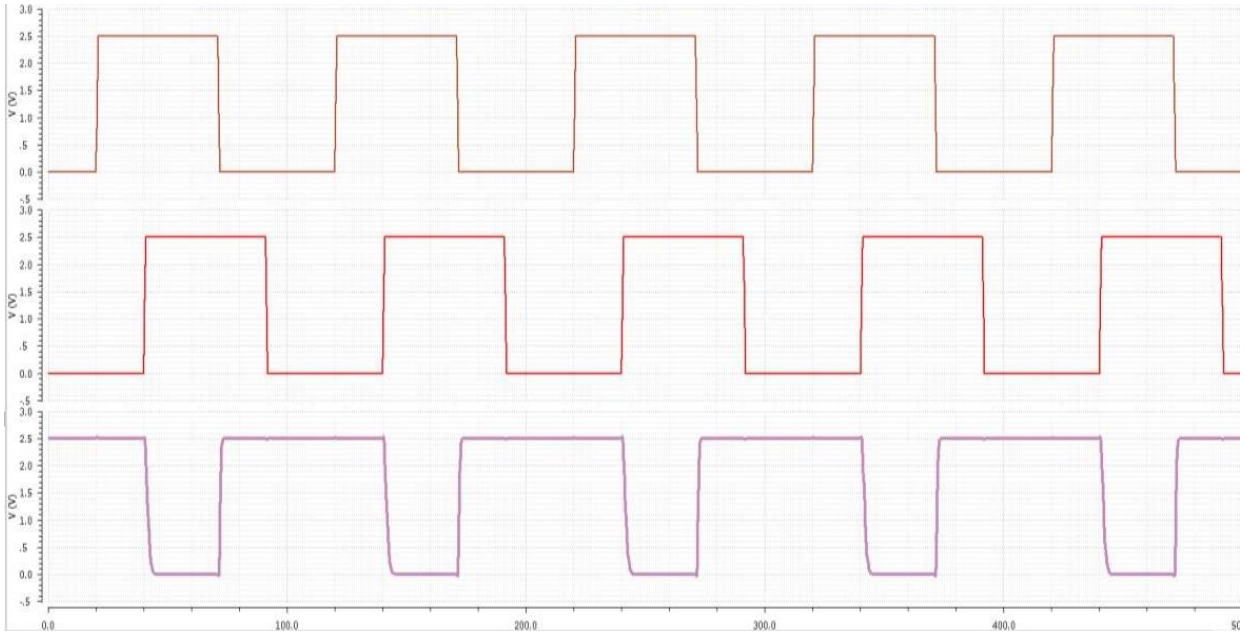


8. Post Layout Simulations

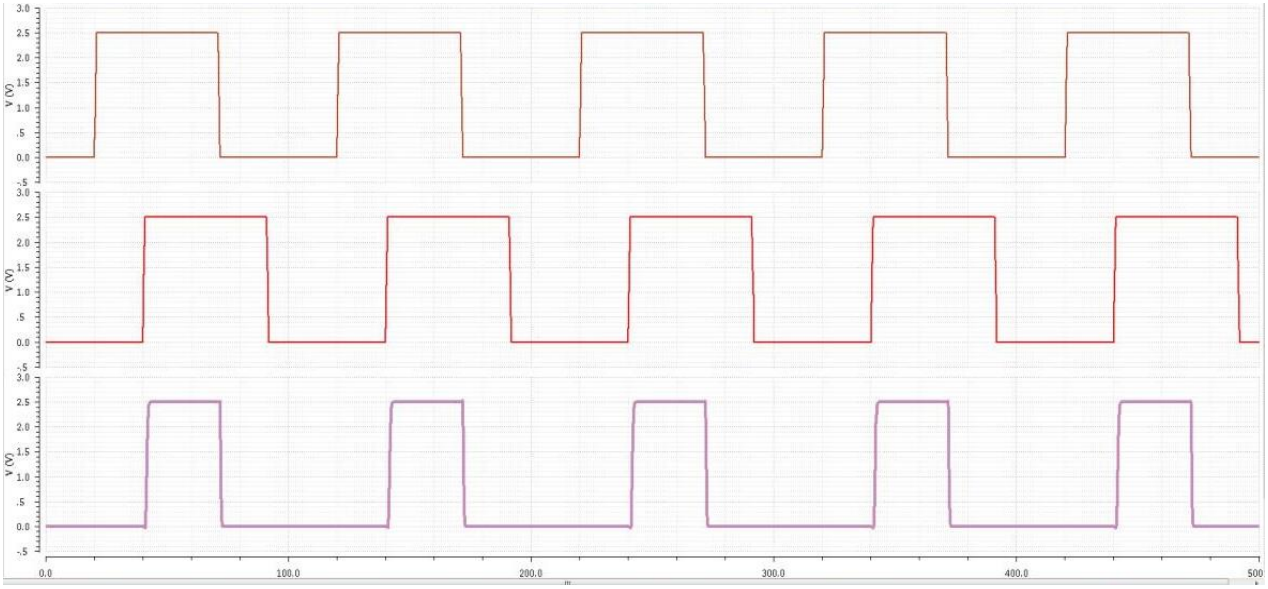
NOT Gate



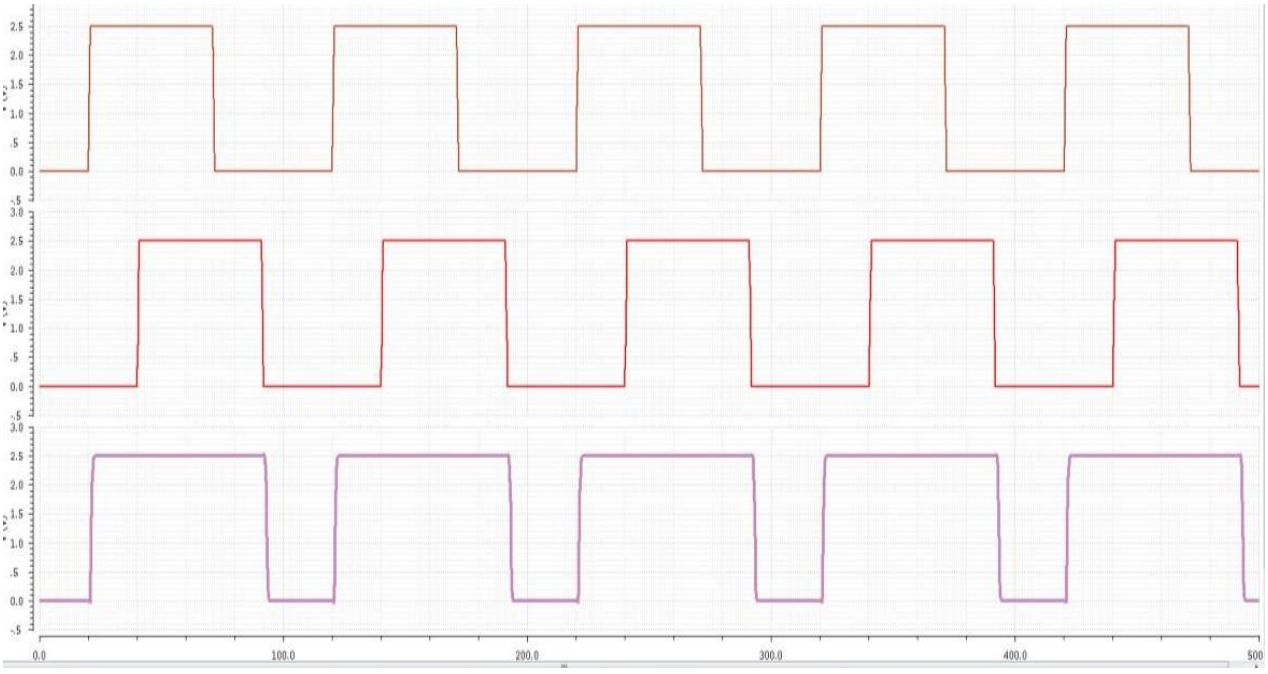
NAND Gate {CMOS Logic}



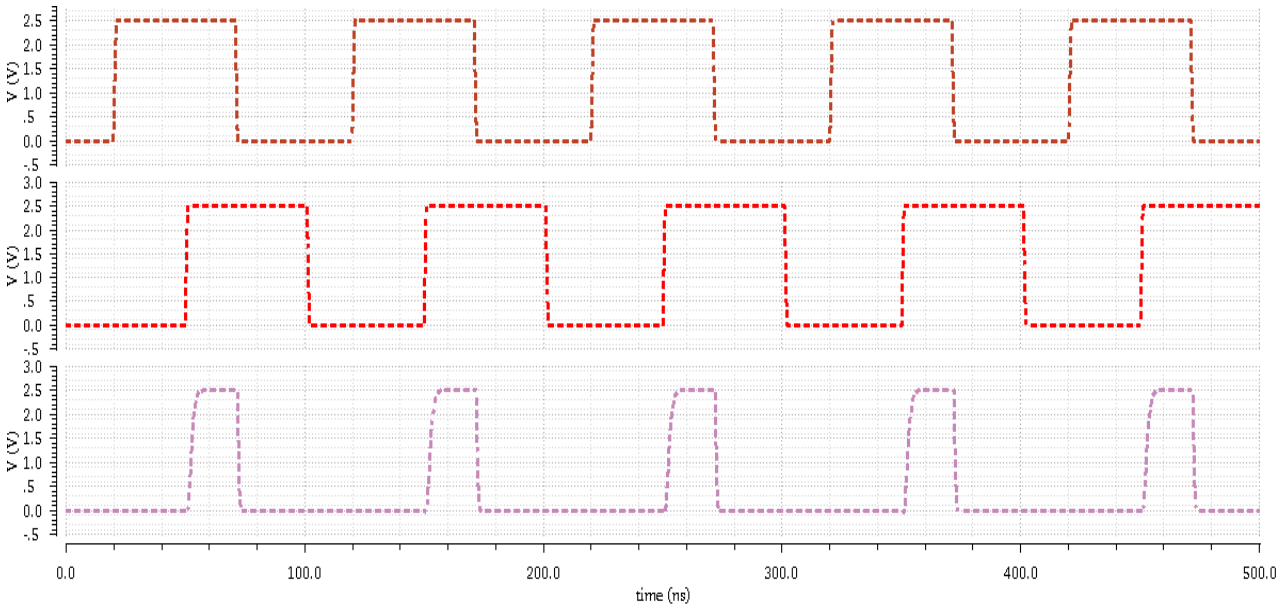
AND Gate {CMOS Logic}



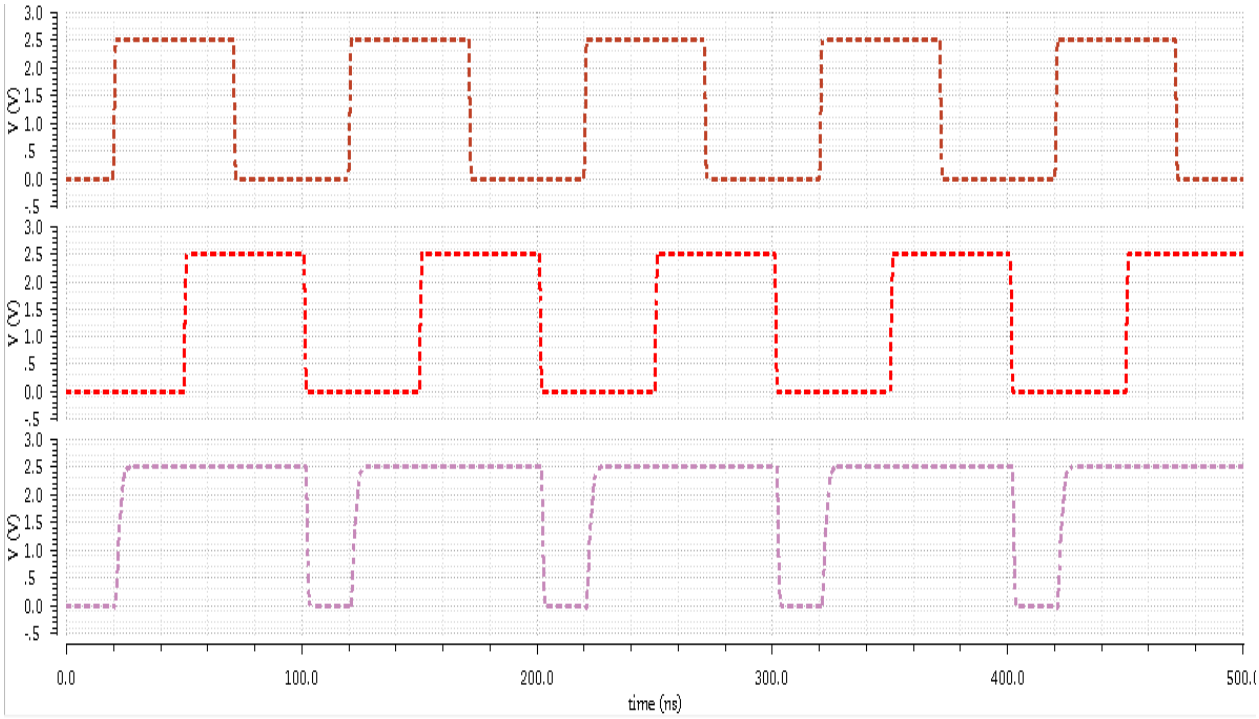
OR Gate {CMOS Logic}



AND Gate {Transmission Logic}



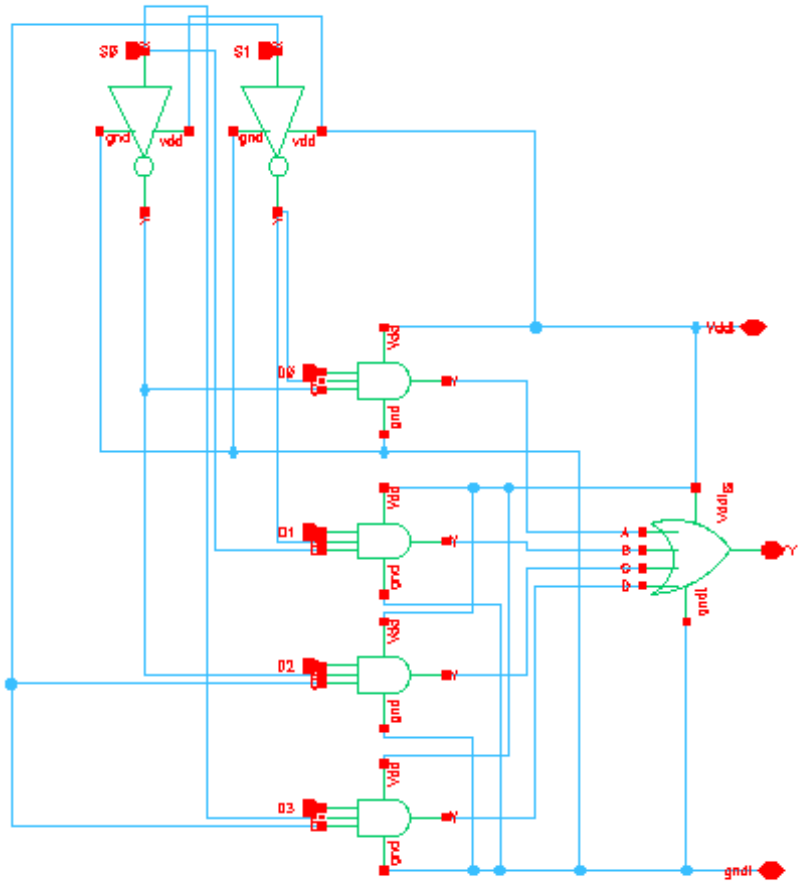
OR Gate {Transmission Logic}



9. Schematic, Symbol, Simulation and Layout of a 4:1 MUX

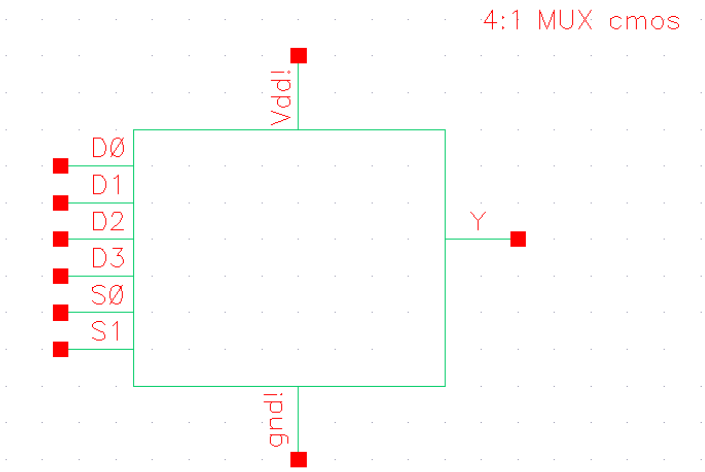
4:1 mux using CMOS logic

Schematic

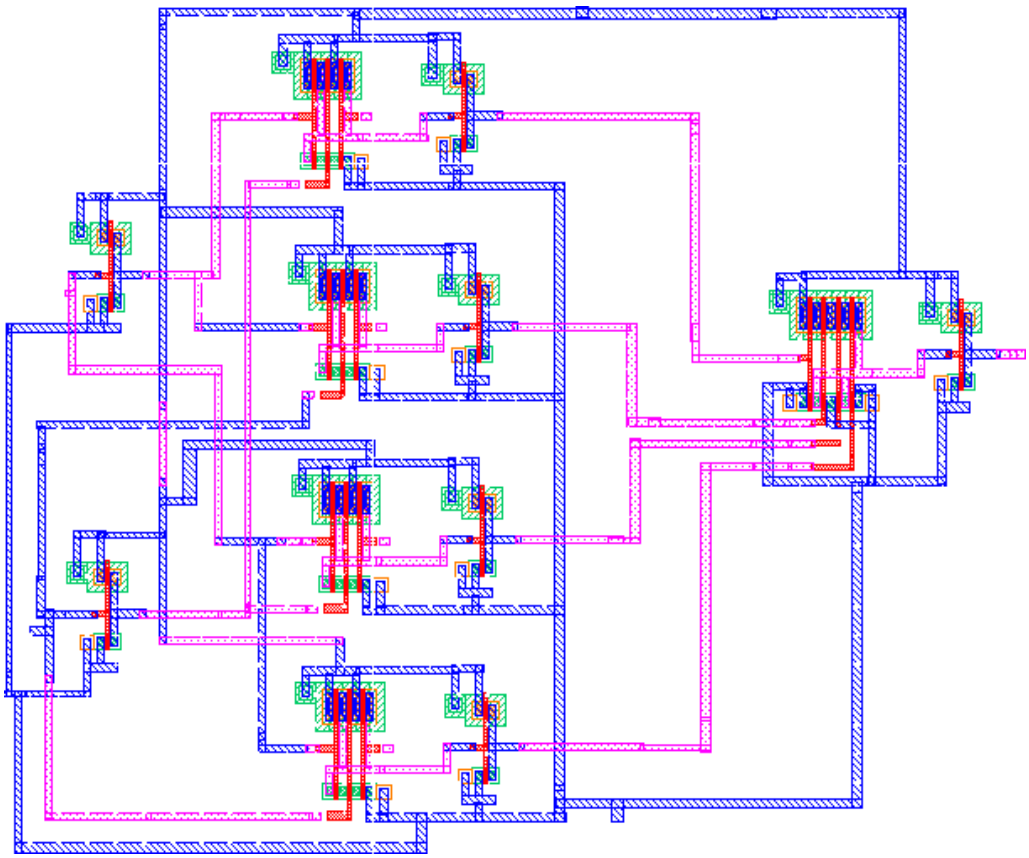


The schematic for a 4:1 Multiplexer with inputs D_0, D_1, D_2, D_3 and selection lines S_0, S_1 with output Y will be as shown above and the symbol and the layout will be as follows.

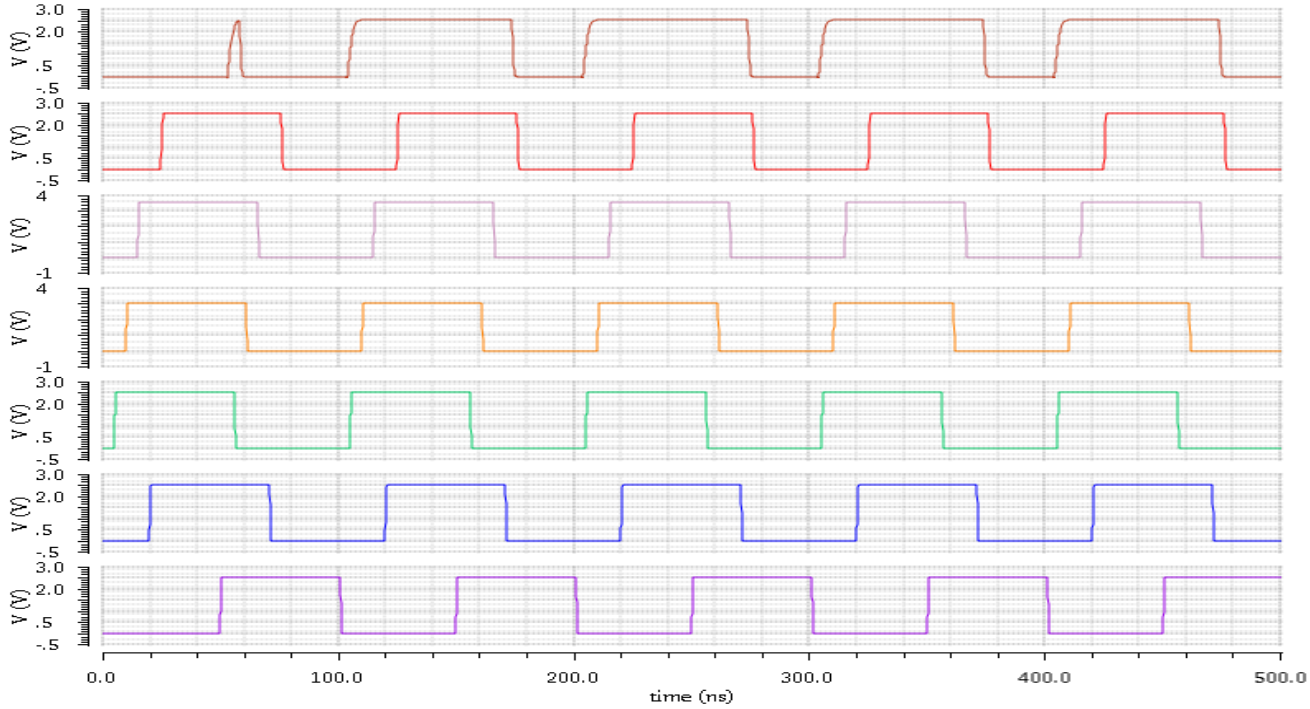
Symbol



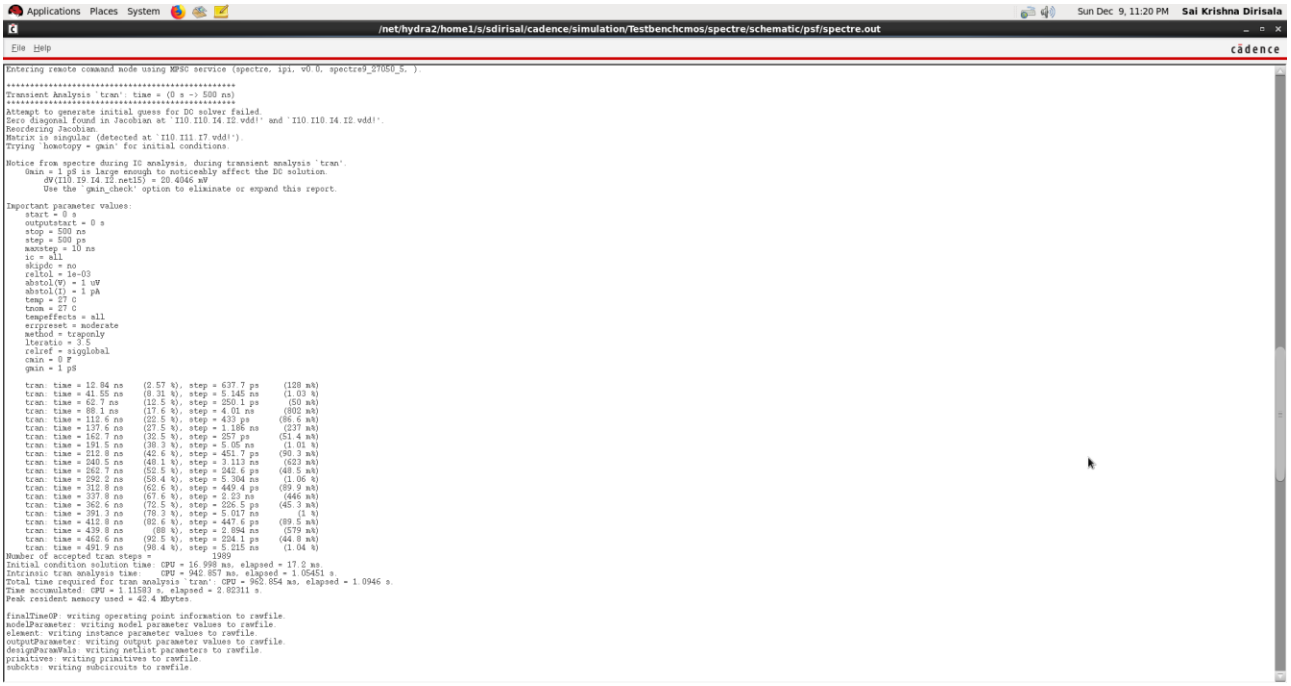
Layout



Simulation

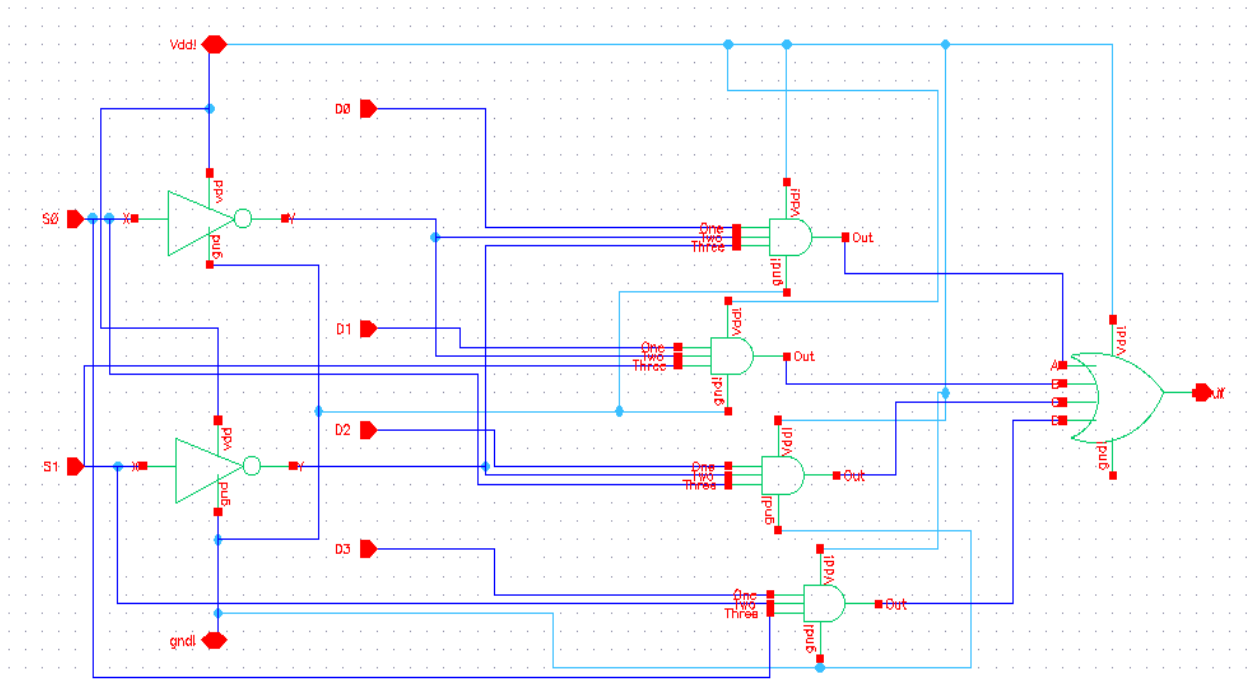


Netlist

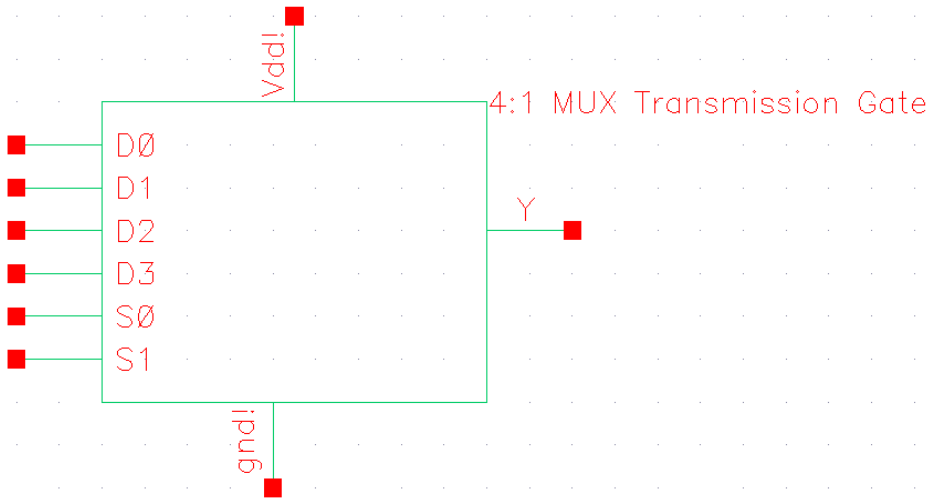


4:1 MUX using Transmission Logic

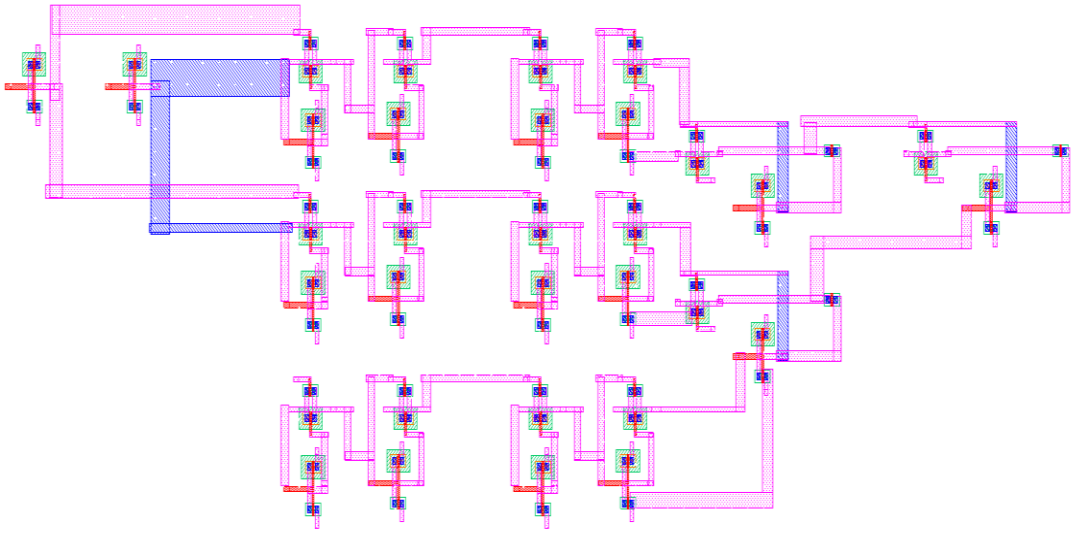
Schematic



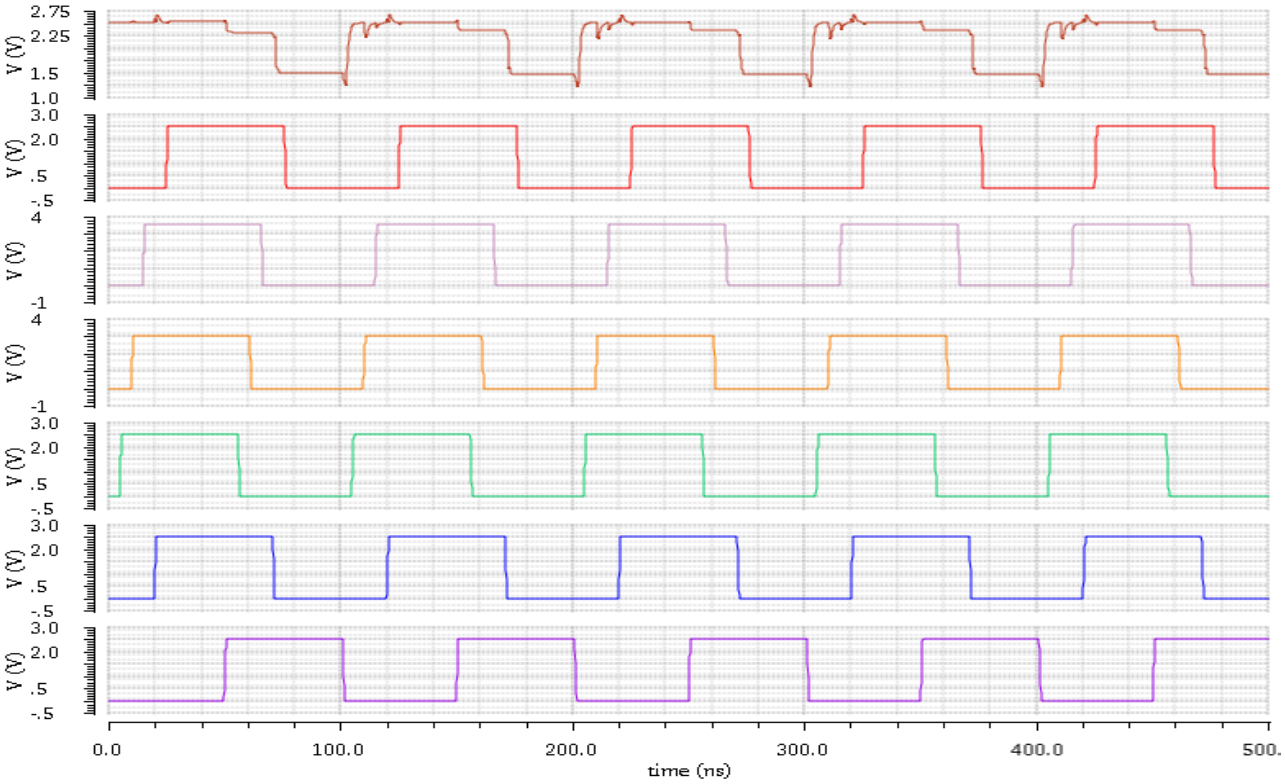
Symbol



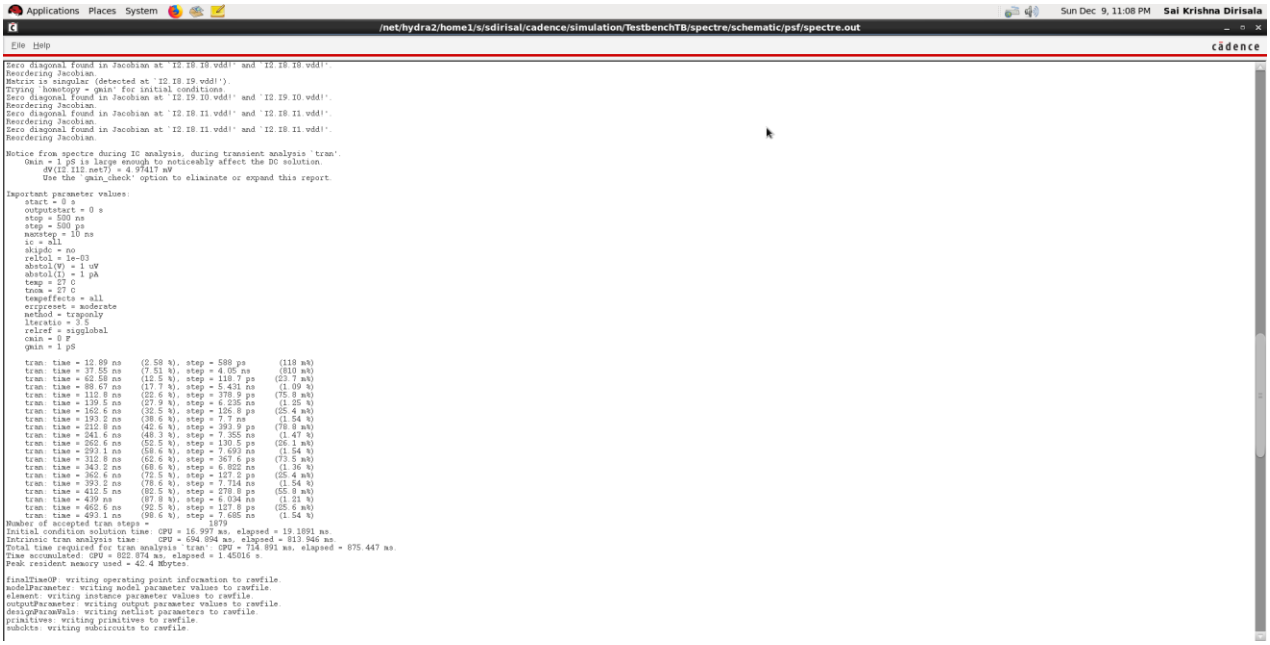
Layout



Simulation

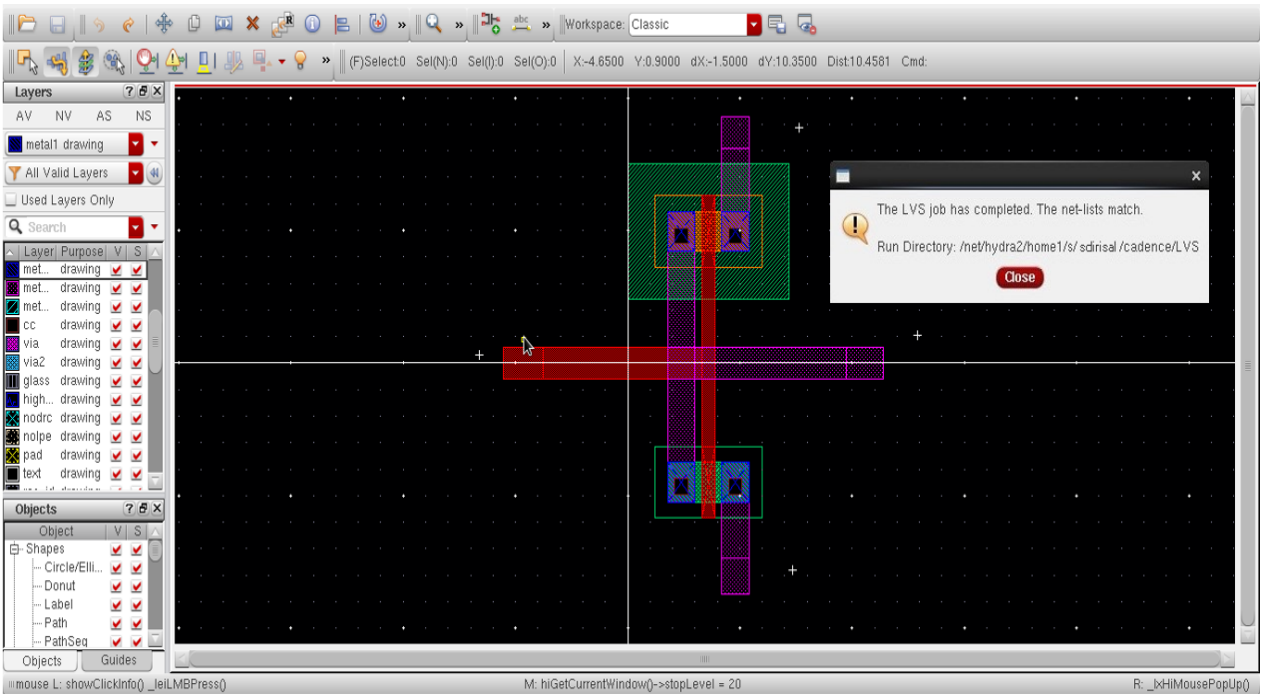


Netlist

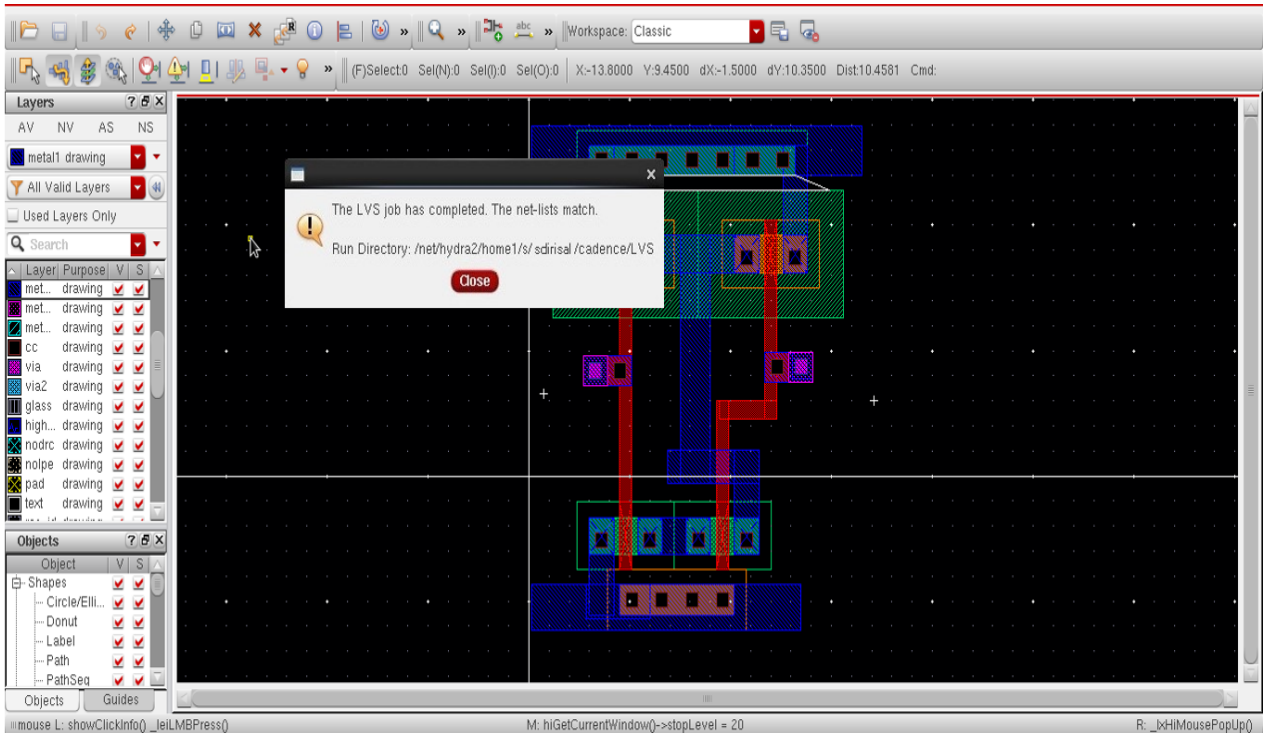


10. LVS Checks

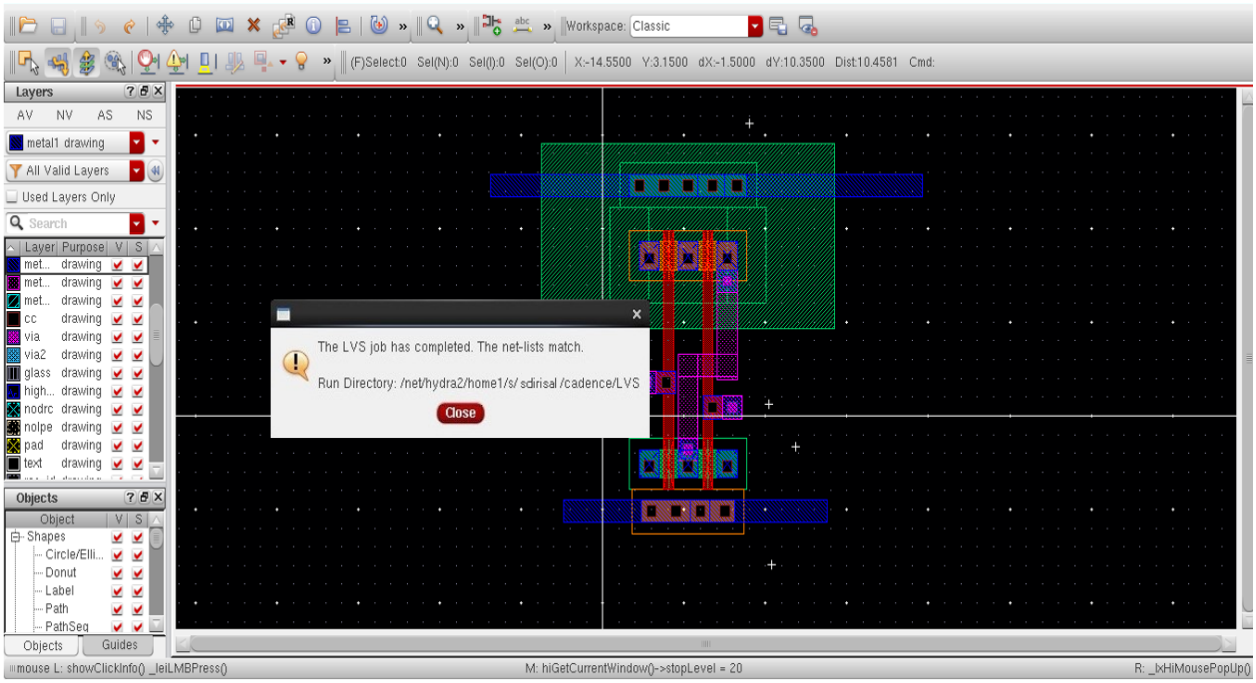
Not Gate



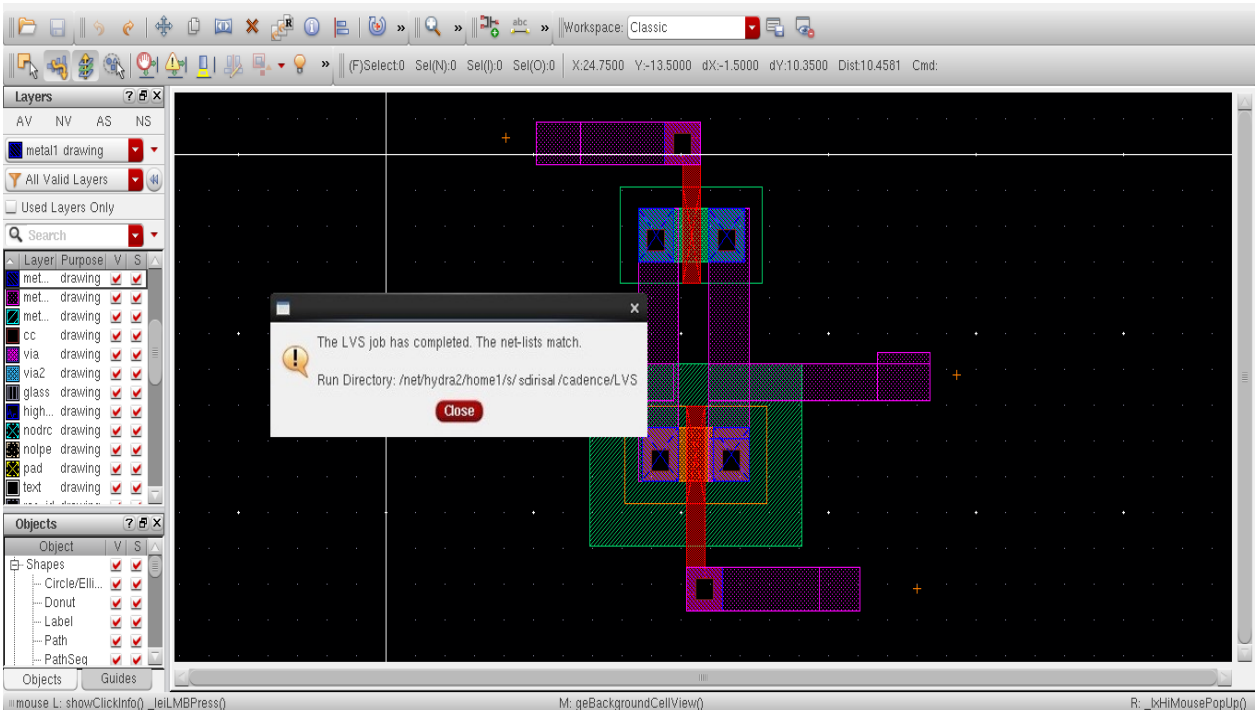
NAND gate using CMOS Logic



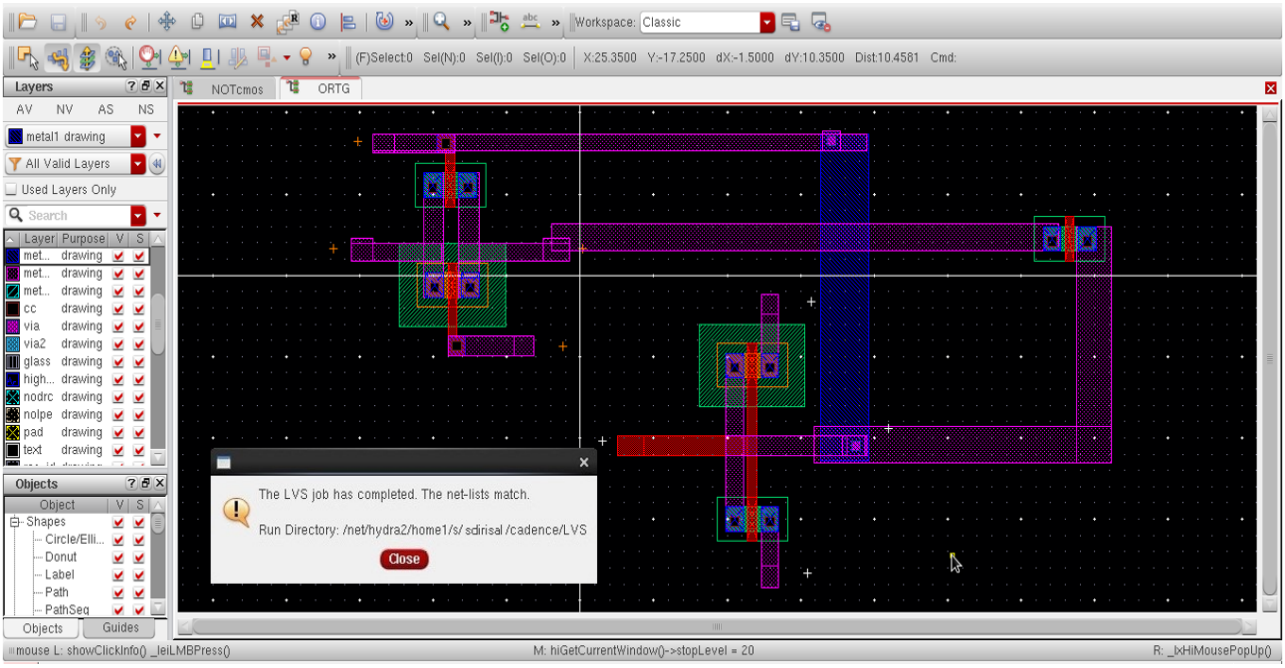
NOR gate using CMOS Logic



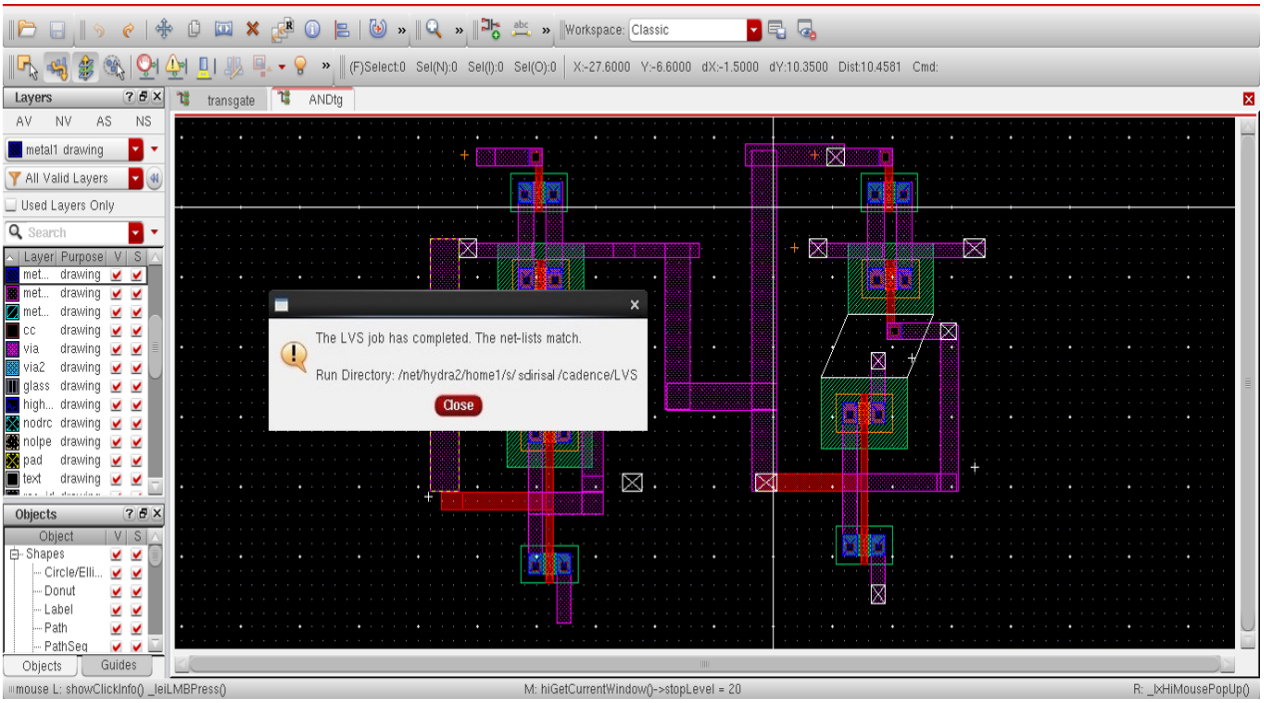
Basic Transmission Gate



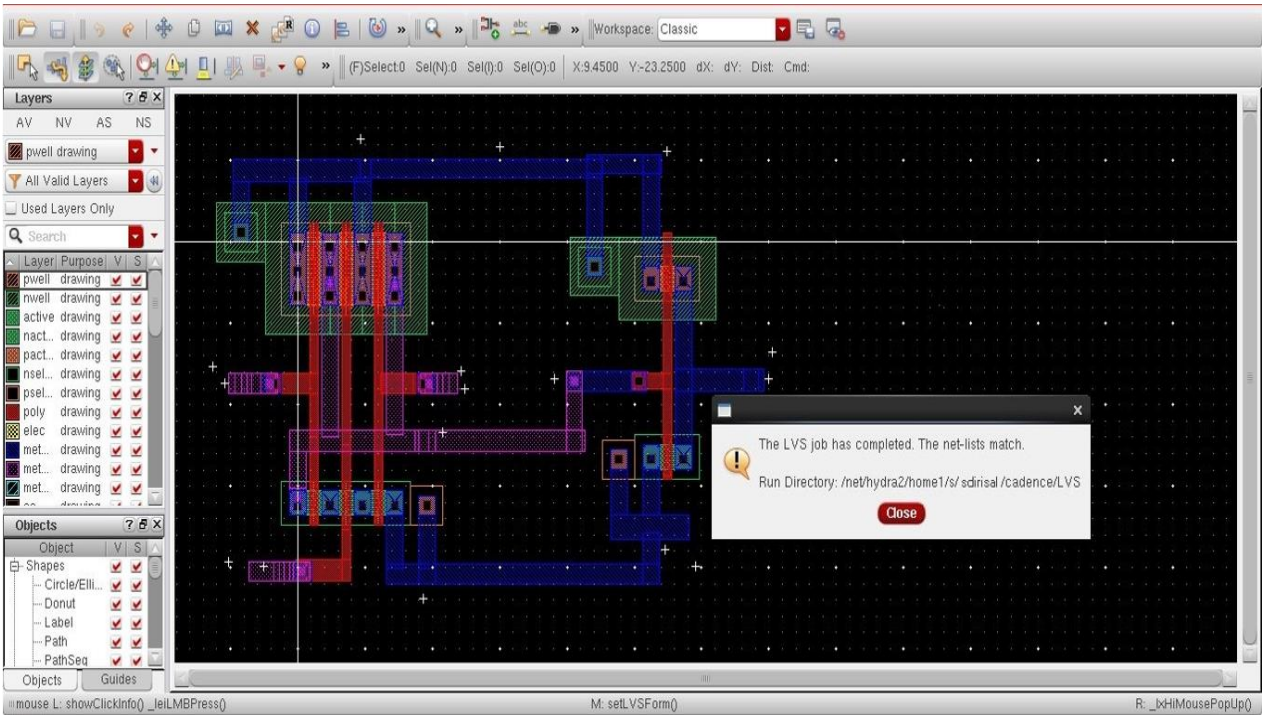
OR gate using Transmission Logic



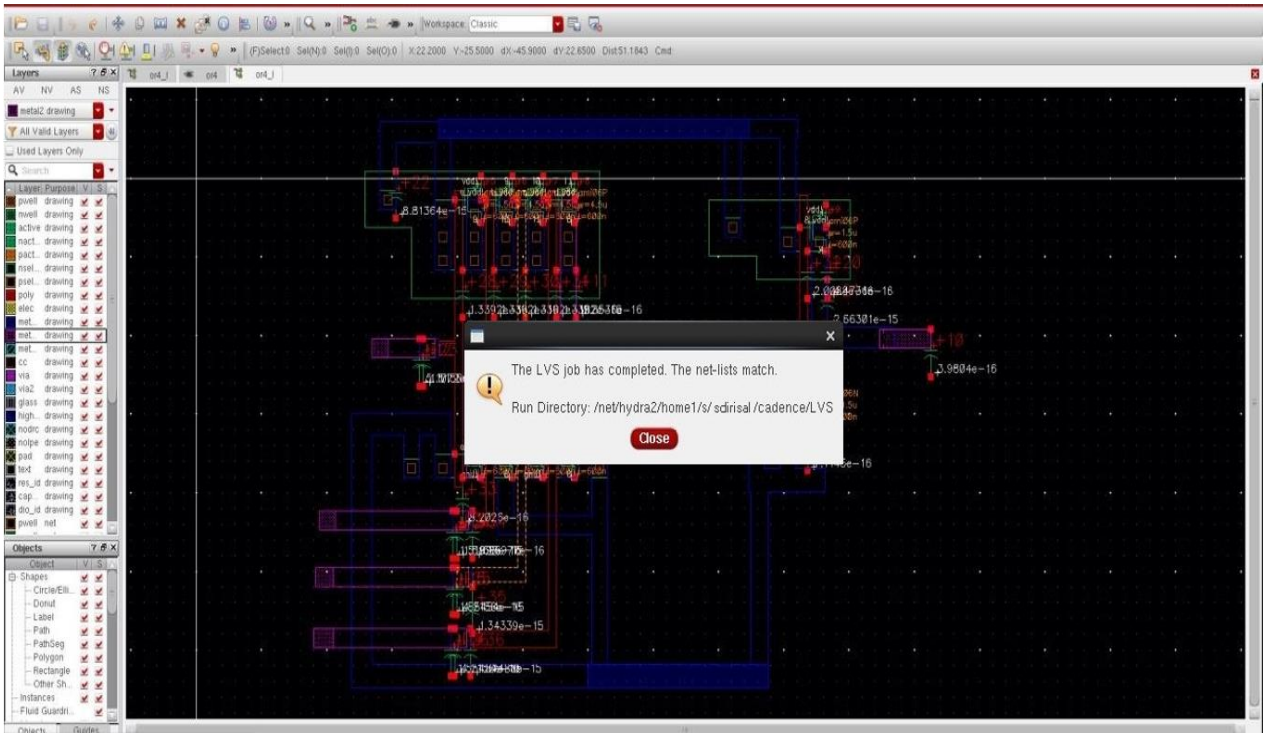
AND gate using Transmission Logic



3 input AND gate using CMOS Logic



4 input OR gate using CMOS logic



11. Comparing both design technique logics - CMOS and Transmission gate

The number of transistors in a device can be termed as transistor count. It is the most common measure of integrated circuit size. As the number of transistor count in the digital circuits increases the area also increases and vice-versa. Usually the CMOS circuits have a number of transistors compared with the one in TG circuits, resulting in which they occupy more area. This also leads to the further concept of Power Dissipation, which is of two types namely Static and Dynamic. The Static Dissipation is due to sub threshold conduction through OFF transistors, tunneling current through gate oxide, leakage through reverse-biased diodes and contention current in rationed circuits. The charging and discharging of load capacitances, also the short circuit current while both PMOS and NMOS networks are partially ON leads to the Dynamic Dissipation.

The other important factor effecting the digital circuits is the speed which can be found by the time delay in the circuits, wherein the delay depends on maximum clock frequency. All these mentioned parameters can be tabulated as shown below for the better understanding of the results.

Parameter	Technique Used	
	CMOS	Transmission
Transistor Count	High	Low
Speed	Low	High
Power Dissipation	High	Low
Area	More	Less
Frequency	Equal	Equal
Complexity	More	Less

11. Conclusion

The advances in CMOS processes are generally a bit complex and hence they inhibit the visualization of all the mask levels that are used in actual fabrication process. The advantage of MUX design had a remarkable gain in terms of transistor count by the introduction and usage of transmission gate logic. Thus is the design of transmission gate type MUX structure implemented is compared to conventional CMOS based design in our project. We started with schematic design of both the logics and implemented their layouts, to finally compare the simulations obtained in both the cases resulting in a conclusion showing that usage of transmission gates are a bit productive comparing with the traditional back to back connected PMOS and NMOS circuits.

12. References

[1] Uyemura, John P. (John Paul), 1952- “Introduction to VLSI circuits and systems”, Wiley; 1st edition (July 30, 2001)

[2] A.G.Dickinson and J.S.Denker,(1995) "Adiabatic Dynamic Logic," IEEE. Journal of Solid-state Circuits, Vol. 30 No.3, pp 311-315.

[3] R. Zimmermann and W. Fichtner, \Low-power logic styles: CMOS versus pass- transistor logic,"IEEE J. Solid-State Circuits, vol. 32, no. 7, pp. 1079{1090, Jul. 1997.

[4] Woong CJ, Roger CY. Transistor and pin reordering for leakage reduction in CMOS circuits. Microelectronics Journal (ELSEVIER). 2016; 53:25–34.

[5] Neha Pannu and Neelam Rup Prakash, “A Power-Efficient Multiplexer using Reversible Logic” Indian Journal of Science and Technology, Vol 9(30), DOI: 10.17485/ijst/2016/v9i30/94689, August 2016

[6] https://link.springer.com/article/10.1007/s13204-013-0206-0

[7] M.Padmaja, V.N.V.SatyaPrakash, “Design of a Multiplexer In Multiple Logic StyleS. for Low Power VLSI” International Journal of Computer Trends and Technology- volume3Issue3- 2012.