EN 2110

ELECTRONICS III

CIRCUIT DESIGN & SIMULATIONS

GROUP 32

Sandeepa H.K.C.A.	180564F
Vitharana N.	180672J
Hansanganie K.H.	180220A

CONTRIBUTIONS

1.0 Parasitic effect in Timing analysis

- 1.1 Introduction - Hansanganie K.H.
- 1.2 Waveforms & Results - Hansanganie K.H.
- 1.3 Discussion - Hansanganie K.H.

2.0 Programmable Logic Device (PLD)

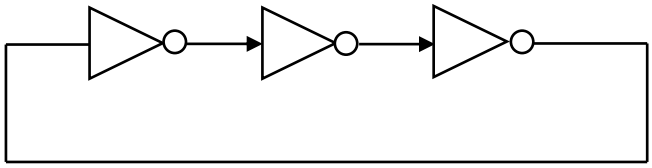
- 2.1 Design a Programmable Logic Block
  - 2.1.1 Introduction - Sandeepa H.K.C.A. / Hansanganie K.H.
  - 2.1.2 Waveforms & Results - Vitharana N.
  - 2.1.3 Discussion - Vitharana N.
    - 2.1.3.1 Problems Faced & Solutions - Vitharana N.
- 2.2 Design a Single Switch Matrix
  - 2.2.1 Introduction - Vitharana N.
  - 2.2.2 Waveforms & Results - Vitharana N.
  - 2.2.3 Discussion - Vitharana N.
    - 2.2.3.1 Problems Faced & Solutions - Vitharana N.
- 2.3 Design a PLD.
  - 2.3.1 Introduction - Sandeepa H.K.C.A.
  - 2.3.2 Waveforms & Results - Sandeepa H.K.C.A.
  - 2.3.3 Discussion - Sandeepa H.K.C.A.
    - 2.3.3.1 Problems Faced & Solutions - Sandeepa H.K.C.A.

1.0 Parasitic effect in Timing analysis

1.1 Introduction

In this assignment, the given task is to design a 3-stage ring oscillator (RO) and to analyze the correlation between parasitic effect and oscillation time.

A conventional ring oscillator is basically an odd number of inverters connected as a chain; the output is connected to the input as feedback. The below figure shows the basic 3-stage ring oscillator.



So, we used 3 basic CMOS inverters for our schematic.

In the analysis of MOSFETs, we considered the transitive work regime (AC). Further, the parasitic capacitances should be considered which affect the speed of operation of the MOSFETs based devices and digital circuits. Based on the physical structure of MOSFET, the parasitic capacitances can be classified into two major groups:

- The gate capacitive effect (indicated by Cox)
- The junction capacitances drain-body and source-body.

These two capacitive effects can be modeled by introducing capacitances to the MOSFET model between the four terminals: G, D, S, and B as shown in Figure01 and Figure02.

1. Gate-to-source capacitance (CGS)
2. Gate to-drain capacitance (CGD)
3. Gate-to-bulk capacitance (CGB)
4. Source-to-bulk capacitance (CSB)
5. Drain to bulk capacitance (CDB)

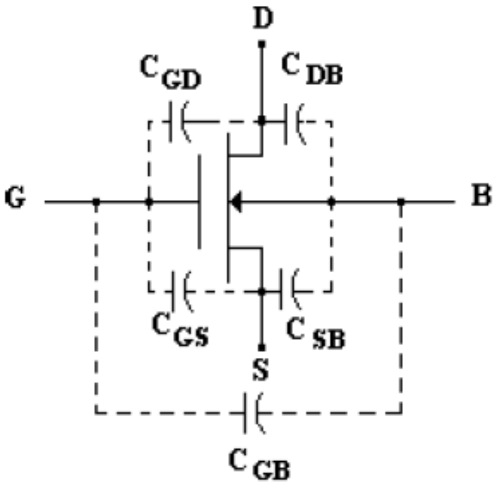


Figure01: Parasites Capacitances of the MOSFET

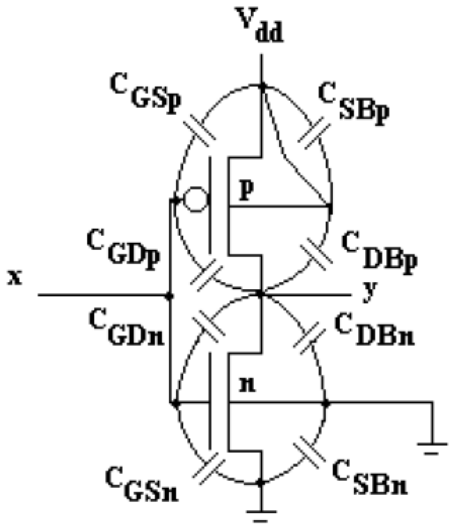


Figure02: Parasite capacitances of CMOS inverter

1.3 Waveforms & Results

The schematic we designed for the 3-stage ring oscillator is shown in Figure03. In this part, every stage consists of a basic CMOS inverter whereas M1-M3 and M4-M6 denote the MOS transistors having the W/L ratios of 10.

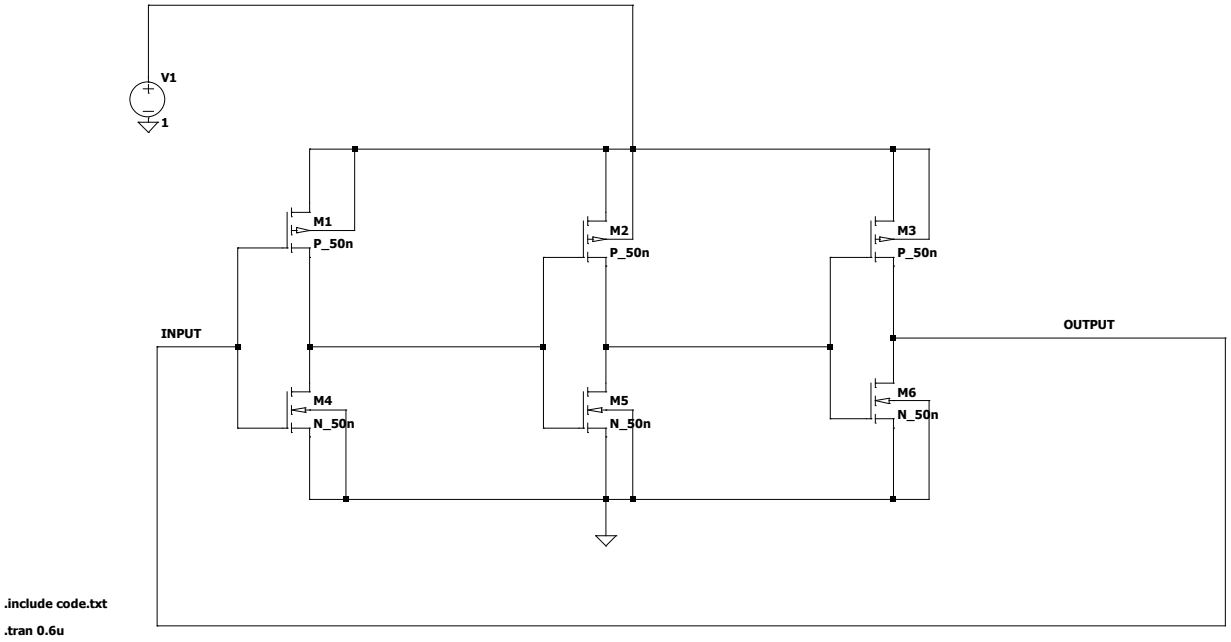


Figure03: Schematic for the Ring Oscillator

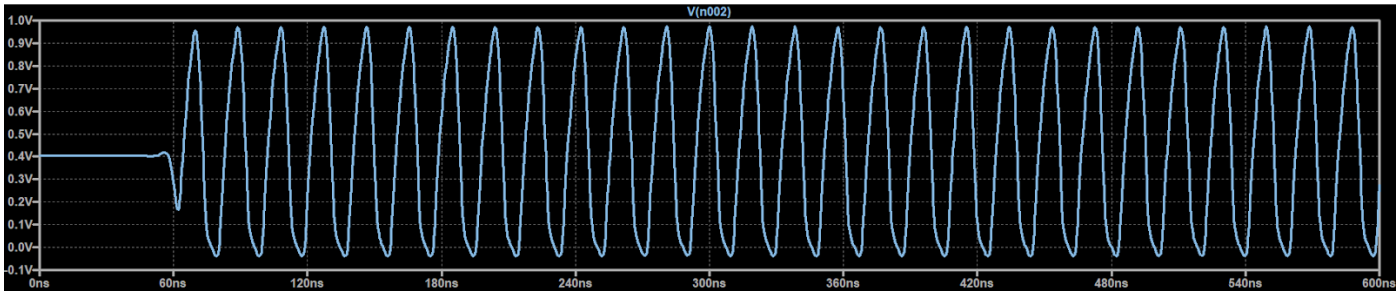


Figure04: Simulation result for the Ring Oscillator

Increase channel dimensions.

$$L = 1\mu, W = 10\mu$$

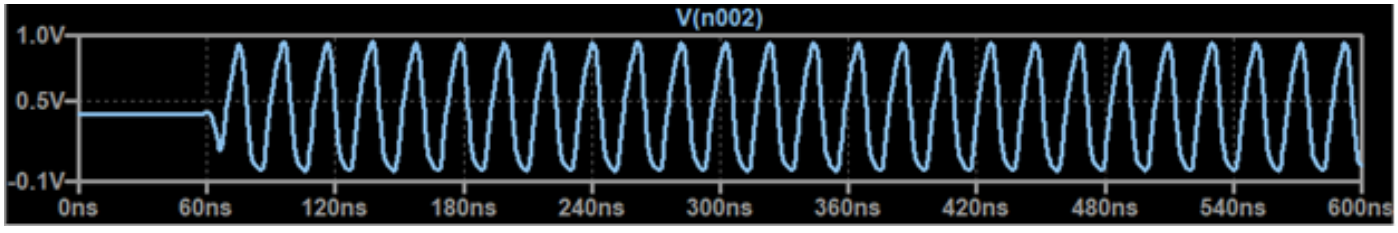


Figure05: Simulation result for L = 1u, W = 10u

$$L = 2\mu, W = 10\mu$$

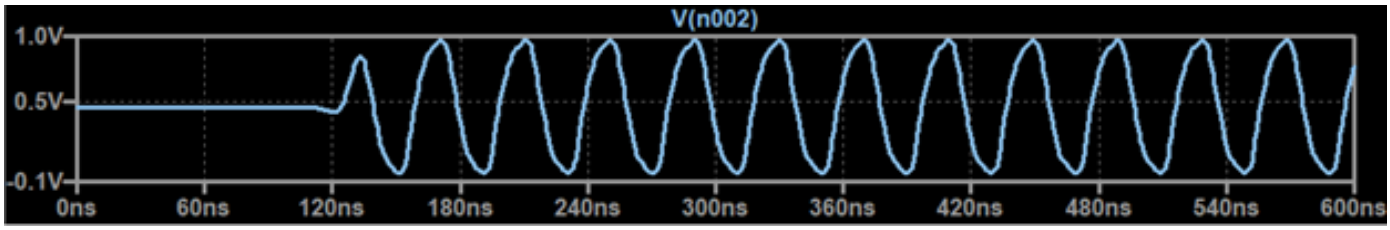


Figure06: Simulation result for L = 2u, W = 10u

$$L = 3\mu, W = 10\mu$$

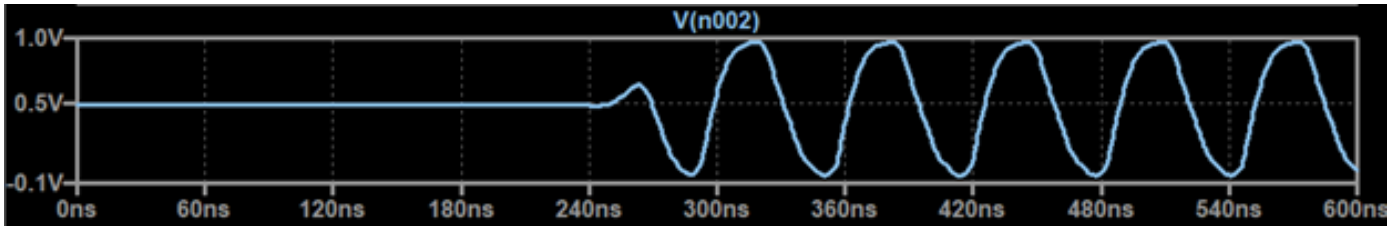


Figure07: Simulation result for L = 3u, W = 10u

When the channel dimensions are increased, gate capacitive effect increases. Due to the increase in the capacitive effect, the delay is increased. As Shown in Figures, the oscillation period is increased. (oscillation period = $2N\tau$)

Furthermore, when the channel dimensions are decreased, the oscillation period decreases.

Therefore, simulation results clearly show how parasitic capacitor values affect the oscillation period of the output waveforms.

1.4 Discussion

For a 3-stage RO, all the output nodes have the same waveform with phase difference; the output of the waveform is phase-shifted by π after passing 3-stages.

According to our simulation results,

$$period\ of\ oscillation = 19.2ns$$

$$therefore,\ \ oscillation\ frequency = 52.08MHz$$

To define internal parasitic capacitance in MOSFETS, we included a predefined dataset into schematic, defined transistors according to the dataset, and assigned W, L values for each transistor.

Because of some power losses in the circuit, the amplitudes of output waveforms are smaller than the input voltage amplitude.

By using given technical parameters given by the manufacturer and the size of the transistor (depends on W, L) the individual value for each of these parasitic capacitances can be computed as follows:

$$C_G = C_{ox}LW + 2C_o$$

$$C_{GD} = G_{GS} = \frac{1}{2}C_{ox}(WL) + C_o$$

$$C_{DB,bot} = K(V_l)C_{jo}A_D$$

$$C_{DB,sw} = K_{\frac{1}{3}}(V_l)C_{jsw}l_D$$

$$C_{DB} = C_{DB,bot} + C_{DB,sw}$$

$$C_{SB,bot} = K(V_l)C_{jo}A_S$$

$$C_{SB,sw} = K_{\frac{1}{3}}(V_l)C_{jsw}l_S$$

$$C_{SB} = C_{SB,bot} + C_{SB,sw}$$

Where,

L = channel length

W = channel width

Co = overlap capacitance

Cj0 = zero bias capacitance per unit area

Cjsw = zero-bias sidewall capacitance per unit perimeter

AD = Area of dreno

lD = perimeter of dreno

AS = Area of source

LS = perimeter of source

These gate capacitance (parasitic capacitance) must be charged before current can flow between source and drain. Thus, the output of every inverter changes within a finite amount of time after the input has changed. This causes propagation delays τ_{PHL} and τ_{PLH}, at the output of the inverter.

When all parasitic capacitances combine into a lumped capacitance C (including load and parasitic capacitances of connection), the above-mentioned propagation delays can be calculated as,

$$\tau_{PHL} = \frac{1.6C}{k_n\left(\frac{W}{L}\right)_n V_{DD}}$$

$$\tau_{PLH} = \frac{1.6C}{k_p \left(\frac{W}{L}\right)_p V_{DD}}$$

$$\tau = \tau_{PHL} + \tau_{PLH}$$

The oscillation frequency of RO can be determined by estimating this delay time τ of each inverter stage. If the number of stages is equal to N, the oscillator will take 2N time to complete the total cycle. So, the oscillation frequency can be written as:

$$f = \frac{1}{2N\tau}$$

According to the above equations, for smaller values of parasitic capacitance C, smaller delays will occur. To reduce the gate capacity effect, channel dimensions must be reduced.

Therefore, by changing channel dimensions (W & L) can change the value of parasitic capacitors and then delays must be changed accordingly. Hence, there should be differences in simulation results for each channel size. Simulation results for different channel dimensions of NMOS transistors can be found in the results section with labels.

2.0 Programmable Logic Device (PLD)

2.1 Design a Programmable Logic Block

2.1.1 Introduction

In this section, the NAND & NOR gates are implemented using a combinational logic circuit with two inputs & one selection bit. The selection bit is used to switch the output gate.

The Truth Table of the combinational logic circuit:

Selection Bit(S)	Input A	Input B	Output	
0	0	0	1	NAND Gate
0	0	1	1	
0	1	0	1	
0	1	1	0	
1	0	0	1	NOR Gate
1	0	1	0	
1	1	0	0	
1	1	1	0	

Table01: Truth Table for 2-input NAND Gate & NOR Gate

This can be simplified using the Karnaugh map.

AB / S	00	01	11	10
0	*1*	*1*	*0*	*1*
1	*1*	*0*	*0*	*0*

Table02: Karnaugh map for 2-input NAND Gate & NOR Gate

$$Output = \bar{S}\bar{A} + \bar{S}\bar{B} + \bar{A}\bar{B}$$

2.1.2 Waveforms & Results

According to the following schematic, any input converts into its invert version using the inverter. Then build the corresponding *MINTERMS* using AND plane that uses NAND gate to build. Then those *MINTERMS* are combined using OR plane; built using NAND gates.

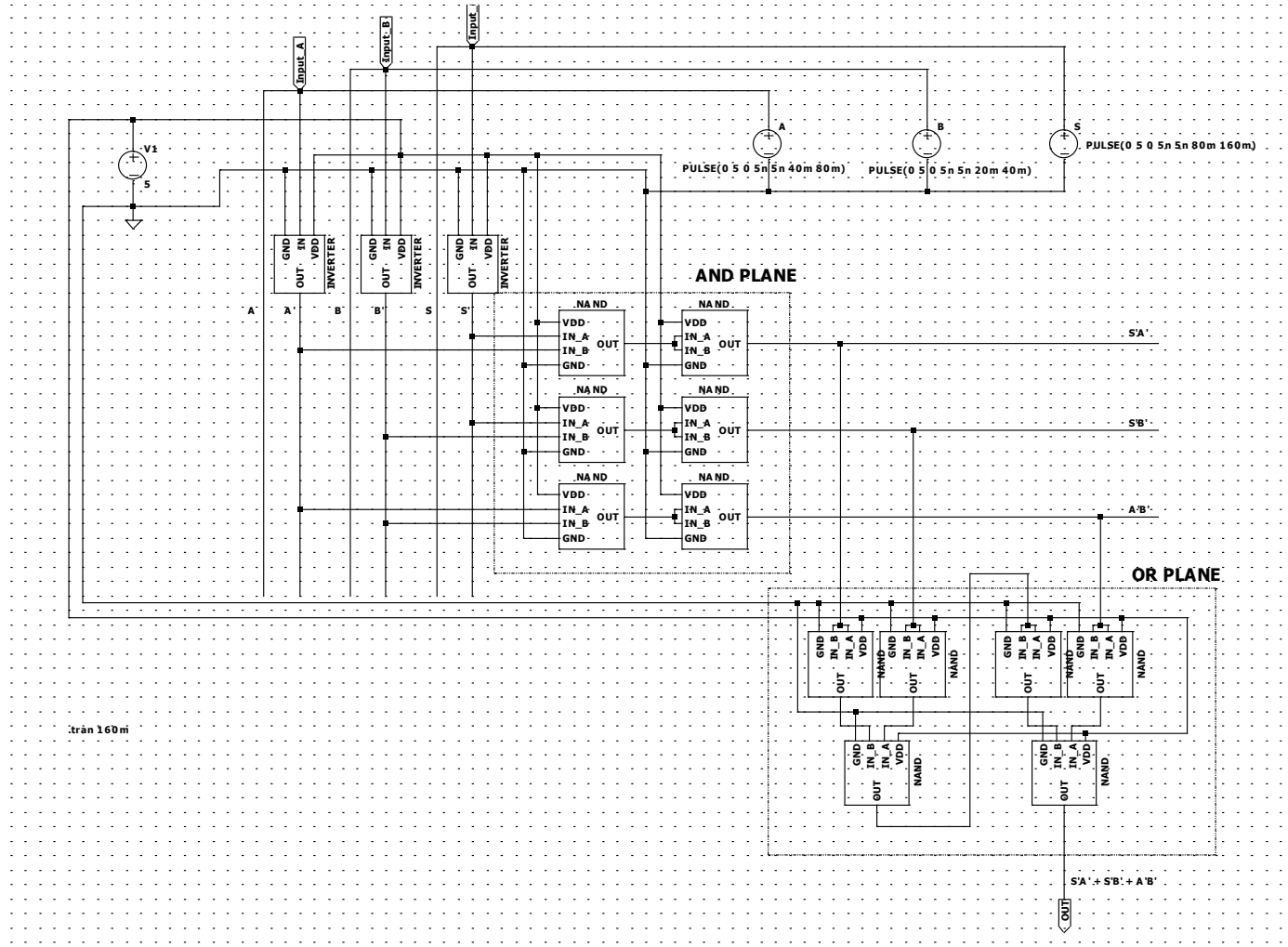


Figure08: Schematic for combinational logic circuit

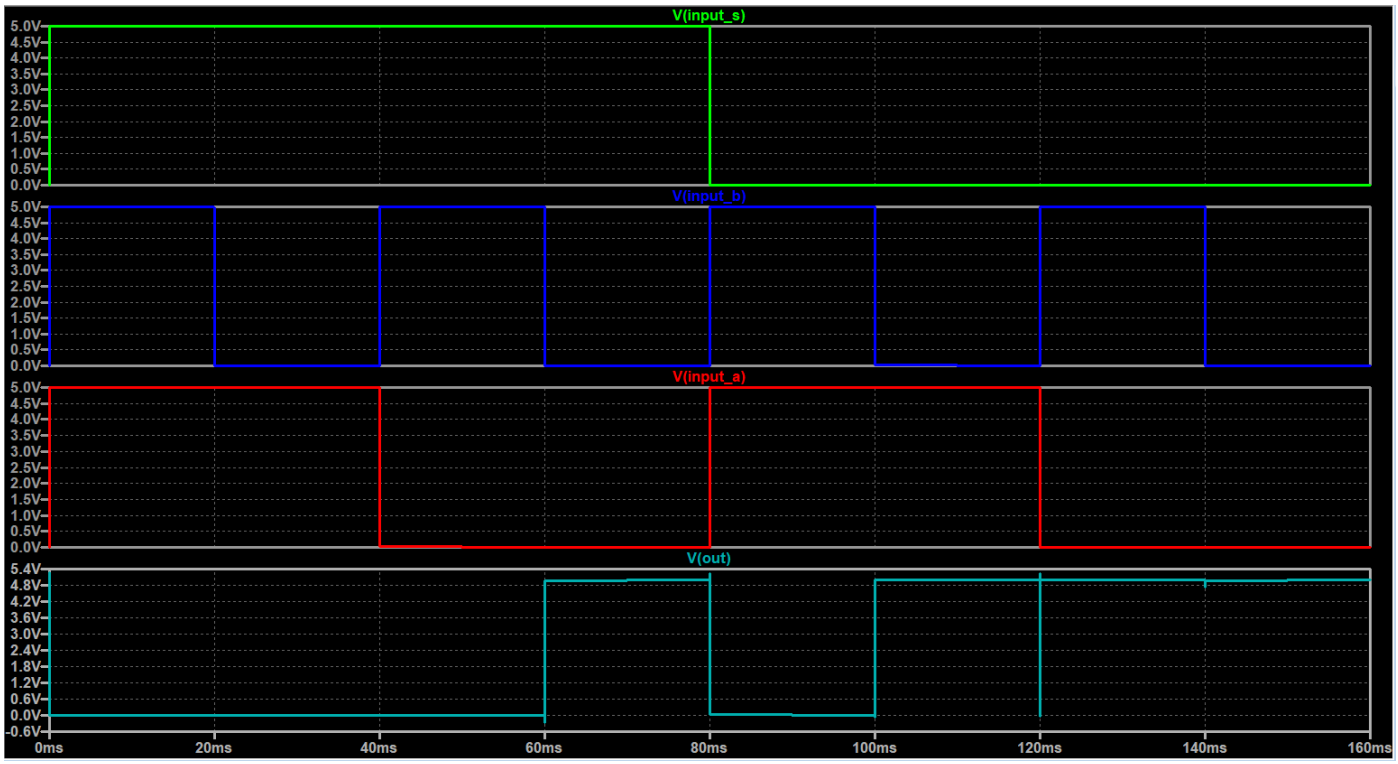


Figure09: plots for the terminals. (1) selection bit pulse pattern, (2) input A pulse pattern, (3) input B pulse pattern, (4) output pulse pattern

2.1.3 Discussion

As the resultant plot for output shows the pulse pattern for both NAND & NOR gates with type of selection. The simulation time was 160miliseconds for the corresponding input pulse pattern with selection state (0 or 1).

As we know the CMOS technology that uses to build gates has more transistors for AND & OR gates while NAND & NOR gates have fewer. It causes delay. In this case, we use NAND as the universal gate to implement both AND & OR gates.

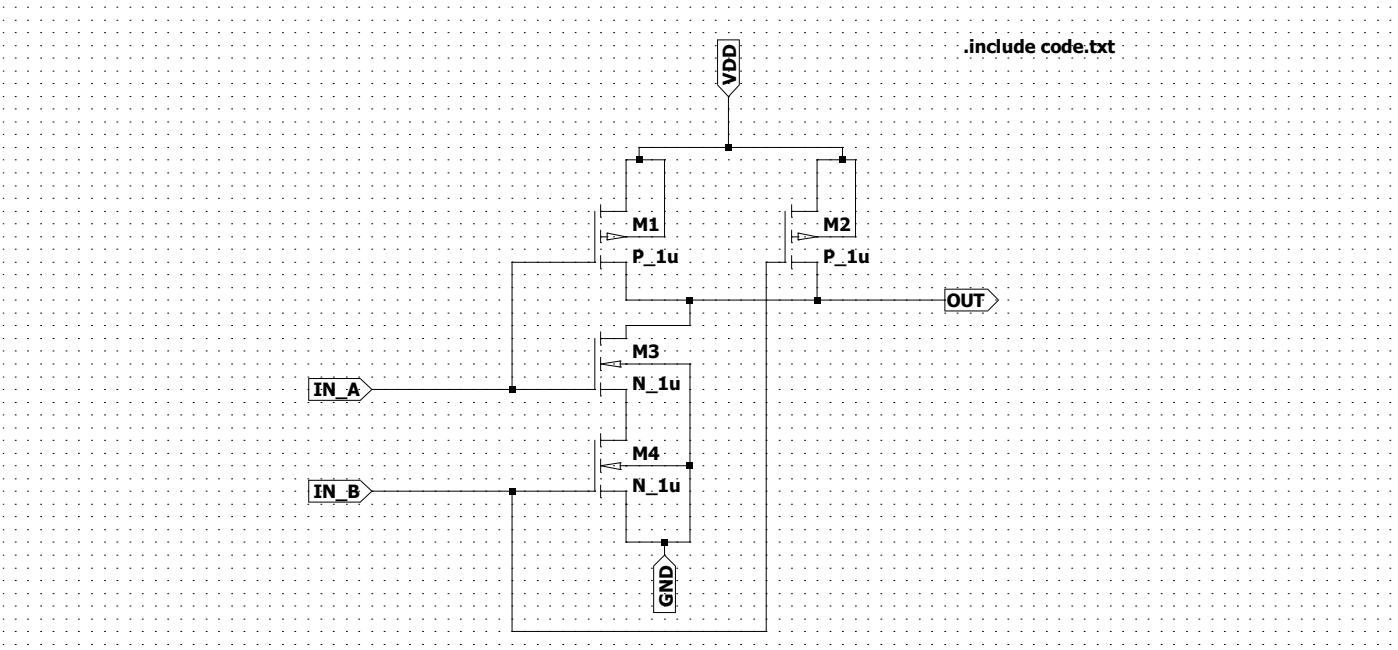


Figure10: NAND gate implementation using MOSFETS.

2.1.3.1 Problems Faced & Solutions.

Problem – Choosing the best MOSFET.

Reason - In this part, we had two choices to use either the MOSFET that has an already connected body terminal or the MOSFET that has a floating body terminal.

Solution - We choose MOSFET with floating terminal and it connected to the most negative terminal (GND) in NMOS & connected to most positive (VDD) terminal in PMOS because the channel width depends on the gate voltage. But the gate potential is measured relative to the substrate (body) potential, and the channel is formed in the substrate material. Therefore, the MOSFET behavior depends strongly on the substrate (body) potential when we use the MOSFET as a switch.

Problem – There is glitch in range of 100ms to 160ms.

Reason – This happens due to the unbalanced gate delays. It can be configured as a static 1 hazard.

2.2 Design a Single Switch Matrix

2.2.1 Introduction

The switch matrix is a switching method mainly used in Field Programmable Logic Arrays (FPGAs). The switch matrix is used to create interconnections between logic cells.

Following benefits and be gained by using switch matrix rather than using fully interconnected logic cells,

1. 6 pass switch matrices can allow all the possible routes in a 4-way junction of a circuit
2. Reduces the complexity of the circuit.
3. Delays of each path are almost the same and predictable.
4. The number of I/O pins can be fixed, because the number of input and output bits are fixed.

In this assignment, a switch matrix must be created using 6 transistors. The following circuit is used to demonstrate.

2.2.2 Waveforms & Results

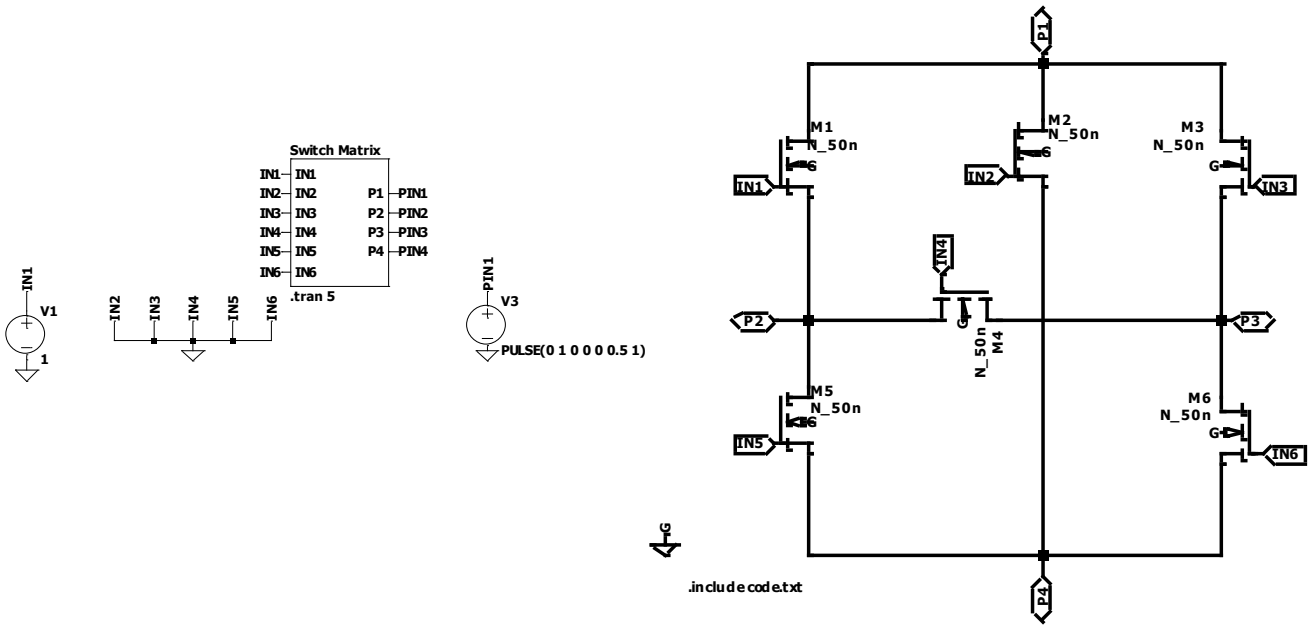


Figure11: SWITCH MATRIX implementation using NMOS & testing circuit.

Using the above circuit, the following waveforms are resulted.

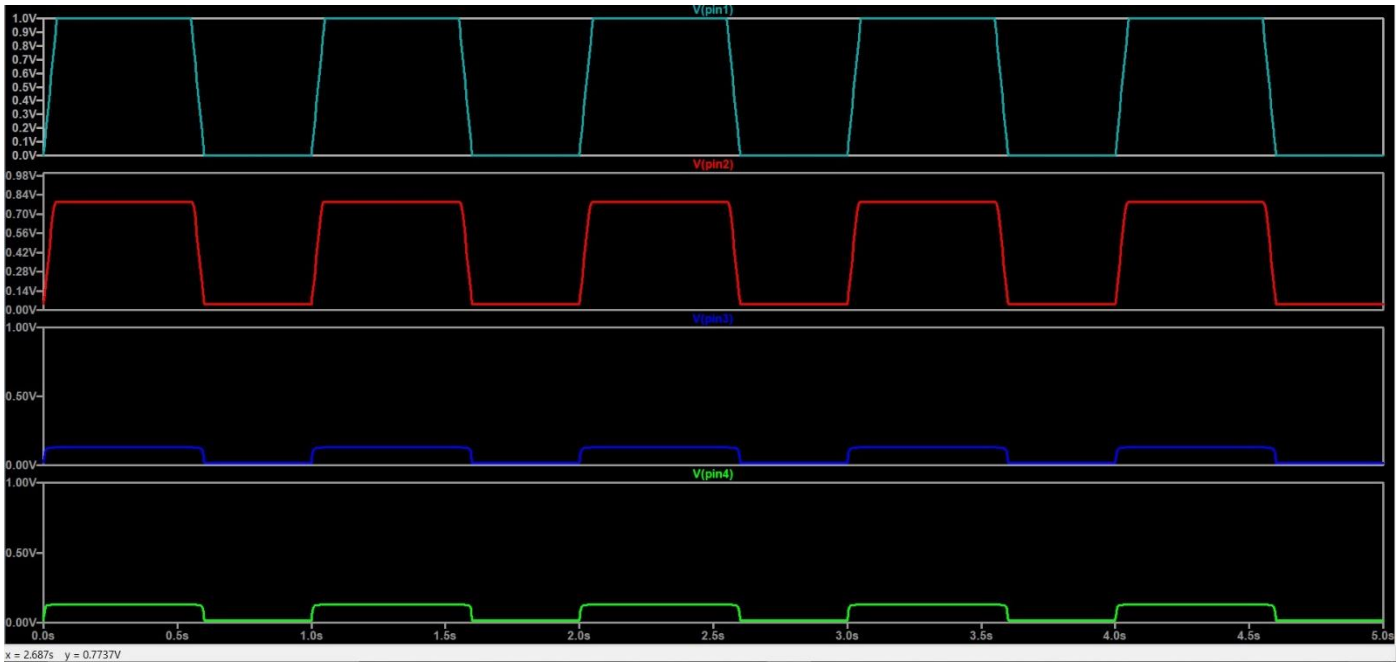


Figure12: Resultant waveforms.

2.2.3 Discussion

The switch matrix has a simple switching procedure. In this case, all 6 MOSFETs are NMOS transistors. For NMOS transistors, if the gate value is high, the transistor act as a short circuit between Drain and Source. If the gate value is low, NMOS works as an open switch, a closed route.

Since the switch matrix containing 6-NMOS, the switch matrix should have input of 6-bits.

2.2.3.1 Problems Faced & Solutions.

Problem - 3 terminal NMOS was not worked properly.

Reason - In this part of the assignment, NMOS transistors are used as bidirectional switches. To accomplish the requirement, PN junctions between Source - Body and Drain - Body much be always reverse biased to work as a bidirectional switch. But if the 3 terminal NMOS is used, always source and body will remain at the same potential, which will stop the functionality as a switch.

Solution - Using 4 terminal NMOS transistors instead of 3 terminal transistors.

Problem - According to the above Figure11 connected routes are given the output with small reduction in amplitude of the input waveform. Because there are some internal resistances and capacitances dissipate power. but the closed routes are resulting a considerably high amplitude waveform instead of expected zero voltage: approximately 200mV. Since this is a digital circuit, keeping the output value below and above the threshold accurately is sufficient but there can be some errors because the output value is near to the threshold.

Reason - There is a leakage current flowing through the closed routes when the NOMS is turned off. Due to that leakage current, a voltage is appearing on the output side.

Solution - Using load resistors at 4-connecting points of the switch.

When the NMOS is turned off, the resistance building up in the NMOS is around kΩ region. Therefore, to reduce the effect of leakage current output terminals, a load resistors of 50MΩ are connected. By using those load resistors following nearly ideal results are being generated.

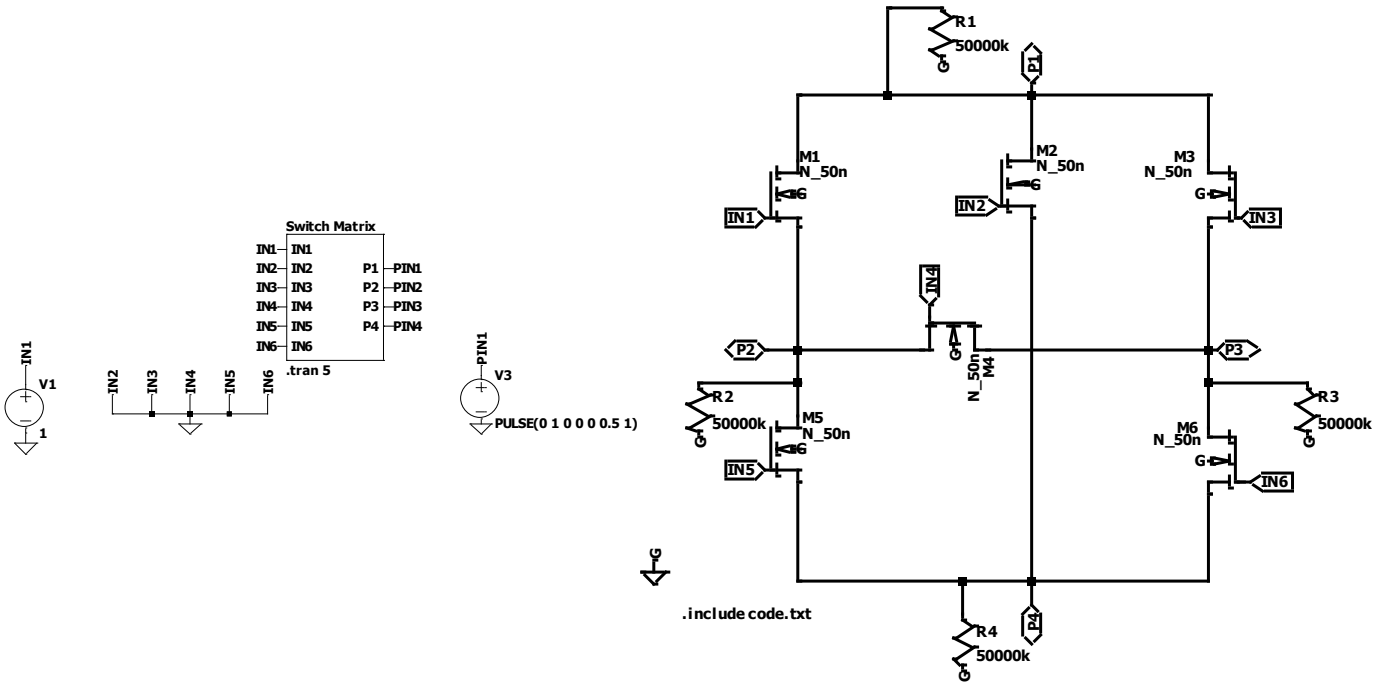


Figure13: SWITCH MATRIX implementation using NMOS with load & testing circuit.

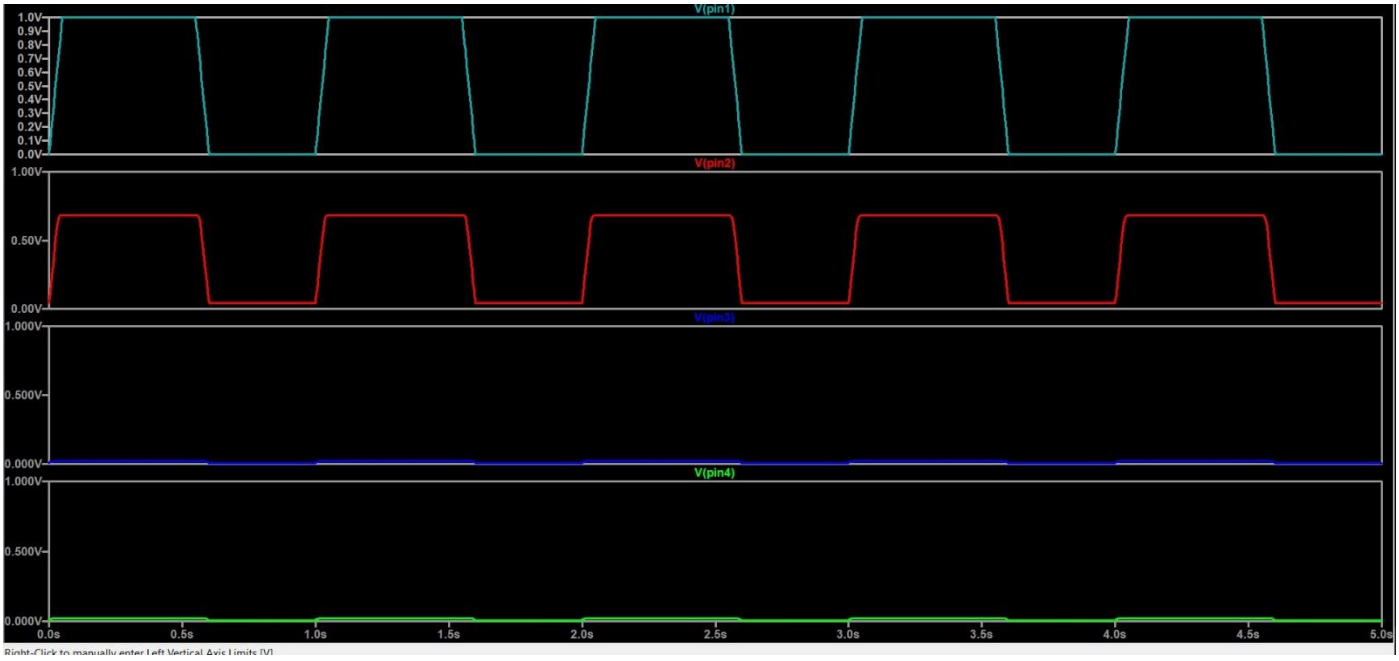


Figure14: Resultant waveforms with load.

2.3 Design a PLD

2.3.1 Introduction

PLD stands for Programmable Logic Devices. Instead of using prebuild output circuit, this can be used to program any desired output value using technologies like FUSE and ANTIFUSE while making the connections in programmable planes. In case of a Simple PLD (SPLD), there are mainly three types. In this part of the assignment we use PAL.

i. Programmable Read Only Memory (PROM) – Fixed AND plane & Programmable OR plane.
ii. Programmable Array Logic (PAL) – Programmable AND plane & fixed OR plane.
iii. Programmable Logic Array (PLA) – Programmable AND & OR planes.

2.3.2 Waveforms & Results

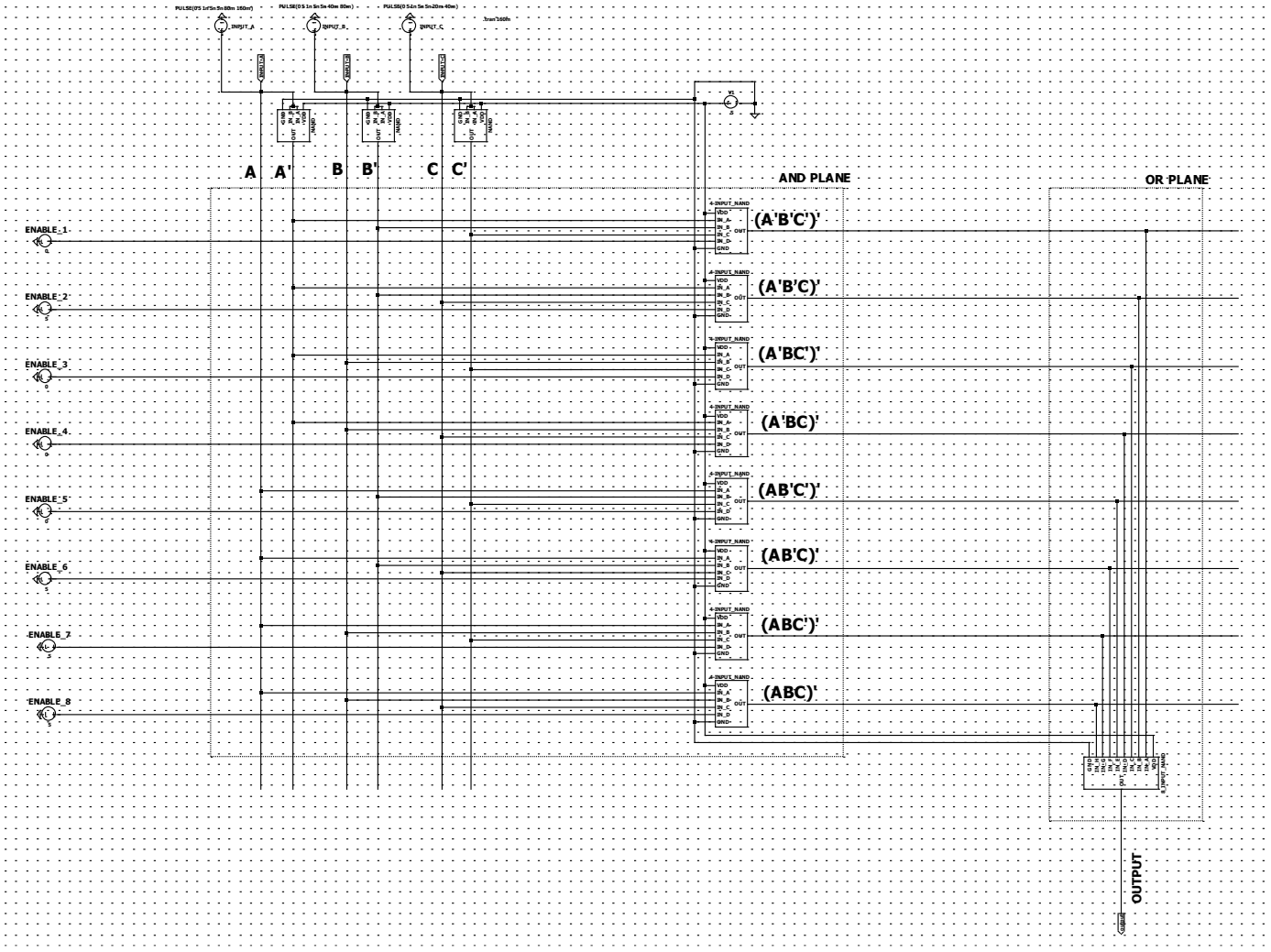


Figure15: PLD schematic for any 3-input combinational logic circuit.

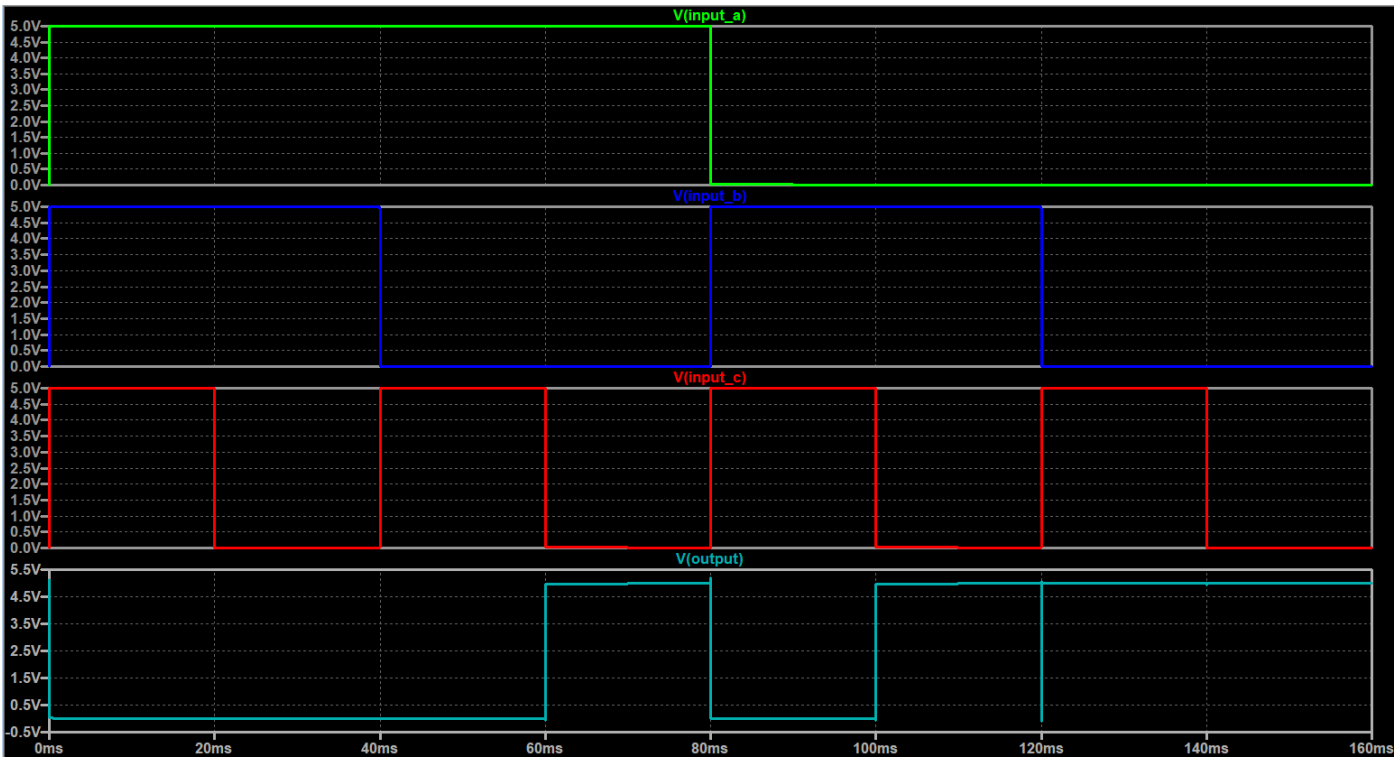


Figure16: plots for the terminals. (1) input A pulse pattern, (2) input B pulse pattern, (3) input C pulse pattern, (4) output pulse pattern for predefined combinational logic number 01.

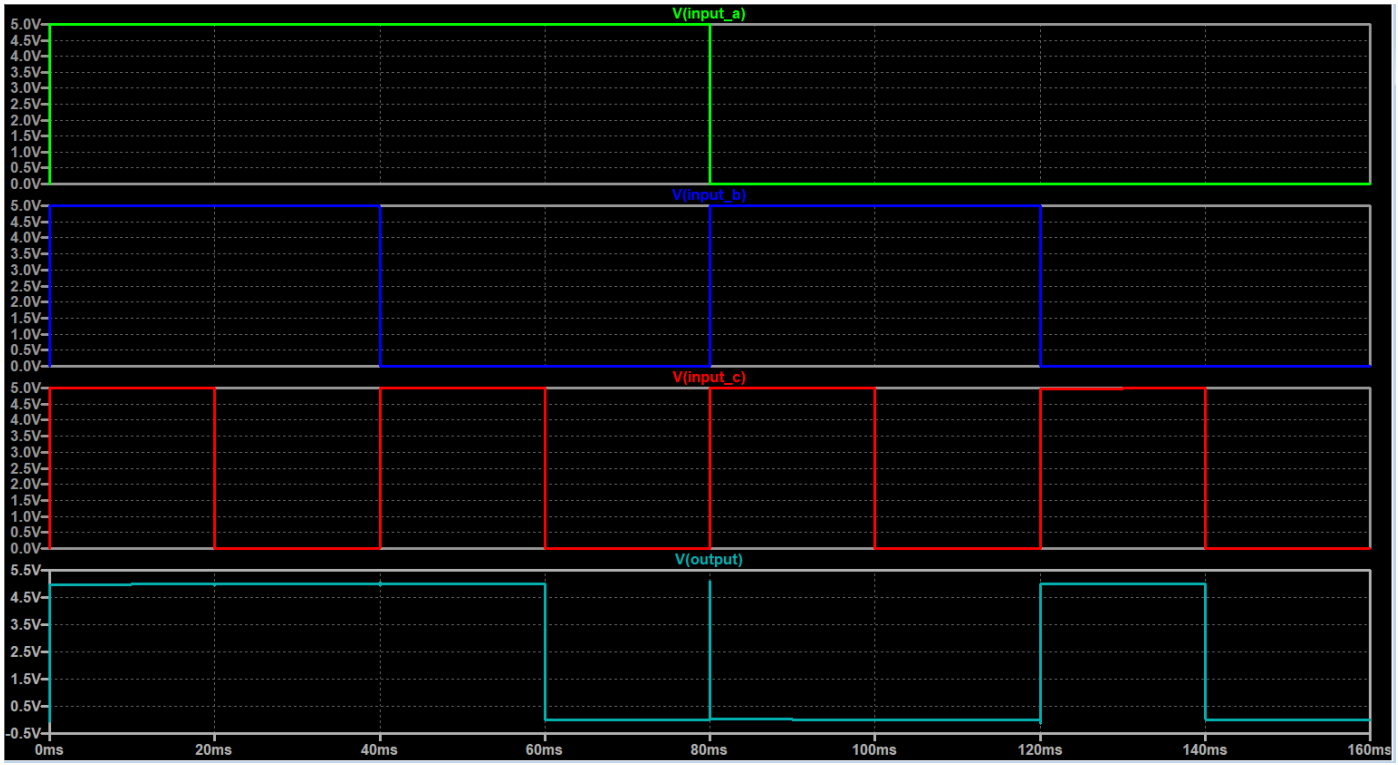


Figure17: plots for the terminals. (1) input A pulse pattern, (2) input B pulse pattern, (3) input C pulse pattern, (4) output pulse pattern for predefined combinational logic number 02.

2.3.3 Discussion

We decided to use programable AND plane & fixed OR plane. That means to use PAL, which is simpler, cheaper, and has high speed than PLA. For that purpose, we find all possible *MINTERMS* for 3-inputs and build them using AND plane. To pass only corresponding *MINTERMS* to the OR plane, we used enabling method to control the output of the corresponding 4-input AND gate with 3-input pins and the ENABLE pin. That means for ENABLE value 1, gate passes the output & for ENABLE value 0 it does not pass the *MINTERM*.

As we know any combinational logic expression can be expanded to 3-input *MINTERMS* using Boolean algebra. Then using enable pin we can program AND plane to pass only the above *MINTERMS* to the OR plane while other terms are equal to logic zero. Then using fixed OR plane they are combined and gives output as a Sum of Product.

Here we also created the logic gates as blocks using MOSFETS.

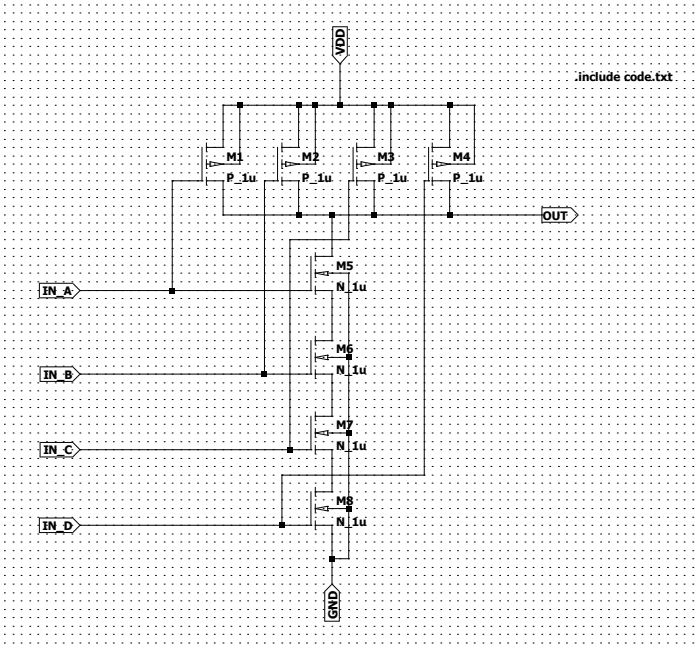


Figure18: 4 input NAND gate implementation using MOSFETS.

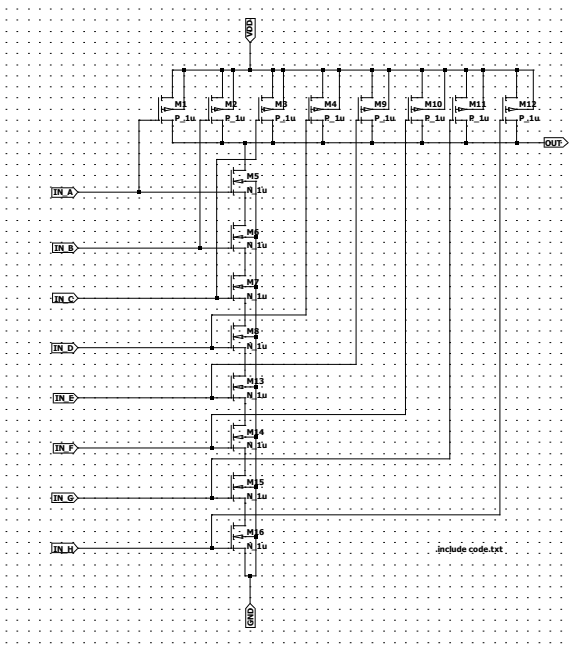


Figure19: 8 input NAND gate implementation using MOSFETS.

When we try to implement those gates using universal gates there are two solutions for both AND & OR planes.

i. NOR-NOR Solution
ii. NAND-NAND Solution

For checking our circuit, we used two pre-defined combinational logic expressions resulted the plot in Figure16 and Figure17.

$$output = \bar{A}\bar{B} + \bar{A}\bar{C} + \bar{B}\bar{C}$$

This can be expressed using all three inputs using Boolean Algebra.

$$output = \bar{A}\bar{B}(C + \bar{C}) + \bar{A}\bar{C}(B + \bar{B}) + \bar{B}\bar{C}(A + \bar{A})$$

$$output = \bar{A}\bar{B}C + \bar{A}B\bar{C} + A\bar{B}\bar{C} + \bar{A}\bar{B}\bar{C}$$

Using this expression, we can check the output. Figure 16 shows the resulting plots. Further, this is an expression we used in part 2.1.1 . Therefore, the result is the same as the previous plot. It helps to check the accuracy of our circuit. Further we can check another logic expression.

$$output = \bar{B}C + AB + AC$$

Use Boolean algebra to expand this and resulting expression is,

$$output = \bar{A}\bar{B}C + A\bar{B}C + AB\bar{C} + ABC$$

Using this expression, we can check the output. Figure17 shows the resulting plots. The validity of result can be checked using a Truth Table. This proves the accuracy of the circuit.

2.3.3.1 Problems Faced & Solutions.

Problem – Choice between NAND-NAND technique & NOR-NOR technique.

Solution – When we are using NAND-NAND plane, both inversion of AND plane output and inversion of OR plane input can be removed. As we know to build the AND plane using NAND plane can be easily done by inverting the output of NAND plane ($\overline{\overline{AB}} = AB$). By using inverted input of the NAND plane, we can easily substitute it with conventional OR plane. ($\overline{\bar{A}\bar{B}} = A + B - using\ de\ morgan's\ law$)

Problem – Choosing a technique to connect inputs.

Reason – When the inputs are connected to AND gate to get the required *MINTERMS*, what is the technique that can be used. Can we use a switch?

Solution – We can use the switch, but it consumes unnecessary power. Hence, the easiest and efficient way is to use enable pin to filter the required output from the AND plane. In a 4-input AND gate one pin is used as the enable pin. After building the circuit AND & OR planes are replaced by NAND planes as described.

Here also we faced a problem of glitch.

References

1. Microelectronic circuits by Sedra Smith-7TH ED.pdf
2. (PDF) Design and analysis of 3 stage ring oscillator based on MOS capacitance for wireless applications (researchgate.net)
3. Ring CMOS NOT-based Oscillators: Analysis and Design by F. Sandoval-Ibarra, E. S. Hernández-Bernal.pdf
4. Capacitance and Power Modeling at Logic-Level by João Baptista Martins, Ricardo Reis.pdf
5. (PDF) Influence of MOSFET parameters in its parasitic capacitance and their impact in digital circuits (researchgate.net)
6. http://cmosedu.com/
7. https://www.allaboutcircuits.com/textbook/digital/chpt-3/cmos-gate-circuitry/