

VLSI

(PART-1) (UNITS 3 &4)

END-SEM NOTES

UNIT- 3

PLD Architecture and Applications

Q..3.1What is PLD? List out its advantages. (6 Marks)

PLD is a programmable logic device that can be programmed according to the user's specific requirements. If discrete ICs are used instead of PLDs, a large amount of space is required because many discrete ICs are needed to complete the design.

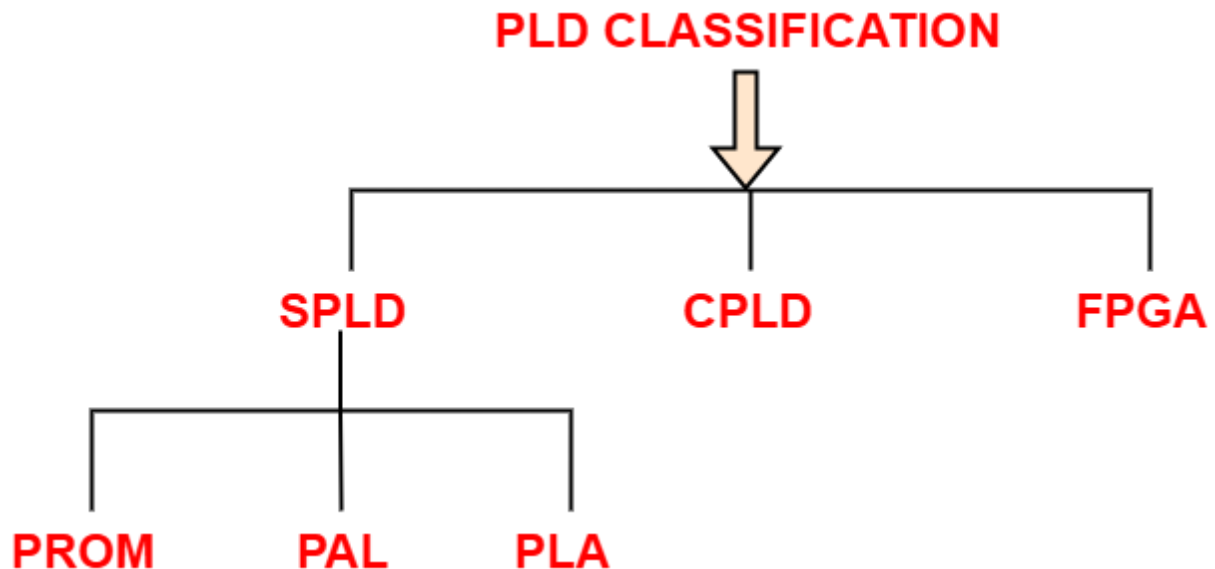
PLD has many discrete ICs combined, which are integrated into a single IC that can be programmed according to the user's requirements. There are different software tools available. By making use of those tools, we can reprogram this again and again. We can verify the design using certain software techniques.

Advantages:

- Less design time is required.
- Low power requirement.
- Reduced space.
- High design security.
- Low production cost.
- The design is flexible.
- It can be reprogrammed according to the requirements quickly.
- It has a high switching speed.

Q.3.2 Explain in brief classification of PLDs..... (MAY-23, 6 Marks)

The classification of PLDs is shown in the following figure.



There are three major types of PLDs:

- 1) SPLD (Simple Programmable Logic Device)
- 2) CPLD (Complex Programmable Logic Device)
- 3) FPGA (Field Programmable Gate Array)

1) SPLD:

These are related to the earlier technology. People were using SPLDs before the discovery of CLD and FPGA. It has a fixed number of functions and is limited by the number of logic gates. But it is a low-cost and high pin-to-pin performance device. Therefore, whenever cost is the major criterion, SPLDs are preferred for some simple applications. There are three subcategories of SPLD as follows:

i) PROM (Programmable Read-Only Memory):

It is basically an array of logical AND and OR gates, implemented on a single chip. AND array is fixed, and OR array is programmable.

ii) PAL (Programmable Array Logic):

In this case, the AND array is programmable, and the OR array is fixed.

iii) PLA (Programmable Logic Array):

In this case, both the AND array and the OR array are programmable.

2) CPLD:

It is an erasable PLD, which means you can reprogram it according to the requirement. It is programmed with the VHDL coding technique. It is basically an arrangement of multiple SPLDs on a single chip. In this case, EPROM and EEPROM Technologies are used.

3) FPGA:

FPGA is suitable for all complex applications. It has a wide density range, from a few thousand to millions of logical Gates. It is a large system interconnected chip.

Q..3.3 Write a short note on PLD and explain each device involved in its classification in brief..... (DEC-24, 8 Marks)

Please refer to Q. 3.1 and 3.2

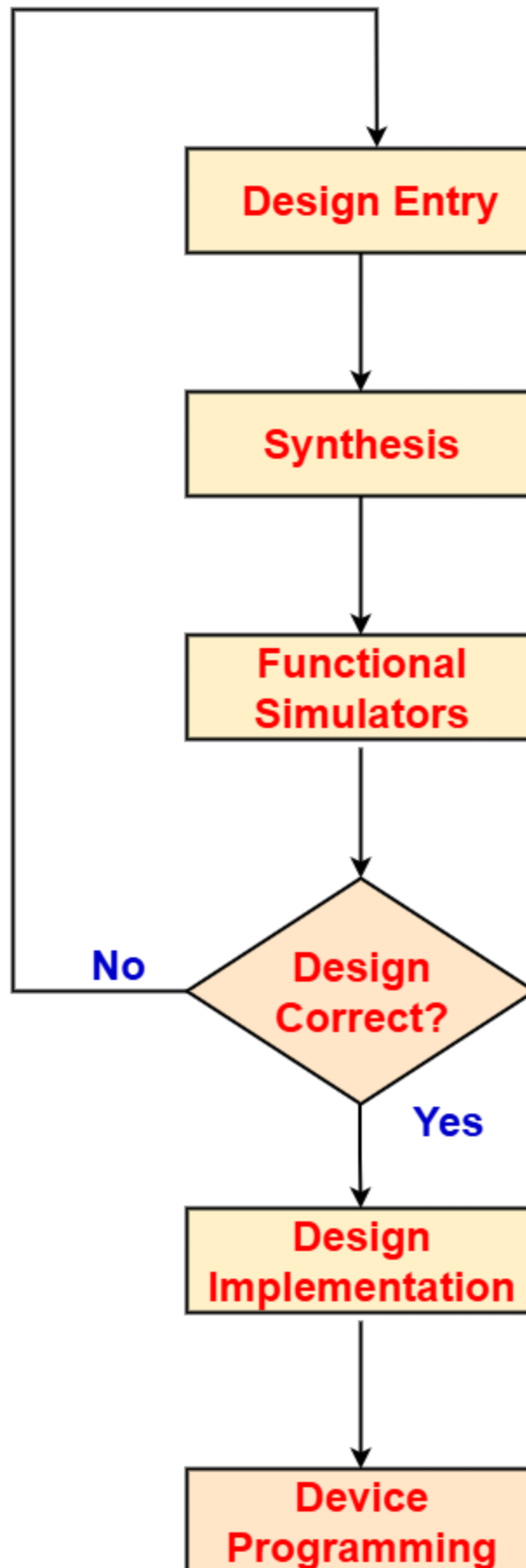
Q..3.4 Draw and explain PLD design flow..... (MAY-24, MAY-25 10 Marks)

The design flow means we have to start from a certain higher level by entering the specifications of the application that we want to design, and we have to work up to the design of the components. The design flow is shown in the following figure.

Design Entry:

We have to enter the specifications of the things that we want to design. These are the things related to the particular application to be designed. So one can enter the design specifications by using a personal computer, or there are other methods of entering the design specifications. The different methods used to enter the design specifications are as follows:

- Schematic entry
- Using HDL (Hardware Descriptive Language).
- Using FSM (Finite State Machines)
- EDIF (Electronic Data Interchange Format)



Synthesis:

This block is used for synthesis purposes, which means it converts the system descriptor from high-level to low-level circuitry. It identifies the syntax errors if you are using hardware descriptive languages. The errors are identified, and we need to correct them before finalizing the component-level design.

Functional Simulator:

The desired specifications are checked by using the functional simulator block. If there are errors, then those errors are again checked and corrected. The decision block indicates if there are design errors. If the design is not correct, then again we have to go back to the design entry. We have to check the design specifications. If required, we need to make changes in the specifications.

Design Implementation:

At this stage, a particular application is implemented using CPLD or FPGA.

Three approaches in the design flow:

The design flow consists of three approaches as follows:

1) Top-Down Approach:

As the name indicates, we have to move from the top design level to the bottom. It is applicable for a larger system. For larger systems, divide the design into different subtypes and then again consider each subtype, and then go on subdividing it till you reach the level of designing a particular component.

2) Bottom-Up Approach:

If the designer knows the library of devices, that means the final devices, then this bottom-up approach is used. We have to start from low-level component design up to the high-level design, which is the final design.

3) Mixed Approach:

It makes use of both approaches. Initially, start with a top-down approach, reaching the medium-level design, and then proceed using the bottom-up approach.

Q..3.5 Draw and explain the architecture of CPLD.Explain Macrocell in detail.... (DEC-22, 10 Marks)

OR

Write a short note on CPLD. (DEC-23, 6 Marks)

OR

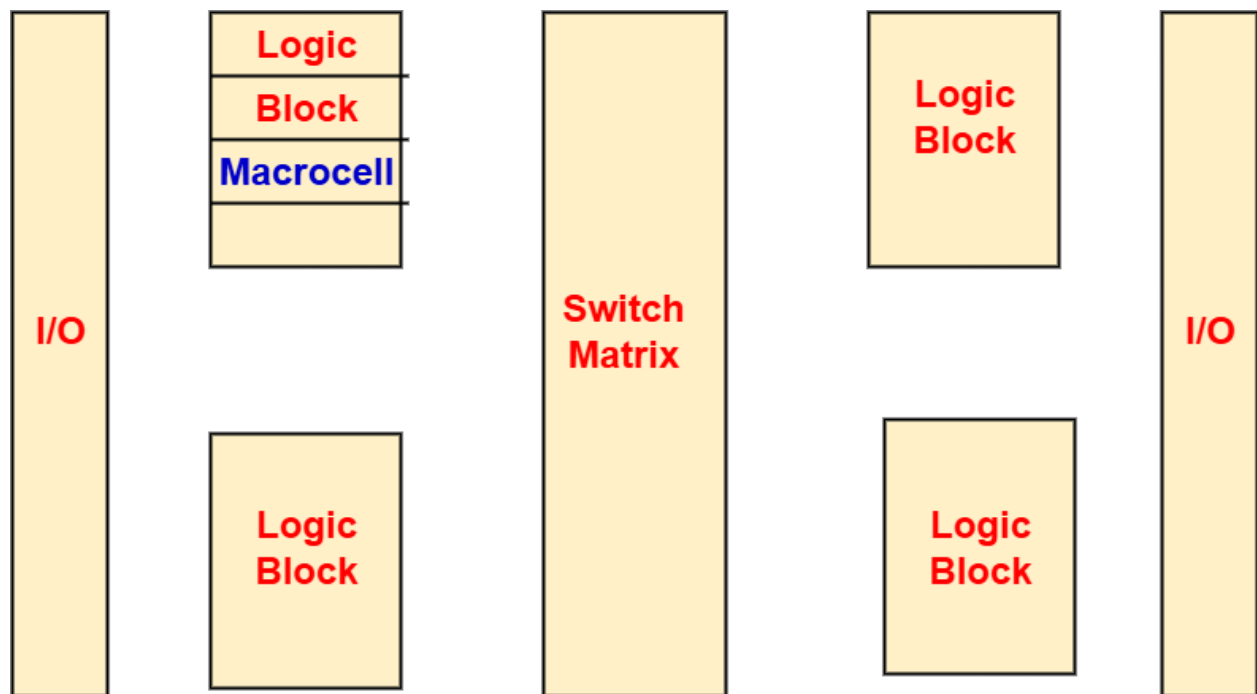
Draw the block diagram of CPLD. Explain the following blocks with respect to CPLD

i) Macrocell ii) Product term allocator. (DEC-24, 9 Marks)

OR

Draw the block diagram of CPLD and explain it in brief. (MAY-25, 9 Marks)

CPLD stands for Complex Programmable Logic Device. If the application is complicated, then depending on the requirement, it is necessary to use a number of SPLDs in a single IC package. That is called complex PLD. The simplified architecture of CPLD is shown in the following figure.



The CPLD consists of many simple PLDs. The different logical blocks are shown in the diagram. A logical block is also called a functional block. Each functional consists of many PLDs. Each PLD is called a macro cell. So there are 'n' number of macro cells in a single logical or functional block.

In the architecture, four logical blocks are shown. It depends on the type of CPLD. All these logical blocks are interconnected with each other. This interconnection is not shown in this diagram just to avoid the complexity.

A switching Matrix is used to perform the switching actions from one functional block to another. The I/O blocks at both ends are used for the connections of input and output devices. So, we can connect 'n' number of inputs and outputs. Thus, the CPLD is basically an integrated SPLD on a single chip.

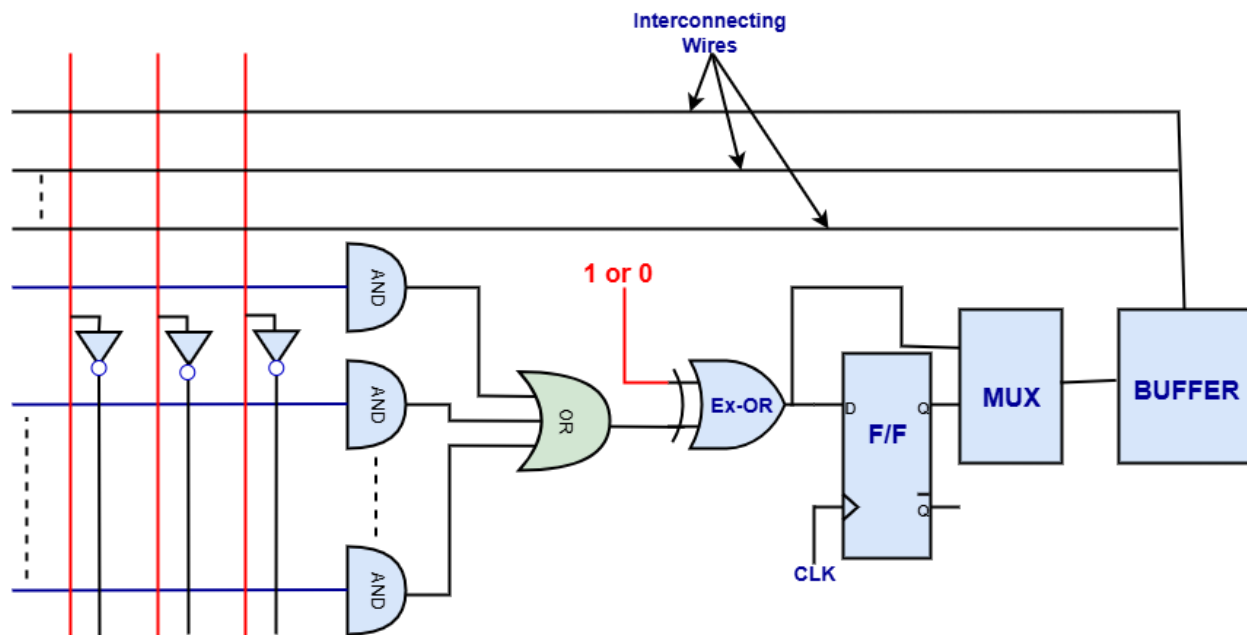
It is an erasable PLD programmed with VHDL. It is based on EPROM and EEPROM technologies. The number of macrocells is used in one functional block, and all these macrocells are internally connected. The macrocell consists of an array of AND and OR gates. So it consists of logical arrays, a product term selection matrix, a multiplexer, and flip-flops. The flip-flops are used to store the data.

Features:

- This device is reprogrammable.
- It has high performance.
- It has nonvolatile configurable memory.
- It has a built-in processor used to perform arithmetic and logical operations.
- It has in-system programmability, which means if you want to reprogram the device, you don't have to disconnect it.

Macrocell of CPLD:

The simplified architecture of Macrocell used in the CPLD is shown in the following figure.



A functional block consists of 'n' macrocells. In the CPLD, all functional blocks are internally connected. So, all the macro cells are also interconnected. The number of logical AND gates is used. It is called an array of AND gates.

It has a fixed OR gate. The outputs of all logical AND gates is applied to the fixed OR gate. The output of the fixed OR gate is connected to one of the inputs of the Ex-OR gate. Another input of

the Ex-OR gate is logic 1 or logic 0. The output of the Ex-OR gate is connected to the Flip-Flop. It is a clocked Flip-Flop. The output of the Flip-Flop is connected to the multiplexer (MUX) and a buffer.

The macro cell has an Arithmetic logic Unit (ALU). The outputs of AND gates are the product terms. To properly control the product terms, the **Product Term Allocator** is used. Each macro cell can be individually configured. There are a number of macrocells in one functional block. But each can be configured depending on the requirement. It has a global Set or Reset terminals. We can set the particular block globally, or we can reset it using this terminal.

As shown in the diagram, one input of the Ex-OR gate is logic 1 or logic 0. If it is logic 1, then it complements, which means it inverts the output of the OR gate. If it is logic 0, then the output of the OR gate remains as it is. The Flip-Flops are used to store the data. The MUX is used to multiplex the data. The tristate buffer acts as a switch.

Q.3.6 Draw and Explain the following for FPGA :

i) Logic Cell

ii) Programmable Switch Matrix

iii) I/O Block..... (DEC-22, 12 Marks)

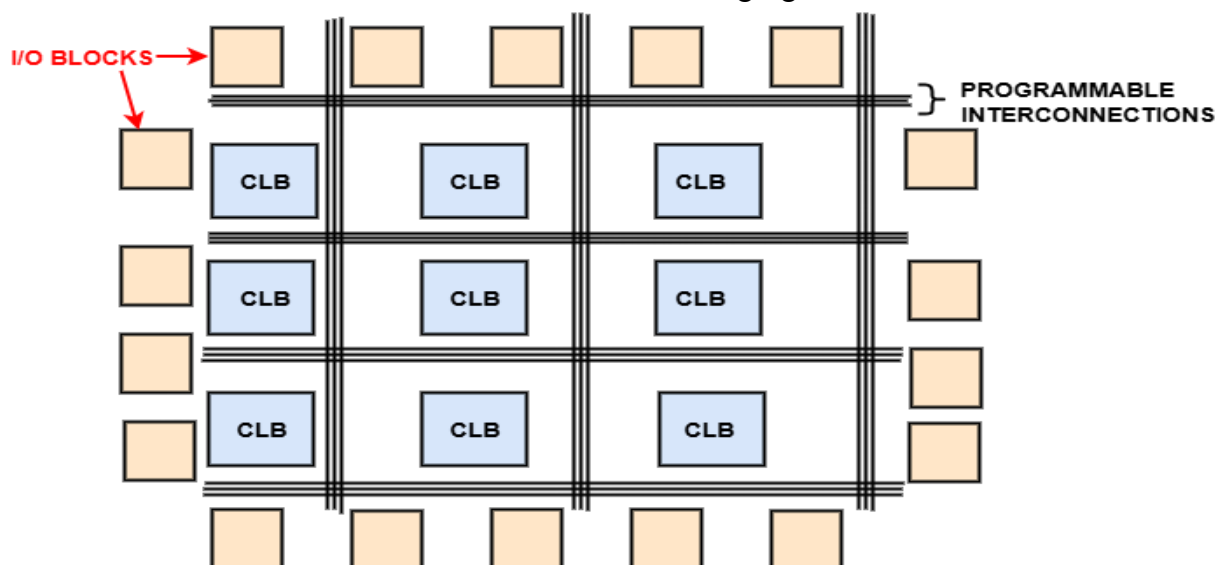
OR

Draw and explain the architecture of FPGA. Explain CLB in detail. (MAY-23, 9 Marks)

OR

Draw the block of FPGA and explain it in brief. (MAY-25, 9 Marks)

FPGA is a Field Programmable Gate Array. It is applicable for designing complicated circuits. It can be reprogrammed, and one can implement a larger logical circuitry on a single uh chip. The architectural details of the FPGA are shown in the following figure.

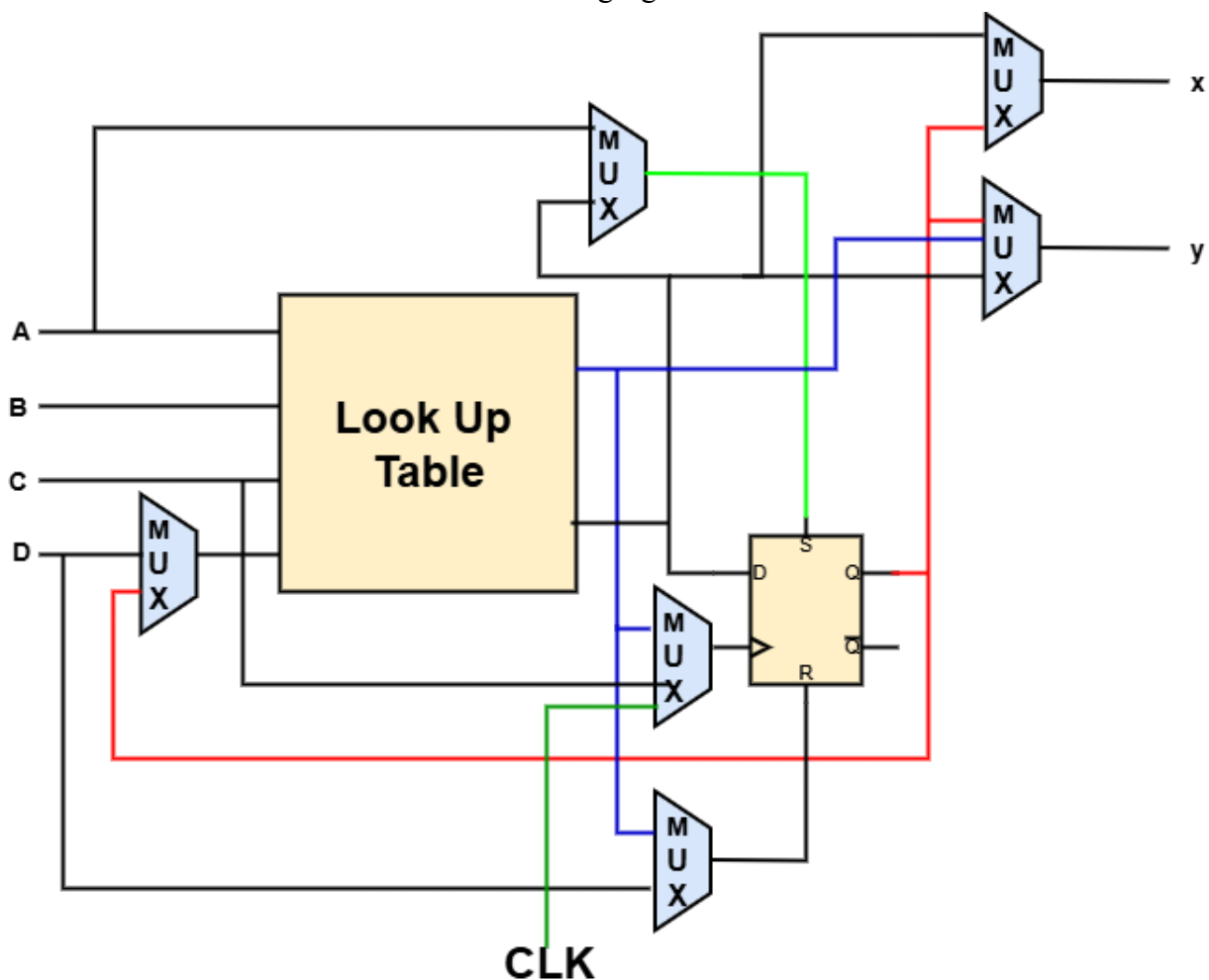


The small squares that are drawn around the structure represent input-output blocks. These input-output blocks are used to provide interconnections from logical devices to the packaging pins. CLB is a Configuration Logic Block.

The vertical and horizontal lines are called programmable interconnections. To interconnect CLBs to the other blocks, programmable interconnections are used. The horizontal lines are also called horizontal routing paths, and the vertical lines are called vertical routing paths. So, programmable interconnection represents the routing path that is used to interconnect CLBs to different things.

CLB (Configuration Logic Block):

An architecture of CLB is shown in the following figure.



It consists of lookup tables. The lookup tables are truth tables. We can implement the required sequential circuits by writing a truth table or a lookup table. It also contains clock signals. It can be a Global or input-output clock network, and it consists of a memory element. So, flip-flops or latches are used to store the data.

The SR flip-flop is used as a memory element to store the data. It consists of different logical blocks and Max, that is, Multiplex. As shown in the figure, four inputs (A,B, C, D) are used. The truth table is prepared accordingly. To implement four input functions, a CD, we need to make use of 16×1-bit lookup table.

i) Logic Cell

A Logic Cell is the fundamental building block of an FPGA. It primarily consists of a Look-Up Table (LUT) and a Flip-Flop. The LUT implements combinational logic functions. The Flip-Flop stores the output for sequential logic. These cells can be interconnected to create complex digital circuits.

ii) Programmable Switch Matrix

The Programmable Switch Matrix allows for flexible connections within the FPGA. It acts like a grid of configurable switches. These switches can route signals between Logic Cells and I/O Blocks. This programmability enables diverse circuit designs.

iii) I/O Block

The I/O Block (Input/Output Block) manages communication between the FPGA and external devices. They are configurable to support various I/O standards. I/O Blocks include buffers and, in some cases, registers. These help to drive signals reliably on and off the chip.

Q..3.7 Give typical features and specifications of FPGA..... (DEC-23, 6 Marks)

OR

Write features of FPGA in detail and write its applications. (MAY-24, 8 Marks)

Features:

- Easy configurations.
- High speed.
- It can be reprogrammed.
- The design cycle is short.
- It consists of a flexible function generator.
- It has high reliability.
- It improves the integration density.
- It can handle more complicated functions.
- It can be configured by end users

Specifications:

The specifications depend on various parameters. CLB (Configuration Logic Block) is the major block that is used to design a sequential circuit. So you need to write a look-up table or truth table, and accordingly design can be implemented. So, the specifications include:

- The system Gates, which means the number of logical Gates that are used in a particular FPGA. How many configuration logic blocks are used in that particular FPGA?
- The number of shift registers.
- How many lookup tables are available?
- Internal frequency, which depends on the clock signal, and it decides the speed of operation of that particular device.
- User input and outputs.
- Operating temperature.
- The number of pins that are used in that particular package of the FPGA.

Applications:

- They are used for Digital Signal Processing (DSP).
- FPGAs are common in Image and Video Processing.
- They are essential in High-Speed Networking and communications.
- They are used for Real-Time System Simulation.
- FPGAs serve as ASIC Prototypes before final chip fabrication.
- They are used in Medical Imaging equipment.
- FPGAs accelerate tasks in High-Performance Computing and Data Centers.
- They are used for Cryptographic algorithms and security.
- FPGAs implement Custom Hardware Accelerators.

Q..3.8 Compare CPLD and FPGA on the basis of features, specifications and applications..... (MAY-23, 6 Marks)

OR

Compare FPGA and CPLD. (DEC-23, DEC-24, MAY-25 6 Marks)

Feature	Complex Programmable Logic Device (CPLD)	Field-Programmable Gate Array (FPGA)
Basic Architecture	Macrocell-based structure.	Look-Up Table and Flip-Flop-based array (Logic Cells/CLBs).
Interconnect	Centralized, simple, and fixed-delay routing matrix.	Highly distributed, flexible, and complex mesh interconnects.
Logic Capacity	Low to Medium density	High to Ultra-High density .
Configuration Memory	Non-Volatile	Mostly Volatile
Start-up Time	Instant-On.	Slow.
Timing Predictability	Highly Predictable.	Less Predictable.
Dedicated Resources	Minimal or none. Primarily for general logic.	Includes dedicated hard blocks.
Flexibility / Complexity	Lower.	Higher.
Power Consumption	Lower Static Power.	Higher Static Power.
Applications	Boot Logic and Configuration Control, Glue Logic, Address Decoding, and Simple State Machines.	Digital Signal Processing (DSP), High-speed Networking, Video/Image Processing, ASIC Prototyping.

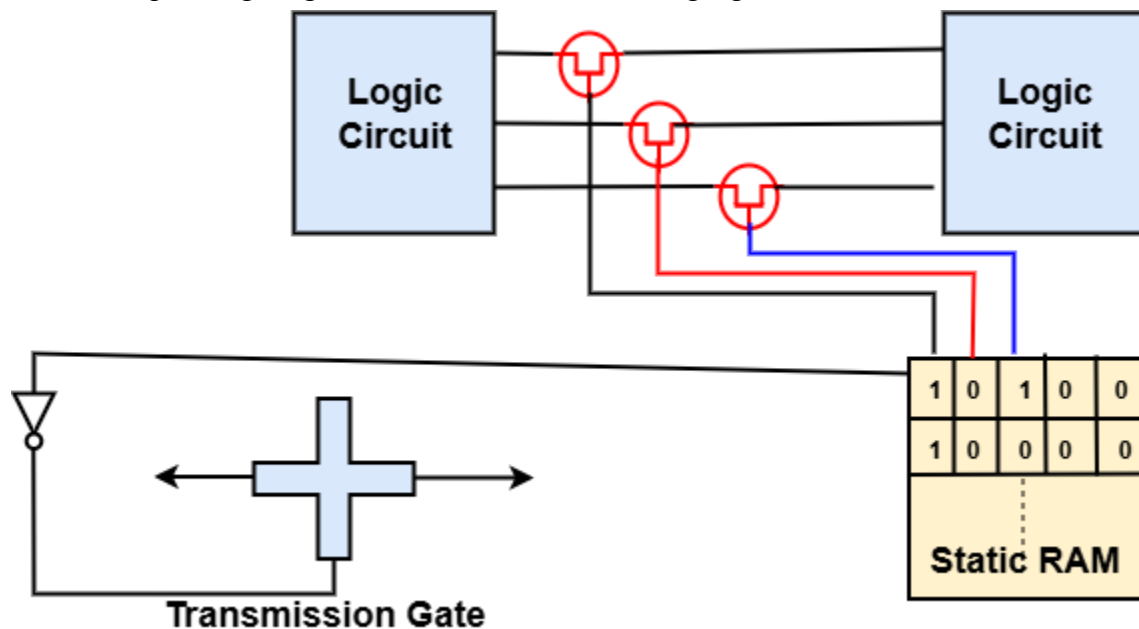
Q..3.9 What are the technologies supported by FPGA? Explain any two in detail.(DEC-22, 8 Marks)
OR
 Explain with diagram SRAM and Anti-fuse programming techniques used in FPGA.
(DEC-23,8 Marks)

There are three major technologies supported by an FPGA.

- 1) Static RAM: It uses transistors or transmission Gates and multiplexers to obtain the interconnected routes.
- 2) Antifuse Technology: In this case, interconnection routes are permanent by making use of fuses.
- 3) EPROM/EEPROM/Flash: In this case, especially in the case of PROM and EEPROM, programmable transistors are used.

1)Static RAM:

The corresponding diagram is shown in the following figure.



As shown in the diagram, two logical circuits are used. In between these two logic circuits, n-MOSFETs are connected. It acts as a switch. Depending on the contents of the static RAM, a path is created. If the corresponding bit in the static RAM is 1 then path is created and if it is 0, the path is not created.

The upper n-MOSFET is connected to the logic 1 bit of the static RAM, so a path is created. The second n-MOSFET is connected to the logic 0 bit of the static RAM, so a path is not created. The second n-MOSFET is connected to the logic 1 bit of the static RAM, so a path is created. Thus, the n-MOSFETs act as a one-way switch.

If you want a two-way switch, then a transmission gate is used. The transmission gate consists of an n-MOS transistor and a p-MOS transistor, which are connected in parallel with each other. The transmission gate is used as a bidirectional or two-way switch.

Advantage:

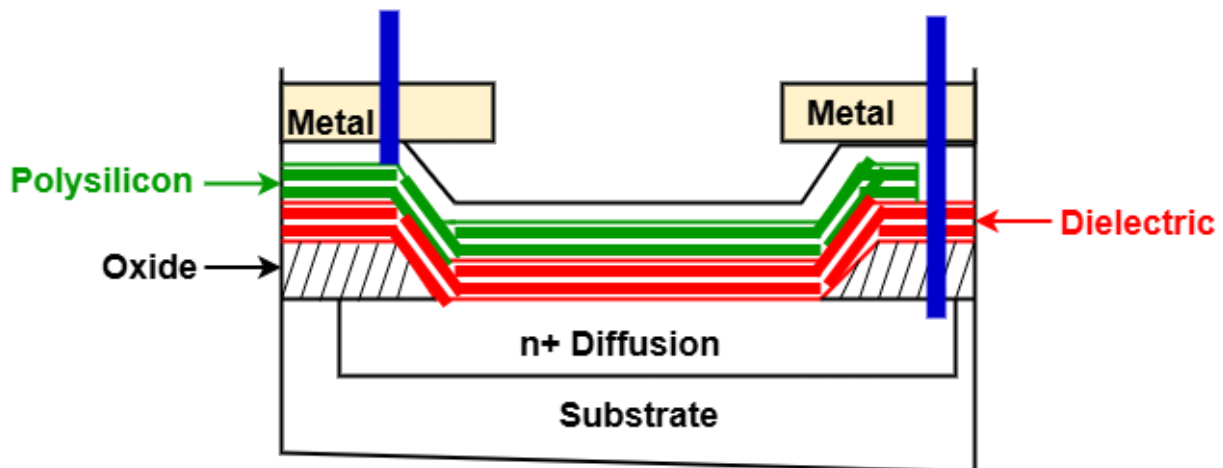
- The major advantage of this technology is that you can reprogram it ideally for infinite times. That means, depending on the requirements of the user, the changes can be made in the programming.

Disadvantages:

- Whenever the power is switched off, it loses the stored program. To retain the program, some non-volatile memory is required.
- Each memory cell requires at least five transistors, so a larger area is required.

2) Antifuse Technology:

This is used whenever only permanent connections are required. But, if it is required to make changes after implementation, it is not possible with this technology. The structure is shown in the following figure.



As shown in the figure, it consists of two metal contacts. Three major layers are shown in this diagram. The bottom layer is a substrate, above it there is a layer of n+ diffusion and a polysilicon layer.

Initially, the impedance of this device is very high. This impedance can be made low by applying a very high voltage. By applying this voltage, a high impedance is reduced, and the connection can be made. Whenever high voltage is applied, a breakdown of the dielectric layer takes place, and a contact is established between the n+ diffusion layer and the polysilicon layer.

The contacts are created for poly silicon as well as n+ diffusion layers. But this makes the permanent connection, and whenever the breaking of the dielectric takes place, then it is said that the fuse is blown. This creates a connection.

Advantages:

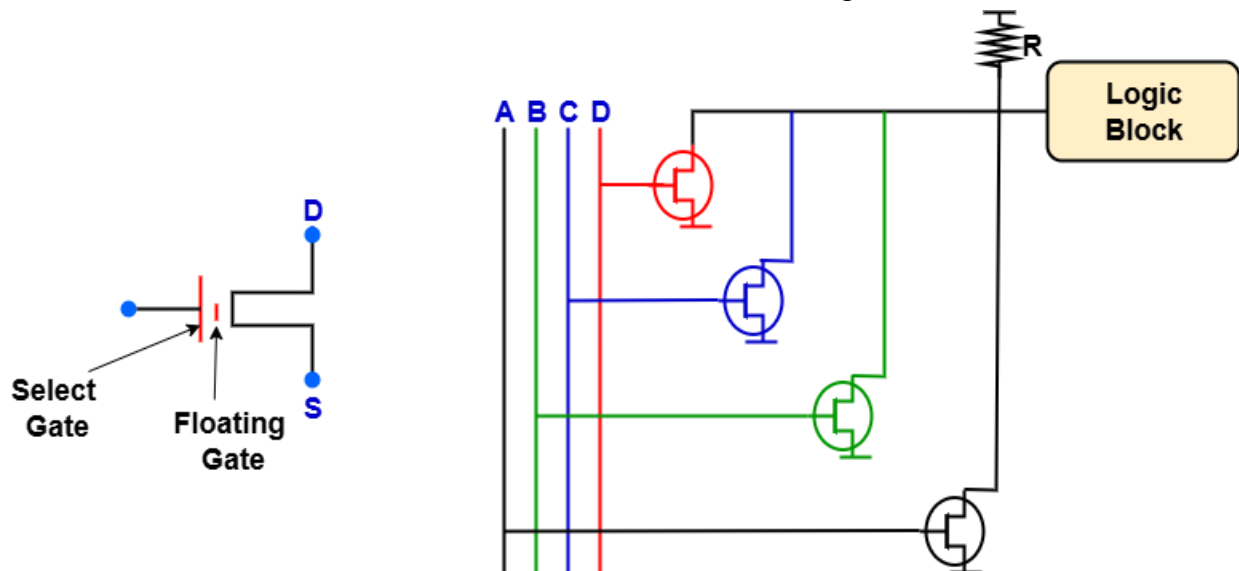
- It requires less area.
- Low ON Resistance
- Low parasitic capacitors.

Disadvantage:

It cannot be reprogrammed. Therefore, this type of technology cannot be used whenever reprogramming is required repeatedly.

3) EPROM/EEPROM/Flash:

The basic symbol of an n-MOS transistor is shown in the following figure. Another diagram shows that the four such n-MOS transistors are connected to the logic block.



The n-MOS transistor has two terminals: one is the Select gate terminal, and the second is the floating gate terminal. The floating gate terminal is not electrically connected to the select gate.

If the heavy current flows through a Drain and Source terminal, then electrons are trapped under the floating gate. This is possible due to the injection of electrons. Once the electrons are trapped, they remain for several years. That means the data can be stored permanently for several years.

In the figure, four n-MOS transistors are shown. If you want to store the data, then you will have to switch ON the particular transistor. That means you will have to pass a heavy current through the drain and source, and electrons will get trapped at the particular floating gate. This is about EPROM.

In the case of EEPROM, instead of using n-MOS, floating gate tunneling metal oxide semiconductor devices are used. The work remains the same. The flash technology makes use of both EPROM and EEPROM technologies. Once the electrons are stored, that means data remains stored for many years. If you want to erase the data, then such devices are exposed to ultraviolet light.

Q..3.10 Explain Clock Management Techniques in FPGA. (MAY-23,8 Marks)

There should be proper clock management, which means the distribution of the clock signal should be proper, or equal to all parts of the circuit. This is achieved by utilizing clock management techniques. These techniques are also applicable to high-speed designs and those with high densities. It provides reliable clock distribution throughout the entire system.

There is one term, which is H-tree clock distribution. This technique is used to make the delay the same for all the elements, all the parts of the circuit. Some Modern FPGA generates new clock signal if required, and this clock signal is distributed in the system.

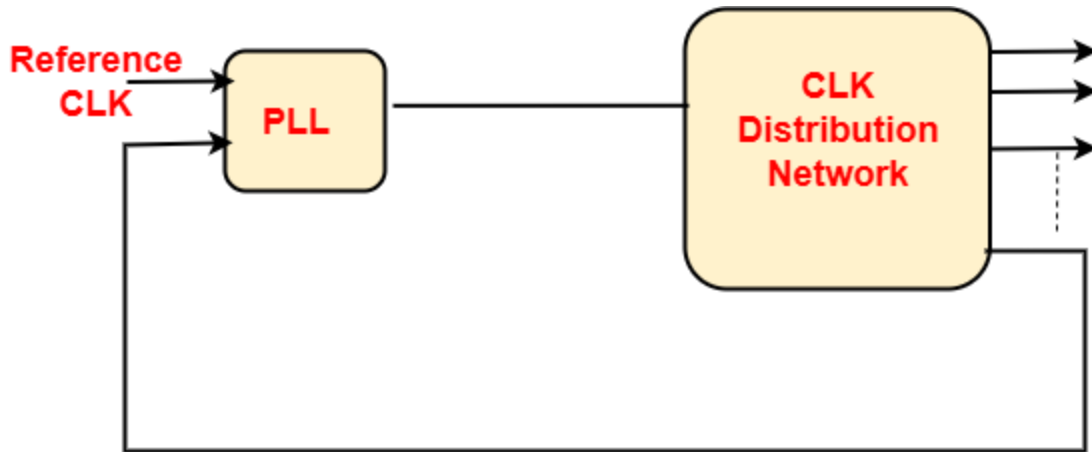
Many times, the clock buffers are used. These are dynamic devices, used to dynamically enable or disable the clock signal applied to the particular part of the circuit. There are clock enable options. Using these options, we can enable or disable the clock signals. So it is used to reduce the power consumption.

Two more devices are used for the clock management techniques.

1) PLL (Phase Locked Loop):

The following block diagram shows the use of a PLL for clock management. PLL generates an internal clock. It generates a high-frequency clock from the external clock. It provides frequency multiplication. So, we can generate the clock signals at faster or slower speeds depending on the requirement.

As shown in the block diagram, a reference clock is applied as one of the inputs to the PLL. The output of the PLL is applied to the clock distribution network. The clock distribution network is used to distribute the clock signals to various parts of the circuit. One output of the clock distribution network is applied back to the input side. This signal is used to lock the PLL device.



2) DLL (Delay Locked Loop):

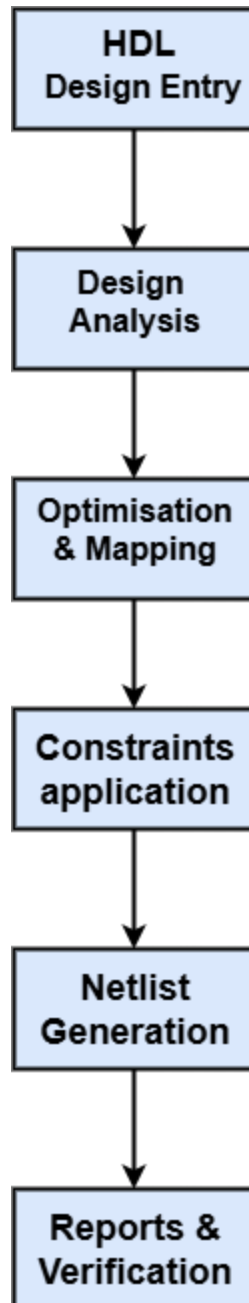
Similar to PLL, we have a DLL that is a Delay Locked Loop. It aligns the output clock pulse with the reference clock pulse. It removes the delay introduced by the internal FPGA clock distribution network. This creates a zero propagation delay clock signal. All parts of the FPGA receive the clock at virtually the same time as the external input.

A DLL can create an output clock with a precise, programmable phase shift relative to the input clock. It helps align the data signal with the clock edge for reliable sampling. Some DLLs can fix a non-50% duty cycle on the input clock signal. This generates a clean, 50% duty cycle clock for internal logic.

Q..3.11 Explain various stages of Synthesis in FPGA with suitable diagram.(MAY-23,6 Marks)

Synthesis is the process of converting a high-level hardware description (written in VHDL or Verilog) into a low-level hardware structure. So that it can be implemented on an FPGA. It translates behavioral code into digital circuit elements like logic gates, multiplexers, and flip-flops.

The synthesis process is shown in the following flow chart.



1. HDL Design Entry:

The designer writes the logic using a Hardware Description Language (HDL) such as VHDL or Verilog. At this stage, the design is written in behavioral or structural form. The Functional simulation can be done to verify logic before synthesis.

2. Design Analysis and Elaboration

The synthesis tool reads and checks the HDL code. It verifies syntax, data types, and hierarchical structure. The design hierarchy (modules, ports, connections) is expanded into a single detailed representation. This ensures the design is ready for logic synthesis.

3. Optimization and Technology Mapping

This stage converts the HDL description into logic gates and flip-flops compatible with the target FPGA family.

4. Constraint Application

Design constraints are applied here to guide synthesis. These may include: Timing constraints (clock period, setup/hold times), Area or power constraints, and I/O pin assignments. The constraints ensure that the final design meets the required performance and resource limits.

5. Netlist Generation

The net list means it provides a complete circuit with logical elements. The synthesis tool now generates a technology-mapped netlist. This netlist describes: Logic gates and flip-flops, Their interconnections, and FPGA-specific primitives

6. Reports and Analysis

Finally, synthesis tools generate detailed reports showing: Resource utilization (LUTs, FFs, DSPs), estimated delay and timing, Power, and area usage. Designers use these reports to check whether the design meets all constraints.

Q..3.12 List and Explain in brief various Simulation and Synthesis Tools.(DEC-22,DEC-23 6 Marks)

Simulation Tools

The simulation tools are useful for checking the functions of the logical circuits. The various simulation tools are as follows:

1) Aldec active HDL:

It is a full HDL and graphic design tool. It is basically an RTL and gate-level simulator, and it verifies VHDL, Verilog, and EDIF-based designs.

2) Mentor Graphic Questa Advanced Simulator:

It provides a Global compiler and simulation optimization algorithm. It is used for verifying Verilog and VHDL. It supports almost all design languages. It partitions the design to run in parallel, but it makes use of a single database for debugging purposes.

3) Model Sim HDL:

It provides an optimized direct compiler for faster compiler time. It provides mixed language simulation and platform-independent compilation. Its features include performance analysis, code conversion, and waveform comparison.

4) Candence Incisive:

It supports gate-level and RTL design, and it evaluates multiple faults and supports VHDL, Verilog, and SystemVerilog.

Synthesis Tools

1) FPGA Express:

It is a PC based tool used for logic synthesis and optimization. It creates a netlist for VHDL and Verilog HDL code.

2) Lenardo Spectrum:

It is a high-level design tool for CPDL and FPGA. It has certain tools that can capture VHDL and Verilog entries. It is useful for debugging operations and for timing analysis. It places and routes, and it is also used for schematic viewing.

3) Altera Max Plus-2:

It is a multiple-array matrix programmable logic user system. It provides a complete design environment to adopt the required or specific design changes. We can easily enter the design as well as it is a fast processing tool. It offers a full spectrum of logic design capabilities and we can read or write VHDL and Verilog netlists.

UNIT- 4

Digital CMOS Circuits

Q..4.1 Define Scaling and explain any one type of scaling and its effect on at least 4 parameters.
..... (DEC-22, 9 Marks)

OR

What is technology scaling? Explain any one type of scaling and its effect on at least 4 parameters.
..... (MAY-25, 8 Marks)

Suppose we are using MOSFET transistors and we want to accommodate a large number of transistors in a single IC package; in that case, scaling technology is used. Thus, scaling means reducing the dimensions by applying certain criteria so that the maximum number of devices can be placed.

But when performing the scaling, care should be taken that the characteristics after scaling should not be affected. So many times, it is required to change the fabrication technology or the bias voltage.

Types of Scaling:

There are two types of technology scaling:

1) Constant Field Scaling:

In this case, the dimensions of the transistor, device voltage, and doping concentration densities are reduced by a scaling factor $S = 1/\alpha$. This scaling factor is greater than one. This keeps the electric field constant across the device. As a result, power density remains nearly constant while performance improves.

2) Constant Voltage Scaling:

In this case, the supply voltage is kept constant; it is not scaled down, but the process is scaled. Here, only the device dimensions are scaled by $1/S$, but the supply voltage remains unchanged. This increases the electric field, which improves speed but also raises power density and leakage.

The effect of scaling on the major parameters is shown in the following table.

Parameter	Scaling Factor	Effect
1) Gate Area (A)	$(1/\alpha) \times (1/\alpha) = 1/\alpha^2$	Decreases (Transistor density increases by α^2)
2) Gate Capacitance (Cg)	$1/\alpha$	Decreases
3) Gate Delay (τ_d)	$1/\alpha$	Decreases (Speed improves by α)
4) Power Dissipation per Gate (P)	$1/\alpha^2$	Decreases

1. Gate Area (A): The area occupied by a single transistor, $A = L \times W$, scales by $1/\alpha^2$. This quadratic reduction is the key factor that allows the **transistor density** (number of transistors per unit area) to increase by a factor of α^2 .

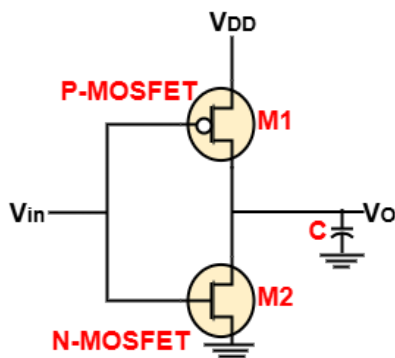
2. Gate Capacitance (C_g): Gate capacitance scales by $1/\alpha$. The reduction in capacitance is beneficial as it requires less charge to switch the transistor.
3. Gate Delay (τ_d): Gate delay scales by $1/\alpha$. This means the speed, or maximum operating frequency, increases by a factor of α , directly contributing to faster microprocessors.
4. Power Dissipation per Gate (P): Dynamic power dissipation scales by $1/\alpha^2$. This significant reduction in power per gate is crucial for maintaining manageable chip power consumption despite the massive increase in the number of transistors.

Q.4.2 Draw and explain CMOS Inverter transfer characteristics with all regions in detail..... (DEC-23,MAY-25 8 Marks)

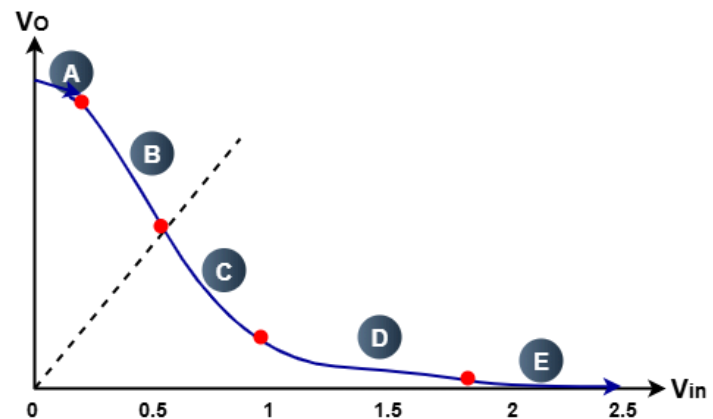
OR

Explain the working of CMOS inverter with the help of Voltage Transfer curve. (MAY-23 8 Marks)

The circuit diagram and transfer characteristics of the CMOS inverter are shown in the following figure.



Circuit Diagram



Transfer Characteristics

It is an Inverter, which means that if you apply a high input, the output will be low. On the contrary, if we apply a low input, the output will be high. This is the inverter action that is generated using MOSFETs.

As shown in the figure, two MOSFETs are used. M1 is a p-MOSFET and M2 is an n-MOSFET. M1 and M2 are connected one after the other. V_{in} is the input voltage and V_o is the output voltage.

If a low input voltage is applied, then the p-MOSFET will be ON. Once the p-MOSFET is switched ON, the supply voltage gets connected to the output. It will charge the capacitor. In this

case, M2, which is an n-MOSFET, is switched OFF. Since it is OFF, there is a direct connection between the input voltage and the output is available. So it will charge the capacitor C. Since the output is available, it will be logic high. This is an inversion action.

When the high input voltage is applied, then M1 is OFF and M2 is ON. M2 is an n-MOSFET. It is connected to the ground. Thus, the capacitor gets a path to get discharged. So, the output will become zero. Again, this is the inverter action.

The transfer characteristics is the graph of output voltage versus input voltage. The graph is divided into different regions.

1) Region A:

In this region, the condition is $0 < V_{in} \leq V_{th}$. Here V_{th} is the threshold voltage. In that case, M2 is switched OFF and M1 is in the linear mode. So, the output is equal to V_{DD} , because V_{DD} is directly available at the output.

2) Region B:

In this case, M1 is in linear mode and M2 goes into saturation. Earlier, M2 was in the cut-off condition. When M2 goes into saturation, the output voltage starts decreasing.

3) Region C:

In this case, both M1 and M2 are in saturation mode.

4) Region D:

In this region, M1 is in saturation and M2 is in linear mode.

5) Region E:

In this region, M1 is OFF and M2 is in linear mode. That means the p-MOSFET is switched OFF and the n-MOSFET is switched ON. So, the discharging of the capacitor takes place.

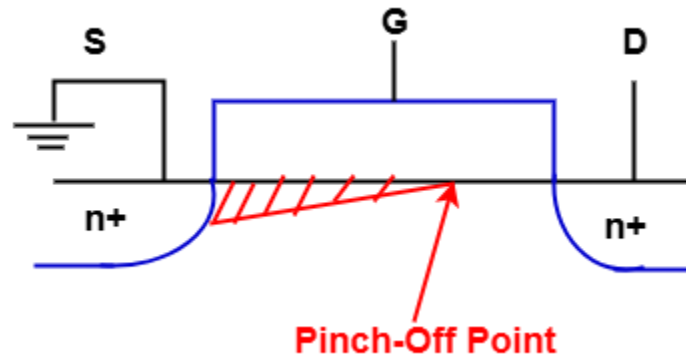
Q..4.3 Write a short note on: Channel length modulation..... (DEC-23, MAY-25 4Marks)

As the name indicates, the length of a channel is changed due to certain conditions. This is the channel length modulation. Actually, in the saturation mode, the transistor acts as a perfect current source. That means in that case, I_{DS} (Drain to Source current) remains constant.

Since this current is constant, the MOSFET transistor acts as a constant current source, and this value is not dependent on V_{DS} (Drain to Source voltage). That means even if we vary, V_{DS} then also I_{DS} remain constant. So, the transistor acts as a constant current source.

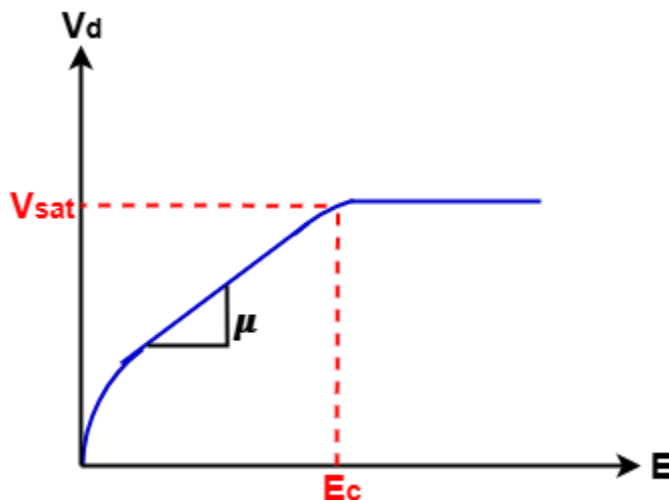
But practically, it's not true. In Practical cases, as V_{DS} (Drain to Source voltage) increases, the depletion region at the drain junction increases and the effective channel length reduces. Actually

there is one condition which is called a pinch-off condition. In that case, as the V_{DS} increase, the pinch-off point shifts towards the source terminal. It will reduce the effective length of a channel. This particular thing, in which as we go on increasing V_{DS} , the length of a channel decreases, is called channel length modulation. This concept is shown in the following figure.



Q..4.4 Explain the following term : Velocity Saturation..... (DEC-22, MAY-25 4Marks)

The velocity (V_d) of a charged carrier is directly proportional to the applied electric field (E), and the proportionality constant is μ . Here μ is the mobility of a charged carrier. The graph of the velocity of a charged carrier (V_d) versus the electric field (E) is shown in the following figure.



So, the velocity of a charged carrier is linearly proportional to the applied electric field, but there is a certain limit. As shown in the diagram, up to a certain point, the value of the velocity of charge carriers increases with an increase in the electric field.

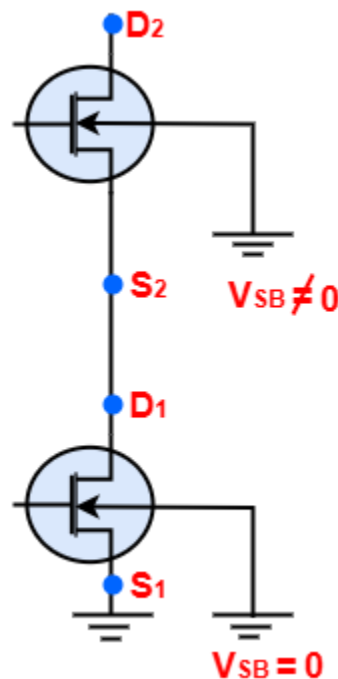
When this value of E reaches a critical value (E_c), then in the case of a MOSFET, the velocity of a charged carrier does not increase beyond this point. Initially, the velocity of a charge carrier changes linearly up to a certain limit. Once the value of the electric field is reached E_c , the velocity remains constant. And this value is denoted by V_{sat} . So, when E reaches E_c , the velocity of a charged carrier tends to saturate, and because of this, the value of mobility (μ) decreases.

Q..4.5 Explain the following term: Body Effect..... (MAY-25, 4Marks)

OR

Write a short note on: Body effect. (DEC-23, MAY-24 4Marks)

If there are similar types of MOSFETs, it means that if there are only n-type or only p-type MOSFETs, then a common substrate is used. That means these similar types of MOSFETs are fabricated by using a common substrate. Therefore, the substrate voltage of each device remains constant. But many times it is required to connect the MOSFET devices in series, as shown in the following figure.



Here, V_{sb} stands for source to substrate or source to bulk voltage. For the lower MOSFET, the S1 terminal is directly connected to ground. So, V_{sb} is ideally zero. But as you move in an upward direction, this V_{sb} is not equal to zero. That means the value of source to bulk or source to substrate voltage is not kept at zero. It is changing.

If there is a series connection one above the other, then as we move further, the channel substrate junction potential increases. This is because V_{SB} is not zero. It increases the gate channel voltage drop. If there are many MOSFET devices, then the value of will keep on changing. So, the net effect is, there is a change in the threshold voltage. That means the threshold voltage increases due to the nonzero value of V_{SB} . This change of the threshold voltage with respect to V_{SB} is called the body effect.

If we are using an n-type MOSFET, then the substrate is connected to the most negative terminal of that particular circuit. When connecting p-MOSFETs in series, the body terminals of the transistors are connected to the highest potential in the circuit.

Q..4.6 Explain the following term: Hot Electron Effect..... (DEC-22, MAY-25 4 Marks)

OR

Write a short note on: Hot Electron Effect (MAY-24 6 Marks)

It is basically an effect in which the electrons gain high kinetic energy because of which they penetrate the oxide layer of a structure, and it may damage the entire IC. This effect is the hot electron effect.

Under normal circumstances, that means in case of normal thermal equilibrium conditions, both electrons and holes absorb as well as emit photons continuously. That means if there is a thermal equilibrium condition, then both electrons and holes are absorbing energy levels as well as emitting energy levels. So, the average gain in kinetic energy remains zero.

But if there is a presence of electric field intensity (E), then kinetic energy increases. If the value of the applied electric field reaches typically up to 100 kV/cm, then the carriers (electrons or holes) will not remain in the equilibrium conditions. These carriers cause impact ionization.

This is the process in which a larger number of electron-hole pairs are generated. It is also called Avalanche Multiplication. So, existing electron and holes causes impact ionization with the lattice structure. It produces more and more electrons and holes. These carriers are injected into the oxide layer and may damage the structure.

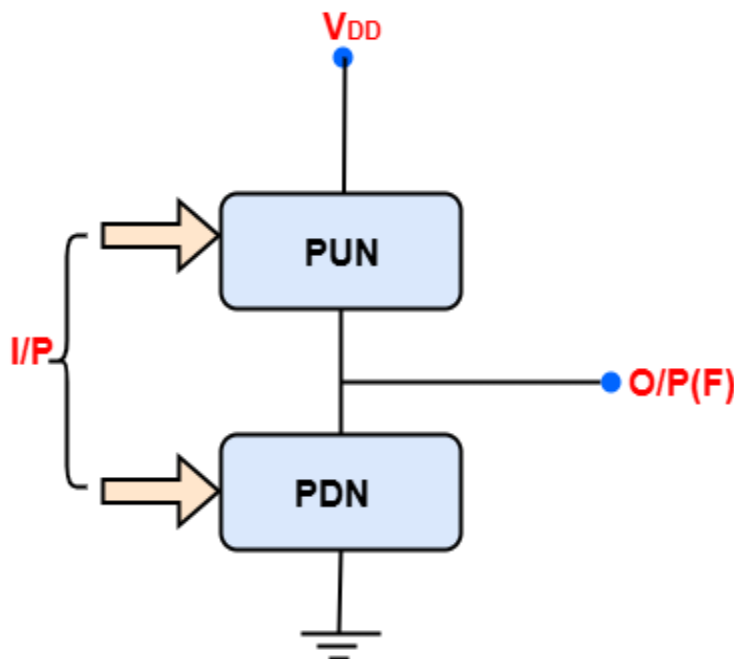
These carriers are called hot carriers, and this is the hot electron effect. To obtain protection from such a hot electron effect, one obvious solution is to reduce the electric field at the drain junction. To reduce the electric field at the drain junction, lower the doping concentration at the drain junction or use an additional MOSFET in series with the actual MOSFET, which is going to get affected. So that the total voltage gets distributed among the two MOSFETs.

Q..4.7 Explain any two non-ideal transistor I-V effects. (DEC-24 6 Marks)

Refer to Q. 4.4 & 4.6

Q..4.8 Explain the designing procedure of a combinational logic circuit using CMOS. (4 Marks)

Complementary MOS is a combination of two networks: The first is PUN, which is a pull-up network second is PDN, which is a pull-down network. Two networks, PUN and PDN, are connected one after the other, as shown in the following diagram.



An input is applied to both pull-up as well as pull-down networks. V_{DD} is the supply voltage. The pull-up network (PUN) is used to fetch the output level to logic high, which means logic one. So pull-up network is used to connect V_{DD} to the output. The output is denoted by F.

The pull-down network (PDN) is used to produce a logic low at the output. So, the output is connected to the ground through PDN. Input is applied to both PUN as well as PDN. PUN pulls output to logic one, and PDN pulls output to logic zero.

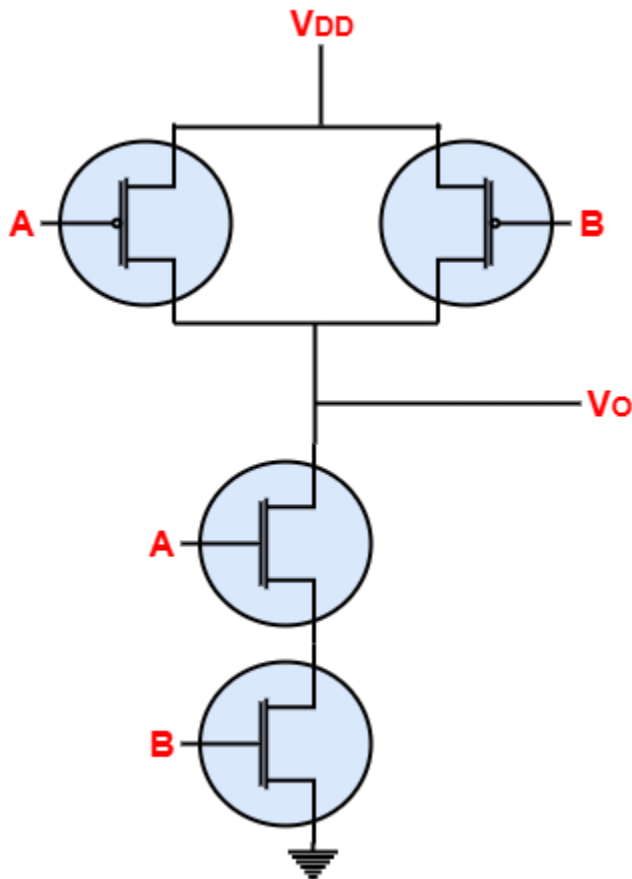
To design the PDN network, n-MOSFETs are preferred because they produce strong logic zeros. For PUN, p-MOSFETs are used because they produce strong ones.

Q..4.9 Draw NAND, NOR, AND, OR gates using CMOS. (DEC-23 9 Marks)
OR
Draw two input AND and OR gates using CMOS. (DEC-24 6 Marks)

NAND Gate Using CMOS:

The Boolean expression for the NAND gate is $Y = \overline{A \cdot B}$. We have to use PUN and PDN. The dot (.) between A and B indicates a parallel connection, and for PUN, we have to use p-MOSFET. So, for PUN, we will draw two p-MOSFETs in parallel.

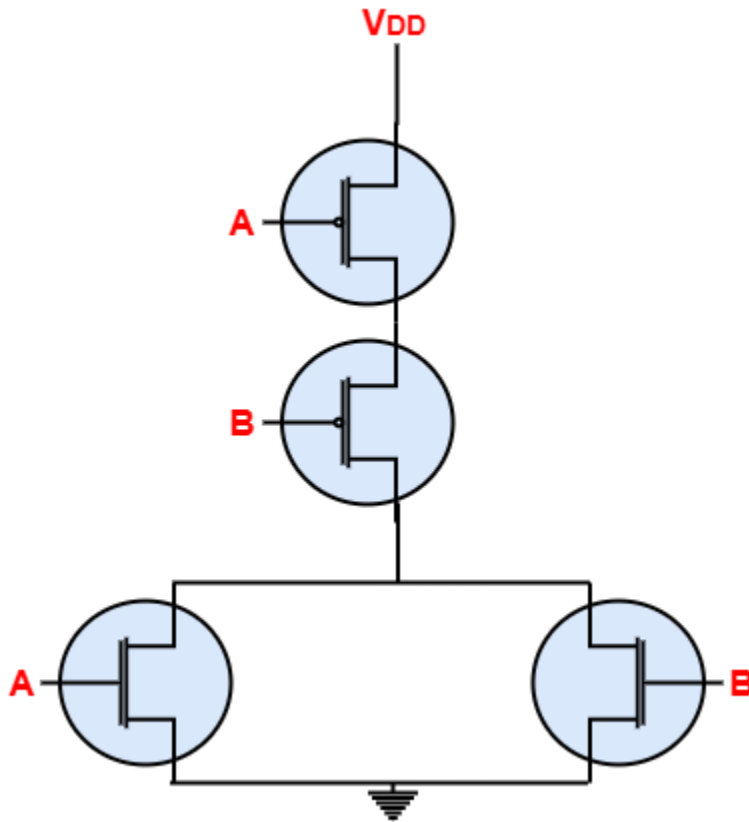
The lower part will be PDN. PDN is the inverse of PUN. For PDN, instead of a parallel connection, use a series connection. Similarly, for PDN use n-MOSFET. The two-input NAND gate is drawn by combining PUN and PDN, as shown in the following figure.



NOR Gate Using CMOS:

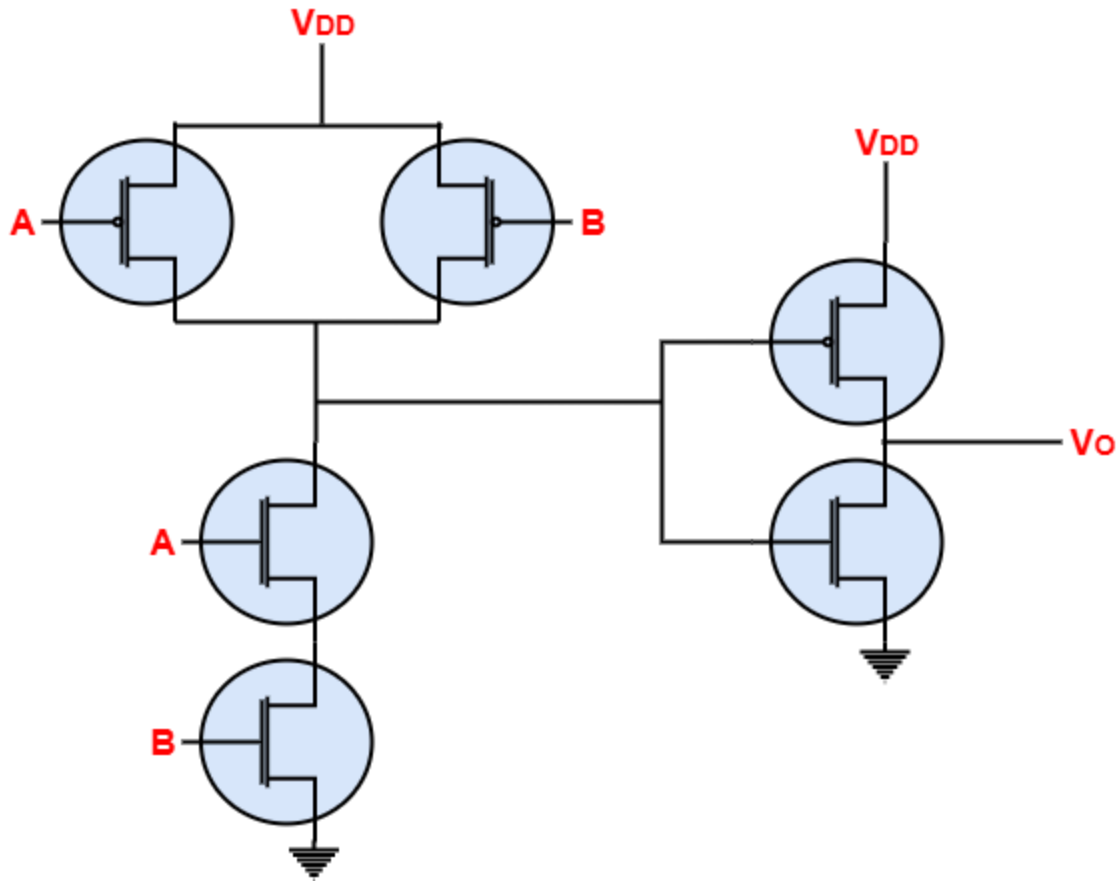
The Boolean expression for the NOR gate is $Y = \overline{A + B}$. We have to use PUN and PDN. The plus sign (+) between A and B indicates a series connection, and for PUN, we have to use p-MOSFET. So, for PUN, we will draw two p-MOSFETs in series.

The lower part will be PDN. PDN is the inverse of PUN. For PDN, instead of a series connection, use a parallel connection. Similarly, for PDN, use n-MOSFET. The two-input NOR gate is drawn by combining PUN and PDN, as shown in the following figure.



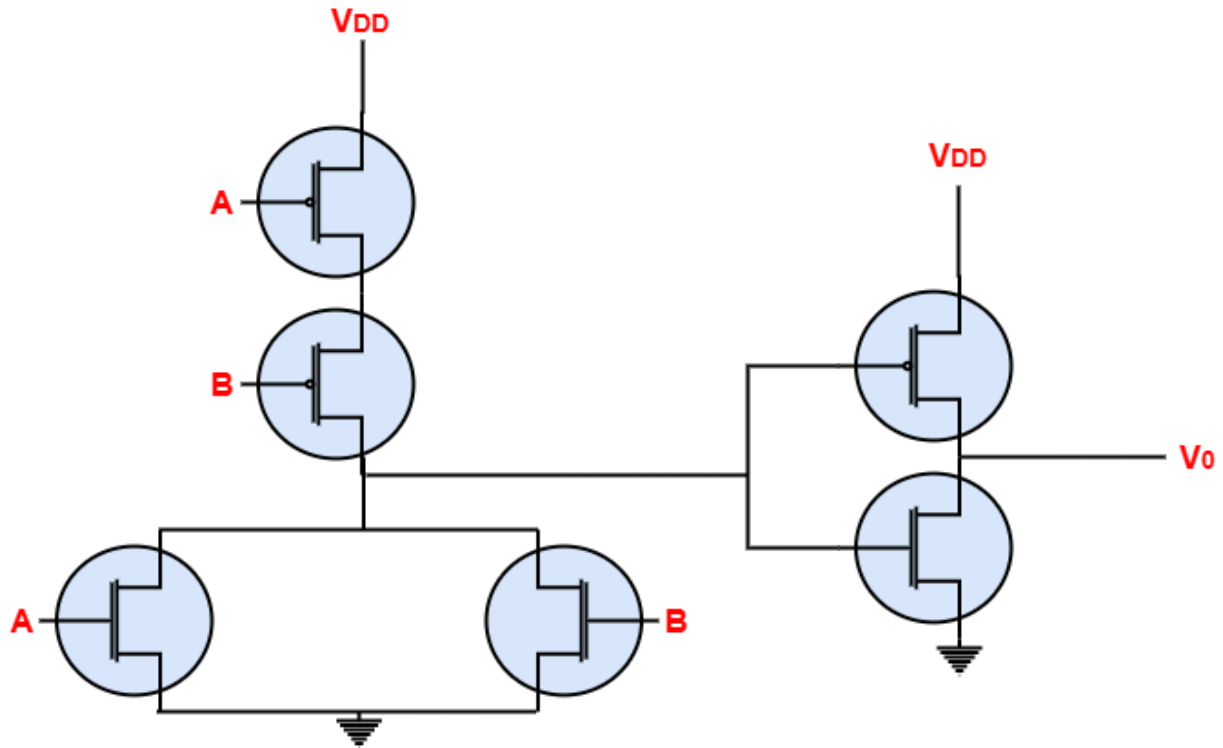
AND Gate Using CMOS:

The Boolean expression for the NAND gate is $Y = \overline{A \cdot B}$, OR gate is the inversion of the NAND gate. So, it is implemented by connecting an inverter at the output of the NAND gate. An inverter is designed by connecting a p-MOSFET and an n-MOSFET one below the other.



OR Gate Using CMOS:

The Boolean expression for the NAND gate is $Y = A.B$, AND gate is the inversion of the NOR gate. So, it is implemented by connecting an inverter at the output of the NOR gate.



Q..4.10 Design cmos logic for $Y = \overline{A}B + A\overline{B}$ (MAY-24 8 Marks)

Given: $Y = \overline{A}B + A\overline{B}$

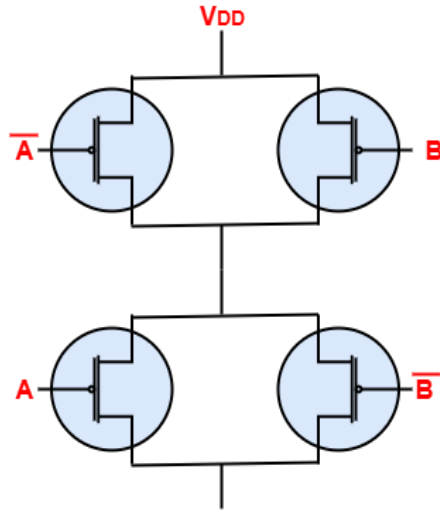
PUN:

The first term is $\overline{A}.B$, here dot (.) indicates a parallel connection of $\overline{A}$ and B. Similarly, for the second term, $A.\overline{B}$, the dot indicates a parallel connection of A and $\overline{B}$. Between the first two terms of equation 1, the plus sign (+) indicates a series connection. For PUN, we have to use p-MOS.

Thus, the given equation can be expressed as

$$Y = (\overline{A} \text{ parallel } B) \text{ series } (A \text{ parallel } \overline{B}) \dots \dots \dots (1)$$

The design of PUN is shown in the following figure.

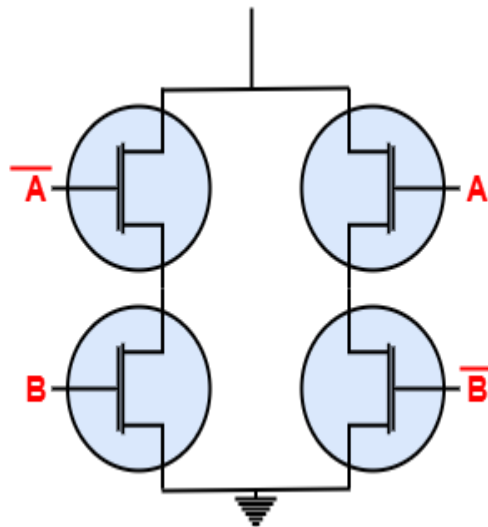


PDN:

It is the inverse of PUN. Thus inverse of equation (1) is,

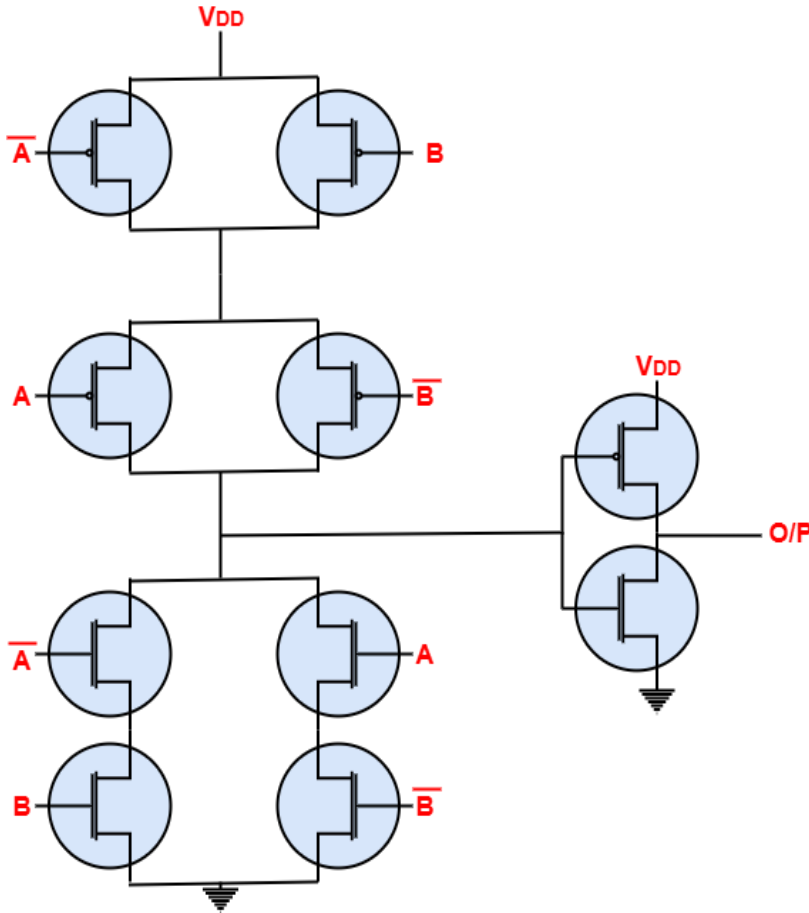
$$Y = (\bar{A} \text{ series } B) \text{ parallel } (A \text{ series } \bar{B}) \dots \dots \dots (1)$$

For PDN, we have to use an n-MOSFET. The design of PDN is shown in the following figure.



The complete design is obtained by connecting PUN and PDN as shown in the following figure.

{NOTE: The CMOS design contains a complementary equation. In the given equation, the bar is not present above the whole equation. So, we will add an inverter at the output.}



Q..4.11 Design CMOS logic for $F = AB + C(D+E)$ (DEC-22 5 Marks)
OR
 Draw CMOS logic for $\overline{AB + C(D + E)}$. Calculate W/L ratio for n-MOS and p-MOS as well as area needed on the chip. (DEC-24 8Marks)

The given equation is,

$$Y = \overline{AB + C(D + E)}$$

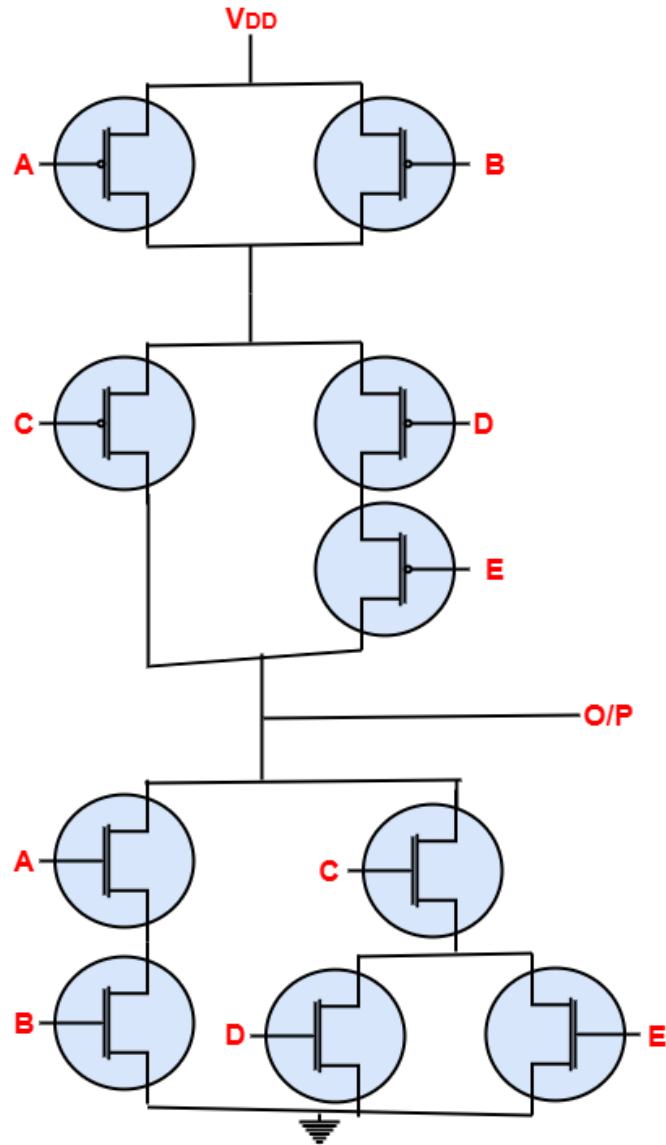
PUN: (Use p-MOSFET)

$$Y = (A \text{ parallel } B) \text{ series } [c \text{ parallel } (D \text{ series } E)]$$

PDN: (Use n-MOSFET)

$$Y = (A \text{ series } B) \text{ parallel } [c \text{ series } (D \text{ parallel } E)]$$

The complement design is obtained by connecting PUN and PDN as shown in the following figure.



W/L ratio:

Let us denote the width of n-MOSFET by W_n and p-MOSFET by W_p . When p-MOSFETs are in parallel, the width of the parallel connection is $1W_p$. This is for the PUN. Now, for PDN, the number of n-MOSFETs = Number of W_n .

For PUN $\Rightarrow$ There are two parallel connections.

$$\therefore \text{Area of PUN} = W_p + W_p = 2W_p$$

For PDN $\Rightarrow$ Number of n-MOSFETs = 5

$$\therefore \text{Area of PDN} = 5W_n$$

$\therefore$ Total area = $2W_p + 5W_n$
 Let L be the length of NMOS and PMOS transistors.
 $\therefore$ W/L ratio of p-MOS transistors = $2W_p/L$
 And W/L ratio of n-MOS transistors = $5W_n/L$

Q..4.12 Design CMOS logic for $Y = AB + CD$. Calculate W/L ratio for NMOS and PMOS area needed on chip. Find the total area..... (MAY-23 8Marks)

The given equation is, $Y = AB + CD$

PUN: (Use p-MOSFET)

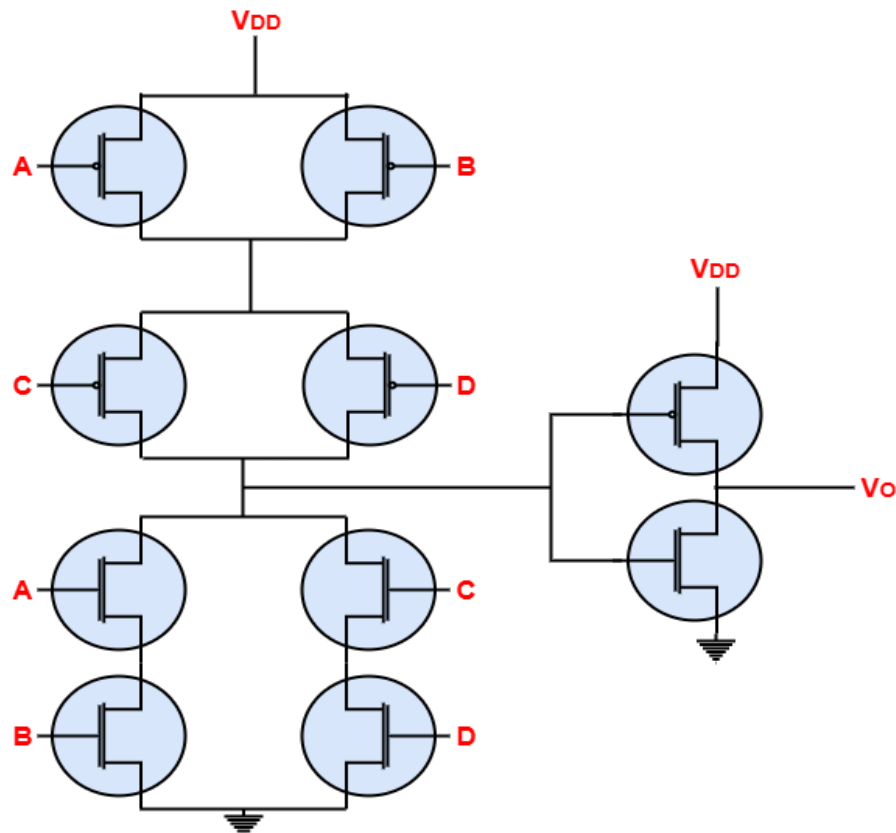
$Y = (A \text{ parallel } B) \text{ series } (C \text{ parallel } D)$

PDN: (Use n-MOSFET)

$Y = (A \text{ series } B) \text{ parallel } (C \text{ series } D)$

The complement design is obtained by connecting PUN and PDN as shown in the following figure.

{NOTE: The CMOS design contains a complementary equation. In the given equation, the bar is not present above the whole equation. So, we will add an inverter at the output.}



Let us denote the width of n-MOSFET by W_n and p-MOSFET by W_p . When p-MOSFETs are in

parallel, the width of the parallel connection is $1W_p$. This is for the PUN. Now, for PDN, the number of n-MOSFETs = Number of W_n .

For PUN $\Rightarrow$ There are two parallel connections.

$$\therefore \text{Area of PUN} = W_p + W_p = 2W_p$$

For PDN $\Rightarrow$ Number of n-MOSFETs = 4

$$\therefore \text{Area of PDN} = 4W_n$$

$$\therefore \text{Total area} = 2W_p + 4W_n$$

Let L be the length of NMOS and PMOS transistors.

$$\therefore \text{W/L ratio of p-MOS transistors} = 2W_p/L$$

$$\text{And W/L ratio of n-MOS transistors} = 4W_n/L$$

Q..4.13 Draw CMOS logic for $Y = \overline{AB + CD + E}$. Calculate W/L ratio for NMOS and PMOS & area needed on the chip. (DEC-23 10 Marks)

The given equation is, $Y = \overline{AB + CD + E}$

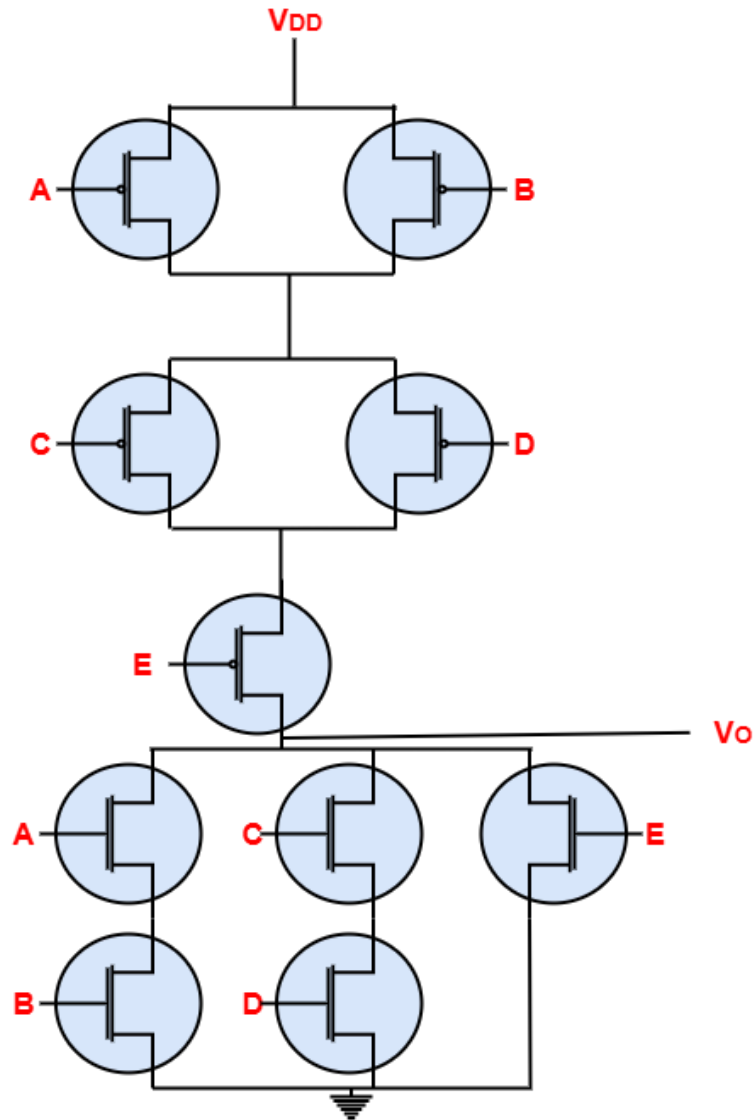
PUN: (Use p-MOSFET)

$$Y = (A \text{ parallel } B) \text{ series } (C \text{ parallel } D) \text{ series } E$$

PDN: (Use n-MOSFET)

$$Y = (A \text{ series } B) \text{ parallel } (C \text{ series } D) \text{ parallel } E$$

The complement design is obtained by connecting PUN and PDN as shown in the following figure.



Let us denote the width of n-MOSFET by W_n and p-MOSFET by W_p . When p-MOSFETs are in parallel, the width of the parallel connection is $1W_p$. This is for the PUN. Now, for PDN, the number of n-MOSFETs = Number of W_n .

For PUN $\Rightarrow$ There are three parallel connections.

$$\therefore \text{Area of PUN} = W_p + W_p + W_p = 3W_p$$

For PDN $\Rightarrow$ Number of n-MOSFETs = 5

$$\therefore \text{Area of PDN} = 5W_n$$

$$\therefore \text{Total area} = 3W_p + 5W_n$$

Let L be the length of NMOS and PMOS transistors.

$$\therefore \text{W/L ratio of p-MOS transistors} = 3W_p/L$$

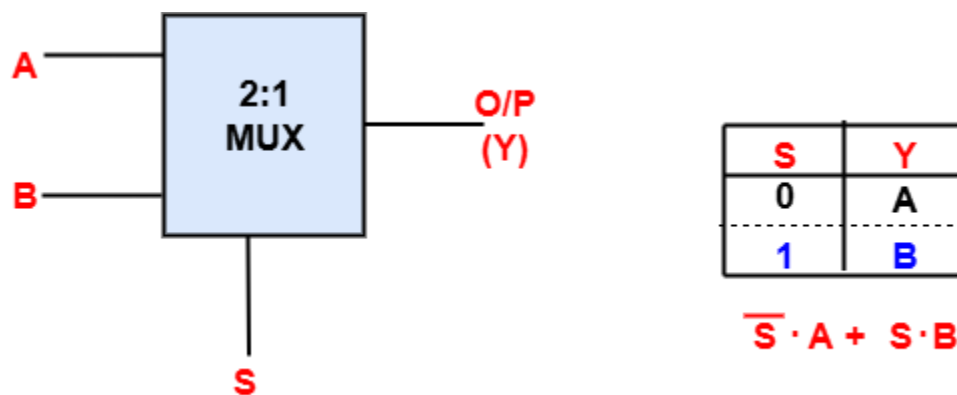
$$\text{And W/L ratio of n-MOS transistors} = 5W_n/L$$

Q..4.14 Draw 2:1 Mux using CMOS as well as TG. Comment on the savings of transistors using TG and CMOS method..... (DEC-22 12 Marks)

OR

What are the merits of transmission gate and design 2:1 multiplexer using transmission gate.
..... (MAY-24 12 Marks)

The transmission gate contains both the transistors: n-MOSFET and p-MOSFET, which are connected in parallel. It acts as a bidirectional switch. The block diagram of a 2:1 multiplexer and its truth table are shown in the following figure.



There are two inputs, A and B. One select line (S) is used. At a time, either of the two inputs will be selected. Refer to the truth table. If S is zero, then the first line A will be selected output will be A. If S is one, then the second line, that is second input B, is selected, so the output will be B.

A select line S can be zero or one. If it is zero, it is written as $\overline{S}$, and the corresponding output is A. So, the first term is $\overline{S} \cdot A$. The second value of S is one, and the corresponding output is B. So the second term is $S \cdot B$. Thus, the equation of output is $Y = \overline{S} \cdot A + S \cdot B$.

CMOS Logic:

The given equation is $Y = \overline{S} \cdot A + S \cdot B$.

PUN: (Use p-MOSFET)

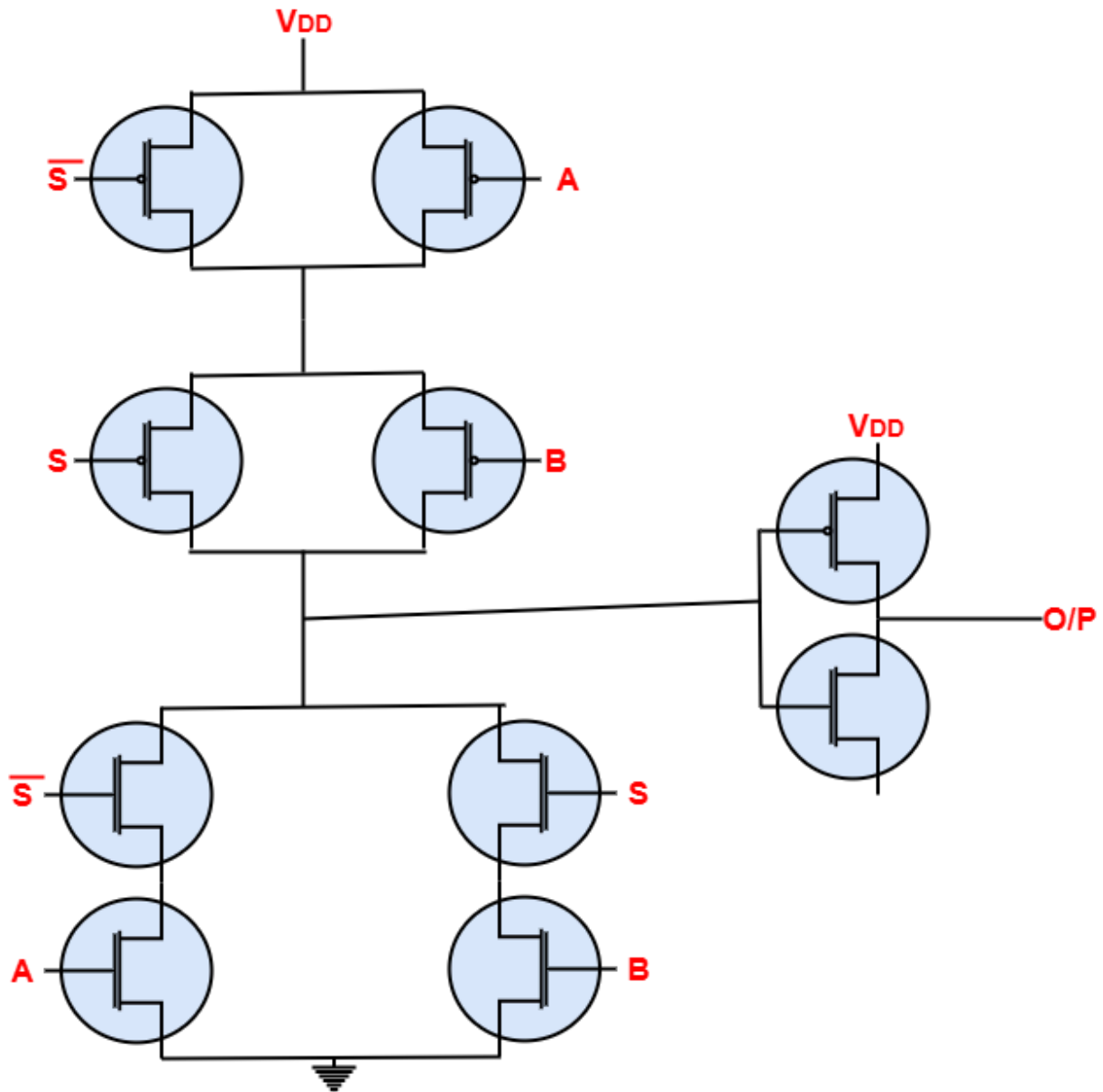
$$Y = (\overline{S} \text{ parallel } A) \text{ series } (S \text{ parallel } B)$$

PDN: (Use n-MOSFET)

$$Y = (\overline{S} \text{ series } A) \text{ parallel } (S \text{ series } B)$$

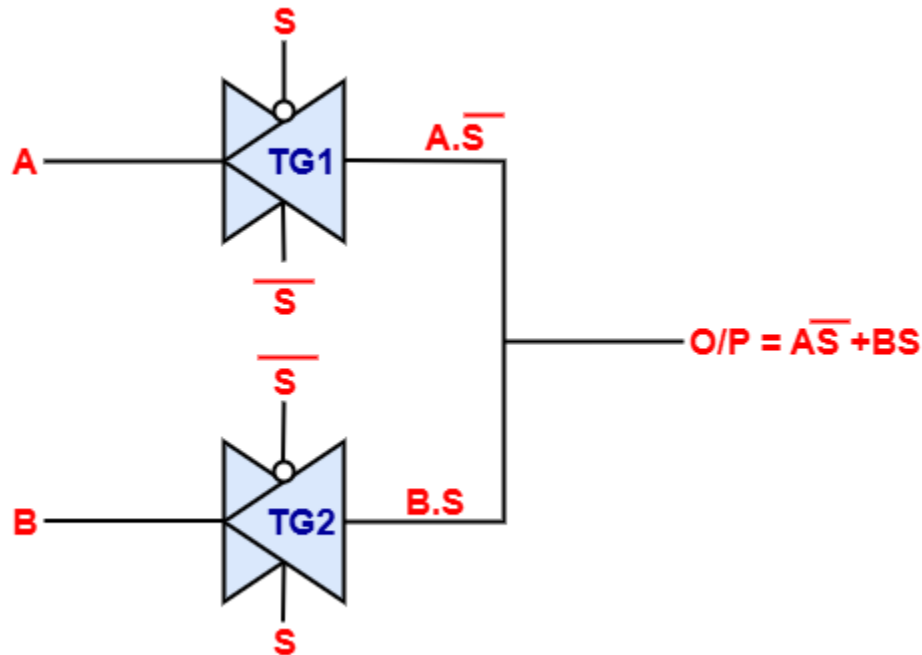
The complement design is obtained by connecting PUN and PDN as shown in the following figure.

{NOTE: The CMOS design contains a complementary equation. In the given equation, the bar is not present above the whole equation. So, we will add an inverter at the output.}



Transmission Gate (TG) Logic:

There are two inputs (A and B), so use two TG. At the input side of TG, inputs A and B are written as shown in the following figure. The first term is $A\bar{S}$, so below TG1 $\bar{S}$ is written. At the upper side of TG1, a bubble is there, so S is written. At the output of TG1, we will get the first term $A\bar{S}$. Similarly, the second term is $B.S$, so below TG2 'S' is written. At the upper side of TG2, due to a bubble, $\bar{S}$ is written. The output of TG2 is $B.S$. The Final output is the addition of two terms. If we need inverted output, we will use the inversion logic at the output.



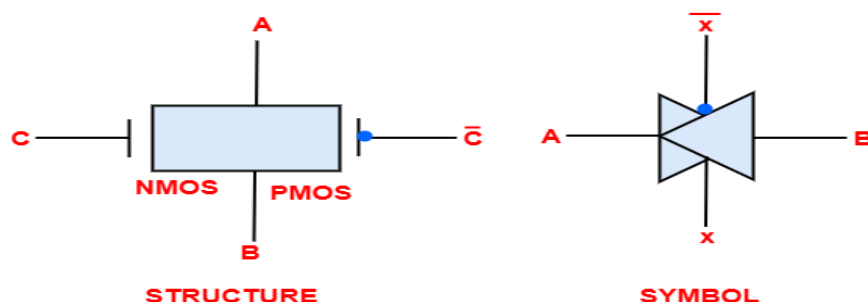
Comment on the savings of transistors:

One Transmission Gate consists of two transistors: one n-MOSFET and one p-MOSFET. So, four transistors are used. And at the output, if an inverter is used, then again two transistors are required. Thus, six transistors are required. Whereas in CMOS logic, 10 transistors are required. Thus, the number of transistors required in the case of a transmission gate is less compared to the CMOS logic.

Q..4.15 Discuss need for transmission gate. Draw 4 : 1 Mux using TG.. (MAY-23, DEC-24 8 Marks)
OR
 Explain the working of transmission gate and implement a circuit 4 : 1 Mux using Transmission Gate.
(MAY-25,10 Marks)

Working of Transmission Gate:

The Transmission Gates utilize both n-MOSFET and p-MOSFET transistors, which are connected in parallel. It acts as a bidirectional switch. The structure of the transmission gate is as shown in the following figure.



At the LHS, an n-MOSFET is used, and at the RHS, a p-MOSFET is used. Both these transistors are connected in parallel. C and $\bar{C}$ are the control signals. These are used to control the transmission through the gate. Now, if $C=1$ n-MOSFET transistor will be switched ON. But if C is 1, $\bar{C}$ (complement of C) becomes zero. There is a bubble at p-MOSFET, so it will again activate p-MOSFET. That means both these transistors are ON, and whatever signal is applied at Point A will appear at Point B. So, if A is zero, B is also zero, and if A is one, B is also one.

If C is equal to zero, the n-MOSFET is switched OFF. $\bar{C}$ becomes one, so p-MOSFET is again OFF. So, both the transistors will be OFF when C is zero. In that case, this switch acts as an open circuit.

In this symbol, x and $\bar{x}$ are the control signals. When x is 1 $\bar{x}$ is zero. So, $A=B$, which means A will be transferred to the output side, because the switch is closed. And when x is 0, $\bar{x}$ is 1, both the transistors are OFF. The switch will not operate, and it will not allow the input to pass through this gate. The truth table is as follows.

x	A	B
1	0	0
1	1	1
0	0	H.I.
0	1	H.I.

If x is 1 and A is zero, B is also zero; B is the output. If x is 1, A is 1, and B is 1, that means whenever x is 1, whatever you are applying at the input will be available at the output. But if x

is 0, $\bar{x}$ becomes 1, and the transmission gate will not operate. It will act as an open circuit. In that case, it is said that the transmission gate is in the high impedance(H.I.)state.

Need for Transmission Gate:

The transmission gate (TG) is used for signal switching and logic implementation. A transmission gate is a bilateral switch. It uses both an NMOS and a PMOS transistor in parallel. A single NMOS transistor passes a strong logic '0' and a weak logic '1'. A single PMOS transistor passes a strong logic '1' and a weak logic '0'. This also causes signal degradation.

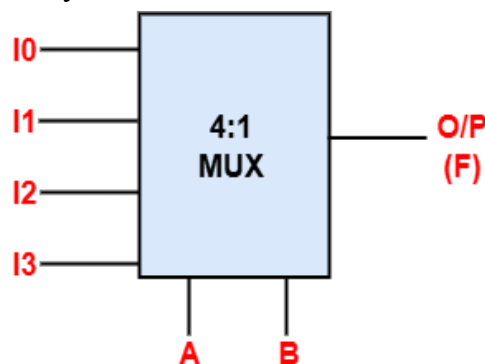
The Transmission Gate combines both to pass strong '0' and strong '1'. The NMOS passes a strong '0'. The PMOS compensates by passing a strong '1'.

Advantages: Transmission gates offer several benefits over conventional logic gates and simple pass transistors.

- A TG can pass a signal in both directions.
- The parallel connection of NMOS and PMOS results in a more constant ON resistance. This provides a faster switching speed.
- Using transmission gate logic can implement complex functions with fewer transistors than conventional CMOS.
- When the control signals are off, the output enters a high-impedance state. This is necessary for bus structures and selecting one of many driving circuits.

4 : 1 Mux using TG

The block diagram of a 4:1 multiplexer is shown in the following figure. There are four inputs. At a time, one particular input is selected. So, one output is there, which is denoted by F. There are four inputs: I0, I1, I2, and I3. To select any one of the four inputs, two logic bits are required. The select bits are denoted by A and B. These are the control bits.



The truth table is as follows.

A	B	F
0	0	I0
0	1	I1
1	0	I2.
1	1	I3

If A and B are two select bits, the combination will be 00, 01, 10, and 11. If A and B are 0 0 first line, I0 will be selected, so the first output is I0. If the next combination is 01, the second line I1 will be selected. For the third combination 10, the next input I2 is available at the output. And for the fourth combination,

To write the expression, whenever zero is there, make use of bar (complement) notation. The first entry in truth is 00I0, so the equation is $\bar{A} \bar{B} I0$. For the second entry 01I1, the equation is $\bar{A} B I1$. For the third entry 10I2, the equation is $A \bar{B} I2$. For the fourth entry 11I3, the equation is, $A B I3$. Thus, the final equation is,

$$F = \bar{A} \bar{B} I0 + \bar{A} B I1 + A \bar{B} I2 + A B I3.$$

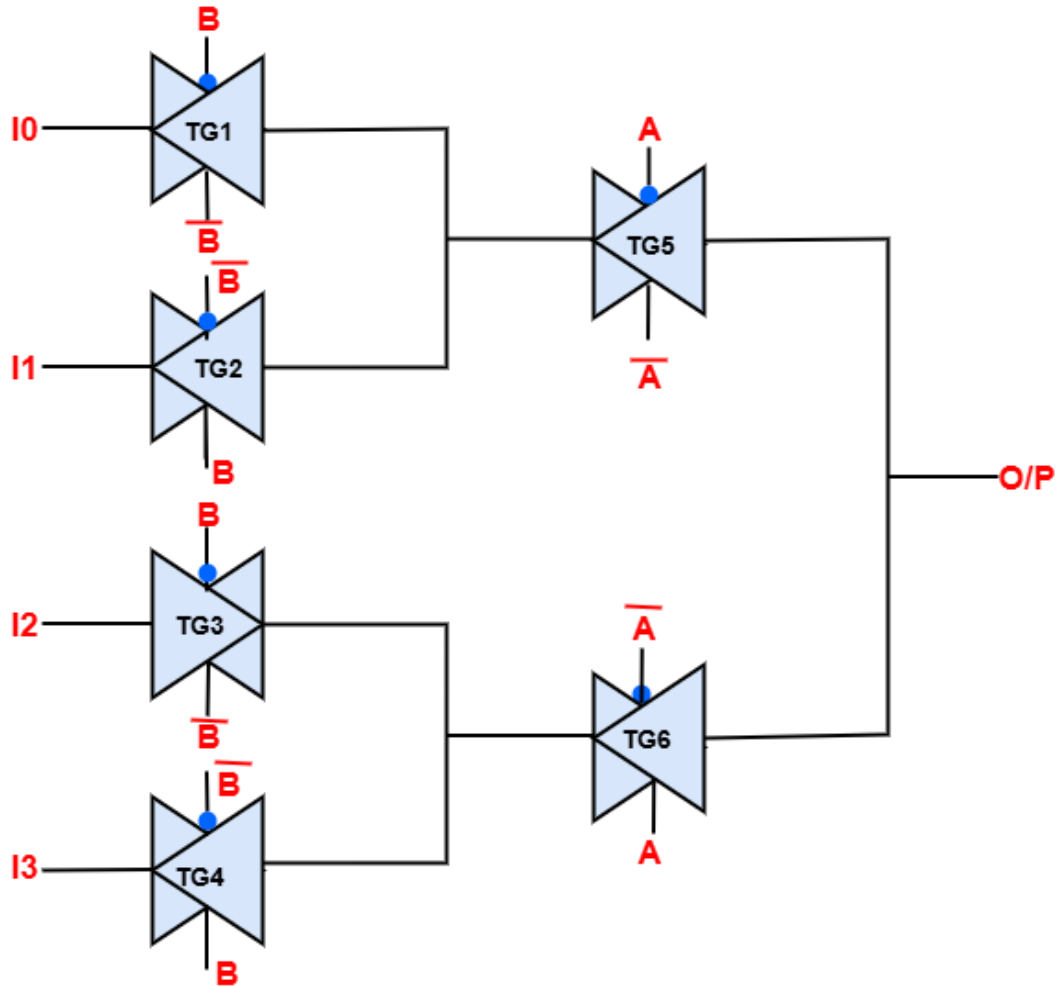
Take $\bar{A}$ the common from the first two terms, and A common from the third and fourth terms.

$$\therefore F = \bar{A} (\bar{B} I0 + B I1) + A (\bar{B} I2 + B I3).$$

The transmission gate logic is drawn in the following figure. Four inputs, I0, I1, I2, and I3, are written at the input side. Consider the bracket term $(\bar{B} I0 + B I1)$. Here I0 is multiplied by $\bar{B}$. So, below TG1 $\bar{B}$ is written. For the second term. I1 is multiplied by B, so the term B is written below the second TG. Now consider the second bracket $(\bar{B} I2 + B I3)$. Here $\bar{B}$ is written below TG3, and B is written below TG4. At the upper side of each TG, the complement of notations is written.

Now, the first bracket $(\bar{B} I0 + B I1)$ is multiplied by $\bar{A}$, so below TG5 $\bar{A}$ is written. Similarly, the second bracket $(\bar{B} I2 + B I3)$ is multiplied by A, so below TG6, B is written. Thus, the final output is

$$F = \bar{A} (\bar{B} I0 + B I1) + A (\bar{B} I2 + B I3).$$



Q..4.16 A CMOS logic is operating at 10 MHz and 3 V with the load of 100 pF. The static power dissipation is 100 μ W. Calculate the total power dissipation if the frequency is increased to 100 MHz.....(DEC-24,10 Marks)

Given: Supply Voltage (V_{DD}) = 3 V

Frequencies: $f_1 = 10 \text{ MHz} = 10 \times 10^6 \text{ Hz}$, $f_2 = 100 \text{ MHz} = 100 \times 10^6 \text{ Hz}$

Load capacitance, $C = 100 \text{ pF} = 100 \times 10^{-12} \text{ F}$

Static power, $P_s = 100 \text{ } \mu\text{W} = 100 \times 10^{-6} \text{ W}$

Calculation of dynamic power at 10 MHz:

Formula $\Rightarrow P_D = C V_{DD}^2 f_1$

$$\therefore P_D = 100 \times 10^{-12} \times 3^2 \times 10 \times 10^6$$

$$\therefore P_D = 9 \times 10^{-3} \text{ W}$$

Calculation of total power at 10 MHz:

Formula $\Rightarrow P_T = P_D + P_S$

$$\therefore P_T = 9 \times 10^{-3} + 100 \times 10^{-6}$$

$$\therefore P_T = 9.1 \times 10^{-3} \text{ W}$$

Calculation of dynamic power at 100 MHz:

Formula $\Rightarrow P_D = C V_{DD}^2 f_1$

$$\therefore P_D = 100 \times 10^{-12} \times 3^2 \times 100 \times 10^6$$

$$\therefore P_D = 0.09 \text{ W}$$

Calculation of total power at 100 MHz:

Formula $\Rightarrow P_T = P_D + P_S$

$$\therefore P_T = 0.09 + 100 \times 10^{-6}$$

$$\therefore P_T = \mathbf{0.0901 \text{ W}}$$

.....**END**.....

VLSI

(PART-2) (UNITS 5 &6)

END-SEM NOTES

UNIT- 5

Application Specific Integrated Circuits

Q.5.1 Draw and explain ASIC design flow. (MAY-23, DEC-24, MAY-25 6 Marks)

ASIC design flow is shown in the following figure. The different blocks are as follows.

Design Entry: We have to enter the design specifications. You may use a system schematic entry or a high-level description language such as VHDL or Verilog.

Logic Synthesis: It is synthesized using certain logical techniques. It contains the synthesis tools, which are used to transform the design description into the net list. That means we have to transform a high-level description into simple logical blocks, using this step.

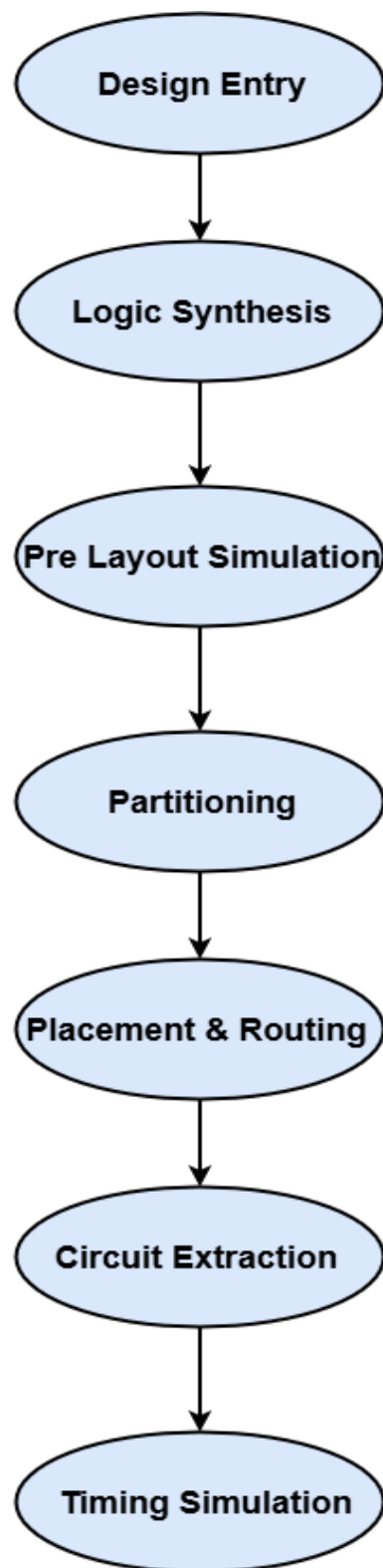
Pre-Layout Simulation: It is used for logical corrections. It consists of simulation tools that are used for making corrections. It first checks the logical connections and then checks for logical correctness.

Partitioning: The entire system is divided into several smaller blocks, and then the floor planning is done. The floor planning means placing different blocks on the chip.

Placement and Routing: This step decides the location of cells in a block, as well as making connections between cells and the block, and care is taken that the interconnection should be as minimal as possible

Circuit Extraction: It determines the parasitic capacitance as well as resistance.

Timing Simulation: This is also called a post-simulation block. It checks the functionality of the circuit after the extraction of the parasitic components, and it uses the timing analysis



Q.5.2 Explain DC and transient analysis using SPICE..... (MAY-25 6 Marks)

The SPICE (Simulation Program with Integrated Circuit Emphasis) is used for checking circuit behavior. DC and Transient analysis are the main tools.

DC Analysis:

DC analysis finds the steady-state voltages and currents in the circuit. It calculates the fixed operating point, or bias point, for all transistors. It ensures transistors are correctly set up (e.g., in saturation or cutoff) to act as working switches or amplifiers.

SPICE Command: Usually, the **.OP** (Operating Point) command is used.

Transient Analysis

Transient analysis shows how circuit voltages and currents change over time. It is essential to see the circuit's dynamic response. It simulates the circuit when inputs are changing, like a digital signal switching from '0' to '1'. This is used to find the following values:

- Propagation Delay: How fast a signal travels through a logic gate.
- Rise/Fall Times: How quickly a voltage level changes.
- Timing: Checking if the circuit meets required speed specifications.

SPICE Command: The **.TRAN** (Transient) command, specifying the total simulation time.

These analyses are critical for:

1. Gate Characterization: Running transient analysis on a basic logic gate (like an inverter) to measure its speed and power consumption.
2. Power Check: DC analysis helps estimate static (leakage) power, and transient analysis helps calculate dynamic power consumption.
3. Circuit Verification: Confirming the fabricated design will work correctly at the required clock speed.

Q.5.3 Write SPICE code for CMOS inverter for AC analysis..... (DEC-23, DEC-24, MAY-25 6 Marks)

- File: cmos_ac_analysis.cir

1. Supply Voltage Definition:

VDD VDD 0 DC 5V; 5V DC Power Supply (Node VDD to Ground/Node 0)

2. Input Signal Definition:

- The input voltage source (VIN) needs a DC bias for the .AC analysis
- to place the inverter in its high-gain region (around VDD/2).
- VDC: DC component (2.5V bias)
- VAC: AC component (1mV small-signal test) VIN IN 0 DC 2.5V AC 1m

3. CMOS Inverter Structure:

- M1: PMOS Transistor (Pull-Up)
- D G S B (Drain Gate Source Bulk/Well)
- M1 VDD IN OUT VDD PMOS W=10U L=1U M1 OUT IN VDD VDD PMOS W=10u L=1u
- M2: NMOS Transistor (Pull-Down)
- D G S B (Drain Gate Source Bulk/Well)
- M2 OUT IN 0 0 NMOS W=5u L=1u
- Note on Transistor Sizing (W/L):
- W_P/L_P (10/1) is typically made larger than W_N/L_N (5/1)
- to balance the rise and fall times.

4. Transistor Model Definitions:

- Note: Real VLSI simulations use complex, technology-specific models. MODEL NMOS NMOS (VTO=0.7 KP=100u GAMMA=0.4 LAMBDA=0.05 LEVEL=1) .MODEL PMOS PMOS (VTO=-0.7 KP=40u GAMMA=0.4 LAMBDA=0.05 LEVEL=1)

5. Analysis Command:

- .AC DEC points_per_decade start_freq stop_freq
- This sweeps 10 points per decade from 1Hz up to 1GHz. .AC DEC 10 1 1G

.END

Q.5.4 What is the need for design rules in VLSI ? (6 Marks)

In the case of IC fabrication, the different layers are present. Certain active components are used. The design rules are used so that the circuit performance will not be degraded, and there will not be any side effects such as open circuit, short circuit, etc. These rules are applicable as follows.

- To decide the minimum spacing between all layers of the circuit. There are different layers, so by making use of these rules, we have to decide the minimum spacing between the different layers of the circuit.
- It provides a set of guidelines to construct different masks in the IC fabrication process.
- It provides minimum spacing and minimum width between objects of different layers. In every layer, there are many objects. These rules provide the required minimum line width.
- These rules are used to transform the circuit concepts into the actual geometry in the silicon material.
- It prevents unwanted short circuits or open circuits in the thin line. If there is a thin line between the two layers, it may happen that after a particular distance, the line becomes an open circuit. So, there will be an unnecessary open circuit in the design. Similarly, if the two lines joining two different layers are very close to each other, then there may be a short circuit. But if you are using such design rules, then there won't be any short circuit or open circuit.
- These are the rules designed in such a way that there will not be any degradation of the circuit performance.

Q.5.5 Which Lambda rules are used for CMOS Layout? Give its significance..... (DEC-22 6 Marks)

OR

Discuss Lambda rules with diagram? (Any 6 rules). (MAY-23 6 Marks)

OR

Explain LAMBDA rules used for CMOS layout design..... (DEC-23, DEC-24, MAY-25 6 Marks)

OR

Explain in detail lambda design rules in CMOS VLSI. (MAY-24 6 Marks)

In the case of MOS design, different layers are used. Many metal connections are present. Different contacts (Via) are made. These contacts are used for the interconnection between different things. So certain λ rules are used while designing the particular circuit.

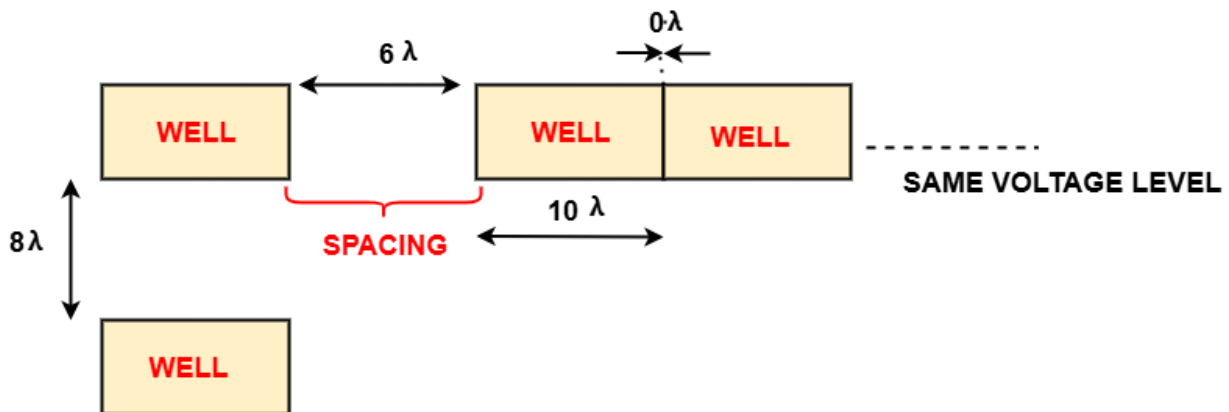
Lambda is a parameter that is set to a certain value. All the dimensions, like the gap between the two layers, the spacing between the contact and a particular polysilicon layer, etc, are expressed in terms of the Lambda parameter (λ). The different λ rules are as follows:

1) For n-Well or p-Well:

The word well stands for substrate. In the case of an n-MOSFET, a p-type substrate is used. And in the case of a p-MOSFET, an n-type substrate is used. For these substrates, the rules are as follows:

- The minimum width = 10λ .
- The wells are at the same voltage level with a spacing = 6λ .
- The wells are at the same voltage level without spacing = 0λ .
- The wells of different types, the spacing = 8λ .

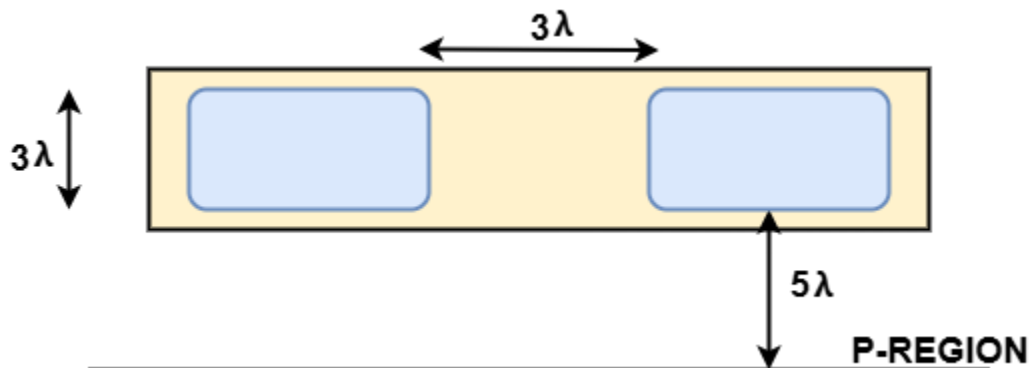
It is shown in the following figure.



2) For active area:

- The minimum width and minimum spacing for the active area = 3λ .
- Source or drain active to well edge spacing = 5λ .
- Substrate or well contact active to well edge spacing = 3λ .

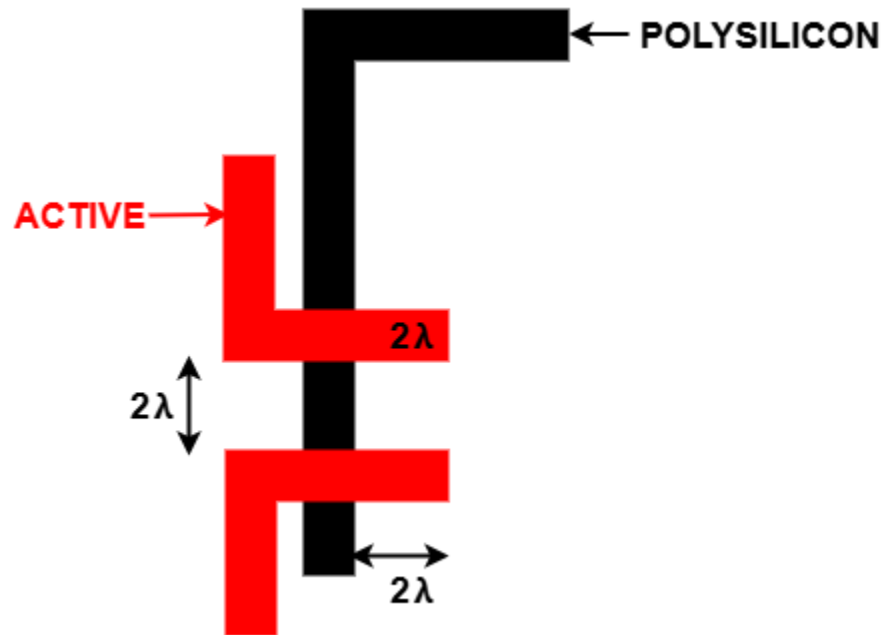
It is shown in the following figure.



3) For poly silicon layer:

- The minimum width and minimum spacing for the active area = 2λ .
- The minimum gate extension = 2λ .
- The minimum spacing between the active area and the polysilicon layer = 1λ .

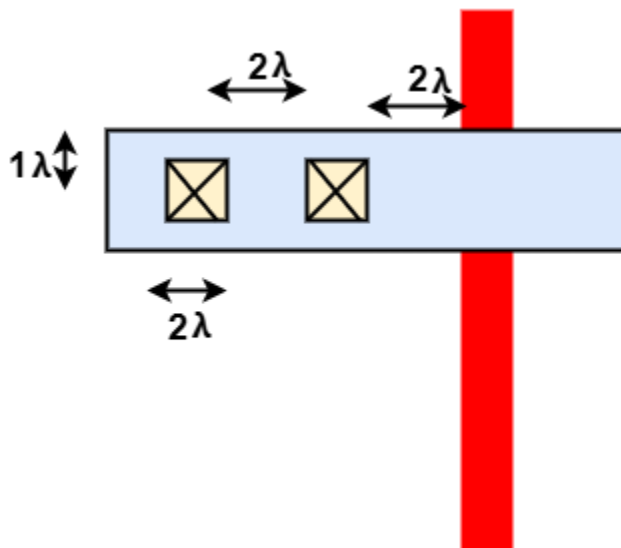
It is shown in the following figure.



4) For contacts:

- Exact contact size = 2λ
- Minimum spacing = 2λ
- Minimum spacing to gate = 2λ
- Minimum active overlap = 1λ

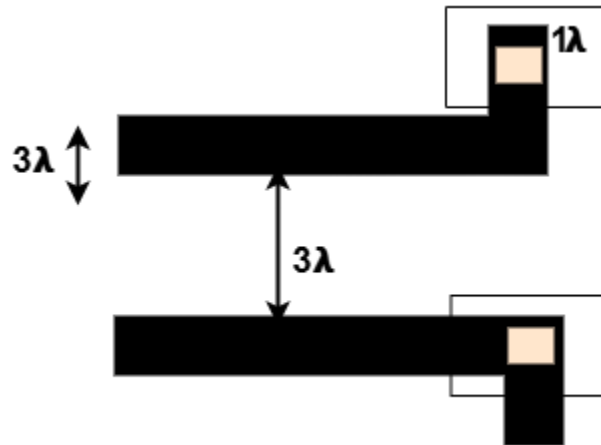
It is shown in the following figure.



5) For Metal-1:

- The minimum width and minimum spacing = 3λ
- Minimum overlap of paducontact and active contact = 1λ

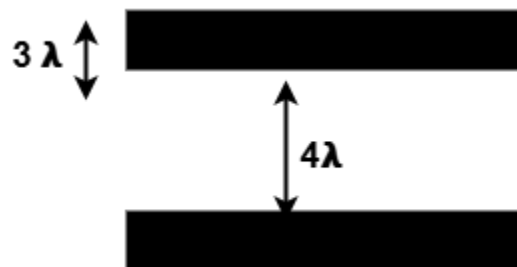
It is shown in the following figure.



6) For Metal-2:

- Minimum size = 3λ
- Minimum spacing = 4λ

It is shown in the following figure.



7) For Metal-3:

- Minimum size = λ
- Minimum spacing = 4λ

8) For Via-1:

- Exact size = 2λ
- Minimum spacing = 1λ
- Minimum spacing to contact = 2λ

- Minimum overlap by metal-1 = 2λ

9) For contact with the active area:

- Minimum active overlap = 1λ
- Minimum contact spacing = 2λ
- Exact contact size = $2\lambda \times 1\lambda$

Significance:

They use a basic unit, λ , to define all parts of the chip. If the manufacturing factory changes (improves or deteriorates), you simply adjust the size of λ , and the entire chip layout scales up or down automatically without requiring a redraw of everything. Essentially, λ rules enable chip design to be flexible and easy to manage across various manufacturing technologies.

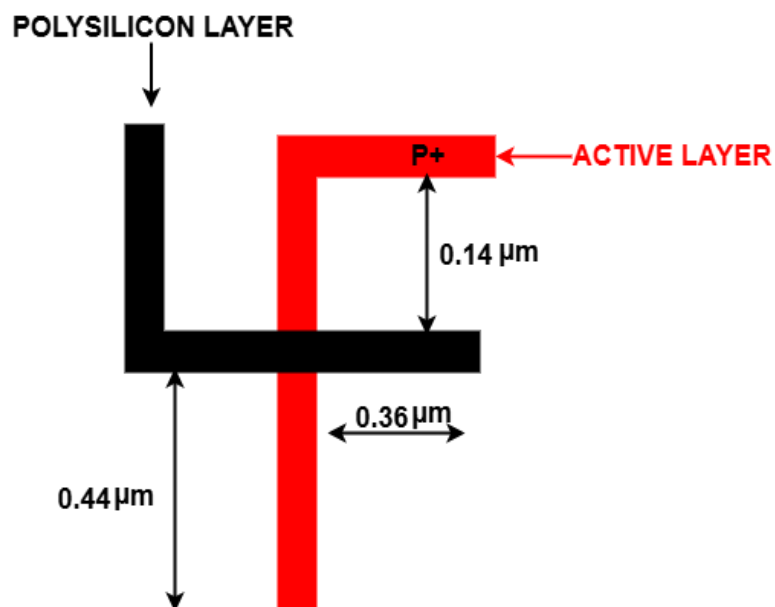
Q.5.6 Explain the layout design rules. (6 Marks)

There are three major types of layout rules.

- 1) Transistor rules
- 2) Contact and via rules
- 3) Well and substrate contact rules

1) Transistor rules:

The schematic for the P-transistor is shown in the following figure.



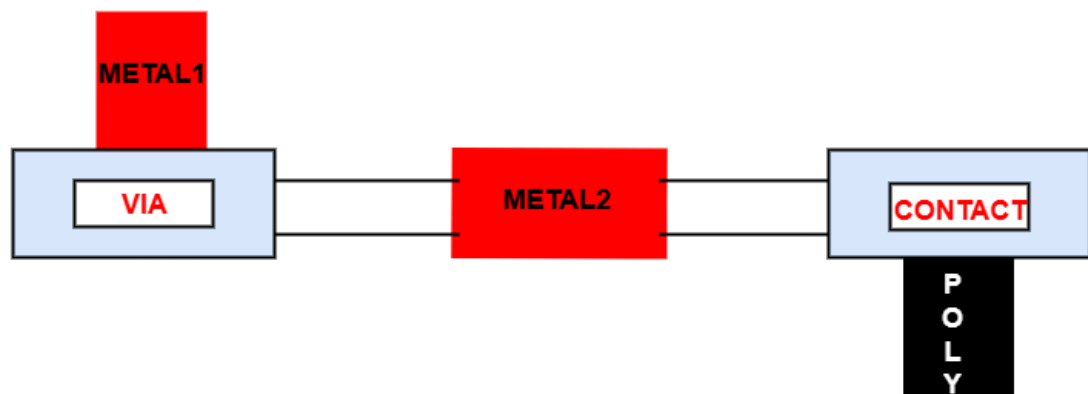
The transistor is formed by the overlapping of the substrate and the polysilicon layer, as shown in the diagram. A P+ substrate is used. The rules are as follows:

- The minimum length of a transistor = The minimum width of poly silicon = $24\mu\text{m}$.
- The width of the transistor = The minimum width of the active layer = $0.3\mu\text{m}$.

2) Contact and via rules:

These rules are explained in the following diagram.

POLYSILICON LAYER



The contacts and vias are used for interconnections. The contacts are used to make an interconnection between the polysilicon layer and the metal. Whereas the vias are used to provide interconnection between two metals.

3) Well and substrate contact rule:

The well and substrate must be properly connected to the power supply. If they are not properly connected, then a resistive part is created between the transistor contact with the substrate. It produces a parasitic effect.

Q.5.7 Write a short note on Design Rule Check (DRC)..... (6 Marks)

Design Rule Check (DRC) is an automatic process. DRC verifies a chip's physical layout. The layout must adhere to specific rules. These rules are called design rules. The main purpose is to

ensure the design is manufacturable. It prevents design issues. These issues could cause yield loss or device failure. Design rules reflect the limitations of the manufacturing process.

DRC software automatically scans the entire layout. It checks for many different types of errors. The major checks include:

- Minimum Width: Ensures all shapes (like wires or transistors) are wide enough.
- Minimum Spacing: Checks that different shapes are not too close together.
- Overlap/Enclosure: Verifies that one layer correctly covers or encloses another.
- Antenna Rule: Checks for metal lines that are too long before connecting to a protective diode.

The DRC Process is as follows:

- The designer finishes the physical layout of the circuit.
- They run a specialized DRC software tool.
- The tool compares the layout against the design rule deck. This file contains all the rules.
- The tool generates an error report. This report highlights all violations.
- The designer must fix all reported errors.
- This process is repeated until the layout is error-free.

A clean DRC is essential. It enables the design to proceed to the next stage, which is layout versus schematic (LVS) checking.

Q.5.8. Write short note on Electrical Rule Check..... (MAY-24 6 Marks)

Electrical Rule Check (ERC) is a verification step in VLSI design. It is often performed after the physical layout is complete. The main goal of ERC is to ensure the circuit is electrically sound and functional.

ERC focuses on the connectivity and power distribution of the design. It checks for common electrical issues. These issues could cause the chip to malfunction or be damaged. It goes beyond the purely geometric checks of DRC (Design Rule Check).

ERC software examines the circuit's netlist and layout connectivity. The major checks include the following:

- Floating Gates or Nodes: Ensures that all transistor gates or internal nodes are connected to a signal source. A floating node can lead to unpredictable behavior.

- Power and Ground Shorting: Verifies that the power (VDD) and ground (VSS) rails are not accidentally shorted together. This is a critical error.
- Net Continuity: Confirms that a single electrical net is not broken or improperly connected.
- Well or Substrate Connectivity: Checks that N-wells and P-substrates are correctly tied to the appropriate power rails (VDD or VSS).

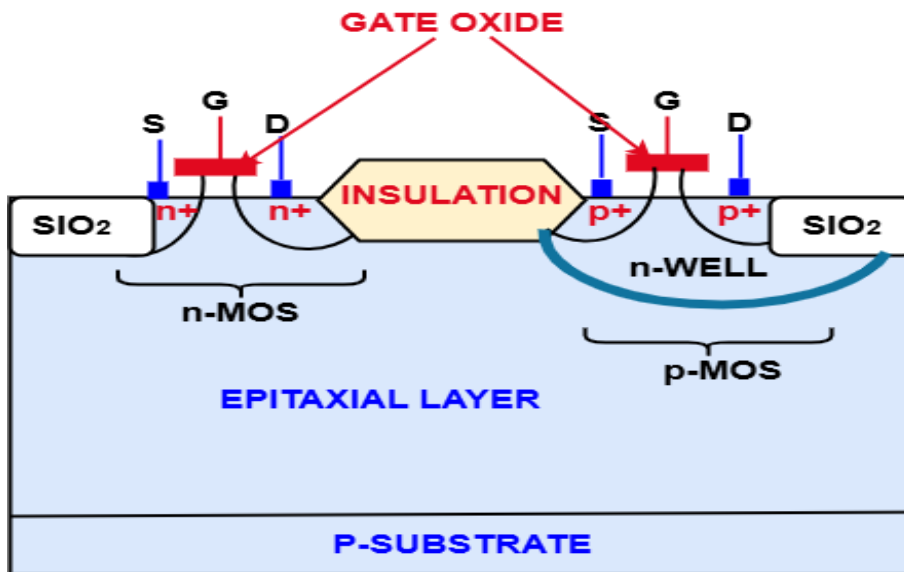
The ERC Process is as follows:

- The layout is extracted to create a netlist. This netlist shows all electrical connections.
- The ERC tool analyzes this netlist. It applies a set of electrical rules.
- The tool looks for violations of these rules.
- It generates a report of all ERC violations.
- The designer must review and fix all errors. This ensures the circuit will function reliably when manufactured.

Q.5.9 Explain the steps involved in fabrication of CMOS transistor with suitable diagram.

(DEC-22, 12 Marks)

In the case of CMOS, both n-MOS and p-MOS exist. The cross-sectional view is illustrated in the figure below.



The n-MOS region is shown in the diagram. Here, n⁺ regions are used. The Source, Gate, and Drain terminals are taken out of the N MOS device. Some isolation is provided, and after that, there is a p-MOS device. Two SiO₂ layers are used.

The substrate requirement for both n-MOS and p-MOS is different. To fabricate an n-MOS device, a P-type substrate is used, and to fabricate a p-MOS device, an n-type substrate is used. So, if you want to obtain both devices on the same chip, then a different technology is used. For this, WELL technology is used to implement the p-MOS device.

The n-MOS is implemented directly on the P substrate. Whereas p-MOS is implemented using a specially created n-WELL region. The isolation is provided using a silicon dioxide layer. There are different process flows that depend on the type of WELL structure used for the fabrication. So, it can be n-WELL process flow, p-WELL process flow, or a dual WELL process flow in which both n and p-WELL techniques are used.

The different steps are as follows:

- Define the active areas.



- Etch and fill trenches. The trenches mean a certain path (a hole) is created.



- Implant WELL regions (n-WELL or p-WELL) depending on the requirements.



- Deposit polysilicon layer for the gate terminal.



- Implant Source and Drain regions.



- Deposit the contacts for the source, drain region, and substrate terminal.

Q.5.10 Explain cell design specifications..... (DEC-24, 6 Marks)

Cell design specifications are the rules for the basic building blocks (cells). These specifications ensure the circuit works correctly and efficiently. These specifications are listed in the following table.

Specification	Description
Physical Dimensions	Defines the height and width of the cell. This is important for placing and routing cells on the chip..
Pin Locations	Specifies the exact locations of the input and output connections (pins). This ensures that wires (interconnects) can easily connect to the cell.
Functional Behavior	Describes the logical operation the cell performs. For example, an 'AND' gate or a 'Flip-Flop'. This is usually defined in a truth table or hardware description language (HDL).
Timing Characteristics	Details how fast the cell operates. It includes propagation delay (the time it takes for a signal to pass through) and setup or hold times for memory elements.
Power Consumption	Specifies the amount of power the cell uses. This includes static power (leakage) and dynamic power (while switching).
Driving Strength (Sizing)	Defines the cell's ability to drive a load or connect to other cells. Stronger cells can drive more connections or longer wires without excessive delay.
Area	The amount of silicon space the cell occupies. A smaller area generally means a lower manufacturing cost.
Layout Rules (DRC)	Ensures the cell's physical layout strictly follows the Design Rule Check (DRC) for the specific fabrication process. This prevents manufacturing errors.

Q.5.11 What is stick diagram? Draw stick diagram of CMOS Inverter..... (MAY-23, 6 Marks)

Every time, it is not possible to draw a complicated diagram. So, a stick diagram is the simplest form of diagram to represent the actual circuit. The stick diagrams are used to plan the layout of transistor cells. It makes use of sticks or lines for the representation of devices and conductors. The different color codes are used for drawing the stick diagram. The color codes for different layers are shown in the following table.

Layer	Color Code
1)N- diffusion	Green
2)P- diffusion	Yellow
3)Polysilicon	Red
4)Metal 1	Blue
5)Metal2	Purple
6) V_{DD} or V_{sat} contacts	Black
7)Via	Black
8)Implant	Yellow

Steps:

- Write the Boolean equation. It must be in the complementary form.
- From the Boolean equation, draw the CMOS diagram.
- From the diagram, draw the corresponding stick diagram.

CMOS Inverter:

An inverter is a circuit in which the output is the complement of the input. There are two parts in the circuit: The upper part is PUN, and the lower part is PDN. PUN is designed using p-MOS, and PDN is designed using n-MOS. The corresponding diagram is shown in the following figure. Here V_{in} is the combined input applied to both p-MOS and n-MOS. The output is the inversion of the input.

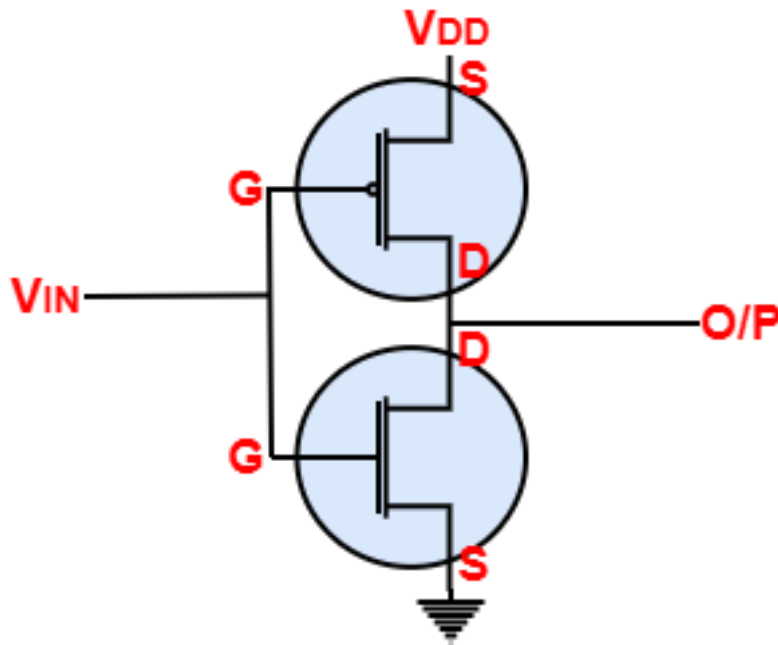


FIG. A

From the circuit (FIG.A), draw the stick diagram. First, draw two horizontal lines. The upper horizontal line is for V_{DD} , and the lower horizontal line is the ground (GND). Now, we have two MOSFETs, p-MOS and n-MOS. For the p-MOS, draw the p-diffusion layer below V_{DD} . Similarly, for the n-MOS, draw an n-diffusion layer above the GND line.

There is one input denoted by V_{in} and one output denoted by V_o . Since there is only one input, draw one polysilicon layer vertically. The source, gate, and drain terminals of both MOSFETs. Observe the connections in fig.A. The drain terminal of p-MOS and n-MOS are connected, and from this common connection, the output is taken.

Whenever the polysilicon layer intersects the diffusion layer, either the p or n diffusion layer, a transistor is formed. So, at the intersections, S (source) and D (drain) points are marked in Figure B. We may interchange the positions of S and D. The drain terminals of p-MOS and n-MOS are connected as shown in Figure A. The common connection is taken,

The source of p-MOS is connected to V_{DD} . So, mark the cross and connect the line to V_{DD} . The source terminal of n-MOS is connected to the ground. Mark the cross and make the connection as shown in Figure B.

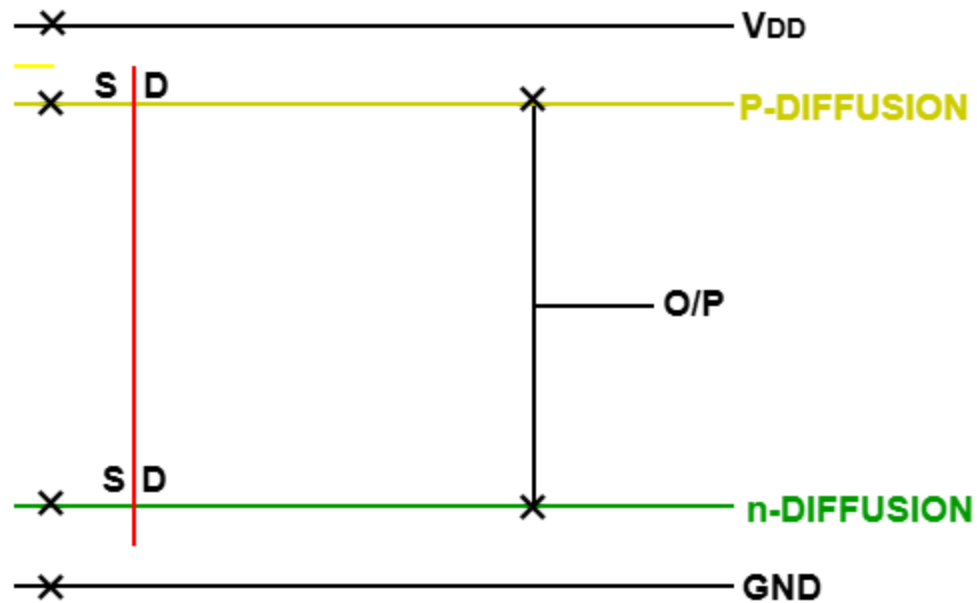


FIG. B

Q.5.12 Draw stick diagram of CMOS inverter, 2 input NAND and NOR gates.
 (DEC-23, MAY-24 8 Marks)

CMOS Inverter:

Refer to Q. 5.10

NAND Gate:

The Boolean expression for the NAND gate is $Y = \overline{A \cdot B}$. We have to use PUN and PDN. The dot (.) between A and B indicates a parallel connection, and for PUN, we have to use p-MOSFET. So, for PUN, we will draw two p-MOSFETs in parallel.

The lower part will be PDN. PDN is the inverse of PUN. For PDN, instead of a parallel connection, use a series connection. Similarly, for PDN, use n-MOSFET. The two-input NAND gate is drawn by combining PUN and PDN, as shown in the following figure A.

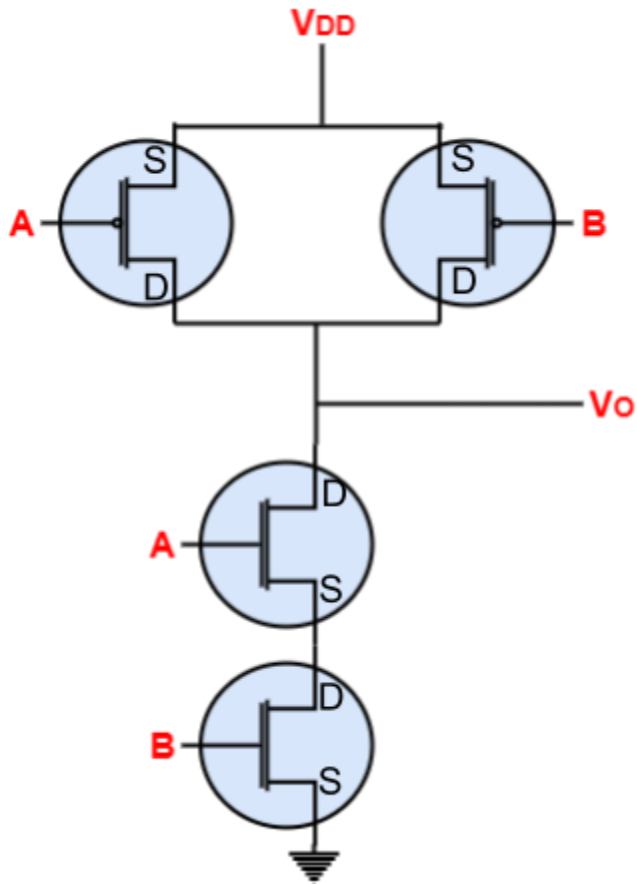


FIG. A

Stick Diagram:

There are two inputs, A and B. So, we need to draw two polysilicon layers. Whenever the polysilicon layer crosses the diffusion line, the transistor is formed. At the upper side, two p-MOS transistors are formed, and at the lower side, two n-MOS transistors are formed. All connections are drawn by referring to Figure A. The complete stick diagram is shown in Figure B.

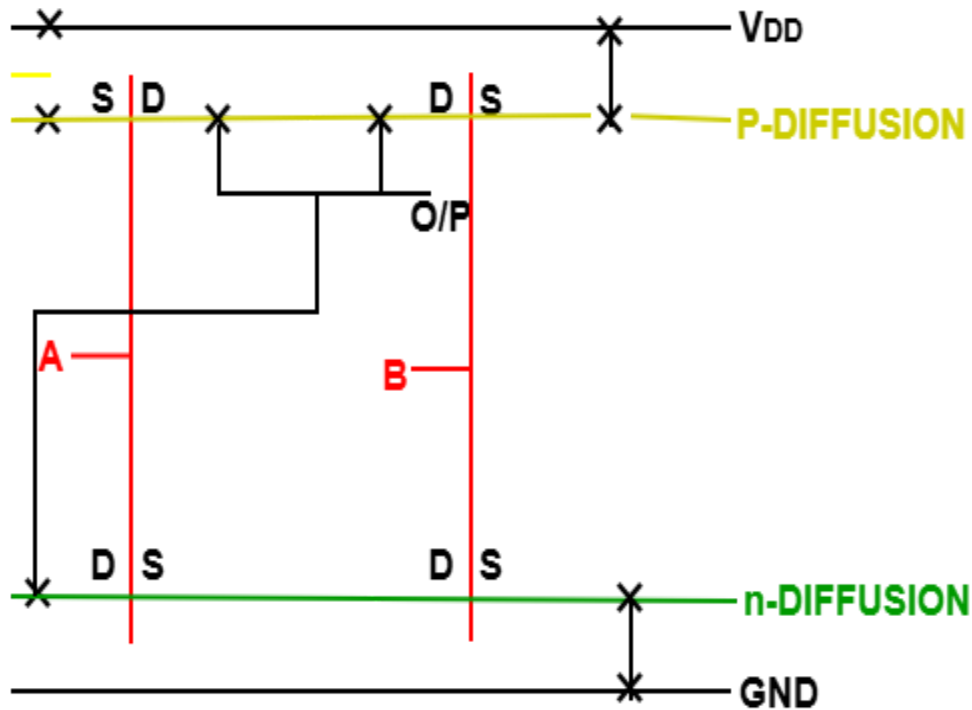
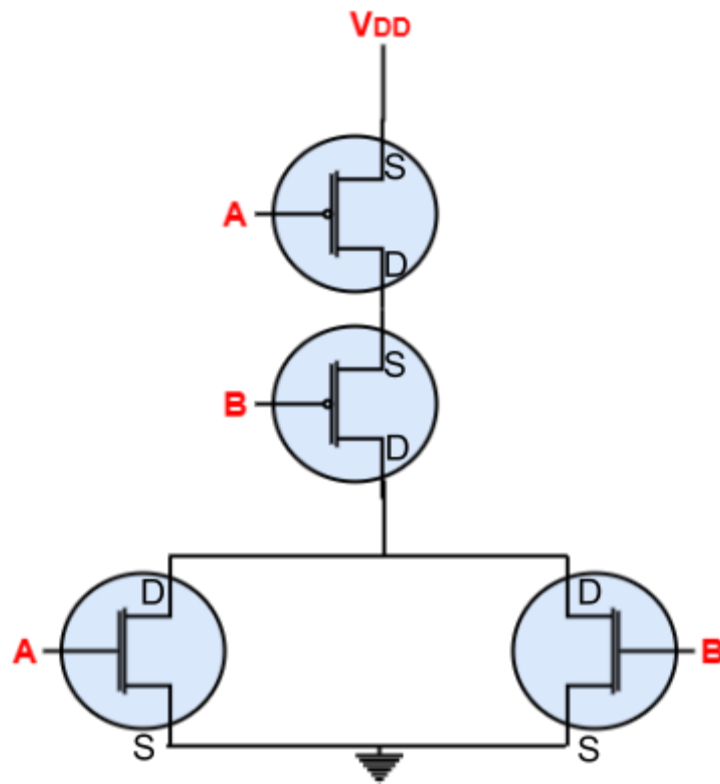


FIG. B

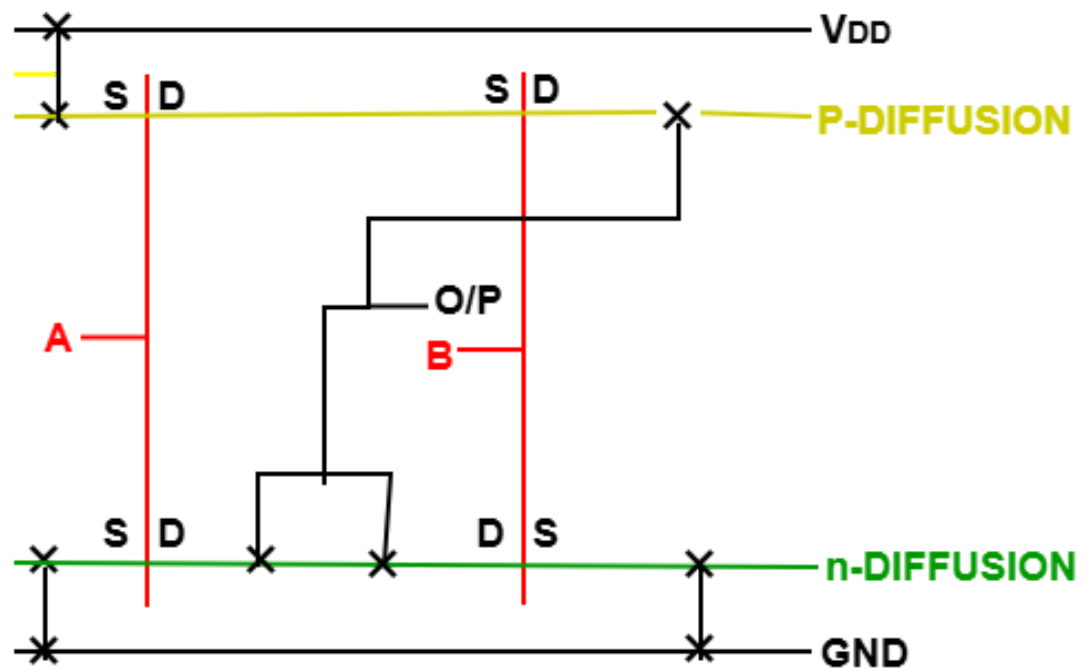
NOR Gate:

The Boolean expression for the NOR gate is $Y = \overline{A + B}$. We have to use PUN and PDN. The plus sign (+) between A and B indicates a series connection, and for PUN, we have to use a p-MOSFET. So, for PUN, we will draw two p-MOSFETs in series.

The lower part will be PDN. PDN is the inverse of PUN. For PDN, instead of a series connection, use a parallel connection. Similarly, for PDN, use n-MOSFET. The two-input NOR gate is drawn by combining PUN and PDN, as shown in the following figure.



The stick diagram is shown in the following figure.

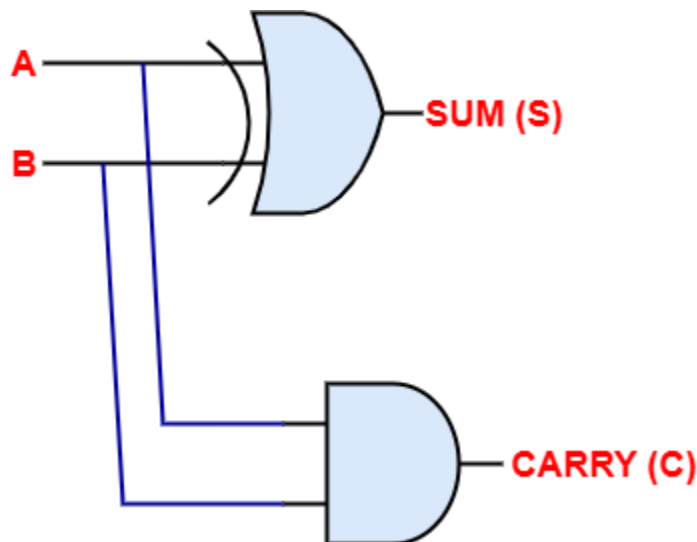


Q.5.13 Draw stick diagram for COMS Boolean equation: $Y = \overline{A + B}$ (DEC-24, 6 Marks)

Refer to Q. 5.11 (NOR Gate)

Q.5.14 Draw the block diagram of half adder and its truth table. Draw it CMOS transistor level circuit. Draw Euler's Path for both networks. Draw the stick diagram for sum and Carry (outputs) of HALF ADDER.(DEC-22, 18 Marks)

The diagram of a half adder is shown in the following figure. It has two inputs, A and B. It adds two single-bit binary numbers and outputs a sum (S) and a carry (C).



The truth table is as follows:

Input A	Input B	Sum (S)	Carry (C)
0	0	0	0
0	1	1	0
1	0	1	0
1	1	0	1

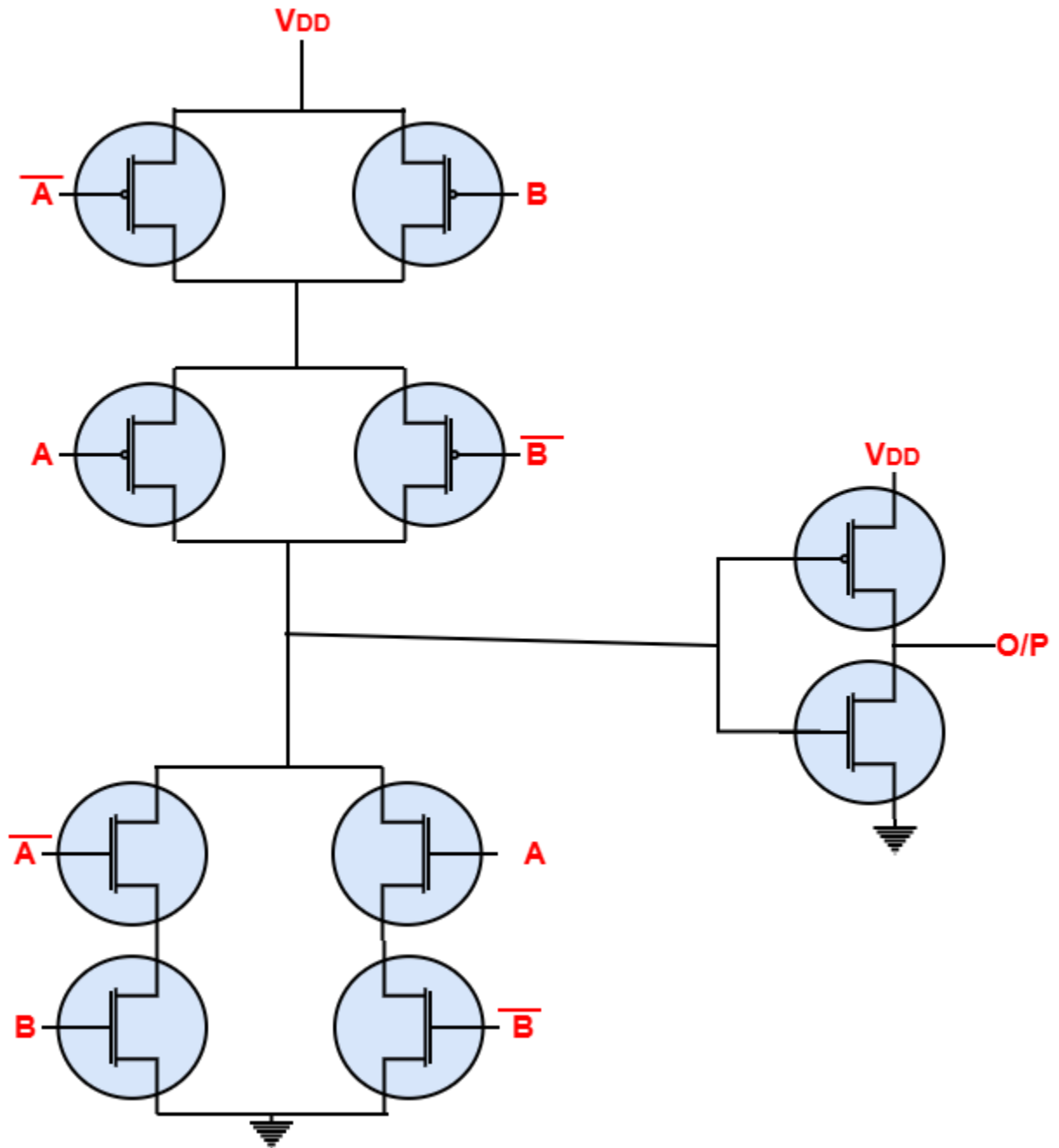
CMOS transistor-level circuit:

SUM: The expression is $S = \bar{A} B + A \bar{B}$

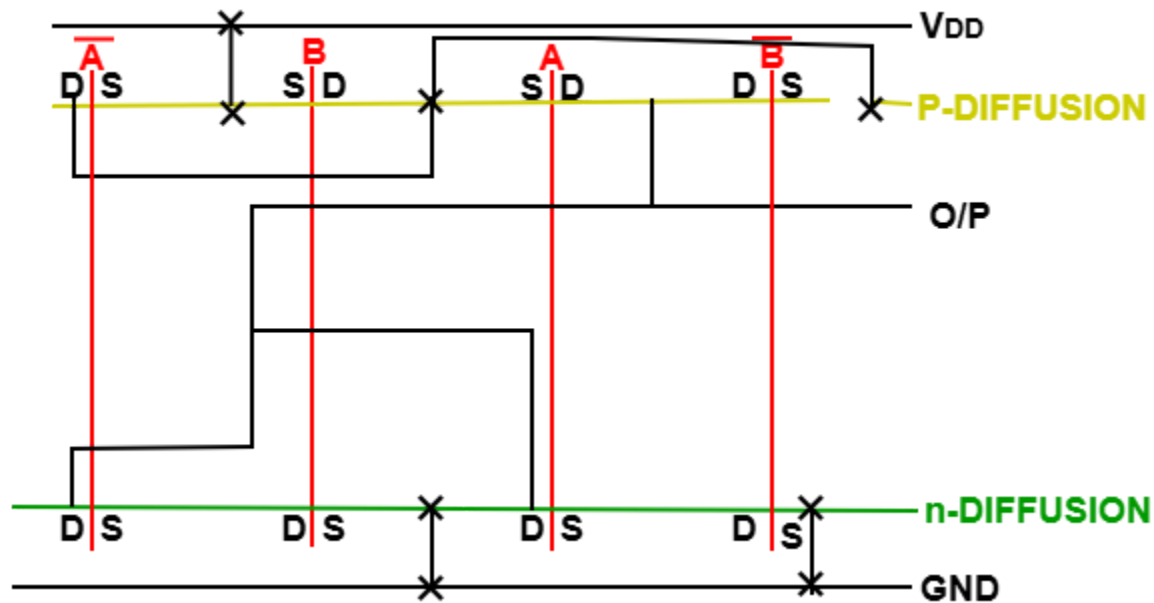
p-MOS $\Rightarrow (\bar{A} \text{ parallel } B) \text{ series } (A \text{ parallel } \bar{B})$

n-MOS $\Rightarrow (\bar{A} \text{ series } B) \text{ parallel } (A \text{ series } \bar{B})$

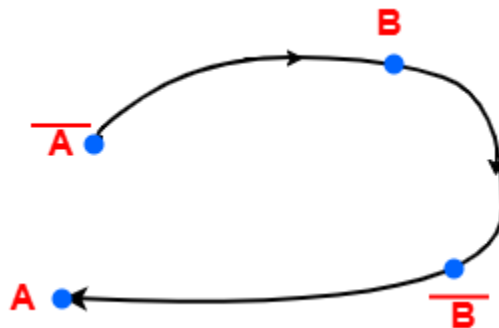
The CMOS transistor logic is shown in the following figure.



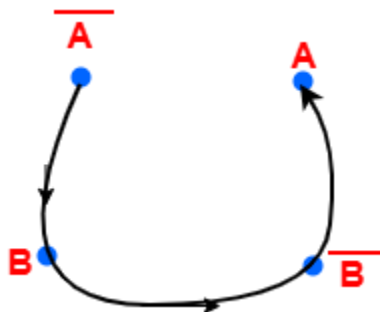
The corresponding stick diagram is shown in the following figure.



Euler's Path: To draw Euler's path, trace the inputs in the CMOS logic diagram. For PUN, trace the inputs in the clockwise direction. So, Euler's path for PUN is shown in the following figure.



For PDN, trace the inputs in the anticlockwise direction. So, Euler's path for PDN is shown in the following figure.

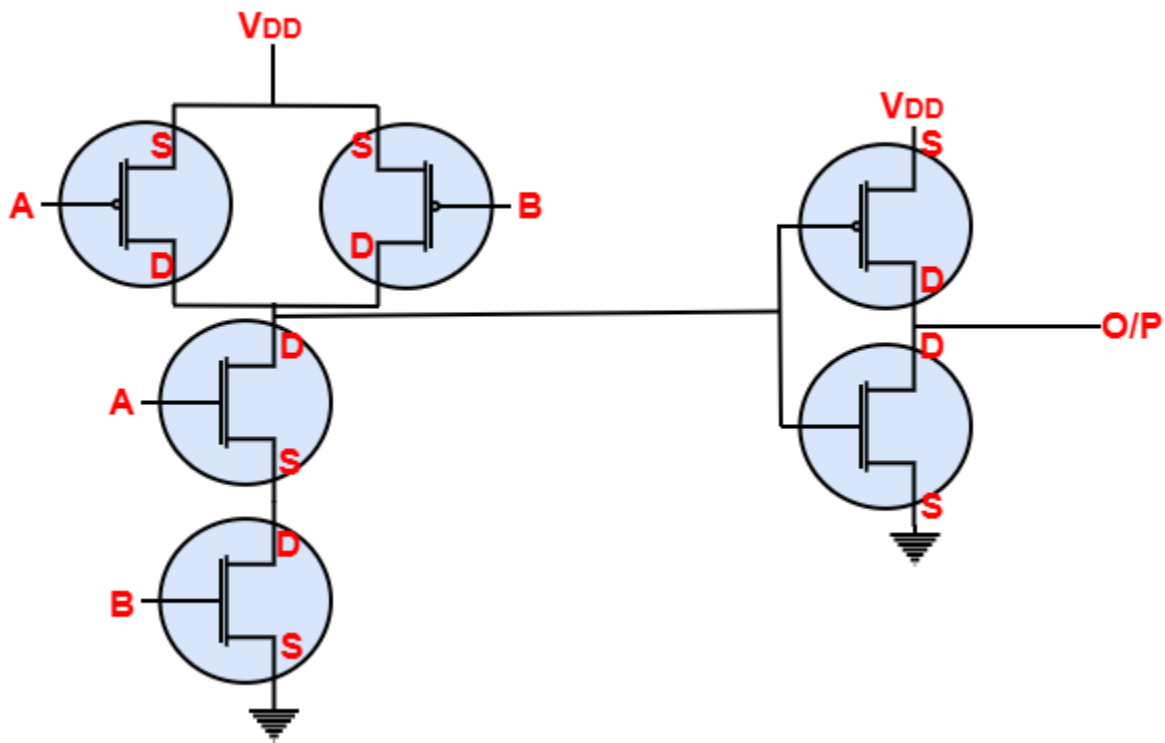


CARRY: The expression is $C = A.B$

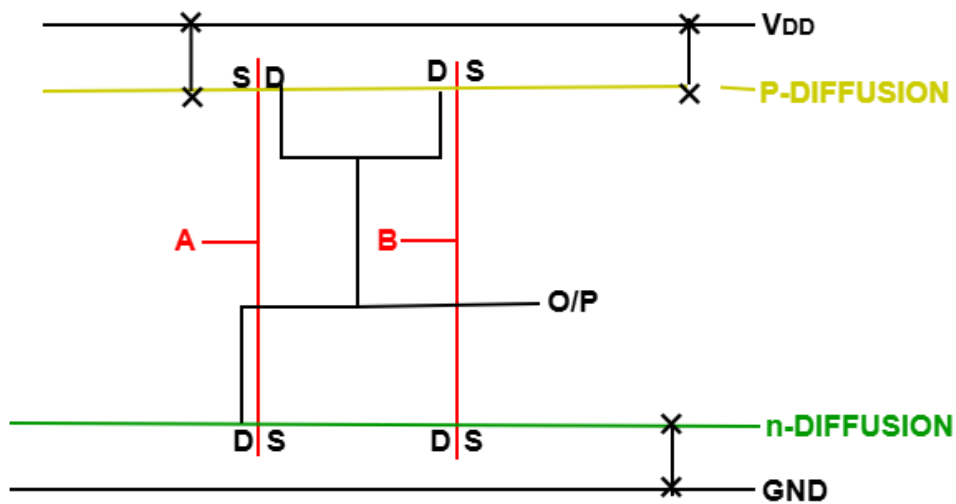
p-MOS $\Rightarrow$ (A parallel B)

n-MOS $\Rightarrow$ (A series B)

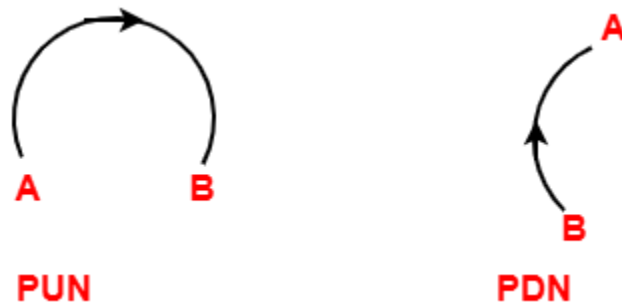
The CMOS transistor logic is shown in the following figure.



The corresponding stick diagram is shown in the following figure.



Euler's Path: To draw Euler's path, trace the inputs in the CMOS logic diagram. For PUN, trace the inputs in the clockwise direction. For PDN, trace the inputs in the anticlockwise direction. So, Euler's path for PUN and PDN is shown in the following figure.



Q.5.15 Write short note on : Antenna Effect..... (MAY-23, DEC-23, MAY-24, MAY-25 6 Marks)

In the case of the fabrication of CMOS devices, plasma processing is used. So, during the plasma processing, the wires that are connected to the gate terminal of the MOSFET will accumulate charges. Such wires act as an antenna, which is called the antenna effect. Another name for this antenna effect is plasma-induced gate oxide damage. Because this effect is related to the plasma, which is getting induced.

There is an accumulation of charges on the conductor, which is connected to the gate terminal of the MOSFET. So, the interconnected wire to the gate terminal acts as an antenna. There is a diode that is formed between the Drain and Source diffusion layers. This diode starts conducting. This particular effect is called the antenna effect.

Effect on the total structure:

- The threshold level gets shifted.
- There is a degradation in the mobility of charges.
- In the worst-case scenario, there is the GATE failure.
- It will degrade the current-voltage (I-V) characteristics of the device.

Steps to reduce antenna effects:

To reduce the antenna effect, the following steps are required.

- Plasma processing is used while fabricating the CMOS device. This plasma processing is used for etching and the deposition process. Reduce the built-up charges by managing or optimizing the plasma processing steps.

- Connect a reverse-biased diode at the gate terminal. So that the accumulated charges will pass through the reverse-biased diode. This reverse-biased diode will not affect the normal performance of the device.
- Use jumpers instead of using the long wires for interconnection. So that the total length of the wire gets reduced.

Q.5.16 Write short note on : Electro Migration..... (MAY-23, 6 Marks)
OR

Explain Electro migration effect in detail. (MAY-23, 6 Marks)

Electromigration is essentially the movement of atoms within a material, like a metal wire or layer. It is caused by the force exerted by a flowing electric current. The major cause is a high current density. When a high amount of current flows through a small wire, the current density is high. This high density pushes the atoms.

A high current density forces electrons to move rapidly and collide frequently with the metal atoms. These collisions generate heat. This generated heat dramatically increases the vibration and energy of the metal atoms, making them easier to push. The constant bombardment from the high-speed electrons then gradually forces the metal atoms to move away from their original positions.

Effects of Electromigration

- **Voids and Breaks (Open Circuits):** As atoms are pushed away, they leave behind empty spaces, called voids. If enough voids cluster together, they can create a break in the wire. This is an open circuit.
- **Piles and Short Circuits:** The displaced atoms accumulate somewhere else, forming piles. If it hillock grows large enough, it can bridge the gap between two nearby wires, creating a short circuit.

Factors Responsible for Electromigration

- 1) **Temperature:** Higher temperatures significantly accelerate electromigration. Heat makes it much easier for atoms to move, so a small increase in temperature can cause a large increase in the effect.
- 2) **Current and Current Density:** A large current results in a stronger pushing force on the atoms, leading to faster migration.
- 3) **Crystal Lattice Structure:** The internal arrangement of atoms in the metal affects how easily atoms can move.

Steps to Reduce Electromigration

- Increase the Metal Width

- Reduce the Power Supply
- Reduce the Supply Frequency
- Use Short Wires
- Use Better Materials

Q.5.17 Write short note on : Crosstalk. (MAY-23, MAY-24, DEC-24, MAY-25 5 Marks)

Crosstalk is a major design issue in high-density Very Large Scale Integration (VLSI) circuits. It is an unwanted signal coupling between adjacent wires or interconnects.

Crosstalk happens when an electric signal on one wire induces noise onto a neighboring wire. This signal coupling occurs due to parasitic capacitance and mutual inductance between the closely placed interconnect lines.

Effects of Crosstalk:

Crosstalk introduces the following major problems:

- Noise: The induced voltage spike can be interpreted as an incorrect logic signal, causing functional failure or data corruption.
- Timing Skew: It can alter the delay of a signal, either speeding it up or slowing it down. This affects the overall circuit timing and can lead to setup and hold time violations.
- Increased Power Consumption: The extra switching activity caused by the coupled noise can slightly increase power use.

Steps to reduce crosstalk:

- Increasing the distance between parallel wires (interconnects).
- Place the ground or power lines (called shield lines) between sensitive signals. This acts as a barrier.
- Arrange the wires to place less active or non-critical signals next to highly switching or critical signals.
- Add buffers to restore signal integrity and drive strength.

Q.5.18 Write short note on : Drain Punch. (MAY-24 6 Marks)

Drain Punch-Through is a significant design issue that affects Metal-Oxide-Semiconductor Field-Effect Transistors (MOSFETs). It is an unwanted electrical connection between the transistor's source and drain. This connection bypasses the gate's control.

It occurs when the depletion regions surrounding the drain and source junctions of a MOSFET merge. This effect is severe in modern ASICs due to short channel lengths.

When punch-through happens, a current path is created deep in the bulk silicon. This path allows a large leakage current to flow, even when the gate voltage is set to turn the transistor OFF.

Effects of Punch-through:

- It dramatically increases the OFF-state current of the transistor. This leads to higher static power consumption and reduced battery life in portable devices.
- The transistor loses its ability to turn off properly. The gate can no longer fully control the current flow.
- In extreme cases, the uncontrolled current can cause a permanent electrical breakdown of the device. This is a complete circuit failure.
- It limits the highest voltage that can be applied to the transistor.

Steps to reduce Punch-through:

- Increase the doping concentration in the substrate or channel. Higher doping reduces the width of the depletion regions.
- Use localized doping profiles like Halo or Pocket implants. These implants create a barrier that prevents the drain's depletion region from extending to the source.
- Ensure all device dimensions and voltages are scaled correctly.

UNIT- 6

VLSI Testing and Analysis

Q.6.1 Explain different types of faults existing in VLSI chips. (DEC-22, 8 Marks)

OR

What are the types of fault? Explain each in brief. (DEC-24, 7 Marks)

A fault means a certain error is generated at the output. This error may be developed during the manufacturing of the circuit or the processing of the circuit.

There are three main types of fault models.

1. Stuck-At Faults (Logical)

This is the most common and simplest fault model. It assumes a wire is permanently fixed at a single logic value.

- **Stuck-at-0 (s-a-0):** The signal line is permanently fixed at a logic '0' (Ground). It cannot change to '1'.
- **Stuck-at-1 (s-a-1):** The signal line is permanently fixed at a logic '1' (Supply Voltage). It cannot change to '0'.

The part of the circuit controlled by this line stops working correctly.

2. Bridging Faults (Shorts)

This fault happens when two signal lines that should be separate are accidentally connected (short-circuited). A physical short, often due to excess metal or contamination during fabrication. The two lines interact. The resulting logic value is often a wired-AND (both must be '1') or a wired-OR (either can be '1') of the two signals. It causes incorrect logic and can even lead to the circuit oscillating.

3. Delay Faults (Timing)

These faults mean the chip's logic is correct, but the signal is **too slow**. The output does not arrive within the required time window.

- **Transition Fault:** A signal fails to switch from '0' to '1' (slow-to-rise) or from '1' to '0' (slow-to-fall) within the time limit. This models a localized, slow spot.
- **Path Delay Fault:** The total delay along a specific signal path from input to output is longer than allowed. This models a cumulative delay over many gates.

The circuit fails at its maximum operating frequency, leading to timing violations.

Other Transistor-Level Faults

These faults describe issues inside the transistors.

- **Stuck-Open Fault:** A transistor is permanently **off (open)**, regardless of the voltage on its gate. This can turn a logic gate into a sequential circuit, which is difficult to test.
- **Stuck-Short Fault:** A transistor is permanently **on (shorted)**, regardless of the voltage on its gate. This often causes a direct path between the power supply and ground, resulting in high current draw.

Q.6.2 Explain Stuck at Fault models in brief. (MAY-23, 9 Marks)
OR
 Explain stuck at fault models with suitable examples. (DEC-23, 9 Marks)
OR
 What is the need for testability? Explain stuck at 0 and stuck at 1 fault with example.
 (MAY-24, 9 Marks)

Need for testability:

The need for testability is essential to ensure a product is functional, reliable, and cost-effective.

1. Detect Manufacturing Defects:

Physical flaws, like a broken wire or a short circuit (known as faults), often happen during the chip fabrication process. Testability is useful to easily check every part of the chip to find these mistakes.

2. Manage High Complexity

Modern chips have billions of transistors. It is impossible to check all possible operations simply by testing the input and output pins. Testability adds special features to the design to check internal wires and nodes.

3. Reduce Costs and Time-to-Market

Testing is a major part of the chip development cost. A difficult-to-test design leads to longer test times and more expensive external testing equipment. With built-in features, testability makes the testing phase much faster and cheaper.

4. Ensure Quality and Reliability

Good testability results in high fault coverage (detecting a high percentage of possible faults). This ensures the final product is reliable.

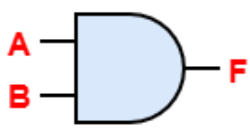
Stack at Fault models:

Stuck at fault means, any input is stuck at a particular level. There are two subcategories:

- 1) Stuck at zero.
- 2) Stuck at one.

1) Stuck at zero:

A particular node in the circuit permanently remains at logic zero. That means permanently stuck at logic zero. Consider an example of an AND gate. The symbol and truth table of an AND gate are as follows.

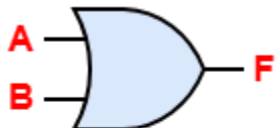


A	B	IDEAL	FAULT
0	0	0	0
0	1	0	0
1	0	0	0
1	1	1	0

We are considering two input AND gate. So, there are two inputs: A and B. It has one output F. For the AND gate, the output is high if both A and B are high. Suppose the input A is remaining zero, is stuck at zero due to a certain error, then in that case, irrespective of the value of B, the output is always zero. Because in the case of an AND gate, if any one of the inputs is zero, then the output is zero. It represents being stuck at zero.

2) Stuck at one:

The node is permanently stuck at logic one. Consider an example of an OR gate. The symbol and truth table of the OR gate are as follows:



A	B	IDEAL	FAULT
0	0	0	1
0	1	1	1
1	0	1	1
1	1	1	1

Two inputs are there: A and B. The output is denoted by F. In the case of an Ideal OR gate, if either of the inputs A or B is high, the output is high. Consider that one of the inputs, let us say input A, is stuck at logic one. That means its value permanently remains at one. Then, irrespective of the value of B, every time, the output will always be one. This is an error called stuck at one fault.

Uses:

1. **Simplification:** Physical defects (like shorts, opens, or broken transistors) are very complex. The Stuck-at Fault Model simplifies these defects into a manageable, reproducible logical error.
2. **Test Generation:** It allows engineers to create test vectors (input patterns) that can easily check if a specific line is stuck at 0 or 1.
3. **Fault Coverage:** Testing all possible physical defects is impossible. By using the Stuck-at Fault Model, engineers can measure the fault coverage, the percentage of all possible stuck-at faults that a test set can detect.

Working in Testing:

- A list is created of all possible stuck-at-0 and stuck-at-1 faults for every internal line in the circuit.
- For a specific fault, a test vector is generated. This vector must do two things:
 - 1) **Activation:** Set the faulty line 'X' to the opposite value (i.e., **1**) if the circuit were working correctly.
 - 2) **Propagation:** Create a path from the faulty line 'X' to one of the circuit's output pins. This ensures the incorrect value (the stuck-at **0**) reaches an observable output.

- If the chip under test is faulty, the output will be different from the expected (fault-free) output. This indicates a stuck-at fault is present.

Q.6.3 Explain the need of Design for Testability. (MAY-23, 9 Marks)
OR
 What is DFT? Explain the need for DFT in brief. (MAY-25, 7 Marks)

Design for Testability (DFT) is one of the simplified methods to test the ICs after manufacturing.

Definition: It is the technique to ensure that the IC is effectively tested after manufacturing, verify IC functionality, and identify the defects.

Need for DFT:

- If the circuit is complex, then it contains even millions of transistors. So it is difficult and time-consuming to test every IC individually. DFT is automatic and optimization techniques, so the required time for the testing is reduced.
- Testing of each component in a complex circuit is difficult because there are n components. DFT makes use of certain automated techniques, so the procedure of testing each component in the complex circuit becomes easy.
- The DFT technique detects the common fault very easily. The common faults include being stuck at zero or being stuck at one. The other common faults, like bridging faults, timing faults, can be easily detected using this method.
- It improves the test coverage. The test coverage is the percentage of faults that can be detected during the testing process.
- Certain regions are called hard-to-test regions if a traditional method is used. For example, you want to insert the probe, then certain regions on the PCB are hard to test using normal methods. But since this technique makes use of certain automated tools, hard-to-test regions can be easily accessible.
- It improves the yield. The meaning of yield is the percentage of properly functioning ICs compared to the total number of ICs that you are testing.
- DFT improves the reliability because we are getting the idea of the fault before proceeding with the particular complex circuit. The Faults are identified earlier in the manufacturing process.
- It includes one of the important techniques, which is Built-In Self-Test (BIST). It automatically tests the circuit, and the fault detection can be easily done. It reduces the time required for debugging the fault.

- DFT allows for very high fault coverage (checking nearly every circuit path).
- If a chip fails, DFT features can instantly inform the engineer of the exact physical location of the defect.

Q.6.4 Write short note: Need of design for Testability & Adhoc DFT techniques. (DEC-23, 8 Marks)

For the need of design for testability, refer to Q. 6.3

Adhoc DFT:

These are informal rules or guidelines used by designers. They are not a standardized method like Scan Design. They often involve simple design changes to quickly fix a testability problem.

Let us say we have one complex circuit that consists of pipeline structures. If you want to test the entire circuit at a time, then it is a time-consuming as well as complex process. In Adhoc DFT, the larger circuit is divided into smaller subparts. At a time, each subpart is tested.

It is required to insert the multiplexer (MUX). By using a multiplexer, it is possible to apply the portion of the input signal to that particular part which you want to test. It is also required to include control signals. So that you can easily access a smaller part of the large circuit. The multiplexer provides the test points.

Guidelines for ADhoc DFT:

- Try to avoid the combinational feedback loop. It is easier to test the sequential circuit rather than the combinational circuit.
- Separate the analog and digital parts of the circuit.
- Avoid redundant logic circuits. The redundant means unnecessarily extra logical circuits might be used.
- Avoid the asynchronous logic circuits.
- Split larger counters into smaller ones. If there are larger counters, you need to partition them into smaller counters.
- Reset all the flip-flops that are used in the circuit by applying the reset or clear signals.

Limitations:

- Ad-hoc methods do not guarantee high test coverage.
- They are hard to automate with software tools.
- They are less reliable for very large and complex chips compared to modern, structured DFT methods.

Q.6.5 Explain Controllability, Observability and Predictability in Testability.(DEC-22, 9 Marks)

The testability is the ability that indicates how easy it is to test the given logical circuit.

1) Controllability:

It is the ability to provide a specific signal logic level (either logic zero or logic one) at each node in the given circuit. This can be done by setting the primary input values. The term Low Controllability means a signal deep inside the chip is hard to reach and change. This makes it difficult to activate a fault. The term High Controllability means that a specific state of the circuit can be reached and tested easily. Example: In a logic gate, if inputs directly affect the output, controllability is high.

It becomes easy to provide desired values for internal signals by applying appropriate test Vector input combinations. That means during this process, you need to choose a proper combination of the input signals. Using an appropriate combination, one can assign the desired signal level at each node of the circuit.

2) Observability:

Observability is the ease of seeing or detecting an internal signal at the system's external outputs. The term observability is related to observing or taking a measurement at a particular stage of the circuit. So it is the ability to observe or measure the logical states of a signal during the testing process of the given circuit.

The term Low Observability means a fault might occur, but its effect gets blocked or hidden before it reaches the output. This makes faults undetectable. Example: A node connected to many logic levels has low observability before reaching the output.

The term high observability is related to the circuit output, which is accurately monitored. If a change at an internal node can quickly appear at the output, observability is high. To check the observability, the circuit should be driven to the given state.

3) Predictability:

Predictability is the ease of determining the expected output for a given set of inputs. Using predictability, the output of a particular stage of a circuit can be predicted. So, to predict the expected behavior under the specific test conditions, the predictability is used. The test vector (a

combination of different signals) applied to the circuit should cover all the critical parts of that circuit, which you want to test. It provides information about how the circuit responds to the different applied inputs.

High Predictability: Means the system's operation is consistent and straightforward. Its future behavior can be calculated precisely. Low Predictability: Happens in circuits that use timing-dependent or analog elements (like clocks or power-up sequences) that are hard to model. This makes test creation very difficult. The predictability can be easily achieved by using extensive pre-silicon simulation and modeling.

Q.6.6 Explain the following terms with respect to testability:

i) Fault Coverage

ii) Fault models

iii) Observability

iv) Contrability

v) Design for Testability.....(DEC-24, 10 Marks)

i) Fault Coverage:

It is a measure of how good the test patterns are at finding manufacturing errors.

- It is the percentage of possible faults that a set of tests can successfully detect.
- A test is considered good if it achieves high fault coverage.
- High coverage means you are more likely to ship a reliable, working chip.
- Low coverage means many faulty chips might slip past testing and fail later on.

ii) Fault Models:

A Fault Model is a simple, imaginary representation of a physical defect that happens during manufacturing. Since physical defects are complex, we use simple models to make testing possible.

- It helps engineers predict how a defect will affect the circuit's logic.
- We use these models to create test patterns that target specific types of errors.
- Example: Stuck-At Fault Model

This is the most common model. It assumes a signal wire is permanently stuck at logic '0' (Stuck-At-0) or stuck at logic '1' (Stuck-At-1). Test patterns are then generated to check if every wire can be set to both '0' and '1'.

iii) Observability:

Refer to Q. 6.5 (2)

iv) Contrability

Refer to Q. 6.5 (1)

v) Design for Testability

Refer to Q. 6.3

Q.6.7 Explain the following terms with respect to testability:

i) Fault Coverage

ii) Fault models

iii) Observability

iv) Contrability

v) Physical Defects.....(DEC-25, 10 Marks)

i) Fault Coverage:

Refer to Q. 6.6 (i)

ii) Fault models:

Refer to Q. 6.6 (ii)

iii) Observability:

Refer to Q. 6.5 (2)

iv) Contrability:

Refer to Q. 6.5 (1)

v) Physical Defects:

A physical defect is any unintended flaw or imperfection in the physical structure of an IC that occurs during the manufacturing process. They are unwanted errors introduced during the highly complex fabrication steps. They are often caused by random events like dust particles, slight misalignment of layers, or chemical contamination. A physical defect is a structural problem on the silicon wafer itself.

Common Examples:

- 1) Open Circuit (Break): A metal connection or wire trace is broken or incomplete. This stops the signal from reaching its destination.
- 2) Short Circuit (Bridge): Two signal lines that should be separate are unintentionally connected.
- 3) Gate Oxide Damage: Tiny damage to the insulating layer in a transistor. This can cause the transistor to leak current or fail.

Q.6.8 What is the scanning method for testing?(6 Marks)

If it is required to test the internal logic of the circuit or the connectivity between different components without making any physical contact, the boundary scanning technique is used. In this method, all the flip-flops are modified into scan cells.

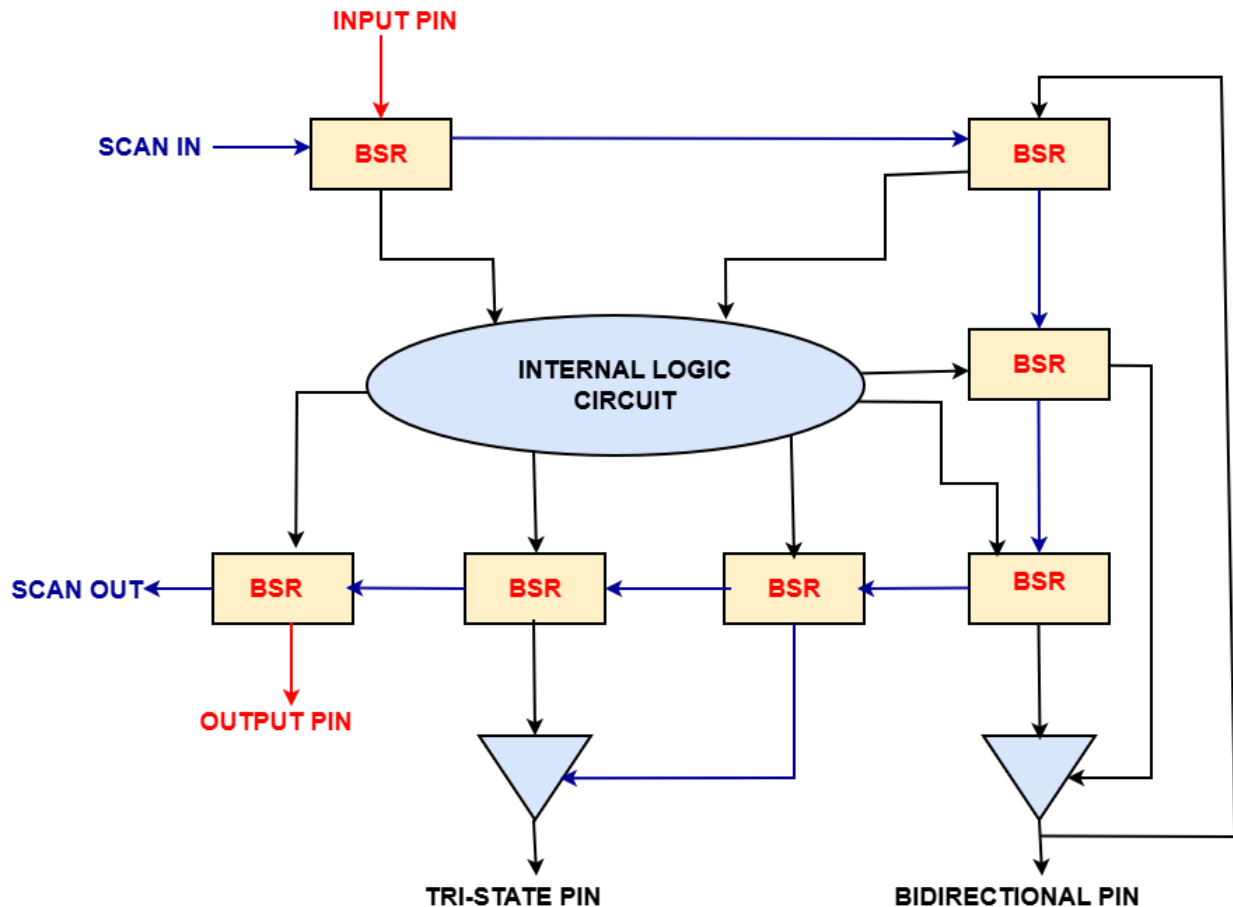
It can be operated in the normal mode as well as in the scan mode. In normal mode, scanning is not performed. In the scan mode, a test pattern is applied, and then the testing is performed. Once the test pattern is loaded, the circuit switches from scan mode to normal mode for one clock cycle.

Once the test data is applied, the circuit behaves like a shift register. This is because different scan cells are combined to form the chain, like a shift register. So data is shifted from one point to another. The test data is applied at the input, and at the output, observations are made. The output signal is available at the scan output pin.

There are two types of scanning. First is the full scan method. In this method, the entire circuit is fully scanned. So, this is the most reliable method. But it is a time-consuming method. The full scan method gives good controllability as well as observability. The second method is the partial scan. Here, only the critical part of the circuit on the PCB is tested.

Q.6.9 Explain JTAG boundary scan. Explain all the pins involved.....(DEC-22, 9 Marks)
OR
Write short note on Boundary Scan Method for testing.(MAY-23, MAY-24 9 Marks)
OR
Explain JTAG boundary scan method for testing.(DEC-23, 9 Marks)
OR

The generalized block diagram of the Boundary scan method is shown in the following figure.



The Boundary Scan Method is a widely used DFT standardized method as a part of JTAG (Joint Test Action Group). In the case of normal circuits, the circuit is complicated. It contains millions of transistors. So, in such a case, it is very difficult to physically access the particular point and then test the output at that point.

The Boundary Scan Method avoids physical access to a particular point. It gives us an effective method to test internal logic. It is used to test interconnections. The interconnections are the connections between different components. It can check whether any particular track on the PCB is broken or disconnected. It is also used to test the internal logic of the particular IC. The input and output pins can be well tested without any physical access.

The major block is BSR (Boundary Scan Register). It contains a group of boundary scan cells. The boundary scan cell is connected to input and output pins, as well as at the different places in the circuit that you want to test. It captures the data from each pin, and after scanning, it produces the data to the output pin. It creates the shift register chain. So, the data is continuously shifted in and out by making use of this boundary cell. The group of such boundaries forms one BSR.

Another important term related to this is Test Access Port (TAP). It is the standard interface. If you want to allow certain external test equipment, then this interface is used. So it is the standard

interface that allows external test equipment to control the boundary scan.

The major pins (Not shown in the generalized diagram) used in this method are as follows:

- **TDI (Test Data Input):**
This pin accepts the input data, which is to be applied to the circuit for testing purposes.
- **TDO (Test Data Output):**
From this pin, the data is taken out.
- **TMS (Test Mode Select):**
There are different modes like normal mode, test mode, bypass mode, etc. These modes are controlled by using the TMS pin.
- **TRST (Test Reset Condition):**
It resets the condition or it clears the scanning.

This method also consists of an Instruction Register. It allows external test equipment to control the different operations. So different modes can be controlled like normal mode, test mode, and bypass mode.

The major applications of this method are as follows:

Interconnect testing: A particular connection can be checked. Similarly, it is used to check whether a particular track on the PCB is disconnected.

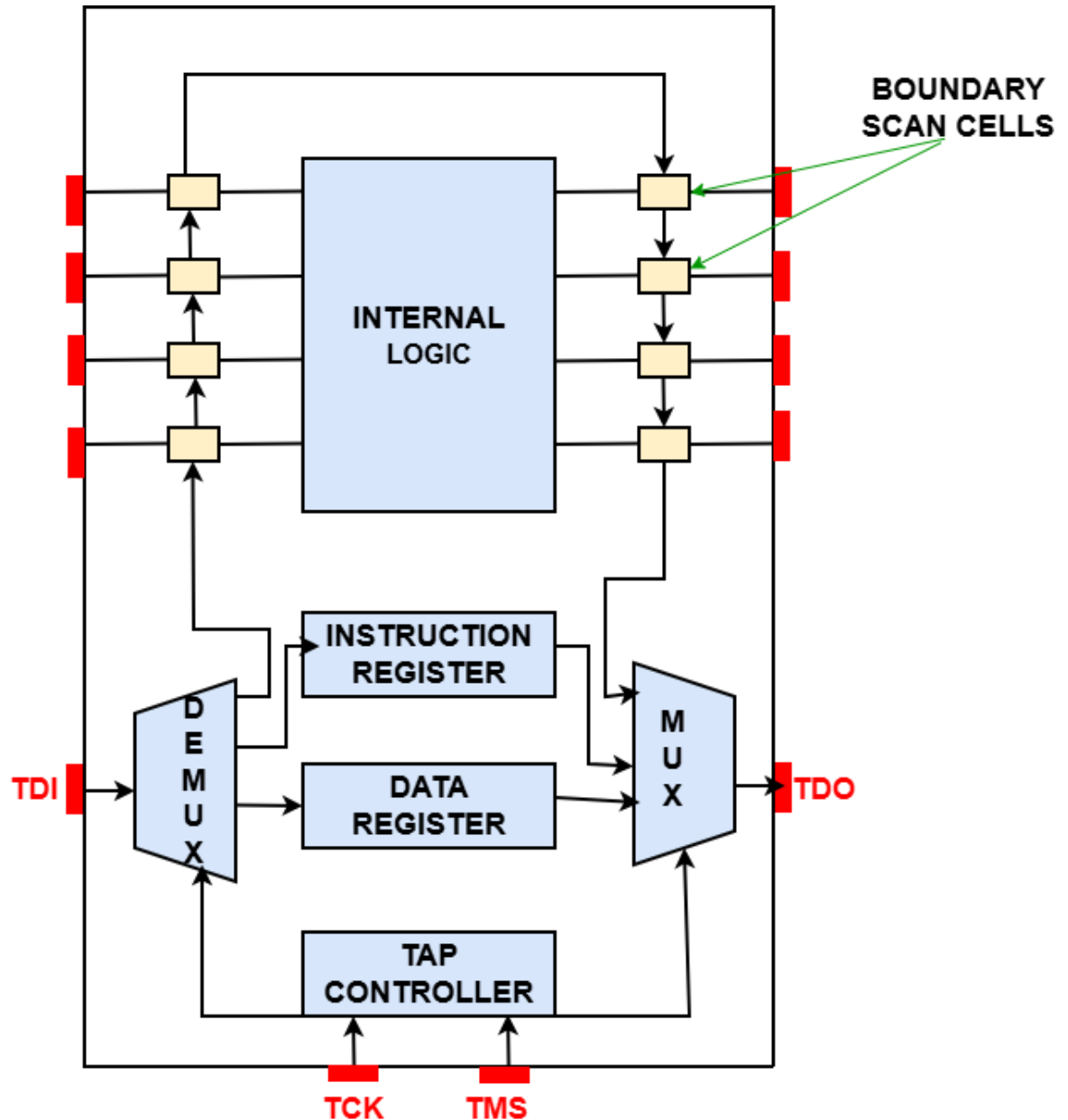
Internal logic testing: An internal logic circuit is used to check the internal logic functioning of a particular IC.

In-system programming: The different components are soldered on the PCB. You can perform the programming, or you can configure these devices.

The function testing: The entire functionality of the circuit can be tested at a stretch using the boundary scanning technique.

Q.6.10 Write a short note on JTAG.....(MAY-24, 4 Marks)

JTAG stands for Joint Test Action Group. The simplified block diagram of the JTAG architecture is shown in the following figure.



The centre block represents internal logic. There are four square boxes at the RHS as well as at the LHS. Each square box represents a boundary scan cell. The number of boundary scan cells is joined, and a chain like a shift register is formed. It creates the boundary scan register BSR. Four physical pins are connected to four boundary scan cells. All the boundary scan cells are connected one after the other.

TDI is the Test Data Input pin. From this end, we have to apply the test data to test the circuit. The output of the demultiplexer is applied to the boundary scan cell, and then this signal goes on sifting one after another. The output of the demultiplexer is also applied to the instruction register and data register. The output of the multiplexer is applied to the TDO (Test Data Output) pin. TCK pin represents the Test Clock Signal, and TMS represents the Test Mode Select pin.

The functions of the major blocks are as follows:

TAP controller:

It is the major block responsible for controlling the entire testing operation. It generates a clock signal and control signals to control the operation of the boundary scanning.

Instruction Register:

Different instructions are required to perform a particular operation. Instruction register is used to generate these instructions. It is like a shift register. It is loaded serially with the different instructions, then, according to that instruction, a particular operation is performed.

Data Register:

The data register has a bank of registers. An external stimulus used to perform different operations are loaded in the data register. The output of the data register is given to the multiplexer, and then it is available at the TDO pin.

Q.6.11 Explain Path sensitization method(DEC-22, 8 Marks)
OR
Explain Path sensitizing algorithm in detail.(MAY-25, 9 Marks)

The Path Sensitizing Algorithm is a technique used in Automatic Test Pattern Generation (ATPG). It generates the test vectors (input patterns) that detect faults in digital circuits. It works on the principle that the effect of an internal fault is visible at an output pin.

The Path Sensitizing Algorithm operates by completing three main steps simultaneously for a specific fault. It works well for the most common stuck-at fault (e.g., a wire permanently stuck at '0' or '1').

1) Fault Activation:

The goal is to set the primary input pins so that the faulty wire is forced to the opposite value of its assumed fault. For example, if a wire 'A' is assumed to be Stuck-At-0, the input signals must be set to drive the wire 'A' to a logic '1'. If there is no fault, wire 'A' will be '1'. If it is faulty, wire 'A' will be '0'. This creates the initial difference. It is called the fault effect.

2) Fault Propagation:

The aim is to create a sensitized path from the faulty wire to one of the circuit's primary output pins. This path must ensure the difference created in Step 1 is carried through. To propagate the fault effect through a logic gate (like an AND or OR gate), the other inputs to that gate (called side inputs) must be set to non-controlling values.

Example: To pass a fault effect through a 2-input AND gate, the other input must be set to a '1' (the non-controlling value). If the side input were '0' (the controlling value), the output would be '0' regardless of the fault effect. This process continues gate-by-gate until the fault effect reaches a primary output.

3. Line Justification:

The aim is to determine the values needed at the primary input pins to satisfy all the internal signal requirements established in Steps 1 and 2. This is a back-propagation process. It works backward from the internal gate inputs to the primary inputs.

- Requirement Check: All gate inputs must be justified back to inputs that the tester can control.
- Conflict: If there is a conflict (e.g., one line requires a '0', but another indicates it must be a '1'), the entire test pattern attempt fails. The algorithm must then backtrack and try a different sensitized path or a different input assignment.

If all three steps succeed without conflict, the final primary input values form the successful test vector for the specific fault.

Q.6.12 Write a short note on Built-in Self-Test. (MAY-23, 8 Marks)
OR
Write a short note on BIST with a block diagram. (DEC-23, 8 Marks)
OR
What is the need of BIST? Explain typical BIST in detail. (MAY-24, DEC-24, MAY-25 9 Marks)

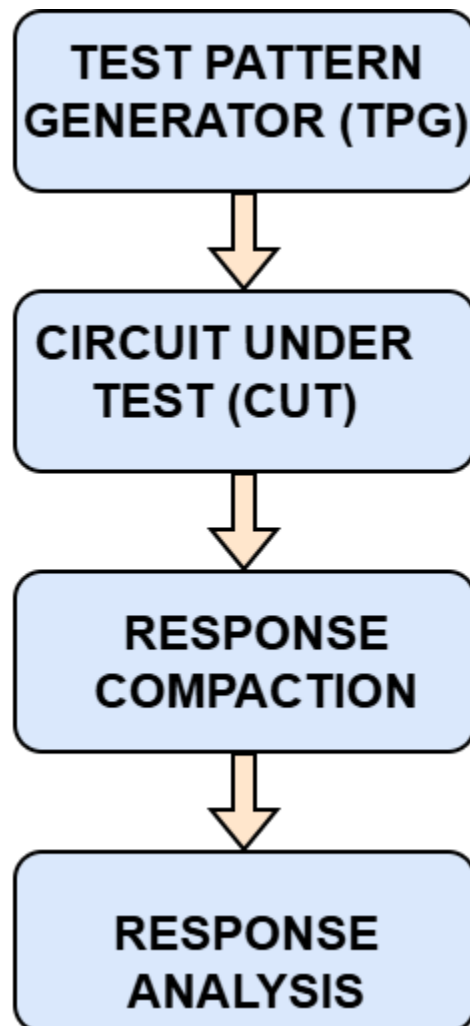
BIST stands for Built-in Self-Test. In the case of VLSI circuits, many ICs are capable of self-testing. That means they can test the functioning by themselves. So, it is not required to use any external circuitry or equipment.

Need for BIST:

1. Reduces Test Cost: BIST moves the testing work from expensive external test equipment (ATE) to the chip itself. This lowers the overall cost.
2. Handles High Speeds: Modern chips operate at very high speeds. External testers often can't run tests fast enough, but BIST operates at the chip's full functional speed.
3. Addresses Complexity: As chips get billions of transistors, the volume of test data becomes massive. BIST dramatically reduces the volume of test data that must be stored and applied externally.
4. Enables System-Level Test: BIST allows a chip to be tested after it is assembled onto a final product. This aids in system-level diagnosis and field repairs.
5. Improves Accessibility: BIST can easily test large internal memory blocks or analog sections that are hard to reach using external pins.

Typical BIST:

The different stages required to perform the BIST operation are shown in the following figure.



First, we need to generate the particular test pattern. Then, after designing the test pattern, it is applied to the Circuit Under Test (CUT). The output of the circuit is obtained, and it is converted into a compact form. There are many techniques to convert the available output into a compact form. One of the most commonly used techniques is signature analysis. The output of response compaction is given to the response analyzer. The response analyzer compares the calculated output with the standard or expected value, and accordingly, the error is calculated.

There are two types of built-in self-test.

1) Logical built-in self-test:

As the name indicates, this self-testing method applies to the logical circuits. It is usually used for identifying the logical faults, like a stuck at zero or stuck at one fault.

2) MBIST:

M stands for memory. So, this built-in self-test applies to memory-related devices like RAM, ROM, etc. It is especially used whenever there is a fault related to the delay propagation.

Advantages:

- External equipment is not required for testing purposes.
- Less testing time.
- Better fault coverage.
- It overcomes the pin limitations and interfacing problems.
- Reduces maintenance.

Disadvantages:

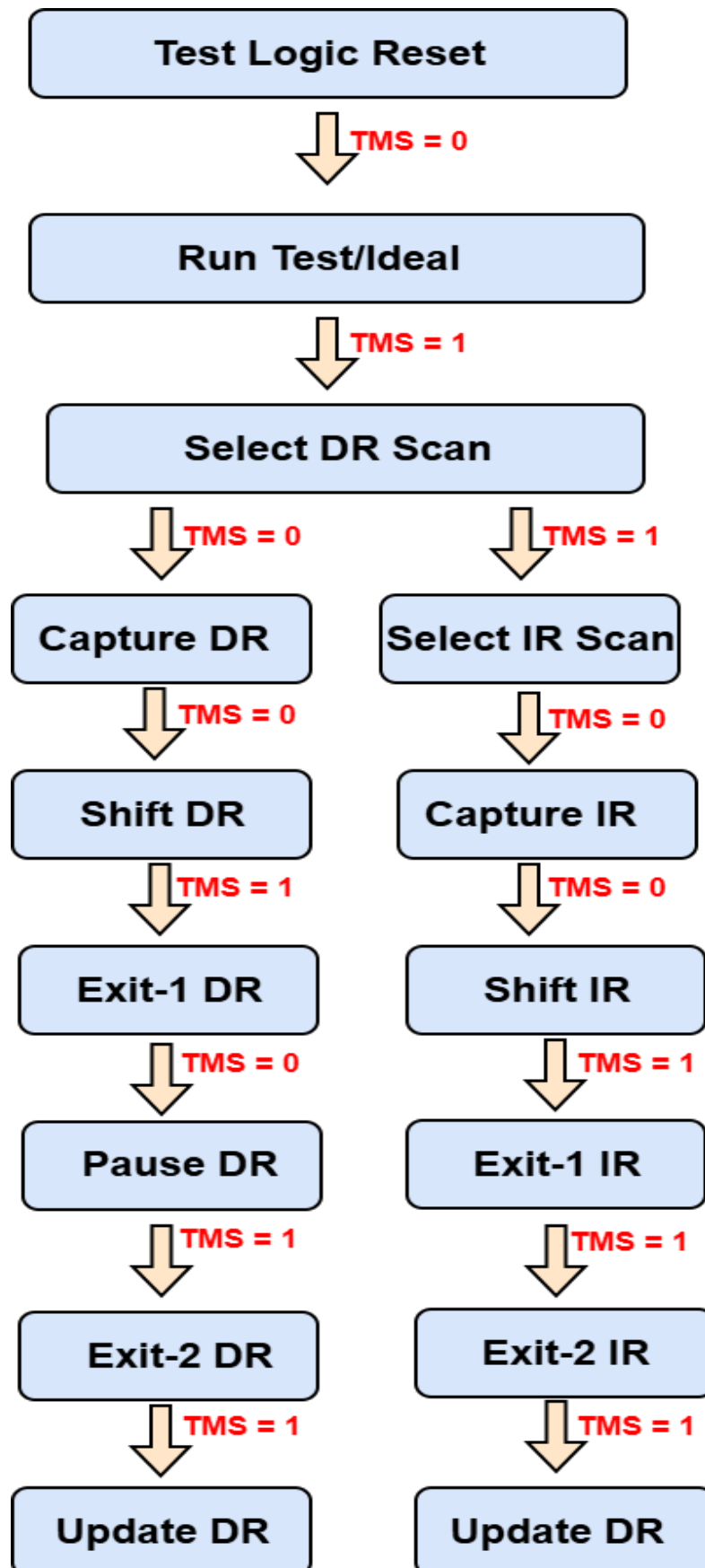
- Extra fabrication (additional silicon area) is required for the fabrication.
- Additional pins are required.

Q.6.13 Draw a TAP controller state diagram and explain. (MAY-24 8 Marks)
OR
Draw and explain the state diagram of TAP controller. (DEC-24 9 Marks)

The TAP controller has a total of 16 states. The state transition depends on the TMS (Test Mode Select) signal, which is sampled at the rising edge of TCK (Test Clock). The TAP Controller ensures a structured way to perform various test operations, such as:

- Scanning test data into the chip.
- Loading test instructions.
- Reading out test results.
- Resetting the test logic.

The state diagram of a TAP controller is shown in the following figure.



The major groups of the states are as follows:

Test-Logic-Reset:

This is the initial state. It puts the JTAG logic into a known, reset condition. It is reached by holding TMS = 1 for 5 TCK cycles.

Run-Test/Idle:

The JTAG logic can perform a test here if an instruction allows it. It is a common state to enter after a reset.

Data Register (DR) Path States:

It includes: Capture-DR, Shift-DR, Exit1-DR, Pause-DR, Exit2-DR, Update-DR. It is used to shift data in or out of data registers.

Instruction Register (IR) Path States:

It includes: Capture-IR, Shift-IR, Exit1-IR, Pause-IR, Exit2-IR, Update-IR. It is used to load and update instructions.

Select States:

It is used for Select-DR-Scan and Select-IR-Scan. It decides whether to work with data or instruction registers.

.....END.....