VLSI DESIGN

Course Code	: 18EC72	CIE Marks	: 40
Lecture Hours/Week	: 3	SEE Marks	: 60
Total Number of Lecture Hours : 40(08 Hrs / Module)		Exam Hours	: 03
CREDITS – 03			

Course Learning Objectives: The objectives of the course is to enable students to:

- Impart knowledge of MOS transistor theory and CMOS technologies
- Learn the operation principles and analysis of inverter circuits.
- Design Combinational, sequential and dynamic logic circuits as per the requirements
- Infer the operation of Semiconductors Memory circuits.
- Demonstrate the concepts of CMOS testing

Module-1

Introduction: A Brief History, MOS Transistors, CMOS Logic
(1.1 to 1.4 of TEXT2)
MOS Transistor Theory: Introduction, Long-channel I-V Characteristics, Non-ideal I-V Effects, DC Transfer Characteristics
(2.1, 2.2, 2.4 and 2.5 of TEXT2)., **L1, L2**

Module-2

Fabrication: CMOS Fabrication and Layout, VLSI Design Flow, Introduction, CMOS Technologies, Layout Design Rules,
(1.5 and 3.1 to 3.3 of TEXT2).
MOSFET Scaling and Small-Geometry Effects, MOSFET Capacitances
(3.5 to 3.6 of TEXT1), **L1, L2,**

Module-3

Delay: Introduction, Transient Response, RC Delay Model, Linear Delay Model, Logical Efforts of Paths **(4.1 to 4.5 of TEXT2, except sub-sections 4.3.7, 4.4.5, 4.4.6, 4.5.5 and 4.5.6).**
Combinational Circuit Design: Introduction, Circuit families
(9.1 to 9.2 of TEXT2, except subsection 9.2.4)., **L1, L2, L3**

Module-4

Sequential Circuit Design: Introduction, Circuit Design for Latches and Flip-Flops **(10.1 and 10.3.1 to 10.3.4 of TEXT2)**
Dynamic Logic Circuits: Introduction, Basic Principles of Pass Transistor Circuits, Synchronous Dynamic Circuit Techniques, Dynamic CMOS Circuit Techniques **(9.1, 9.2, 9.4 to 9.5 of TEXT1)**, **L1, L2, L3**

Module-5

Semiconductor Memories: Introduction, Dynamic Random Access Memory (DRAM) and Static Random Access Memory (SRAM),
(10.1 to 10.3 of TEXT1)
Testing and Verification: Introduction, Logic Verification Principles, Manufacturing Test Principles, Design for testability
(15.1, 15.3, 15.5 15.6.1 to 15.6.3 of TEXT 2). L1, L2

Course outcomes: At the end of the course, the students will be able to:

1. Demonstrate understanding of MOS transistor theory, CMOS fabrication flow and technology scaling.
2. Draw the basic gates using the stick and layout diagrams with the knowledge of physical design aspects.
3. Demonstrate ability to design Combinational, sequential and dynamic logic circuits as per the requirements
4. Interpret Memory elements along with timing considerations
5. Interpret testing and testability issues in VLSI Design

Question paper pattern:

- Examination will be conducted for 100 marks with question paper containing 10 full questions, each of 20 marks.
- Each full question can have a maximum of 4 sub questions.
- There will be 2 full questions from each module covering all the topics of the module.
- Students will have to answer 5 full questions, selecting one full question from each module.
- The total marks will be proportionally reduced to 60 marks as SEE marks is 60.

TEXT BOOKS:

1. "CMOS Digital Integrated Circuits: Analysis and Design" - **Sung Mo Kang & Yosuf Leblebici,** Third Edition, Tata McGraw-Hill.
2. **"CMOS VLSI Design- A Circuits and Systems Perspective"-** Neil H. E. Weste and David Money Harris, 4th Edition, Pearson Education.

REFERENCE BOOKS:

1. Adel Sedra and K. C. Smith, "Microelectronics Circuits Theory and Applications", 6th or 7th Edition, Oxford University Press, International Version, 2009.
2. Douglas A Pucknell & Kamran Eshragian, "Basic VLSI Design", PHI 3rd Edition, (original Edition – 1994).
3. Behzad Razavi, "Design of Analog CMOS Integrated Circuits", TMH, 2007.

MODULE - 1
Introduction

A Brief History:-

In 1958, Jack Kilby built the first integrated circuit flip-flop with two transistors at Texas Instruments.
In 2003, the Intel Pentium 4 microprocessor contained 55 million transistors & a 512-Mbit dynamic random access memory (DRAM) contained more than half a billion transistors. This corresponds to a compound annual growth rate of 53% over 45 years. No other technology in history has sustained such a high growth rate for so long.
This incredible growth has come from steady miniaturization of transistors & improvements in manufacturing processes. Most other fields of engineering involve tradeoffs b/w performance, power and price. However, as transistors become smaller, they also become faster, dissipate less power, and are cheaper to manufacture.

In 1947, John Bardeen, William Shockley & Walter Brattain built the first functioning point contact transistor at Bell laboratories.
"Transistor is a semiconductor device which can amplify electrical signal as they are transferred through it from input to output terminals".
Soon after inventing the point contact transistor, Bell Labs developed the bipolar junction transistor. Bipolar transistors were more reliable, less noisy, and more power-efficient.
Early integrated circuits primarily used bi-polar transistor. Transistors can be viewed as electrically controlled switches with a control terminal & two other terminals that are connected (or) disconnected depending on the voltage applied to

the control. Bipolar transistors require a small current into the control (base) terminal to switch much larger currents between the other two (emitter and collector) terminals. The quiescent power dissipated by these base currents limits the maximum number of transistors that can be integrated onto a single die.

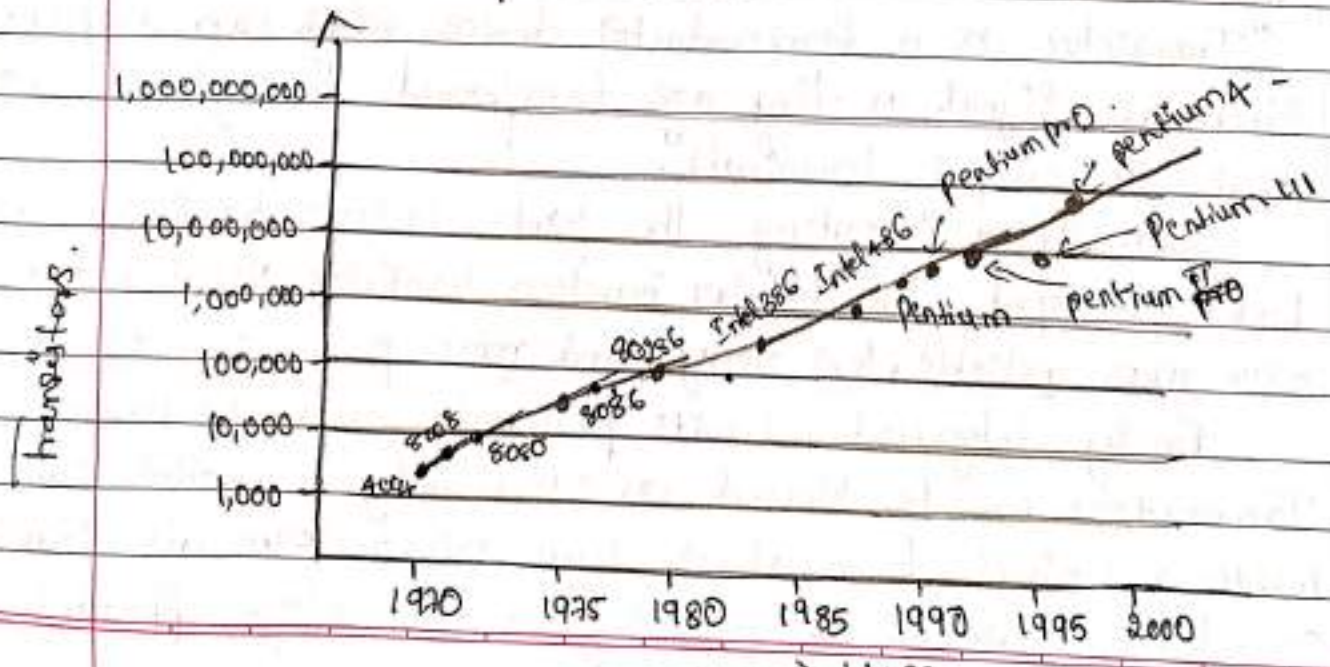
Metal oxide Semiconductor Field Effect Transistors (MOSFET'S) offer the compelling advantage that they draw almost zero control current while idle.

They come in two flavors: nMOS and pMOS, using n-type and p-type dopants, respectively.

Frank Wanlass described the first logic gates using MOSFET'S in 1963. His gates used both nMOS and pMOS transistors, earning the name Complementary Metal oxide Semiconductor, or CMOS.

Gordon Moore observed in 1965 that plotting the no of transistors that can be most economically fabricated on a chip gives a straight line on a semilogarithmic scale. At the times he found transistor count doubling every 18 months. This observation has been called Moore's Law

Figure below shows that the no of transistors in Intel microprocessors has doubled every 26 months since the invention of the 4004.



Moore's Law States that "The no of transistors integrated on a single chip doubles every 18 months".

The level of integration of chips has been classified as,

(a) Small Scale Integration (SSI) have 10 - 100 transistors
Eg:- logic gates.

(b) Medium Scale Integration (MSI) have 100 - 1000 transistors
Eg:- Counters.

(c) Large Scale Integration (LSI) have 1000 - 20,000 transistors
Eg:- 8-bit µp.

(d) Very large Scale Integration (VLSI) have 20,000 - 1,000,000 transistors Eg:- 16-bit µp.

MOS Transistors:-

Silicon (Si), a semiconductor, forms the basic starting material for a large class of integrated circuits.

An MOS (Metal-oxide-semiconductor) structure is created by superimposing several layers of conducting & insulating materials to form a sandwich-like structure.

This structures ~~is~~ are manufactured using a series of chemical processing steps involving oxidation of Si, the diffusion of impurities into the silicon to give it certain conducting characteristics, and the deposition and etching of aluminum (or) other metals to provide interconnection in the same way that a printed wiring board is constructed.

Mos transistor consists of a stack of the conducting gate, an insulating layer of silicondioxide (SiO_2) and the silicon wafer also called the substrate, body (or) bulk. gate is made of polycrystalline silicon (or) polysilicon.

Adjacent to the gate, the Source & drain diffusions are present in the bulk.

There are mainly two types of MOS transistors they are.

(1) NMOS Transistor.

(2) PMOS Transistor.

An nMOS transistor is built with a p-type body & has regions of n-type semiconductor adjacent to the gate called the Source & drain.

A pMOS transistor is just the opposite, consisting of p-type Source & drain regions with an n-type body.

Figure below shows cross-sections & symbols of these transistors.

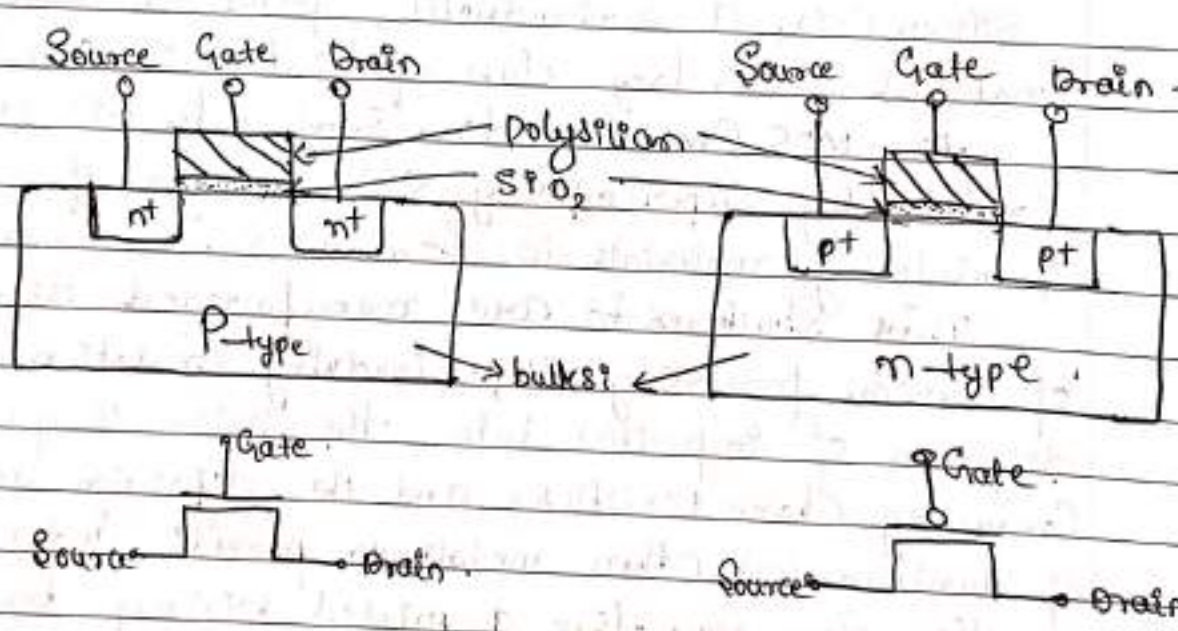


fig (a) :- NMOS transistor

fig (b) :- PMOS transistor

The gate is a control input. It affects the flow of Electrical current between the Source and drain.

The gate of an MOS transistor controls the flow of current b/w the source and drain.

When the gate of an nMOS transistor is '1', the transistor is ON and there is a conducting path from source to drain.

When the gate is low, the nMOS transistor is OFF and almost zero current flows from source to drain.

A pMOS transistor is just the opposite, being ON when the gate is low and OFF when the gate is high.

This switch model is illustrated in fig below, where g, s and d indicates gate, source and drain.

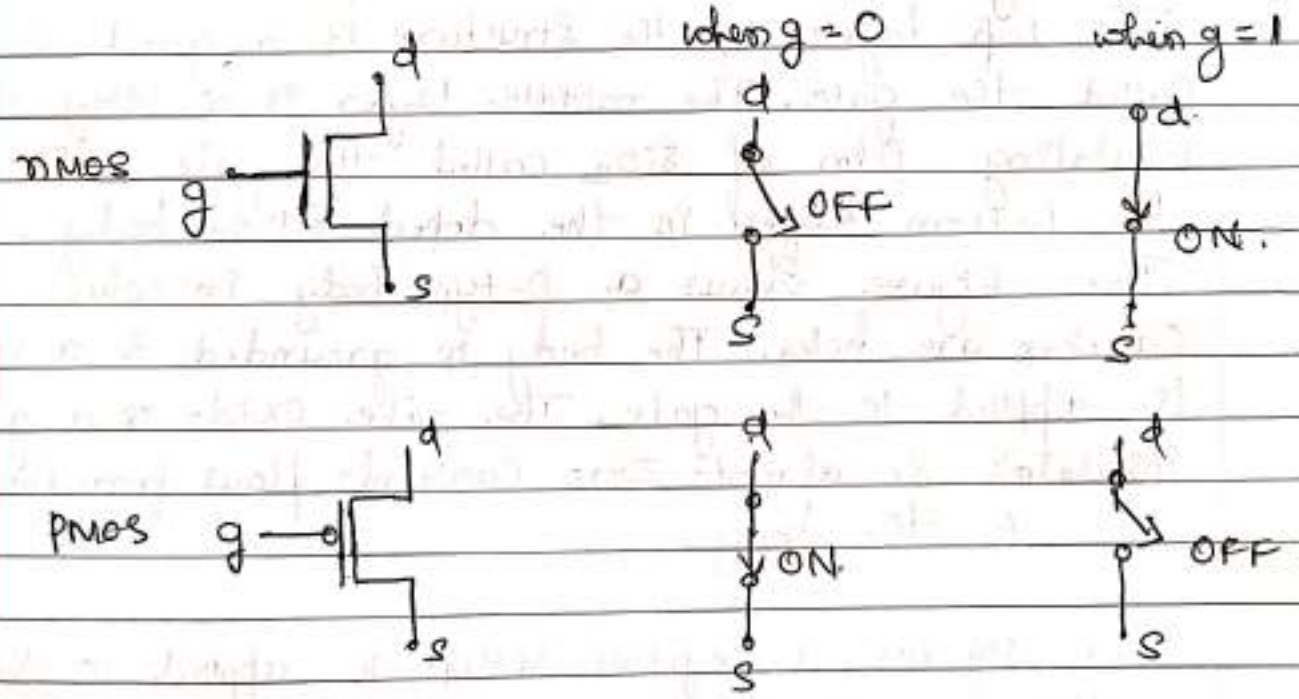


fig:- Transistor Symbols and Switch-level models.

Page No.:

MOS Transistor Theory:-

MOS Structure:-

The MOS transistor is a majority-carrier device in which the current in a conducting channel b/w the source & drain is controlled by a voltage applied to the gate.

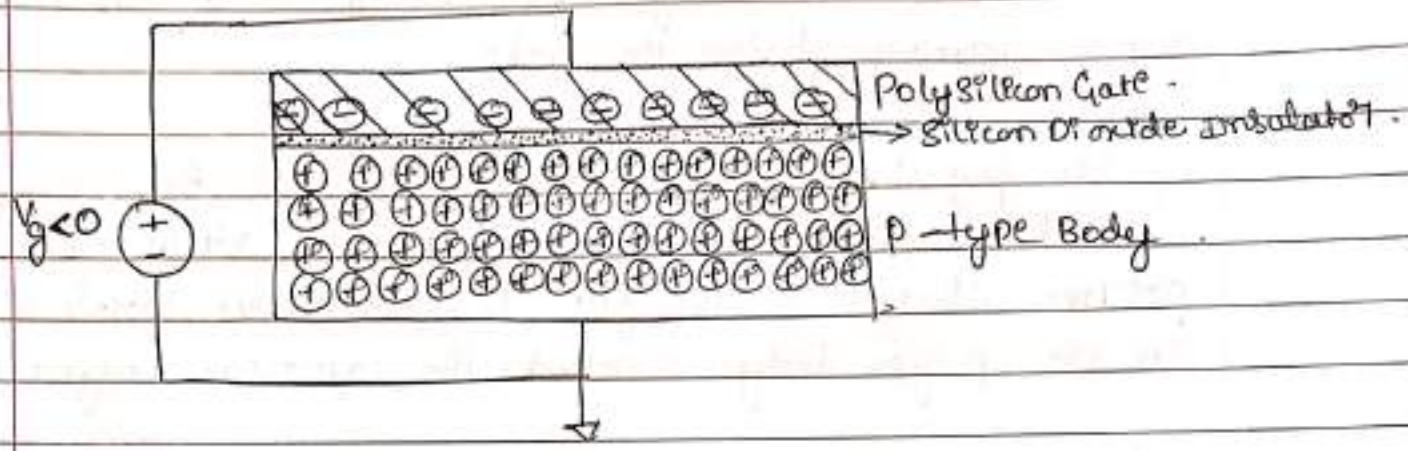
In an nMOS transistor, the majority carriers are electrons, in a pMOS transistor, the majority carriers are holes.

The behavior of MOS transistors can be understood by first examining an isolated MOS structure with a gate & body but no source or drain.

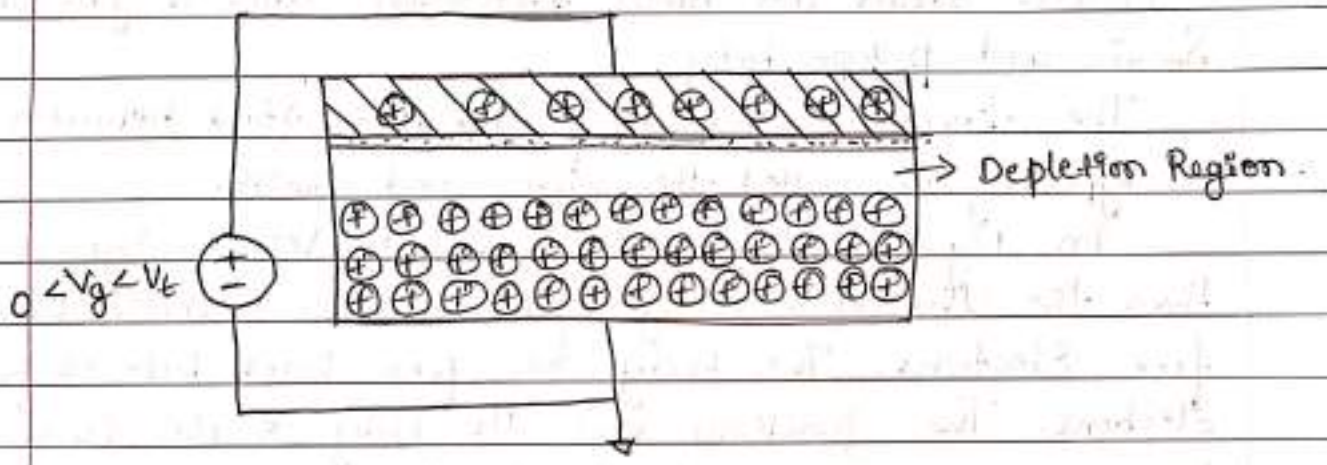
Figure below shows a simple MOS structure. The top layer of the structure is a good conductor called the gate. The middle layer is a very thin insulating film of SiO_2 called the gate oxide. The bottom layer is the doped silicon body. The figure shows a p-type body in which the carriers are holes. The body is grounded & a voltage is applied to the gate. The gate oxide is a good insulator so almost zero current flows from the gate to the body.

In fig (a), a negative voltage is applied to the gate, so there is negative charge on the gate. The mobile positively charged holes are attracted to the region beneath the gate. This is called the accumulation mode.

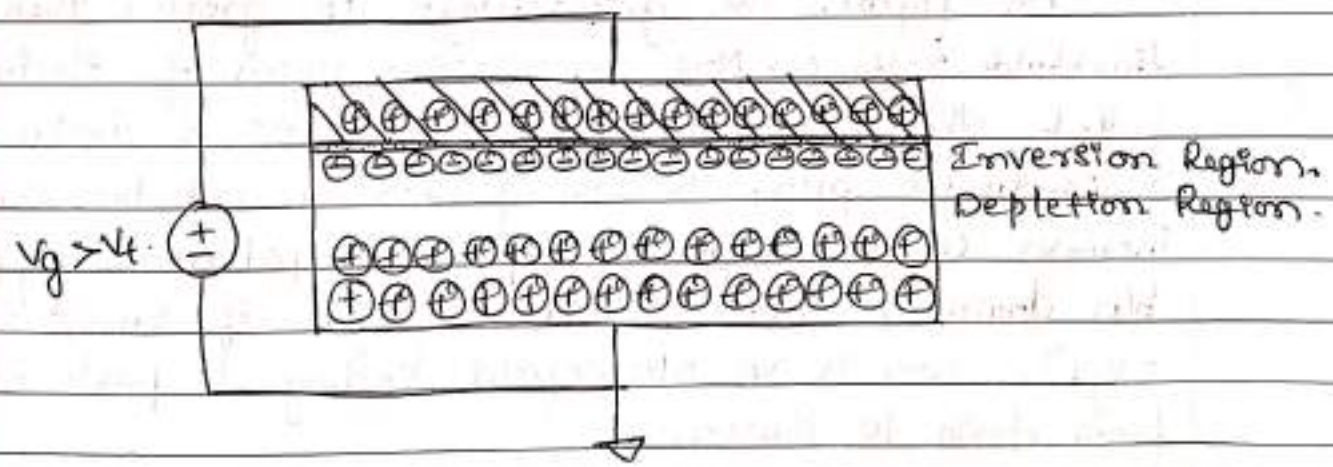
In fig (b), a low positive voltage is applied to the gate, resulting in some positive charge on the gate. The holes in the body are repelled from the region



(a) Accumulation.



(b) Depletion.



(b) Inversion.

Fig:- MOS Structure demonstrating (a) accumulation (b) depletion (c) Inversion.

directly beneath the gate, resulting in a depletion region forming below the gate.

In fig C, a higher positive potential exceeding a critical threshold voltage V_t is applied, attracting more positive charge to the gate. This conductive layer of electrons in the p-type body is called the inversion layer.

Working of nMOS transistor :-

Figure shows an nMOS transistor with a grounded source and p-type body.

The transistor consists of the MOS stack between two n-type regions called the source and drain.

In fig (a) the gate to source voltage V_{gs} is less than the threshold voltage. The source & drain have free electrons. The body has free holes but no free electrons. The junctions b/w the body & the source (or) drain are reverse biased, so almost zero current flows. This mode of operation is called cut-off.

In fig (b), the gate voltage is greater than the threshold voltage. Now an inversion region of electrons called the channel connects the source & drain, creating a conductive path. The no of carriers and the conductivity increases with the gate voltage. The potential difference b/w drain & source is $V_{ds} = V_{gs} - V_{gd}$. If $V_{ds} = 0$ (i.e $V_{gs} = V_{gd}$), there is no electricfield tending to push current from drain to source.

When a small positive potential V_{ds} is applied to the drain (fig(c)), current I_{ds} flows through the channel from drain to source. This mode of operation is termed

linear, the current increases with both the drain voltage & gate voltage.

If V_{ds} becomes sufficiently large that $V_{gd} < V_t$, the channel is no longer inverted near the drain and becomes pinched off (fig (d)). However, conduction is still brought about by the drift of electrons under the influence of the positive drain voltage.
As the electrons reach the end of the channel, they are injected into the depletion region near the drain & accelerated toward the drain. Above this drain voltage the current I_{ds} is controlled only by the gate voltage & ceases to be influenced by the drain. This mode is called saturation.

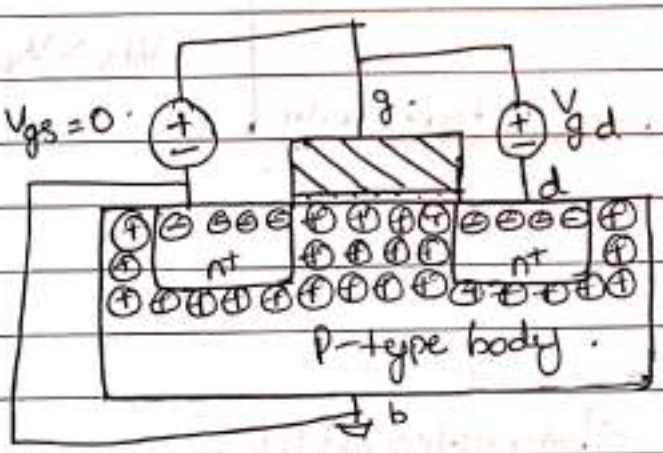


fig (a)

Cut off : No channel
$I_{ds} = 0$.

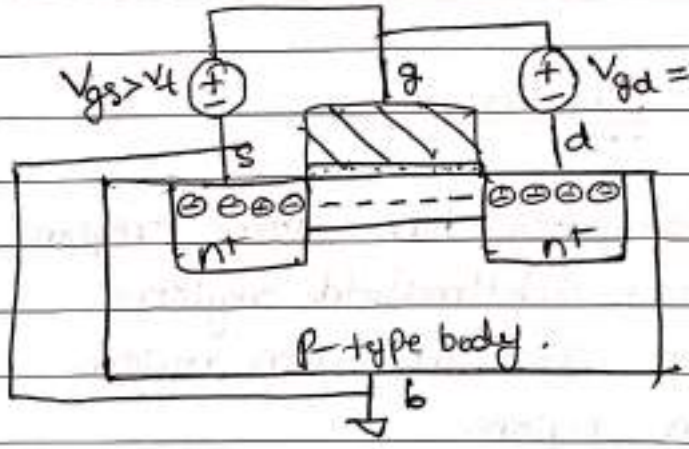
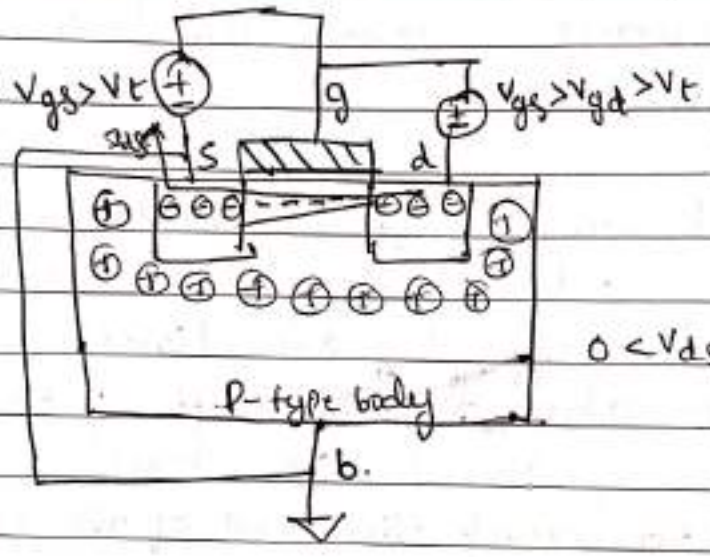


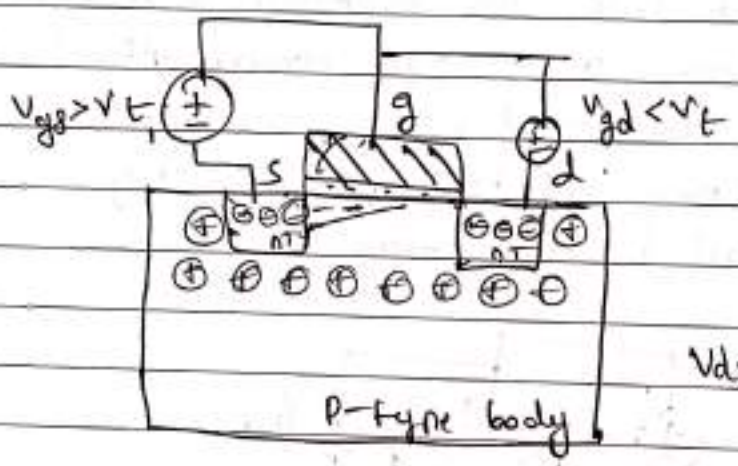
fig (b)

$V_{ds} = 0$.



I_{ds} increases with V_{ds}.

$0 < V_{ds} < V_{gs} - V_t$.



Saturation.
Channel pinched off.
I_{ds} independent of V_{ds}.

$V_{ds} > V_{gs} - V_t$.

Ideal I-V Characteristics:-

Derivation for I_{ds}:-

MOS transistors have three regions of operation.

(1) Cut-off (or) Subthreshold region.
(2) Linear (or) non-saturation region.
(3) Saturation region.

In cut-off regions, ($V_{gs} < V_t$), there is no channel and almost zero current flow from drain to source.

In the other regions, the gate attracts carriers to form a channel. The Electrons drift from source to drain at a rate proportional to the Electric field between these regions. Thus we can compute currents if we know the amount of charge in the channel & the rate at which it moves.

W.K.T the charge on each plate of a capacitor is $Q = C.V$. Thus the charge in the channel. $Q_{channel}$ is,

$$\boxed{Q_{channel} = C_g (V_{ge} - V_t)} \rightarrow (1)$$

where C_g is the capacitance of the gate to the channel and.

$(V_{ge} - V_t)$ is the amount of voltage attracting charge to the channel beyond the minimum required to invert from p to n.

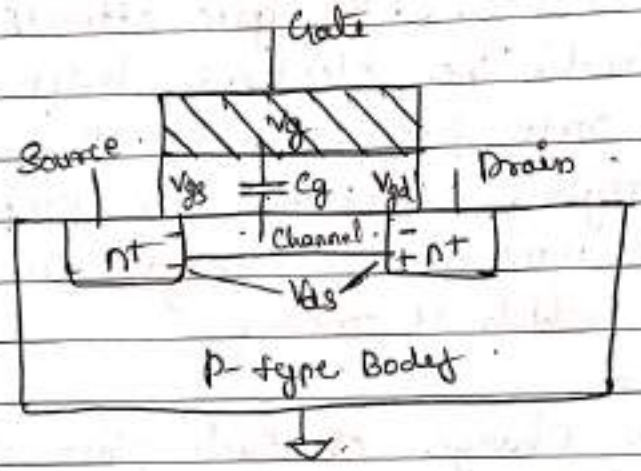
The gate voltage is referenced to the channel, which is not grounded.

If the source is at V_s and the drain is at V_d, the average is $V_c = \frac{V_s + V_d}{2} = V_s + V_{ds}/2$.

$\therefore$ the mean difference b/w the gate & channel potential V_{ge} is ~~V_{gs} is~~ $V_{gs} - \frac{V_{ds}}{2}$.

We can model the gate as a parallel plate capacitor with capacitance proportional to area over thickness. If the gate has length 'L' and width 'W' & the oxide thickness is t_{ox} as shown in figure, the capacitance is given ~~by~~ as,

$$\boxed{C_g = \varepsilon_{ox} \frac{W.L}{t_{ox}}} \rightarrow (2).$$



Average gate to channel potential :

$V_{gc} = V_{gs} - V_{ds}/2$.

where ϵ_{ox} is the permittivity of the oxide.

& $\boxed{\frac{t_{ox}}{\epsilon_{ox}} = C_{ox}}$

Each carrier in the channel is accelerated to an average velocity proportional to the lateral Electric field i.e the field b/w Source & drain.

$\boxed{V = \mu E}$

The Electric field 'E' is the voltage difference b/w drain & Source V_{ds} divided by the channel length.

$$\boxed{E = \frac{V_{ds}}{L}}$$

The time required for carriers to cross the Channel length divided by the carrier velocity: L/V.
∴ the current b/w Source & drain is the total amount of charge in the channel divided by the time required to cross.

$$I_{ds} = \frac{Q}{t}, \quad I_{ds} = \frac{Q_{channel}}{L/V}$$

$$= \frac{C_g (V_{gc} - V_t)}{L} \cdot V$$

$$= \frac{\epsilon_{ox}}{t_{ox}} \cdot \frac{WL}{L} \left(V_{gs} - \frac{V_{ds}}{2} - V_t\right) \cdot \mu E$$

$$= \mu \cdot C_{ox} \cdot W \left(V_{gs} - V_t - \frac{V_{ds}}{2}\right) \cdot \frac{V_{ds}}{L}$$

$$= \mu \cdot C_{ox} \cdot \frac{W}{L} \left(V_{gs} - V_t - \frac{V_{ds}}{2}\right) \cdot V_{ds}$$

$$= \beta \cdot \left(V_{gs} - V_t - \frac{V_{ds}}{2}\right) V_{ds} \quad \text{where } \beta = \mu C_{ox} \cdot \frac{W}{L}$$

$$\boxed{I_{ds} = \beta \left\{ (V_{gs} - V_t) V_{ds} - \frac{V_{ds}^2}{2} \right\}} \rightarrow (3)$$

→ Current Equation for linear region.

However, if $V_{ds} > V_{dsat} \simeq V_{gs} - V_t$, the channel is no longer inverted in the vicinity of the drain, we say it is pinched off.

Substituting $V_{ds} = V_{gs} - V_t$ in Eqn (3) we get.

$$I_{ds} = \beta \left\{ (V_{gs} - V_t)(V_{gs} - V_t) - \frac{(V_{gs} - V_t)^2}{2} \right\}$$

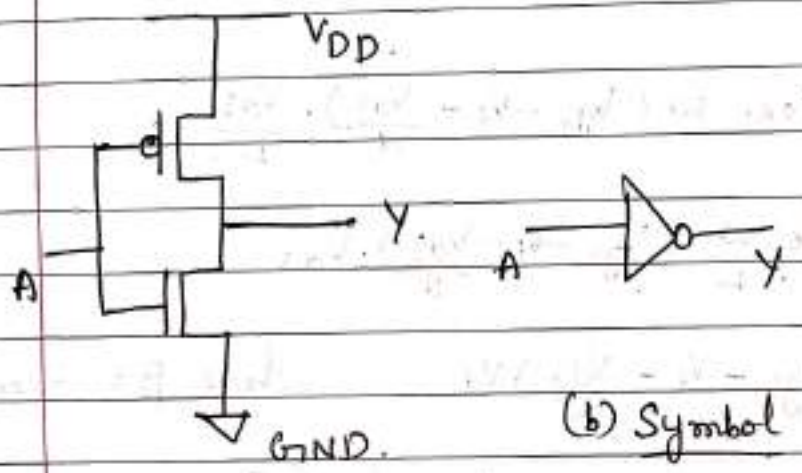
$$\boxed{I_{ds} = \frac{\beta}{2} (V_{gs} - V_t)^2} \quad \text{— (4)}$$

Current Equation for Saturation region.

1.4 CMOS LOGIC :-

THE INVERTER

$Y = \overline{A}$



(b) Symbol

A	Y
0	1
1	0

(c) Truth table -

fig(a) :- Inverter Schematic.

* Figure (a) Shows a CMOS inverter (or) NOT gate using one NMOS transistor & one PMOS transistor.

* The horizontal bar at the top indicates V_{DD} & the triangle at the bottom indicates GND.

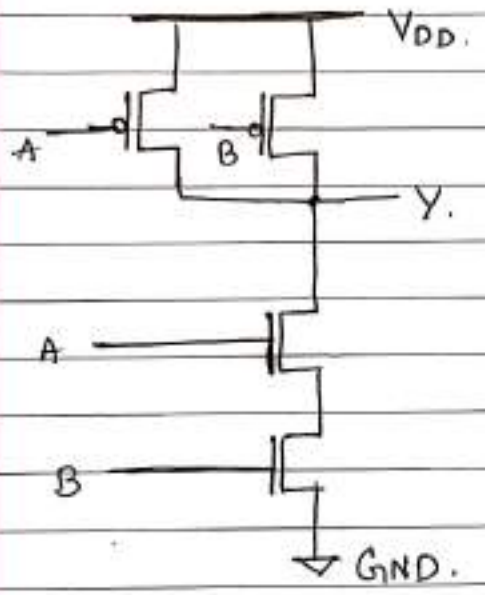
* When the i/p 'A' is '0', the NMOS transistor is OFF & the PMOS transistor is ON. Thus the o/p 'Y' is pulled up to '1'.

* When the i/p 'A' is '1', the NMOS is ON, the PMOS is OFF, & the Y is pulled down to '0'.

* The Symbol & the Truth table is given in figure (b) & (c).

The NAND Gate :-

* Figure (a) shows a 2-input CMOS NAND gate. It consists of two series NMOS transistors b/w Y & GND, & two parallel PMOS transistors b/w Y & V_{DD}.



(b) Symbol.

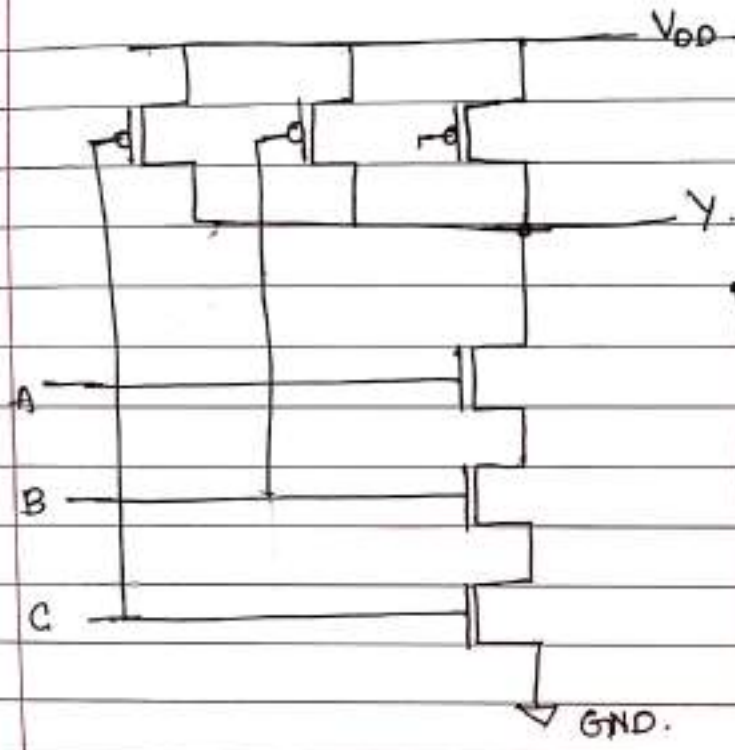
$$Y = \overline{A \cdot B}$$

fig (a) :- 2-i/p NAND Gate Schematic.

* If either input A (or) B is '0', at least one of the NMOS transistors will be OFF, breaking the path from Y to GND. But at least one of the PMOS transistors will be ON, creating a path from Y to V_{DD}. Hence, the o/p Y will be '1'.

* If both inputs are '1' both of the NMOS transistors will be ON & both of the PMOS transistors will be OFF. Hence the o/p will be '0'.

3-input NAND Gate:-



A	B	Pull-down n/w	Pull-up n/w	Y
0	0	OFF	ON	1
0	1	OFF	ON	1
1	0	OFF	ON	1
1	1	ON.	OFF	0

Fig:- 3-input NAND gate Schematic $Y = A \cdot B \cdot C$

NAND gate truth table.

K-input NAND gates are constructed using K-Series NMOS transistors & K-parallel PMOS transistors.

COMBINATIONAL LOGIC:-

In general, a fully complementary CMOS gate has an NMOS pull-down network to connect the o/p to '0' (GND) & PMOS pull-up n/w to connect the o/p to '1' (V_{DD}). as shown in figure.

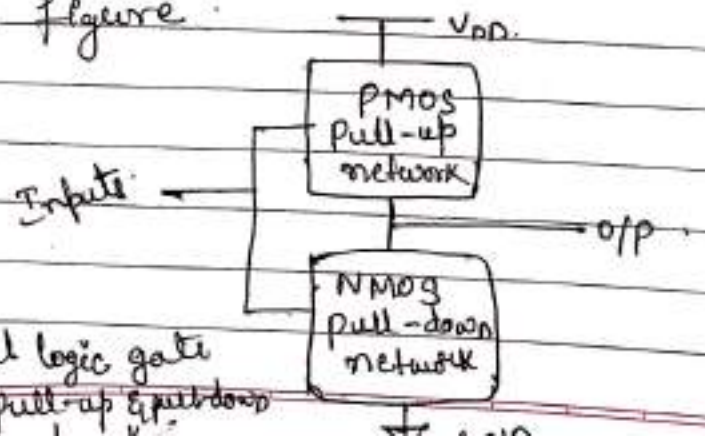


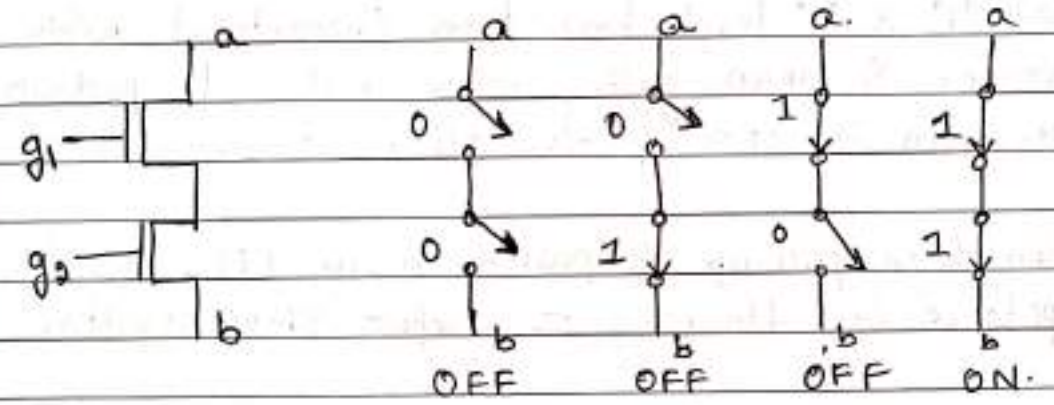
Fig:- General logic gate using pull-up & pull-down

* The inverter & NAND gates are examples of CMOS logic gates, also called static CMOS gates.

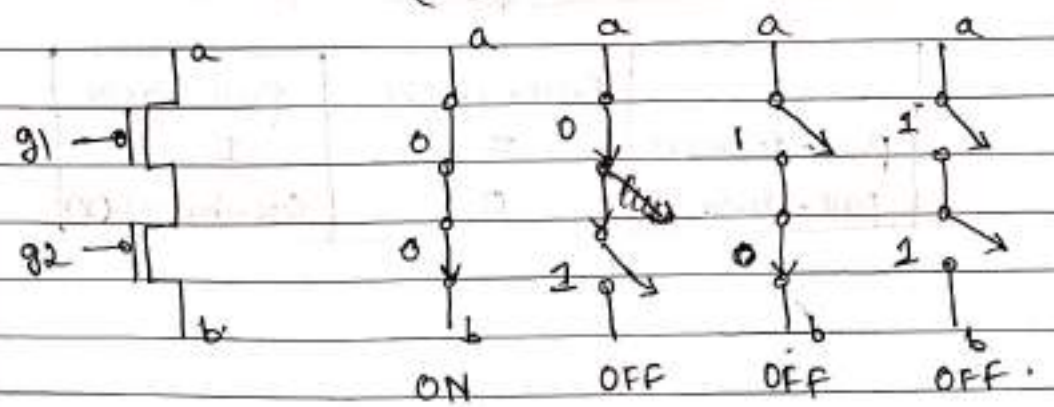
* Two (or) More transistors in series are ON only if all of the series transistors are ON.

* Two (or) More transistors in parallel are ON if any of the parallel transistors are ON.

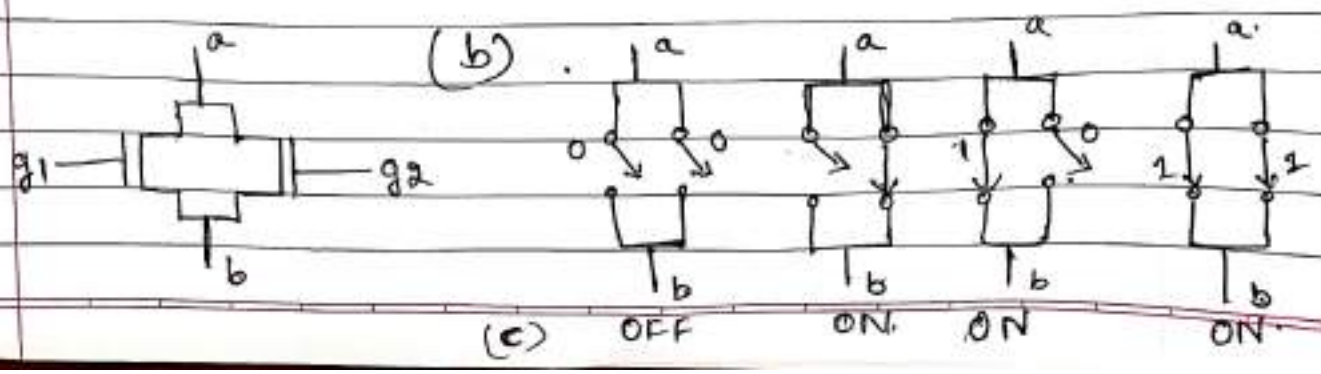
* The NMOS & PMOS transistor pairs are illustrated below in figures.



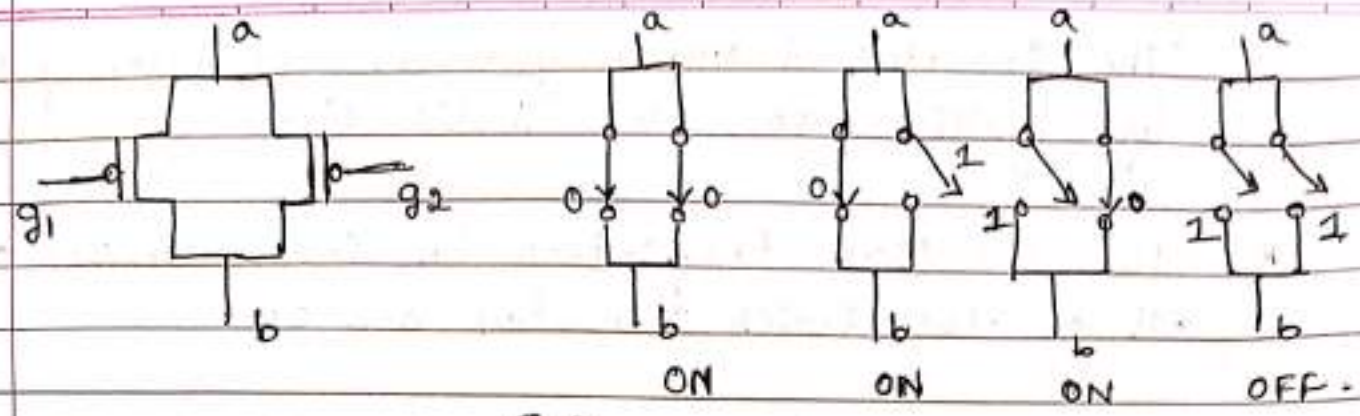
(a)



(b)



(c)



(d)

Connection & behavior of Series & parallel transistors.

* By using Combinations of these Constructions, CMOS Combinational gates can be constructed.

* The '1' & '0' levels have been Encountered with the inverter & NAND gates, where either the pull-up (or) pull-down is OFF & the other Structure is ON.

* when both pull-up & pull-down are OFF, the high-impedance (or) floating Z output State results.

Output States of CMOS logic gate

	Pull-up OFF	Pull-up ON
Pull-down OFF	Z	1
pull-down ON	0	Crowbarred (X)

Example :-

* Sketch a 3-input CMOS NOR gate

Soln:- $Y = \overline{A+B+C}$.

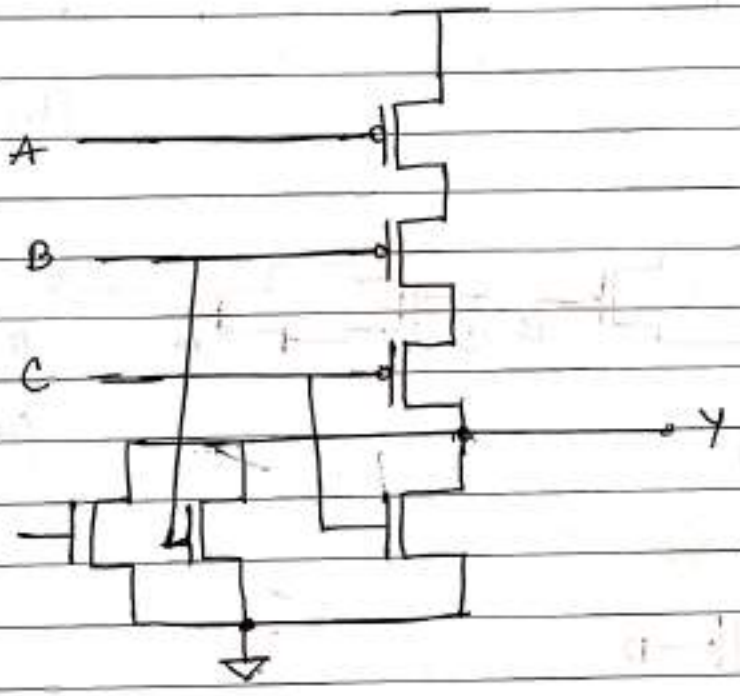


Fig:- 3-i/p NOR gate Schematic

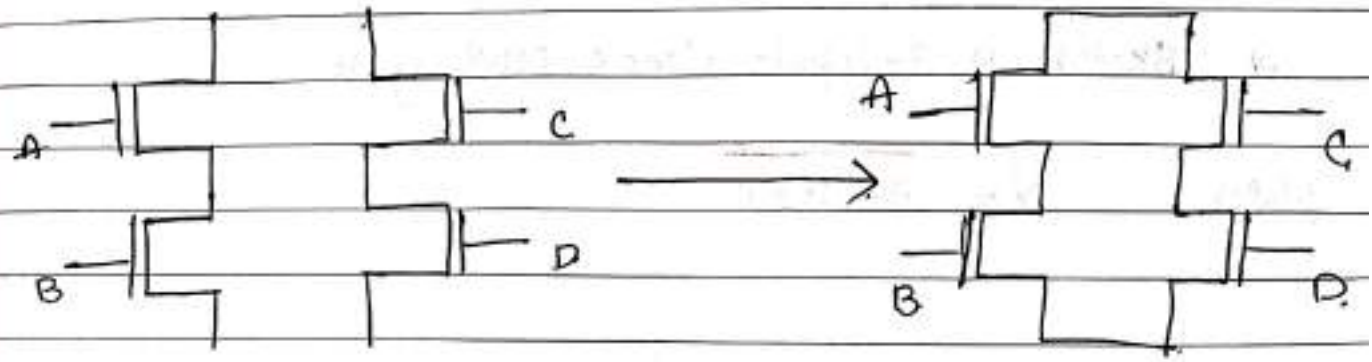
Figure shows a 3-i/p CMOS NOR gate Schematic. If any i/p is high, the o/p is pulled low through the parallel NMOS transistors. If all i/p's are low, the o/p is pulled high through the series PMOS transistors.

COMPOUND Gates :-

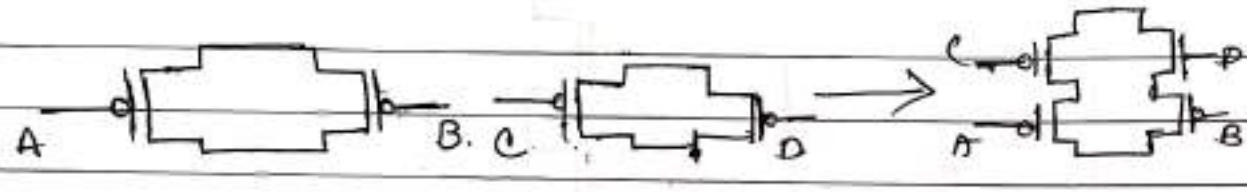
* A Compound gate is formed by using a combination of series & parallel switch structures.

* For Example, the derivation of the switch connection diagram for the function $Y = \overline{(A \cdot B) + (C \cdot D)}$ is

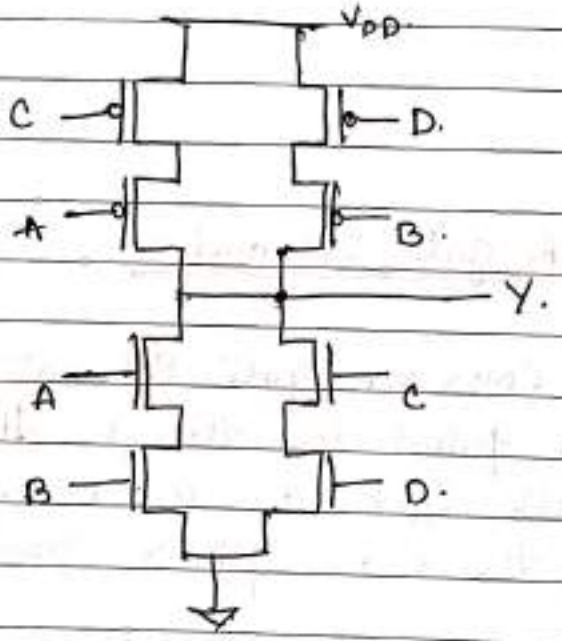
shown in figure below.



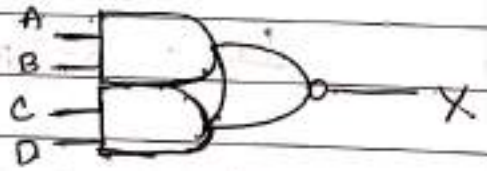
(a). (b).



(c) (d)



(e)



(f).

CMOS Compound gate for function

$$Y = \overline{(A \cdot B) + (C \cdot D)}$$

* This function is sometimes called AND-OR-INVERT-22 (or) AOI22 because it performs the NOR of a pair of 2-input ANDs.

Fig (a) → The AND expressions (A.B) & (C.D) may be implemented by series connections of switches.

Fig(b) → ORing the result requires the parallel connection of two structures.

Fig(c) & (d) → In the pull-up network, the parallel combination of A & B is placed in series with the parallel combination of C & D.

Fig(e) → putting the networks together yields the connection diagram.

Fig(f) → The schematic icon, which shows that this gate can be used in a 2-i/p Multiplexer.

If $C = \overline{B}$, then $Y = A$ if B is true, ~~while Y = D if B is~~ while $Y = D$ if B is false.

PASS Transistors & Transmission Gates:-

* The strength of a signal is measured by how closely it approximates an ideal voltage source.
* In general, the stronger a signal, the more current it can source (or) sink.
* The power supplies (or) rails (V_{DD} & GND) are the source of the strongest 1s and 0s.

N-MOS pass transistor :-

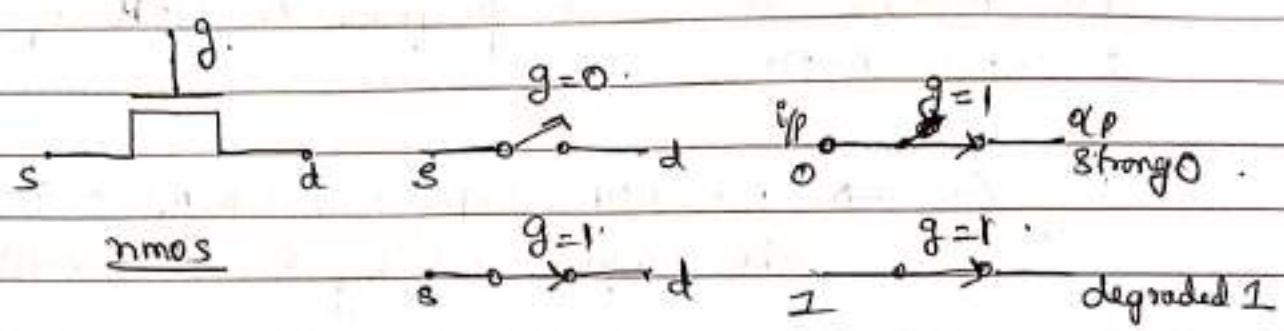


fig (a) fig (b) fig (c)

* An NMOS transistor is an almost perfect switch when passing a 0 and thus we say it passes a strong 0.

* NMOS transistor is imperfect at passing a 1. The high voltage is somewhat less than V_{DD}. It passes a degraded (or) weak 1.

P-MOS pass transistor :-

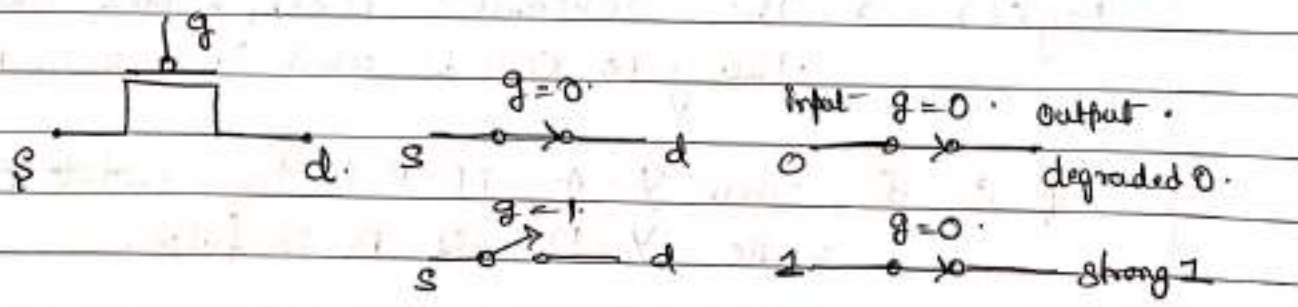
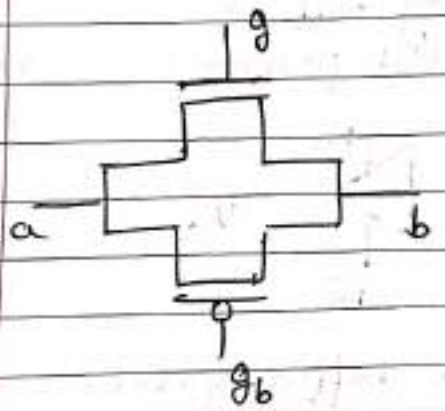


fig (d) fig (e) fig (f)

* A pMOS transistor has the opposite behaviour, passing strong 1's but degraded 0's.
* The transistor symbols & behaviors are shown in figure with g, s, and d indicating gate, source & drain.

Transmission Gates:-

By combining an nMOS and a pMOS transistor in parallel we obtain a switch that turns on when a 1 is applied to g in which 0s & 1's are both passed in acceptable fashion. We term this a transmission gate (or) pass gate.



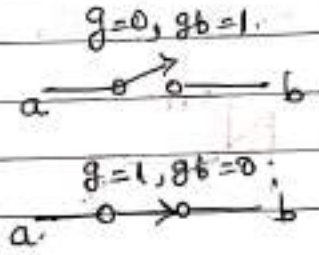


fig ⓑ

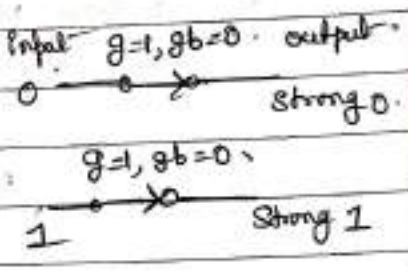


fig ⓒ

fig ⓐ :- Circuit diagram.

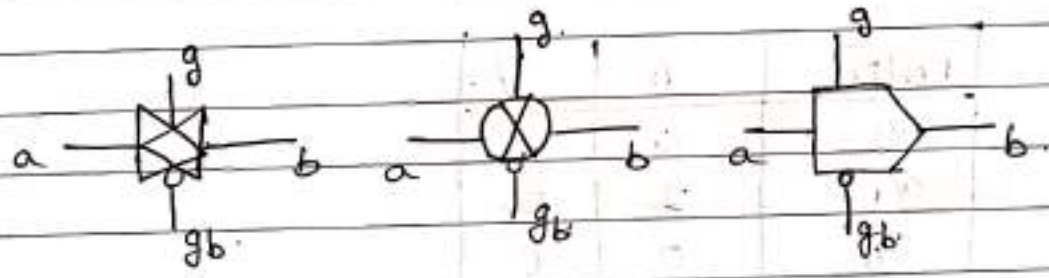


fig ⓓ Symbols of TG.

In a circuit where only a 0 (or) a 1 has to be passed, the appropriate of transistor (n or p) can be deleted, reverting to a single nMOS (or) pMOS device.

Tristates :-

① Tri-state Buffer :-

* Figure below shows symbols for a tri-state buffer.
* When the enable input EN is 1, the output Y equals the input A, just as in an ordinary buffer.
* When the Enable is 0, Y is left floating (a 'Z' value).

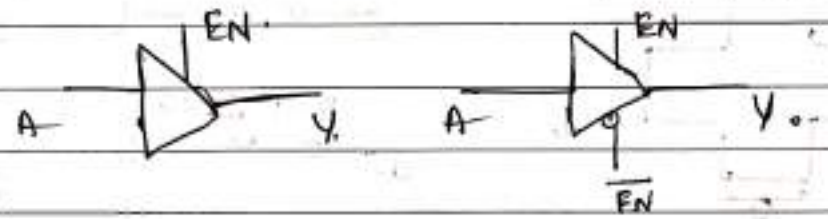
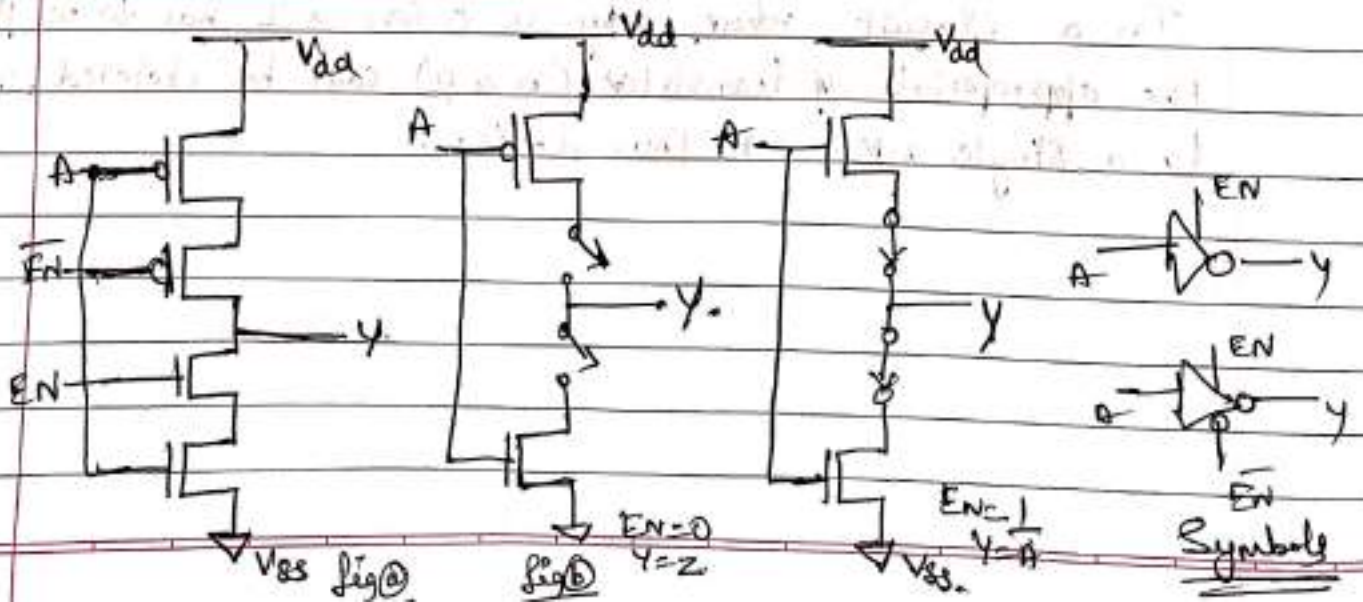


Figure :- Tri-state buffer symbol.

Truth table for tri-state.

EN/$\overline{EN}$	A	Y
0/1	0	Z
0/1	1	Z
1/0	0	0
1/0	1	1

② Tri-State Inverter :-



* Fig (a) shows a tri-state inverter. The output is actively driven from V_{DD} or GND, so it is a restoring logic.
* Tri-state inverter does not obey the conduction complement-is rule because it allows the o/p to float under certain input combinations.
* when EN = 0, as shown in fig (b), both enable transistors are OFF, leaving the output floating.
* when EN = 1, as shown in fig (c), both enable transistors are ON.
* Fig (d) shows symbols for the tristate inverter.

Multiplexers:-

Multiplexers are key components in CMOS memory elements and data manipulation structures.

A Multiplexer chooses the output from among several inputs based on a select signal.

A 2-input, (or) 2:1 Multiplexer, chooses input D0 when the select is 0 and input D1 when the select is 1.

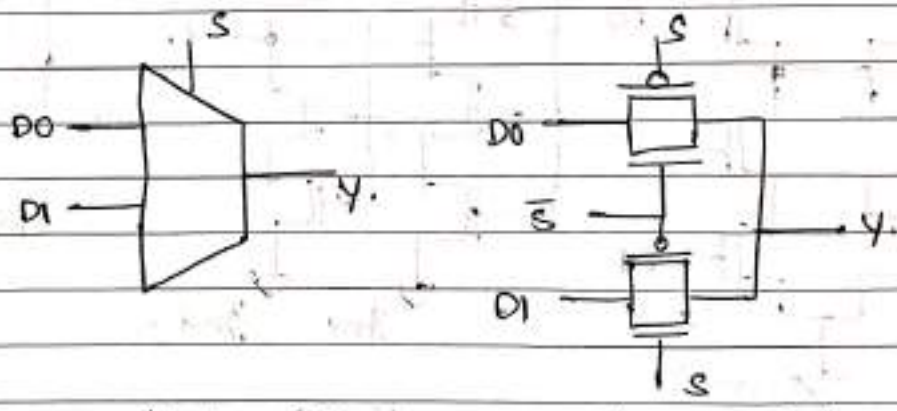


fig:- Transmission gate multiplexer.

The truth table is given below.

Multiplexer Truth table.

$S/\bar{S}$	D1	D0	Y
0/1	X	0	0
0/1	X	1	1
1/0	0	X	0
1/0	1	X	1

The logic function is $Y = \bar{S} \cdot D0 + S \cdot D1$.

~~The~~ Two transmission gates can be tied together to form a compact 2-input multiplexer as shown in figure. The select & its complement enable exactly one of the two transmission gates ~~any~~ at any given time.

We could build a restoring, inverting multiplexer out of gates in several ways.

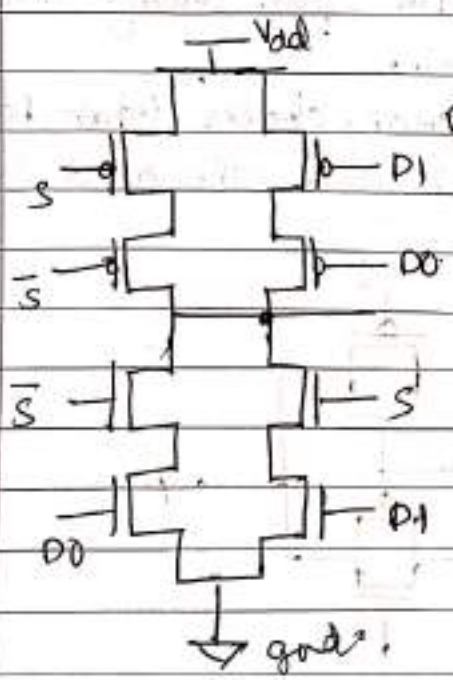


fig ⓐ :- Compound gate multiplexer.

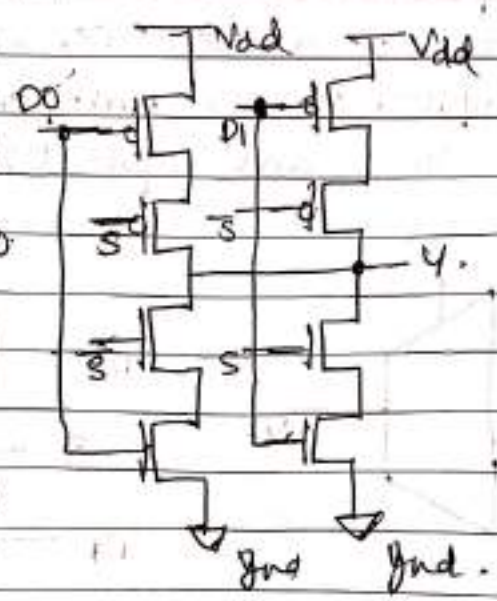


fig ⓑ :- gang together two tristate inverter

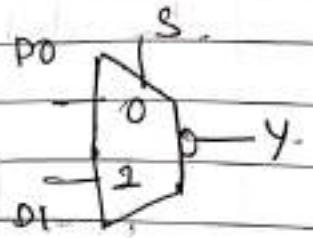
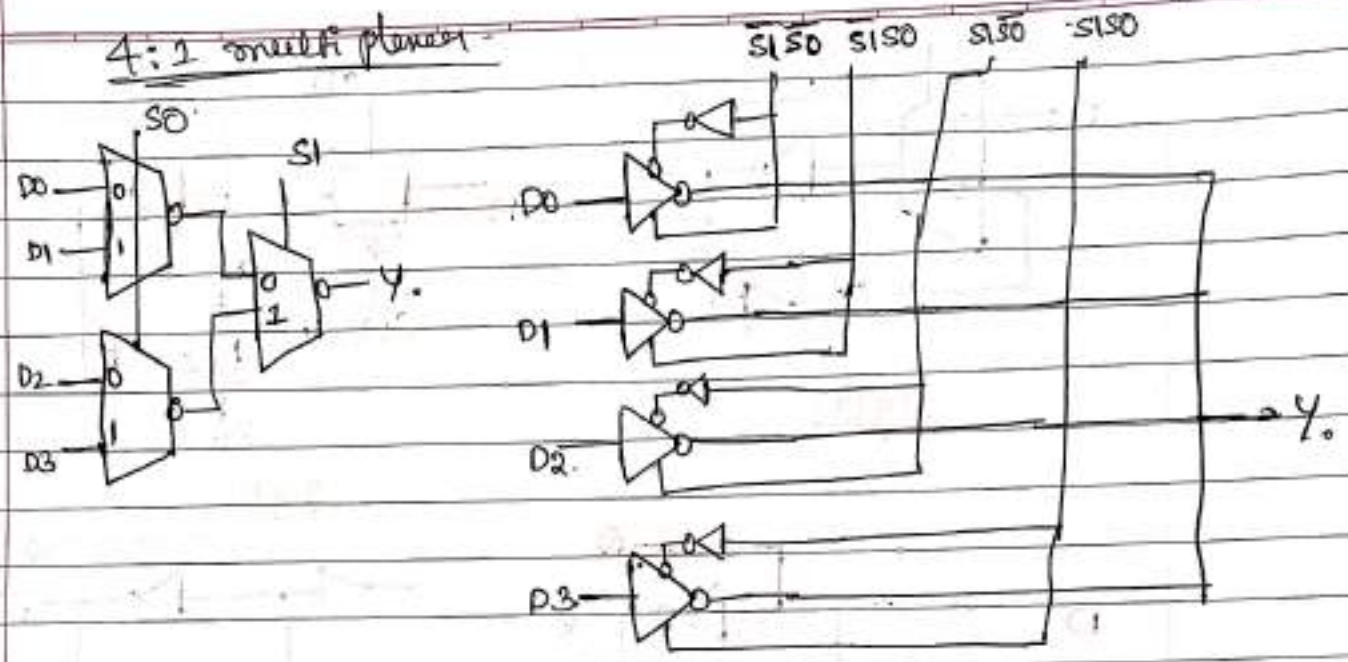


fig ⓒ.

4:1 multiplexer



Larger multiplexer can be built from multiple 2-input multiplexers (or) by directly gang ganging together several tristates.

Sequential Circuits:-

So far, we have considered combinational circuits, whose outputs depend only on the current inputs.
Sequential circuits have memory, their outputs depend on both current & previous inputs.
Sequential circuits such as latches and flip-flops. These elements receive a clock, CLK and a data input, D, and produce an output Q.

Latches:-

A D-latch is transparent when CLK = 1, meaning that Q follows D. It becomes opaque when CLK = 0, meaning Q retains its previous value and ignores changes in D.

A D-latch built from a 2-input multiplexer and two inverters is shown in figure.

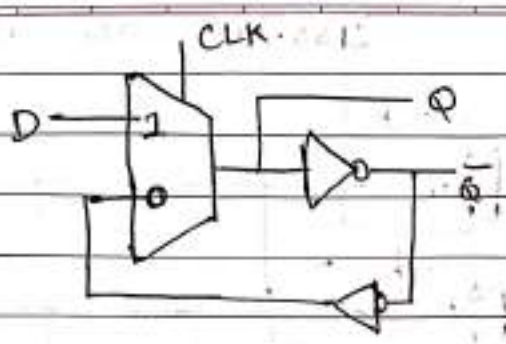


fig ⓐ :-

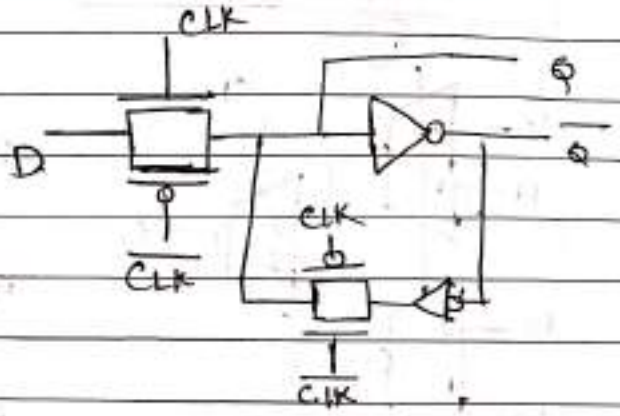


fig ⓑ

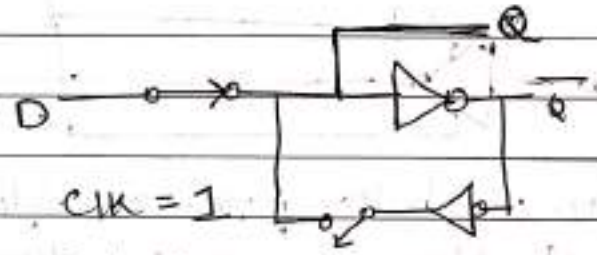


fig ⓒ

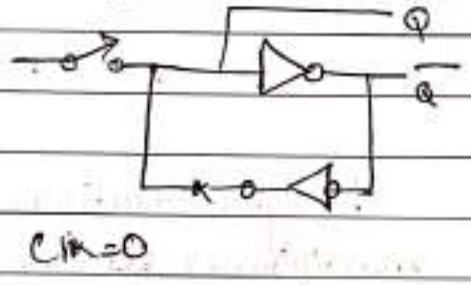


fig ⓓ

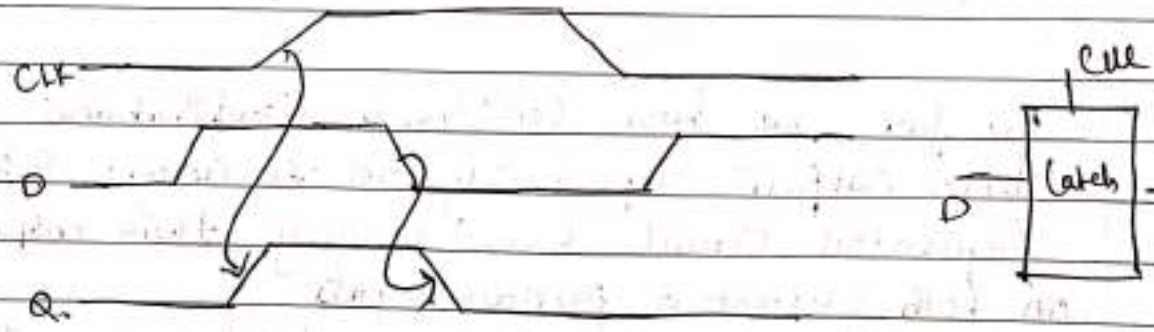


Fig :- CMOS positive-level sensitive D latch.

The multiplexer can be built from a pair of Transmission gates shown in figure ⓑ, because the inverters are restoring. This latch also produces a complementary output, $\overline{Q}$.

when CLK = 1, the latch is transparent & D flows through to Q, shown in fig ⓒ.

when CLK falls to 0, the latch becomes opaque. A feedback path around the inverter pair is established

as shown in figure (d), to hold the current state of 'Q' indefinitely.

The D-latch is also known as a level-sensitive latch because the state of the o/p is dependent on the level of the clock signal, as shown in fig (e).

Flip-Flops:-

By combining two level-sensitive latches, one negative-sensitive and one-positive-sensitive, we construct the edge-triggered flip-flop shown in figure below. The first stage is called the master and the second latch is called the slave.

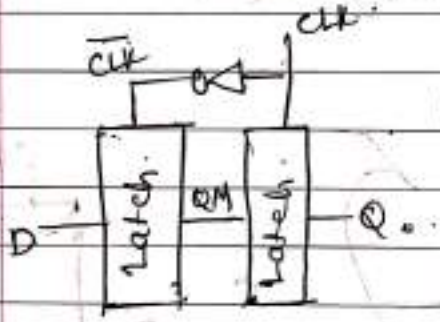


fig (a)

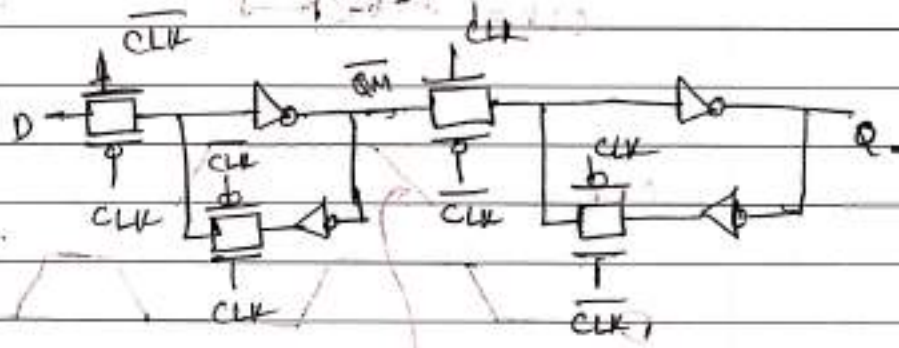


fig (b)

When the CLK is low, the master negative-level-sensitive latch ~~holds~~ output ($\overline{QM}$) follows the D input while the slave positive-level-sensitive latch holds the previous value shown in figure (c) below.

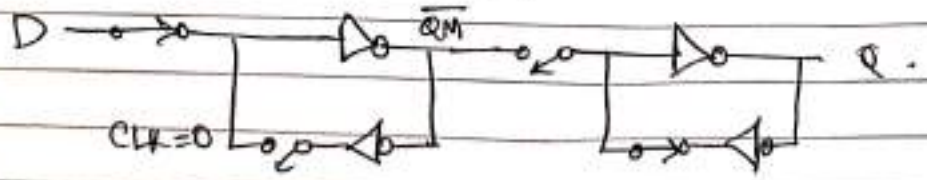
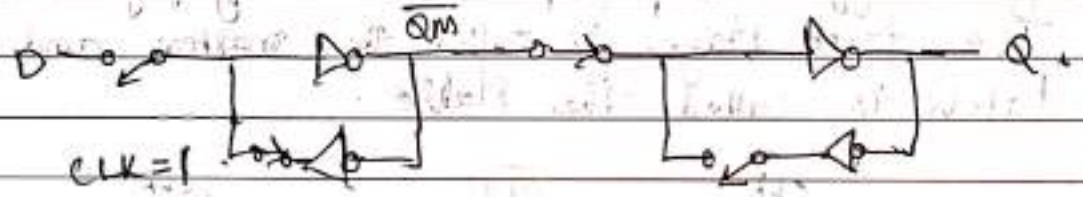


fig (c)

When the Clock transitions from 0 to 1, the master latch becomes opaque and holds the D value at the time of the clock transition.

The Slave latch becomes transparent, passing the stored master value ($\overline{QM}$) to the output of the slave latch (Q).

The D input is blocked from affecting the O/P. because the master is disconnected from the D-input. as shown in fig (d). when the Clock transitions from 1 to 0, the Slave latch holds its value & the master starts sampling the input again.



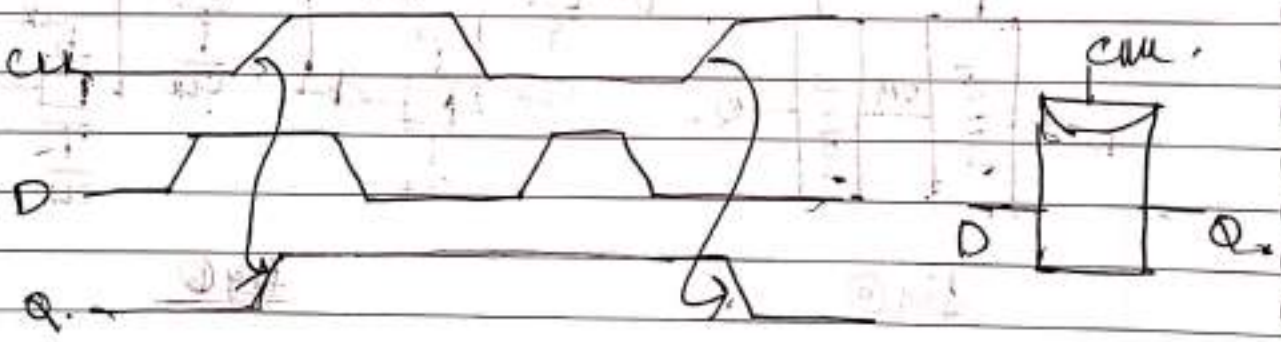


fig:- CMOS positive-Edge triggered D flipflop

Module - 3

Part-1 : Delay

Definitions:-

(1) Propagation delay time (t_{pd}):-

Maximum time from the input crossing 50% to the o/p crossing 50%.

(2) Contamination delay time (t_{cd}):-

Minimum time from the input crossing 50% to the o/p crossing 50%.

(3) Rise time, t_r :-

Time for a waveform to rise from 20% to 80% of its steady-state value.

(4) Fall time, t_f :-

Time for a waveform to fall from 80% to 20% of its steady-state value.

(5) Edge rate, $t_{rf} = (t_r + t_f)/2$.

(6) Slack :- Slack is the difference b/w the required & arrival times.

Two types of slack

(a) positive slack - means that the circuit meets timing.

(b) Negative slack - means that the circuit is not fast Enough.

Timing optimization:-

In most designs there will be a no of critical paths that limit the operating speed of the system & require attention to timing details.

The critical paths can be affected at four main levels:-

(1) The architectural / microarchitectural level.
(2) The logic level.
(3) The circuit level.
(4) The layout level.

(1) The Microarchitectural level:-

The most leverage is achieved with a good microarchitecture. This requires a broad knowledge of both the algorithms that implement the function & the technology being targeted, such as how many gate delays fit in a clock cycle, how quickly addition occurs, how fast memories are accessed, & how long signals take to propagate along a wire.

Trade-offs at the microarchitectural level include the number of pipeline stages, the number of execution units (parallelism), and the size of memories.

(2) Logic level:-

The next level of timing optimization comes at the logic level. Trade-offs include types of functional blocks (Eg ripple carry vs. lookahead adders), the no of stages of gates in the clock cycle, and the fan-in and fan-out of the gates.

(3) Circuit level:-

Once the logic has been selected, the delay can be tuned at the circuit level by choosing transistor sizes (or) using other styles of CMOS logic (4) Finally, delay is dependent on the layout. Good cell layouts can also reduce parasitic capacitance.

Transient Response:-

The most fundamental way to compute delay is to develop a physical model of the circuit of interest, write a differential equation describing the output voltage as a function of input voltage & time, and solve the equation. The solution of the differential equation is called the transient response, & the delay is the time when the o/p reaches $V_{DD}/2$.

The differential equation is based on charging (or) discharging of the capacitances in the circuit.

If capacitance C is charged with a current I, the voltage on the capacitor varies as:

$$\boxed{I = C \cdot \frac{dv}{dt}}$$

Every real circuit has some capacitance. In an integrated circuit, it typically consists of the, ~~gate capacitance of the load~~ the following capacitances,

(a) the gate capacitance of the load

(b) the diffusion capacitance of the driver's own transistors,

(c) wires that connect transistors together often contribute the majority of capacitances.

The transistor current depends on the input (gate) and output (source/drain) voltages.

To illustrate these points, consider computing the step response of an inverter.

Step response of an inverter:

* Figure (a) shows an inverter X1 driving another inverter X2 at the end of a wire.
* Suppose a voltage step from 0 to V_{DD} is applied to node A & we wish to compute the propagation delay, t_{pdf}, through X1, i.e, the delay from the input step until node B crosses $V_{DD}/2$.

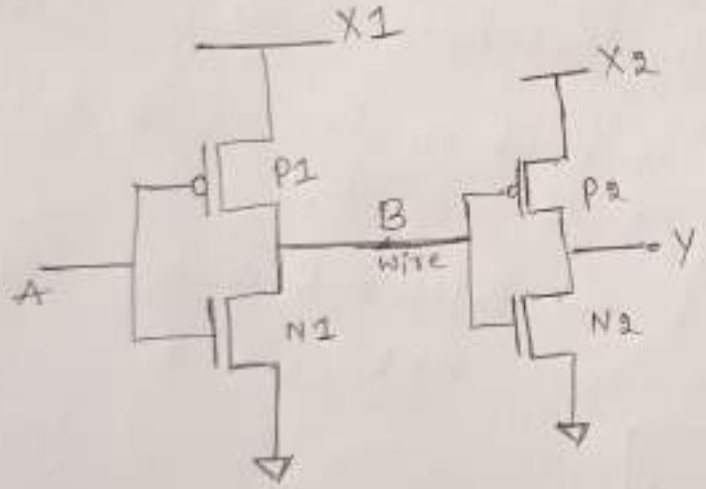


Fig (a).

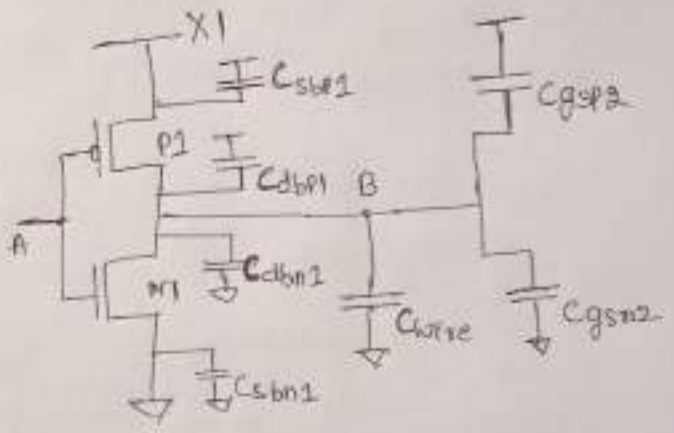


Fig (b)

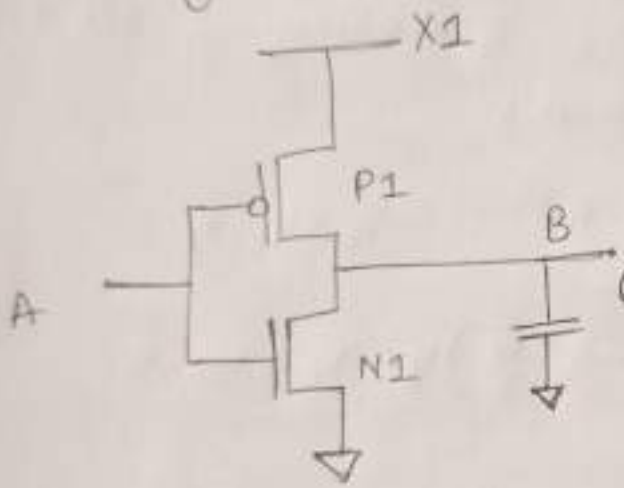
* These capacitances are annotated on Figure (b).

C_{db} and C_{sb} → These are diffusion capacitances b/w the drain & body of each transistor & b/w the source & body of each transistor.

C_{gs} → The gate capacitance C_{gs} of the transistors in X2 are part of the load.

C_{wire} → The wire capacitance is also part of the load.

* The gate capacitance of the transistors in X1 and the diffusion capacitance of the transistors in X2 do not matter because they do not connect to node B.
* The source-to-body capacitors C_{sbn1} & C_{sbp1} have both terminals tied to constant voltages & thus do not contribute to the switching capacitance.
* Figure (c) shows the equivalent-circuit diagram in which all the capacitances are lumped into a single C_{out}.



$$C_{out} = C_{dbn1} + C_{dbp1} + C_{wire} + C_{gsn2} + C_{gsp2}$$

Fig (c) :- Capacitances for Inverter delay Calculations

* Before the voltage is applied, A = 0, N1 is OFF
  P1 is ON.
  & B = V_{DD}.

* After the step, A = 1, N1 turns ON
  P1 turns OFF
  & B drops toward 0.

* The rate of change of the voltage V_B at node B depends on the O/P capacitance & on the current through N1:

$$C_{out}\frac{dV_B}{dt} = -I_{dsn1}$$

* The current depends on whether N1 is in the linear (or) Saturation regime.

The gate is at V_{DD},
The Source is at 0, &
The drain is at V_B.

Thus, $V_{gs} = V_{DD}$ & $V_{ds} = V_B$.

Initially, $V_{ds} = V_{DD} > V_{gs} - V_t$, so N1 is in Saturation.

As V_B falls below $V_{DD} - V_t$, N1 Enters the linear regime.

Substituting in current Eqn's & rearranging, we find the differential Equation,

$$\frac{dV_B}{dt} = \frac{-\beta}{C_{out}} \begin{cases} \frac{(V_{DD}-V_t)^2}{2} & V_B > V_{DD}-V_t \\ \left(V_{DD}-V_t-\frac{V_B}{2}\right)V_B & V_B < V_{DD}-V_t \end{cases}$$

RC Delay Model:-

RC delay models approximate the nonlinear transistor I-V and C-V Characteristics with an average resistance & Capacitance over the Switching range of the gate.

Effective Resistance:-

* The RC delay model treats a transistor as a Switch in Series with a resistor. The Effective resistance is the ratio of V_{ds} to I_{ds} averaged across the Switching interval of interest.
* A unit nMOS transistor is defined to have Effective resistance R.

* The size of the unit transistor refers to a transistor with minimum length and minimum contacted diffusion width (i.e, 4λ).

* An nmos transistor of K times unit width has resistance R/K because it delivers K times as much current.

* A unit pMOS transistor has greater resistance, generally in the range of 2R - 3R, because of its lower mobility.

* The resistance of two transistors in series is the sum of the resistances of each transistor.

Gate & Diffusion capacitance:

Each transistor also has gate & diffusion capacitance. A transistor of K times unit width has capacitance KC. 'C' is the gate capacitance of a unit transistor of either flavor.

Diffusion capacitance depends on the size of the source/drain region. Wider transistors have proportionally greater diffusion capacitance.

Increasing channel length increases gate capacitance proportionally but does not affect diffusion capacitance.

Equivalent RC circuits:-

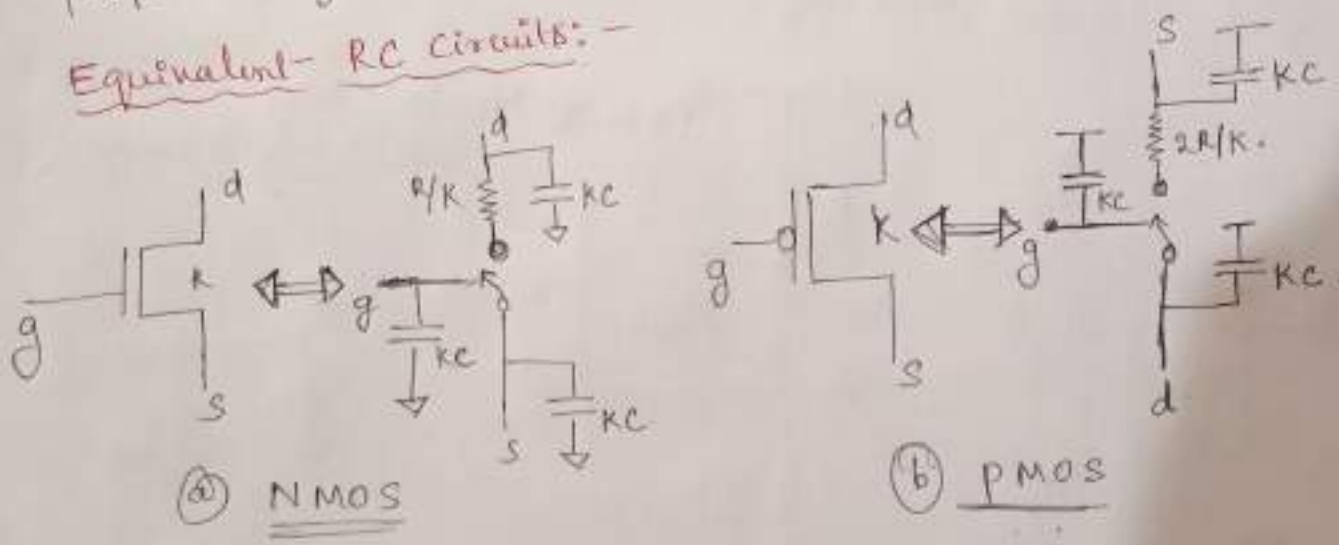


Figure :- Equivalent Circuits for transistors.

* Figure shows equivalent RC circuit models for nMOS and pMOS transistors of width K with contacted diffusion on both source & drain.

* The pMOS transistor has approximately twice the resistance of the nMOS transistor because holes have lower mobility than electrons.

* The pMOS capacitor are shown with V_{DD} as their second terminal because the n-well is usually tied high.

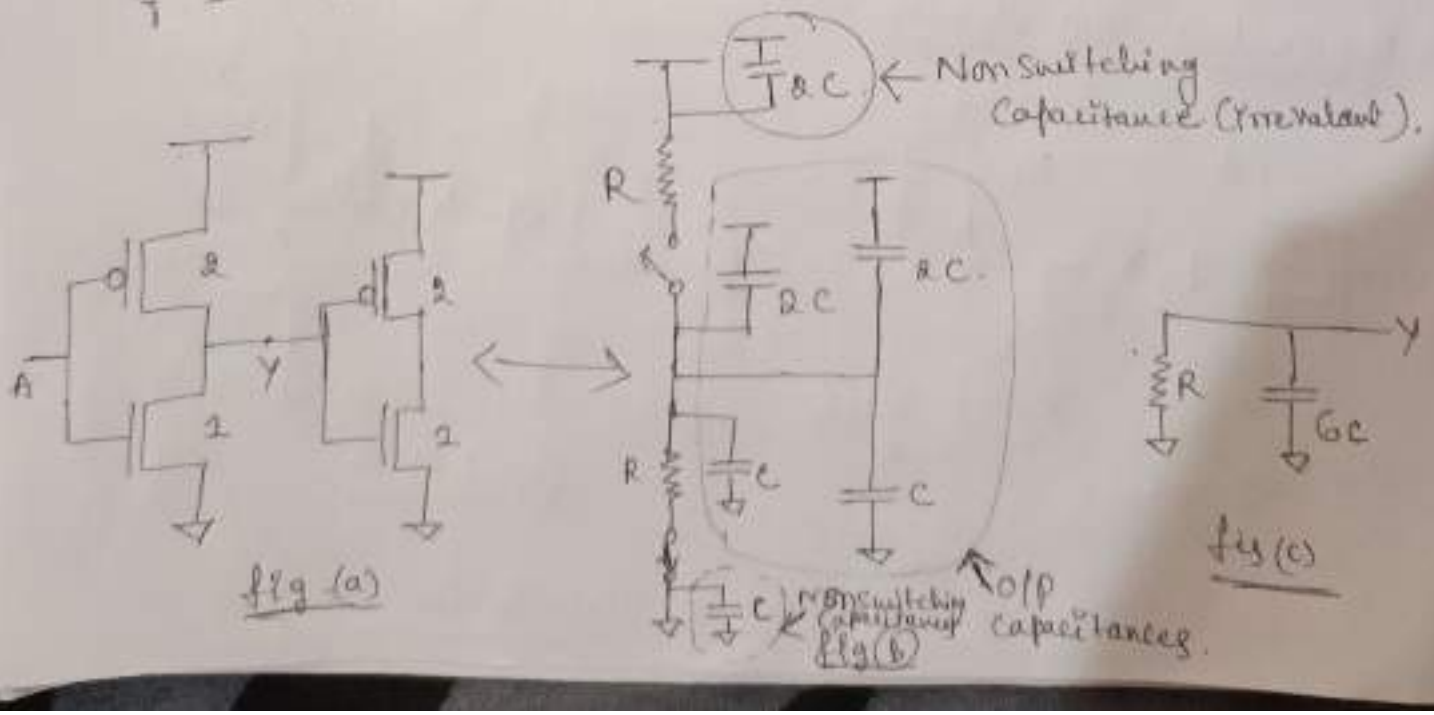
Note :- The behavior of the capacitor from a delay perspective is independent of the second terminal V_{tg} so long as it is constant. Hence we sometimes draw the second terminal as ground for convenience.

Equivalent circuits for logic gates :-

The equivalent circuits for logic gates are assembled from the individual transistors.

Example 1 :- Equivalent circuit for an inverter.

Figure ~~5(a)~~ (a) shows the equivalent circuit for a fanout-of-1 inverter with negligible wire capacitance.



* The unit inverters of Fig(a) are composed from an nMOS transistor of unit size & a pMOS transistor of twice unit width to achieve equal rise & fall resistance.
* Figure (b) gives an equivalent circuit, showing the first inverter driving the second inverter's gate.
* If the input A rises, the nMOS transistor will be ON & the pMOS OFF.
* Figure (c) illustrates this case with the switches removed. The capacitors shorted b/w two constant supplies are also removed because they are not charged (or) discharged.
* The total capacitance on the o/p Y is 6C.

Excersise problem:-

(1) Sketch a 3-input NAND gate with transistor widths chosen to achieve effective rise & fall resistance equal to that of a unit inverter (R). Annotate the gate with its gate & diffusion capacitances. Assume all diffusion nodes are contacted. Then sketch equivalent circuits for the falling o/p transition & for the worst-case rising o/p transition.

Soln:-

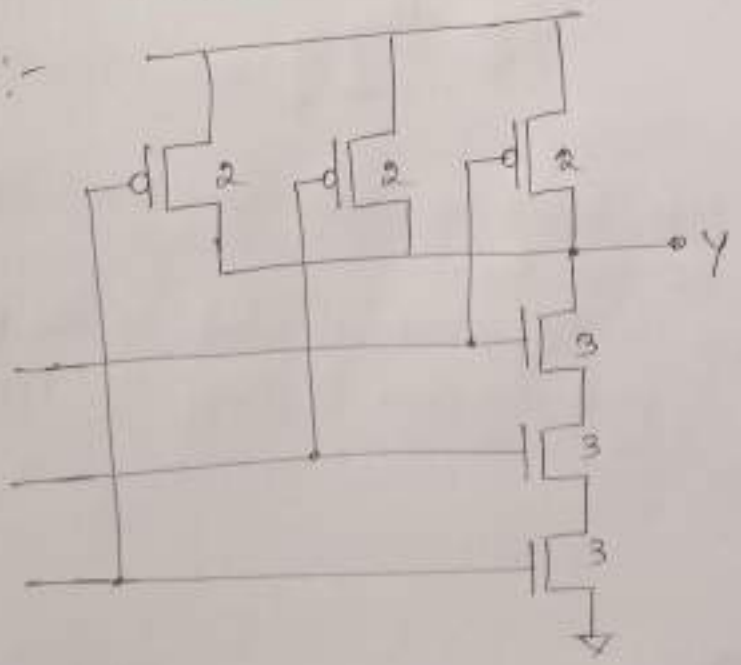


fig (a):- 3 input NAND gate.

* The three nMOS transistors are in series so the resistance is three times that of a single transistor.
∴ Each must be three times unit width to compensate.

* The two pMOS transistors are in parallel. In the worst case (with one of the inputs low), only one of the pMOS transistors is ON.
∴ Each must be twice unit width to have resistance R.

Figure (b) shows the capacitances.

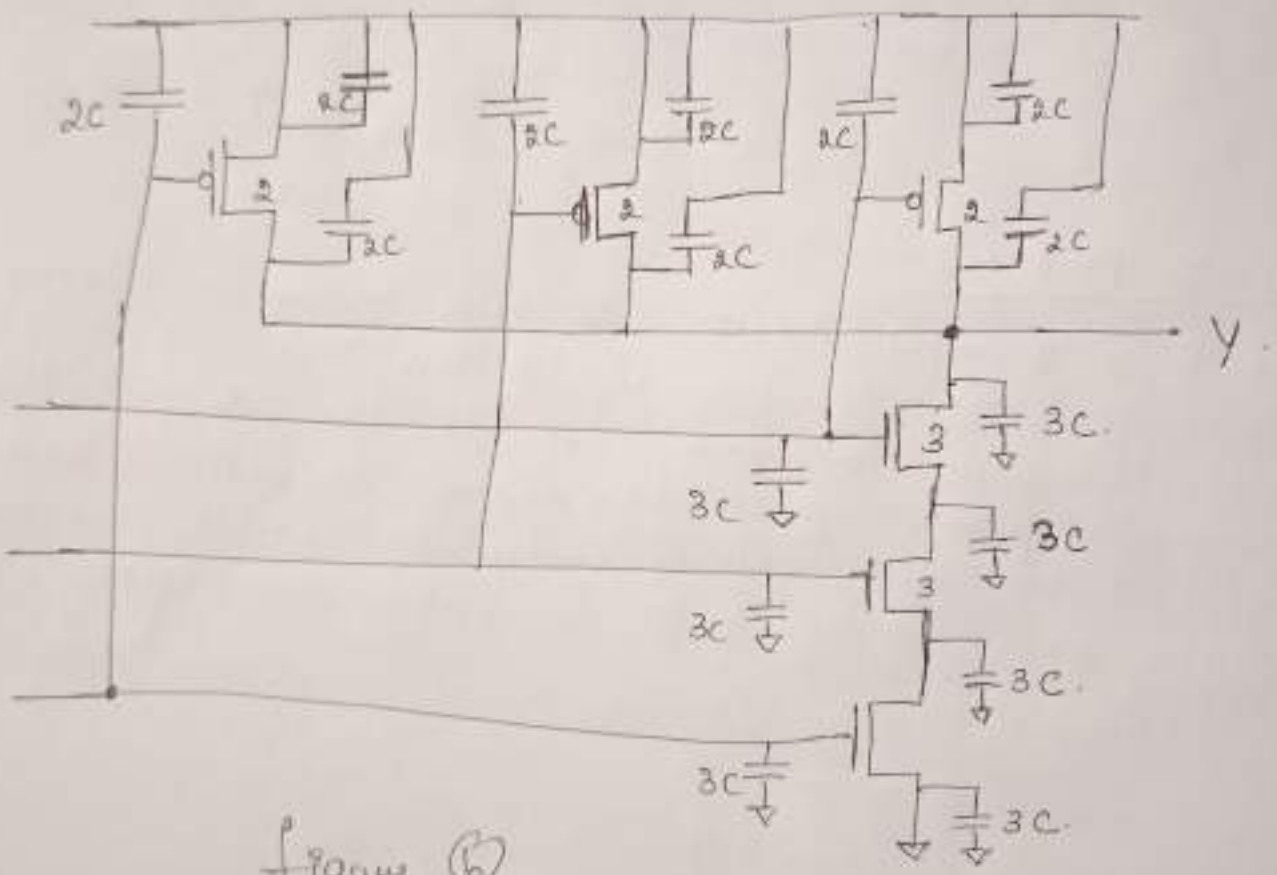


figure (b)

* Each input presents five units of gate capacitance to whatever circuit drives that i/p.
* The capacitors on source diffusions attached to the rails have both terminals shorted together so they are irrelevant to circuit operation.

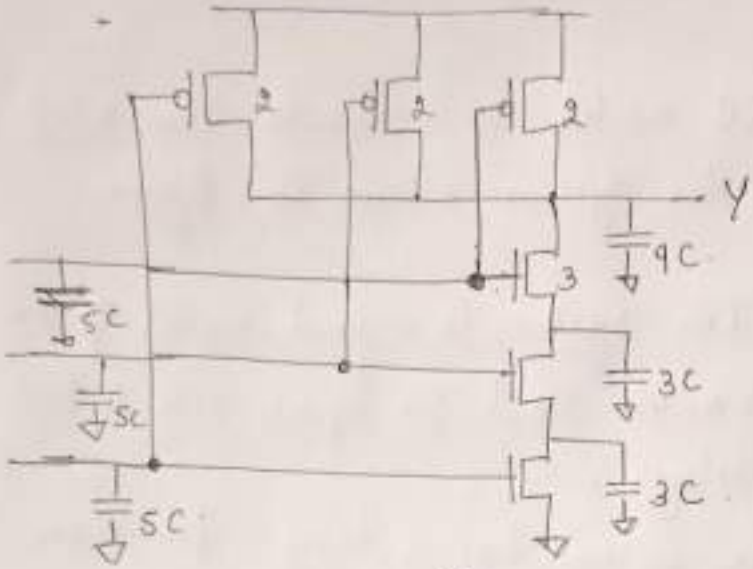
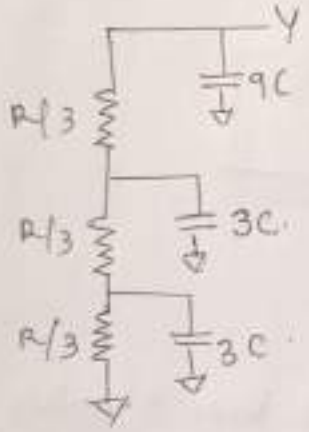


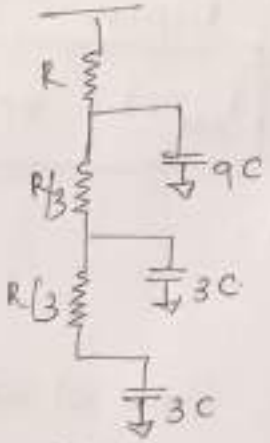
Figure (c)

Figure (c) redraws the gate with these capacitances deleted & the remaining capacitances lumped to ground.



Falling

Figure (d)



Rising

Figure (e)

Figure (d) shows the equivalent circuit for the falling o/p transition. The o/p pulls down through the three series nMOS transistors.

Figure (e) shows the equivalent circuit for the rising o/p transition. In the worst case, the upper two i/p's are 1 and the bottom one falls to 0.

The output pulls up through a single pMOS transistor. The upper two nMOS transistors are still on, so the diffusion capacitance b/w the series nMOS transistors must also be discharged.

Transient Response:-

Consider Applying the RC model to Estimate the Step response of the first-order System shown in figure.

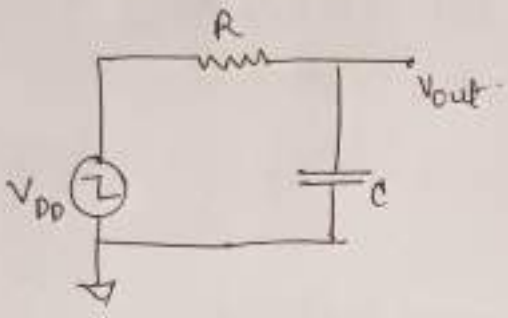


fig:- First order RC System

This System is a good model of an inverter Sized for Equal rise & fall delays.

The System has a transfer function

$$\boxed{H(s) = \frac{1}{1+sRC}}$$

& a Step response

$$\boxed{V_{out}(t) = V_{DD} \cdot e^{-t/\tau}}$$

where $\tau = RC$.

$$\boxed{t_{pd} = RC \ln 2}$$

The propagation delay is the time at which Vout reaches $V_{DD}/2$, as Shown in Fig.

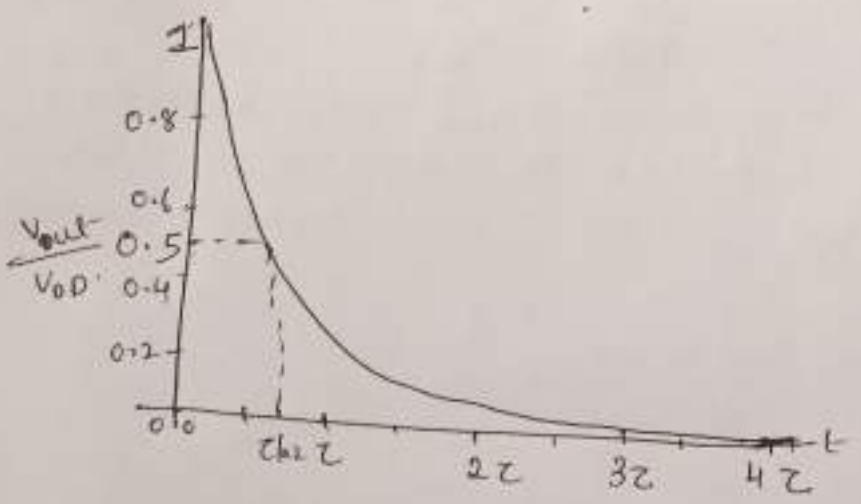


fig:- First order Step response

Figure below shows a second-order system. R_1 and R_2 might model the two series nMOS transistors in a NAND gate (or) an inverter driving a long wire with non-negligible resistance.

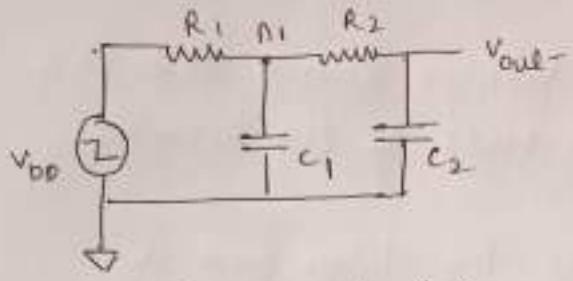


Fig :- Second order RC System.

The transfer function is

$$H(s) = \frac{1}{1 + [R_1C_1 + (R_1+R_2)C_2] + s^2 R_1C_1R_2C_2}$$

The function has two real poles & the step response is

$$V_{out}(t) = V_{DD} \cdot \frac{\tau_1 \cdot e^{-t/\tau_1} - \tau_2 \cdot e^{-t/\tau_2}}{\tau_1 - \tau_2}$$

with $$\tau_{1,2} = \frac{R_1C_1 + (R_1+R_2)C_2}{2}\left[1 \pm \sqrt{1 - \frac{4R^* C^*}{[1+(1+R^*)C^*]^2}}\right]$$

where $R^* = \frac{R_2}{R_1}$, $C^* = \frac{C_2}{C_1}$.

$$\tau = \tau_1 + \tau_2 = R_1C_1 + (R_1+R_2)C_2.$$

This approximation works best when one time constant is significantly bigger than the other.

Elmore Delay:-

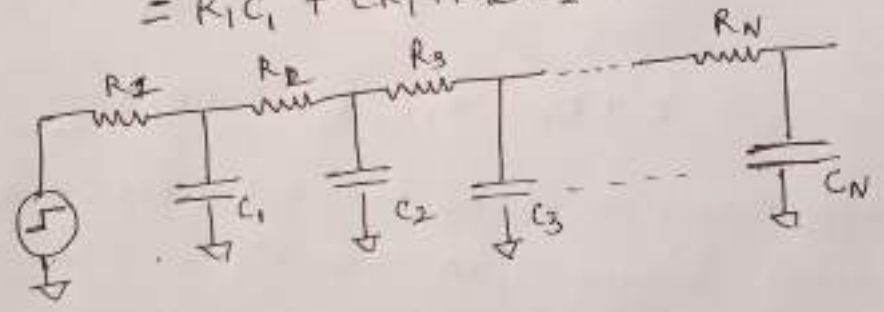
* In general, most circuits of interest can be represented as an RC tree, i.e an RC circuit with no loops.

* The root of the tree is the Voltage Source and the leaves are the capacitors at the ends of the branches.

* The Elmore delay model estimates the delay from a source switching to one of the leaf nodes changing as the sum over each node i of the capacitance C_i on the node, multiplied by the effective resistance R_{is} on the shared path from the source to the node & the leaf.

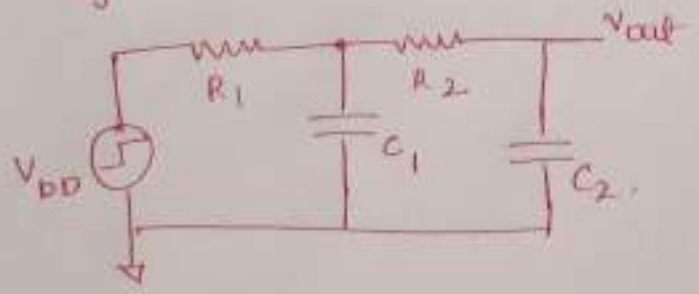
i.e $$\boxed{t_{pd} = \sum_{\text{nodes } i} R_{i\text{-to-source}} \cdot C_i}$$

$$= R_1C_1 + (R_1+R_2)C_2 + \dots + (R_1+R_2+\dots+R_N)C_N.$$

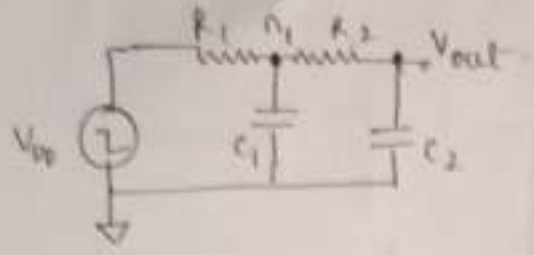


Example:-

① Compute the Elmore delay for Vout in 2nd order RC System as shown below:-



Soln:- The circuit has a source & two nodes.



At node n_1,

The capacitance is C_1 & the resistance to the source is R_1.

At node V_{out},

The capacitance is C_2 & the resistance to the source is (R_1+R_2).

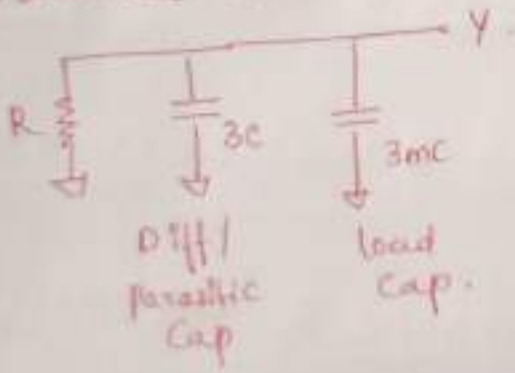
The Elmore delay is, $t_{pd} = \sum_i R_{is} C_i$

$= R_{1s} C_1 + R_{2s} C_2$.

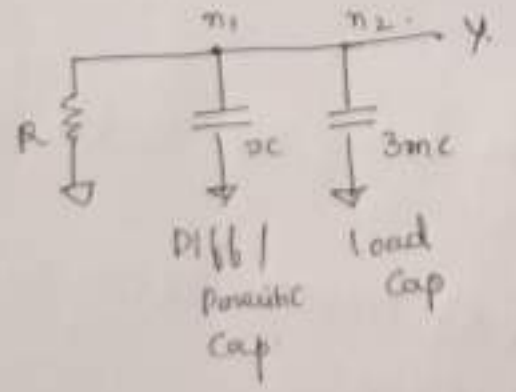
$t_{pd} = R_1 \cdot C_1 + (R_1+R_2) C_2$.

where R_{1s} is the ~~the~~ resistance from node 1 to source &, R_1.

(2) Estimate t_{pd} for a unit-inverter driving m identical unit inverters.



Soln:-



The Elmore delay is,

$t_{pd} = R \cdot 3C + R \cdot 3mC$.

$t_{pd} = (3+3m) RC$.

③ Estimate t_{pdf} and t_{pdr} for the 3-i/p NAND gate, the output is loaded with h identical NAND gates.

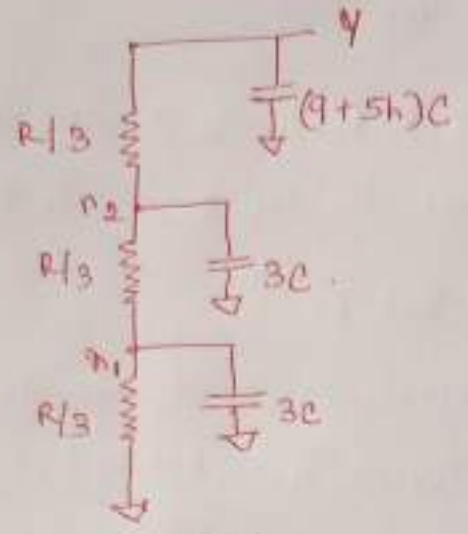


fig ⓐ

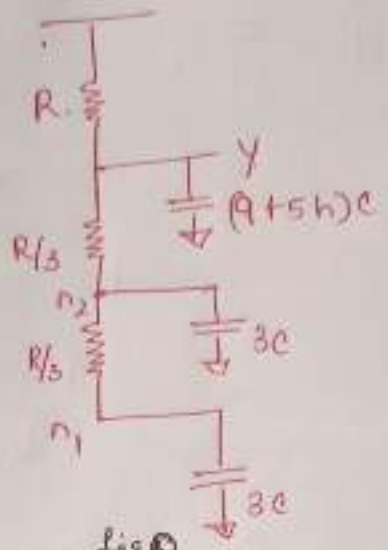


fig ⓑ

Soln :- Each NAND gate load presents 5 units of capacitance on a given input.

For falling ~~transition~~ output:

The circuit has three nodes.

Node n_1 has capacitance 3C & resistance of R/3 to ground.

Node n_2 has capacitance 3C & resistance (R/3 + R/3 + R/3) to ground.

Node Y has capacitance (9+5h) C & resistance (R/3 + R/3 + R/3) to ground.

The Elmore delay for the falling o/p is the sum of these RC products.

$$t_{pdf} = (3C)(R/3) + (3C)(R/3 + R/3) + ((9+5h)C)(R/3 + R/3 + R/3)$$

$$= (12 + 5h)RC.$$

For Rising output-

Fig (b) :- Equivalent circuit for the falling transistor.

- Y is pulled up to V_{DD} through a single pMOS transistor.

* Node Y has capacitance $(9+5h)C$ & Resistance R to the V_{DD} supply.

* Node n_2 has capacitance 3C. The Relevant resistance is only R, not $(R+R/3)$, because the o/p is being charged only through R.

* Node n_1 has capacitance 3C & Resistance R.

∴ The Elmore delay for the rising o/p is

$$t_{pdr} = (15+5h)RC.$$

Module- 3.

PART-2: Combinational Circuit Design.

Introduction:-

Digital logic is divided into combinational & sequential circuits. Combinational circuits are those whose outputs depend only on the present inputs, while sequential circuits have memory.

The Building blocks for combinational circuits are logic gates, while the building blocks for sequential circuits are registers & latches.

In this chapter we examine techniques to optimize combinational circuits for lower delay and/or energy.

Static CMOS - majority of circuits use static CMOS because it is robust, fast, energy efficient & easy to design.

Certain circuits have particularly stringent speed, power, (or) density restrictions that force another solution. Such alternative CMOS logic configurations are called circuit families.

The Most commonly used alternative circuit families are,

(1) Ratioed circuits.

(2) Dynamic circuits &

(3) pass-transistor circuits.

Circuit Families:-

(1) Static CMOS:-

Static CMOS circuits with complementary nMOS pulldown & pMOS pullup networks are used for the vast majoritrity of logic gates in integrated circuits.

Designers accustomed to AND and OR functions must learn to think in terms of NAND and NOR to take advantage of Static CMOS.

(a) Bubble pushing:-

In manual circuit design, the AND & OR functions are represented in terms of NAND & NOR through bubble pushing.

CMOS stages are inherently inverting, so AND and OR functions must be built from NAND & NOR gates.

DeMorgan's Law helps with this conversion:

$$\overline{A \cdot B} = \overline{A} + \overline{B}.$$

$$\overline{A + B} = \overline{A} \cdot \overline{B}.$$

These relations are illustrated graphically in Figure below.

A NAND gate is equivalent to an OR of inverted inputs.

A NOR gate is equivalent to an AND of inverted inputs.

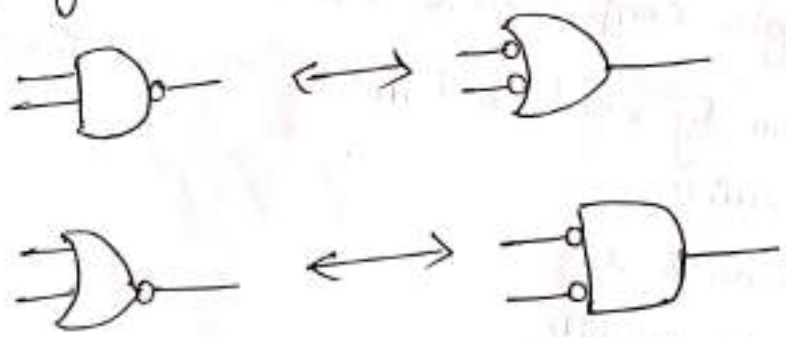


Figure :- Bubble pushing with De Morgan's Law.

The same relationship applies to gates with more inputs. Switching between these representations is easy to do on a white board and is often called bubble pushing.

Example :-

① Design a circuit to compute $F = AB + CD$ using NAND's and NOR's.

Soln:-

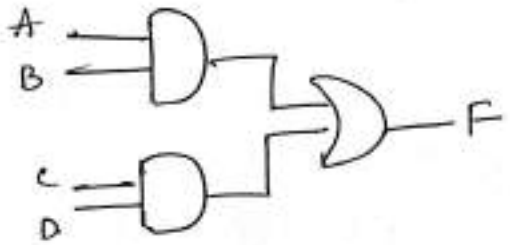


fig ⓐ

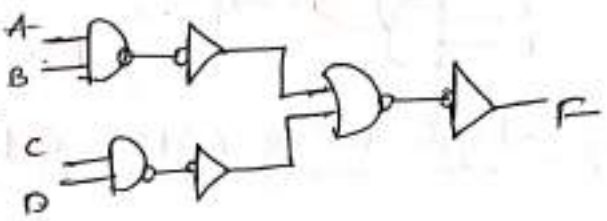


fig ⓑ

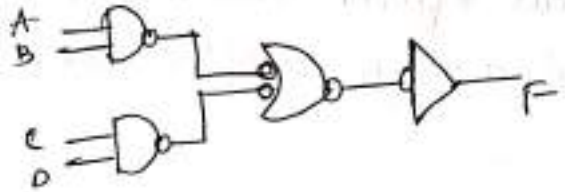


fig ⓒ

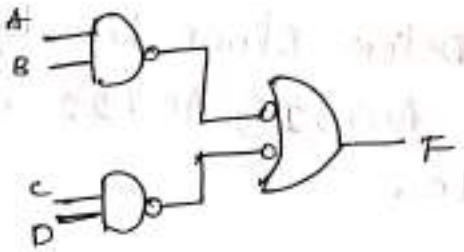
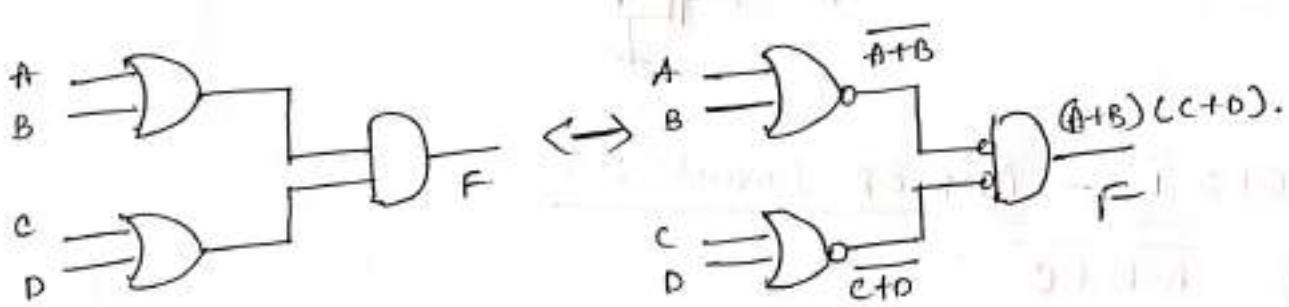


fig ⓓ

Figure:- Bubble pushing to convert AND's & OR's to NAND's & NOR's.

② Design a circuit to compute $F = (A+B)(C+D)$



(b) Compound Gates:-

Compound gates are particularly useful to perform complex functions with relatively low logical efforts.

Static CMOS also effeciently handles compound gates computing various inverting combinations of AND/OR functions in a single stage.

The Function $F = AB + CD$ can be computed with an AND-OR-INVERT-22 (AOI22) gate & an inverter, as shown in Fig.

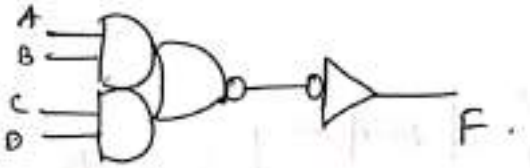


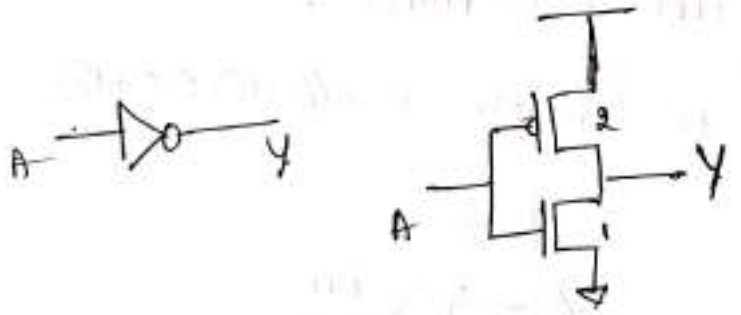
Fig:- Logic using AOI22 gate.

In General, logical Effort of Compound gates can be different for different inputs.

Figure below shows how logical efforts can be Estimated for the AOI21, AOI22 and a more complex Compound AOI gate.

Unit Inverter

$Y = \overline{A}$

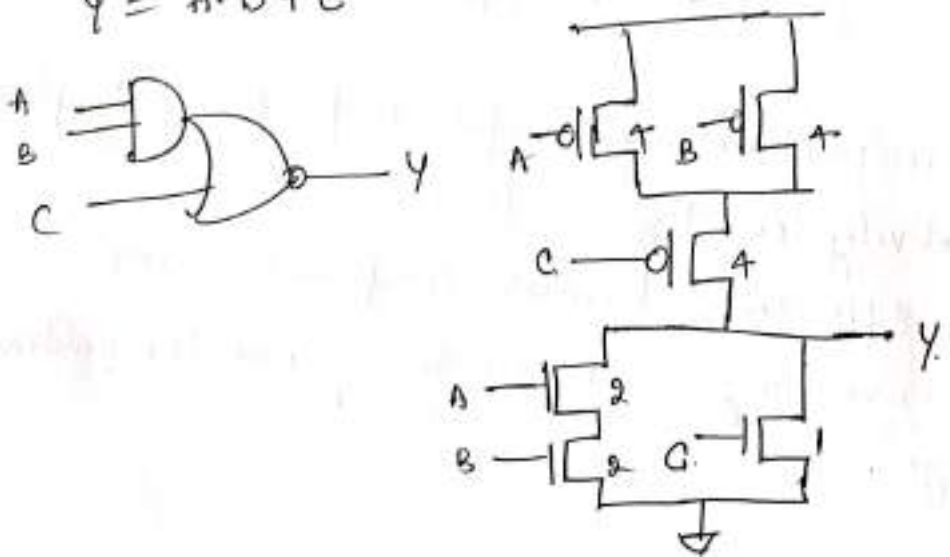


$g_A = 3/3.$

$p = 3/3.$

AOI21 - AND OR Invert 21

$Y = \overline{A \cdot B + C}$



$g_A = 6/3$

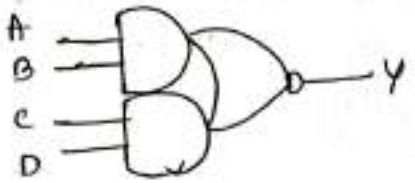
$g_B = 6/3$

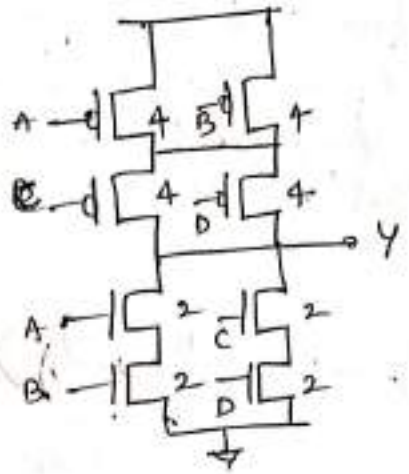
$g_C = 5/3$

$p = 7/3.$

AOI22

$Y = \overline{A \cdot B + C \cdot D}$.





$g_A = 6/3$

$g_B = 6/3$

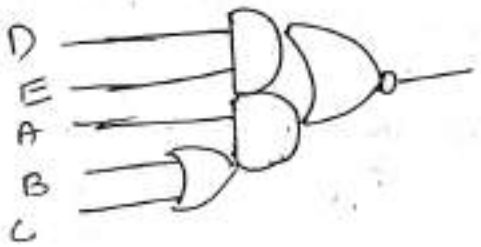
$g_C = 6/3$

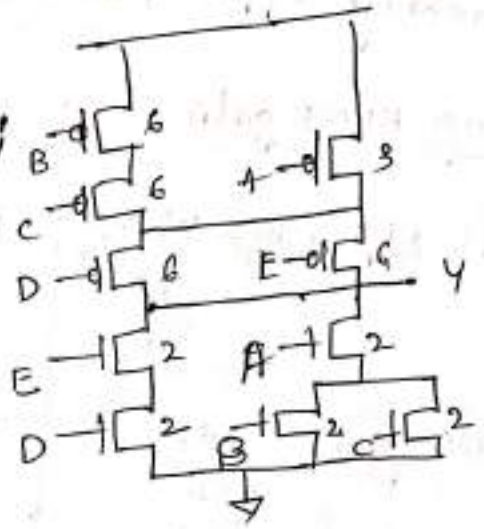
$g_D = 6/3$

$P = 12/3$.

Complex AOI.

$Y = \overline{A \cdot (B+C) + D \cdot E}$.





$g_A = 5/3$

$g_B = 8/3$

$g_C = 8/3$

$g_D = 8/3$

$g_E = 8/3$

$P = 16/3$

NOTE :-

* The logical effort (g) of each input is the ratio of the input capacitance of that input to the input capacitance of the inverter.
* The parasitic delay (P) is estimated from the total diffusion capacitance on the o/p node by summing the sizes of the transistors attached to the o/p.

Example Problem:-

① Calculate the minimum delay, in, τ, to compute $F = AB + CD$ using the circuits ~~fig~~ shown in figures. Each input can present a maximum of 20λ of transistor width. The output must drive a load equivalent to 100λ of transistor width. Choose transistor sizes to achieve this delay.

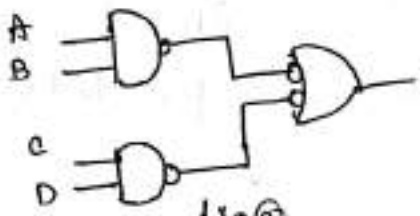


fig ⓐ

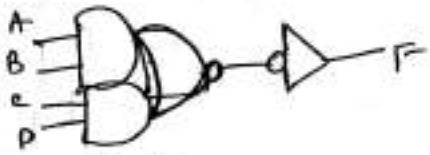


fig ⓑ

Soln:- Path Electrical Effort, $H = \frac{C_{out}}{C_{in}} = \frac{100}{20} = \underline{\underline{5}}$

Branching Effort, $\underline{\underline{B = 1}}$.

Design using NAND gates

Path logical Effor, $G = 4/3 \times \frac{4}{3}$
$= \underline{\underline{\frac{16}{9}}}$

Parasitic delay, $P = 2+2$
$= \underline{\underline{4}}$

Path Efforts, $F = G.B.H.$
$= \frac{16}{9} \times 1 \times 5$
$= \underline{\underline{80/9}}$

The path delays, $D = N.F^{1/N} + P$
$\Rightarrow N = 2$ stages.
$D = 2 \times \left(\frac{80}{9}\right)^{1/2} + 4.$
$= \underline{\underline{10.0\tau}}$

Desing using the AOI22 & INV.

Path logical Effor, $G = (6/3) \times 1$
$= \underline{\underline{2}}$

$P = 12/3 + 1$
$= \underline{\underline{5}}$

Path Effort, $F = GBH$
$= 2 \times 1 \times 5$
$= \underline{\underline{10}}$

Path delay, $D = \underline{\underline{11.3\tau}}$

From the results, it can be seen that simple 2-input NAND gates can be quite fast.

To compute the sizes, we determine the best stage efforts, $\hat{f} = F^{1/N} = 3.0$ for circuit (a) & $\hat{f} = 3.2$ for circuit (b).

The input capacitance can be determine by using capacitance transformation,

$$C_{in_i} = \frac{C_{out_i} \times g_i}{\hat{f}}$$

For the NAND design,

$$C_{in} = \frac{100\lambda \times (4/3)}{3.0} = 44\lambda$$

For the AOI22 design,

$$C_{in} = \frac{100\lambda \times (1)}{3.2} = 31\lambda$$

Asymmetric Gates:-

When one input is far less critical than another, even nominally symmetric gates can be made asymmetric to favor the late i/p at the expense of the early one.

In series n/w → connecting early i/p to the outer transistor & making the transistor wider

In parallel n/w → The early input is connected to a narrower transistor to reduce the parasitic capacitance.

Example :-

Consider the path shown in Fig (a) below.

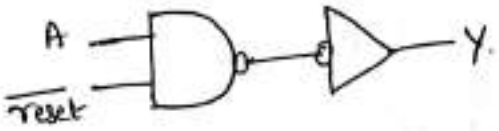


fig (a).

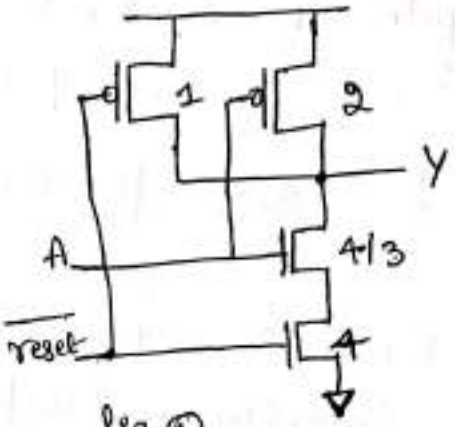


fig (b).

Fig :- Resettable buffer optimized for data input.

⇒ Under ordinary conditions, the path acts as a buffer b/w A & Y. When reset is asserted, the path forces to o/p low.

⇒ If reset only occurs under exceptional circumstances & can take place slowly, the circuit should be optimized for i/p to o/p delay at the expense of reset. This can be done with the asymmetric NAND gate as shown in fig (b).

Skewed Gates :-

Definition :- Logical Effort of a skewed gate for a particular transition is the ratio of the ~~output~~ input capacitance of that gate to the input capacitance of an unskewed inverter delivering the same o/p current for the same transition.

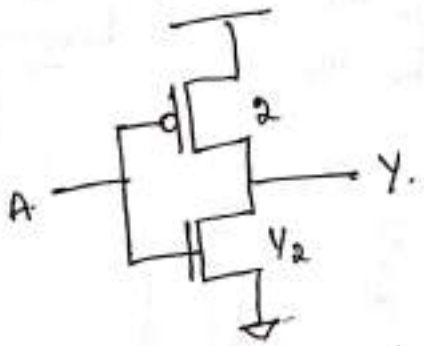
* Skewed gates favor one transition over another.
* Hi-Skew gates favor rising o/p transition (small nMOS).
* Lo-Skew gates favor falling o/p transition (small pMOS).
* This favoring can be done by decreasing the size of the non-critical transistor.

* The logical Effort is smaller for favored direction but larger for the other direction.

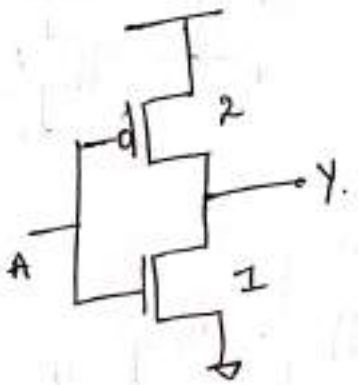
* The logical Efforts for the rising (up) → g_u & The logical Efforts for the falling (down) → g_d.

<u>Example</u>:- Suppose rising o/p of inverter is most-critical. Downsize non critical nMOS transistor.

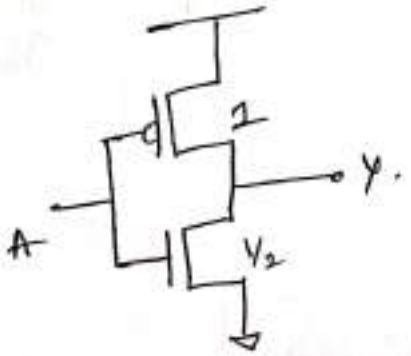
Hi-Skew Inverter



unskewed inverter (Equal rise resistance).



unskewed inverter (Equal fall resistance).



<u>Fig</u>:- Logical Effort calculation for Hi-Skew inverter.

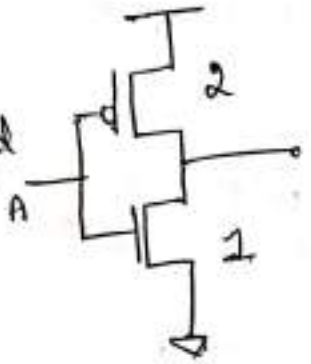
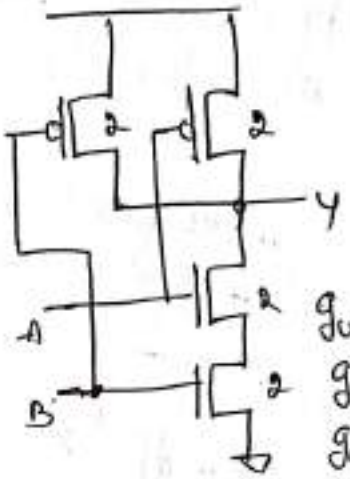
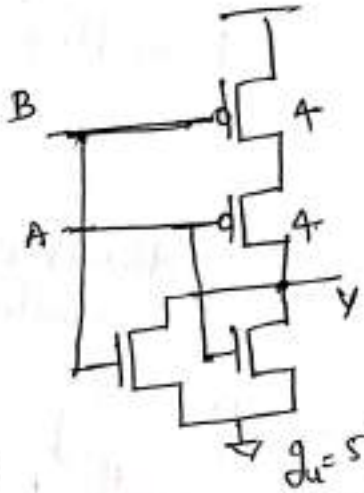
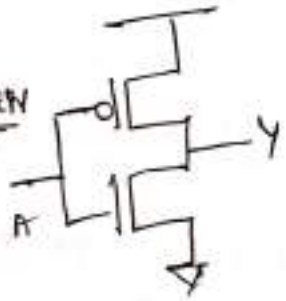
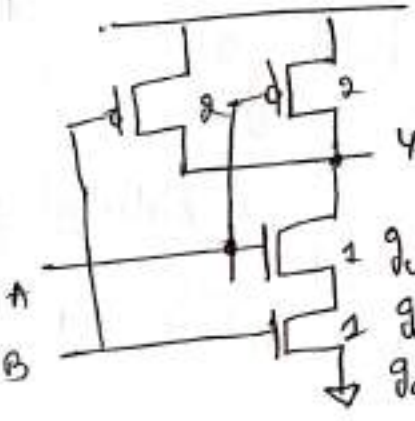
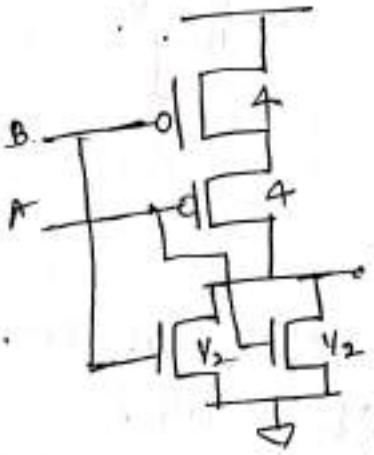
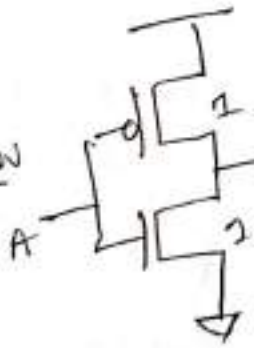
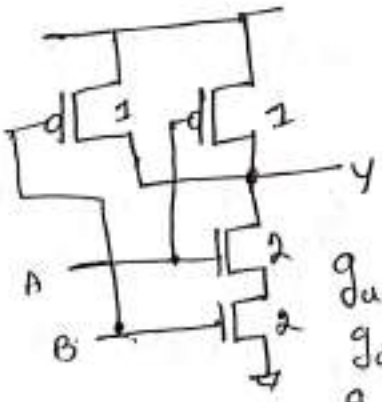
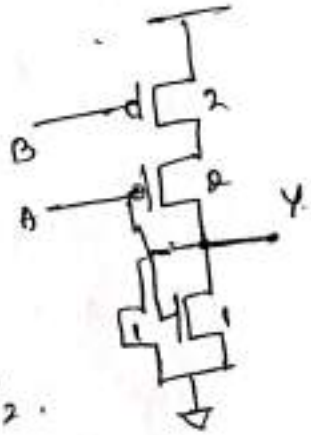
⇒ Hi-Skew inverter is constructed by downsizing the nMOS transistor.

⇒ $g_u = 2.5/3 = 5/6$.

⇒ $g_d = 2.5/1.5 = 5/3$.

⇒ The degree of skewing (eg., the ratio of Effective resistance for the fast transition relative to the slow transition) impacts the logical Efforts & noise margins; a factor of two is common.

Figures below Catalogs Hi-skew & Lo-skew gates with a skew factor of two.

	Inverter	NAND2	NOR2
Unskewed	 $g_u = 1$, $g_d = 1$, $g_{avg} = 1$	 $g_u = 4/3$, $g_d = 4/3$, $g_{avg} = 4/3$	 $g_u = 5/3$, $g_d = 5/3$, $g_{avg} = 5/3$
Hi-skew		 $g_u = 1$, $g_d = 2$, $g_{avg} = 3/2$.	 $g_u = 3/2$, $g_d = 3$, $g_{avg} = 9/4$
Lo-skew	 $g_u = 4/3$, $g_d = 2/3$, $g_{avg} = 2$.	 $g_u = 2$, $g_d = 1$, $g_{avg} = 3/2$.	

P/N Ratios :-

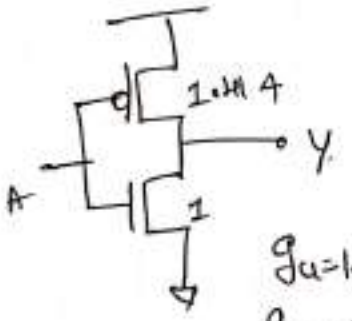
In general, best P/N ratio is sqrt of Equal delay ratio.

⇒ only increases average delay slightly for inverters.

⇒ But significantly decreases area & power.

Inverter:

Fastest P/N Ratio

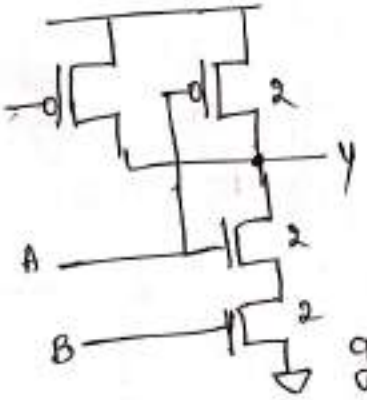


$g_u = 1.44$

$g_d = 0.80$

$g_{avg} = 0.97$.

NAND 2:

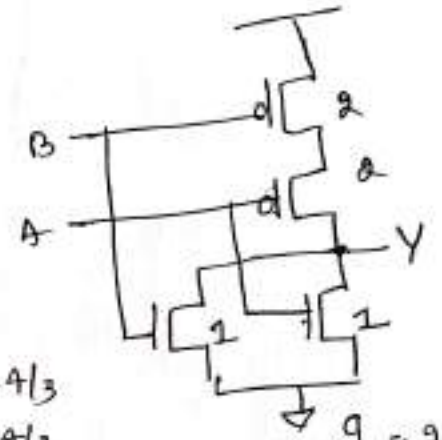


$g_u = 4/3$

$g_d = 4/3$

$g_{avg} = 4/3$

NOR 2:-



$g_u = 2$

$g_d = 1$

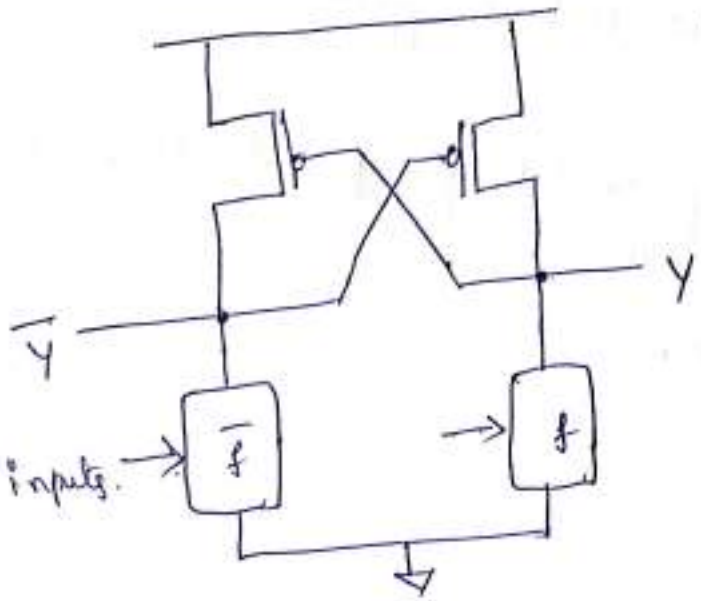
$g_{avg} = 3/2$

Fig :- Gates with P/N ratios giving least delay.

Cascode Voltage Switch Logic :- (CVSL) :-

* CVSL is a gate that performs like a pseudo-nmos without the static power consumption.
* It requires mainly N-channel MOSFET transistors to implement the logic using true and complementary input signals.
* Also it needs two p-channel transistors at the top to pull one of the output high.
* It uses both true and complementary input signals & computes both true and complentary outputs using a pair of nMOS pull down networks.
* CVSL has a good speed advantage because all of the logic is performed with nMOS transistors. Thus reducing the input capacitance.
* It also has very good noise margin.

The figure below shows the general schematic of a CVSL.



* The pulldown network f implements the logic function as in a static CMOS gate, while $\bar{f}$ uses inverted inputs feeding transistors arranged in the conduction complement.

fig:- schematic of CVSL.

Operation:-

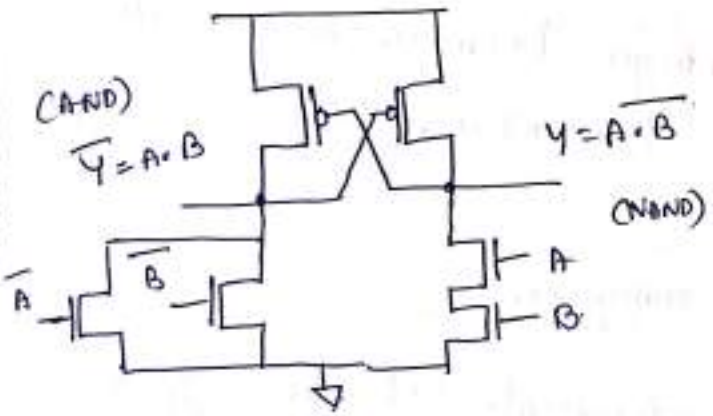
For any given input pattern, one of the pulldown networks will be ON and the other OFF.

The pulldown network that is ON will put that output low. This low output turns ON the PMOS transistor to pull the opposite output high.

When the opposite o/p rises, the other PMOS transistor turns OFF so no static power dissipation occurs.

Example:-

AND/NAND CVSL gate:-



* use pair of Complementary i/p's.
* Apply Demorgan's Law
$\overline{A \cdot B} = \overline{A} + \overline{B}$.

Disadvantages of CVSL:-

* It requires both the low & high going transitions, adding more delay.
* Contention Current during switching period increases. Power Consumption.
* It requires more area.

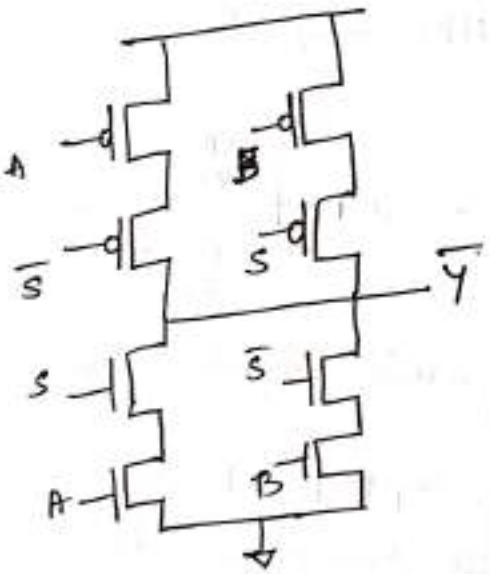
Pass-Transistor Circuits:-

In pass-transistor circuits, inputs are applied to the Source / drain diffusion terminals.

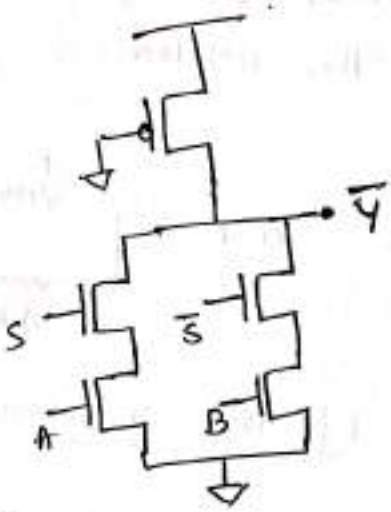
These circuits build switches using either nMOS pass transistors (or) parallel pairs of nMOS and pMOS transistors called transmission gates.

For the purpose of comparision, Figures below shows a 2-i/p Multiplexer Constructed in a wide variety of pass-transistor circuit families along with static CMOS, pseudo nMOS, CVSL, CPL, CMOS TG.

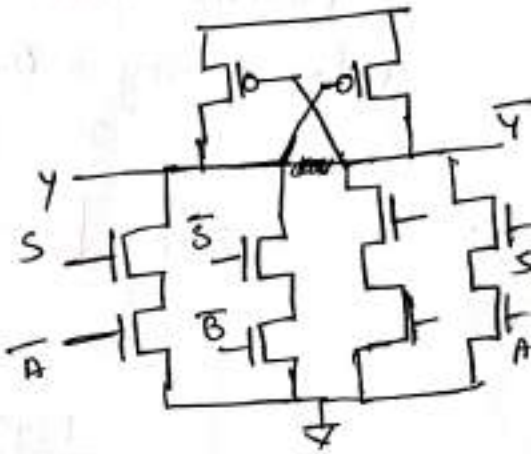
Static CMOS :-



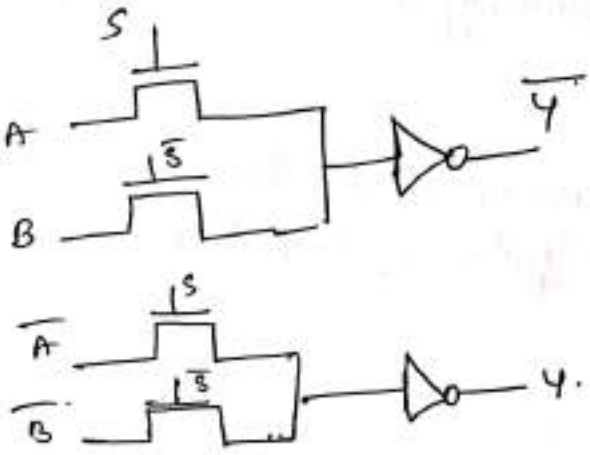
pseudo-nMOS.



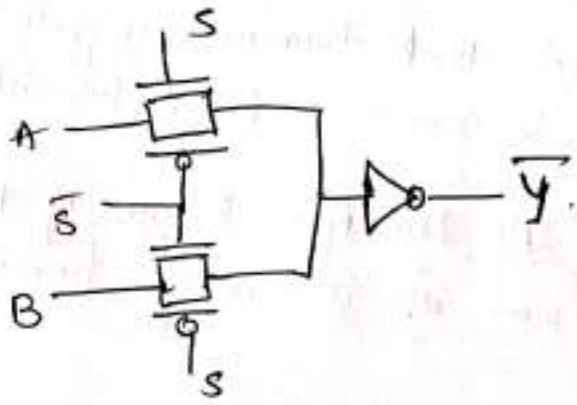
CVSL



CPL



CMOS TG.



(a) CMOS with Transmission Gates:-

A single nMOS (or) pMOS pass transistor suffers from a threshold drop. If used alone, additional circuitry may be needed to pull the output to the rail. Transmission gates solve this problem but require two transistors in parallel.

The Resistance of a unit-sized transmission gate can be estimated as R for the purpose of delay estimation. Current flows through the parallel combination of the nMOS & pMOS transistors. One of the transistors is passing the value well & the other is passing it poorly.

The effective resistance of a unit transistor passing a value in its poor direction as twice the usual value: 2R for nMOS & 4R for pMOS.

Figure below shows the parallel combination of resistances. When passing a 0, the resistance is $R \| 4R = (\frac{4}{5})R$.

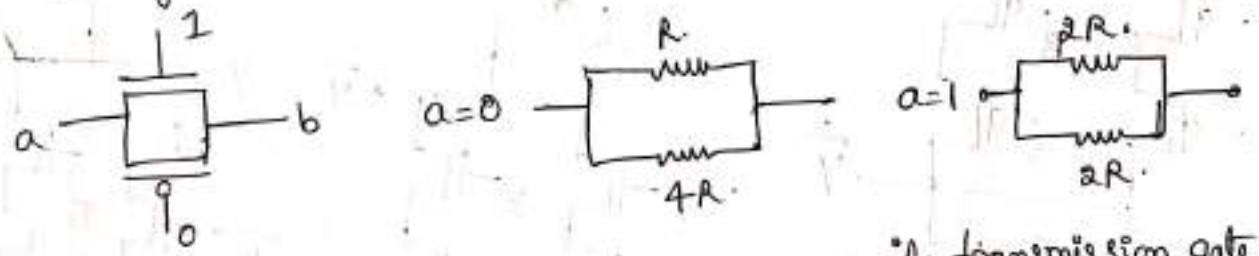


Fig:- Effective resistance of a unit transmission gate

The effective resistance passing a 1 is $2R \| 2R = R$.

Hence, a transmission gate made from unit transistors is approximately R in either direction.

Note that transmission gates are commonly built using equal-sized nMOS & pMOS transistors.

If multiple stages of logic are cascaded, they can be viewed as alternating transmission gates & inverters.

Figure (a) below redraws the multiplexer to include the inverters from the previous stage that drive the diffusion inputs but to exclude the output inverter.

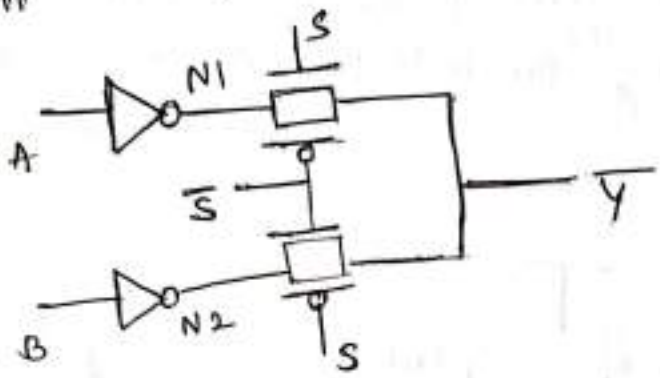


Figure (a)

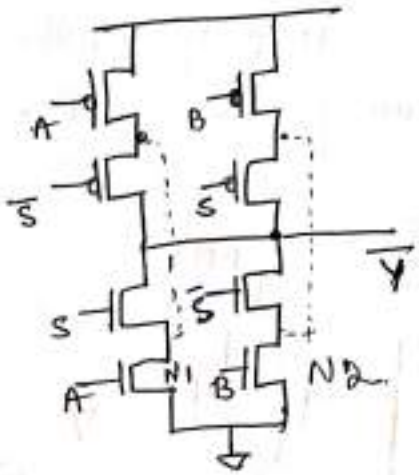


Figure (b)

Figure (b) shows the multiplexer drawn at the transmission level. This is identical to the static CMOS multiplexer except that the intermediate nodes in the pullup and pulldown networks are shorted together as N1 & N2.

The shorting of the intermediate nodes has two effects on delay.

(1) The effective resistance decreases somewhat because the O/P is pulled up (or down through the parallel combination of both pass transistors rather than through a single transistor.

(2) The effective capacitance increases slightly because of the extra diffusion and wire capacitance required for this shorting.

Tri-state inverter:-

Figure shows a similar transformation of a tristate inverter from transmission gate form to conventional static CMOS by anshorting the intermediate node & redrawing the gate.

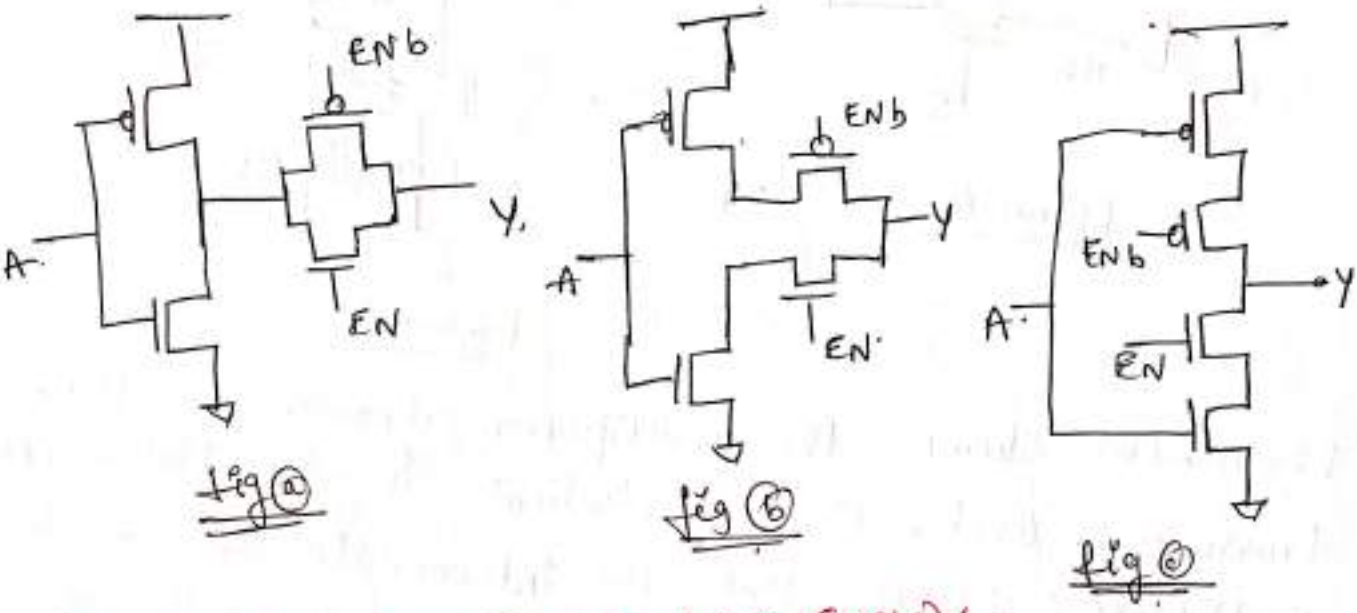


fig (a) fig (b) fig (c)

(b) Complementary pass Transistor Logic (CPL):-

CPL can be understood as an improvement on CVSL. CVSL is slow because one side of the gate pulls down, & then the cross-coupled pMOS transistor pulls the other side up.

The size of the cross-coupled device is an inherent compromise b/w a large transistor that fights the pull down excessively and a small transistor that is slow pulling up. ~~CPL~~ CPL resolves this problem by making one-half of the gate pull up while the other half pulls down.

Figure (a) shows the CPL multiplexer.

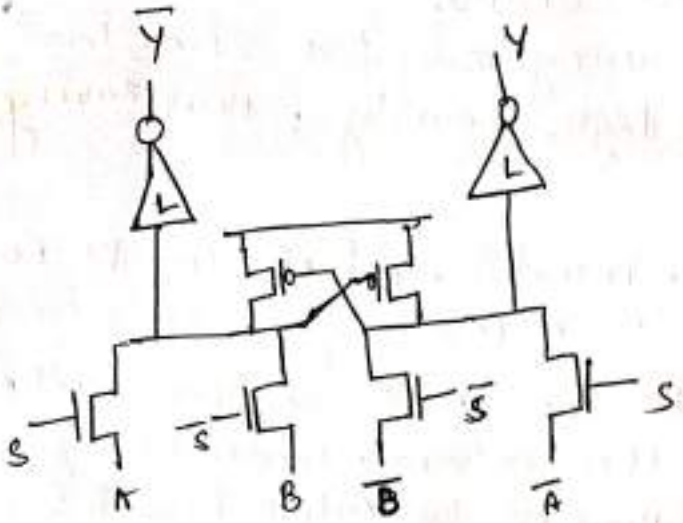


fig (a).

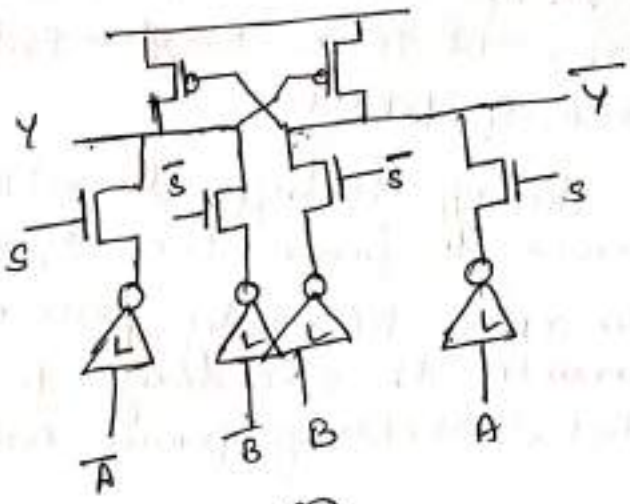


fig (b).

Fig (b) redraws the mux to include the inverters from the previous stage that drives the diffusion input, but to exclude the o/p inverters.

Fig (c) shows the mux drawn at the transistor level. This is identical to the CVSL gate except that the internal node of the stack can be pulled up through the weak pMOS transistors in the inverters.

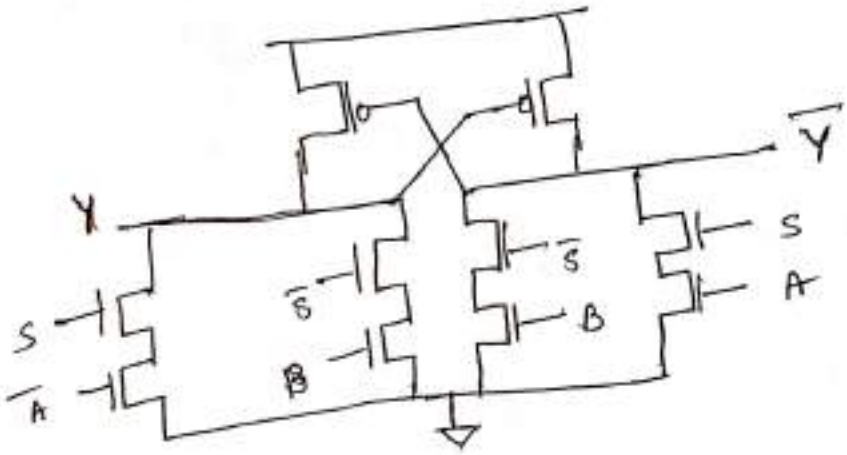


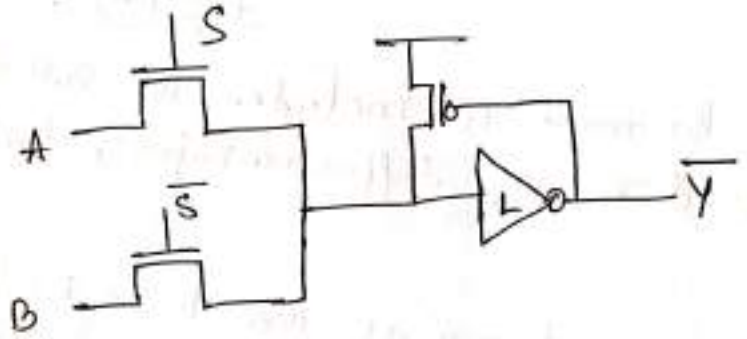
Fig (c)

When the gate switches, one side pulls down well through its nMOS transistors. The other side pulls up. CPL can be constructed without cross-coupled pMOS transistors, but the o/p's would only rise to $V_{DD}-V_t$ (or slightly lower because the nMOS transistors experience the body effect).

(c) Lean Integration with pass Transistors (LEAP):-

LEAP builds logic networks using only fast nMOS transistors. It is a single-ended logic family, thus saving area & power.

The o/p is buffered with an inverter, which can be lo-skewed to favor the asymmetric response of an nMOS transistor. The nMOS t/w only pulls up to $V_{DD} - V_t$ so a pMOS feedback transistor is necessary to pull the internal node fully high, avoiding power consumption in the output inverter.



Module - 4 (Part - A)

Sequential Circuit Design.

Introduction:-

Combinational circuits - in which output is a function of the current inputs.

Sequential circuits - in which the output depends on previous as well as current inputs; such circuits are said to have state.

Eg:- Finite state machines & pipelines are two important examples of sequential circuits.

* Sequential circuits are usually designed with flip-flops (or) latches, which are sometimes called memory elements, that hold data called token.
* Sequencing elements delay tokens that arrive too early, preventing them from catching up with previous tokens.
* Unfortunately, they inevitably add some delay to tokens that are already critical, decreasing the performance of the system. This extra delay is called Sequencing Overhead.
* Static Circuits - refers to gates that have no clock input.

  Eg:- Complementary CMOS, pseudo-nMOS, or pass transistor logic.
* - A sequencing element with static storage employs some sort of feedback to retain its o/p value indefinitely.

* Dynamic circuits - refers to gates that have a clock i/p, especially Domino logic.

- An Element with dynamic storage generally maintains its value as charge on a capacitor that will leak away if not refreshed for a long period of time.

Circuit Design of Latches:-

Conventional CMOS latches are built using pass transistors or tri-state buffers to pass the data while the latch is transparent & feedback to hold the data while the latch is opaque.

A number of alternative latch & flip-flop structures have been used in commercial designs.

The true single phase clocking (TSPC) technique uses a single clock with no inversions to simplify clock distribution.

The Klass semidynamic Flip-Flop (SDFF) is a fast flip-flop using a domino-style input stage.

Differential flip-flops are good for certain applications.

① Conventional CMOS Latches:-

(a)

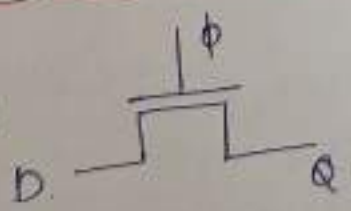


fig (a) shows a very simple transparent latch built from a single transistor.

when $\phi = 1$, $Q = D$.
when $\phi = 0$, Q = hold previous value.

Advantages:-
* It is compact & fast.

Disadvantages:-
* The o/p does not swing from rail-to-rail
* It never rises above $V_{DD} - V_t$.

The output is also dyamic, in other words, the output floats when the latch is opaque.
If it floats long enough, it can be disturbed by leakage. D drives the diffusion input of a pass transistor directly, leading to potential noise issues and making the delay harder to model with static timing analyser.
Finally, the state node is exposed, so noise on the o/p can corrupt the state.

The following figures illustrate improved latches using more transistors to achieve more robust operation.

ⓑ CMOS transmission gate:-

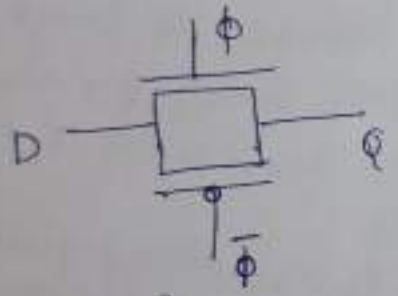


fig ⓑ.

Fig ⓑ uses a CMOS transmission gate in place of the single nMOS pass transistor to offer rail-to-rail o/p swings. It requires a complementary clock $\bar{\phi}$, which can be provided as an additional input (or) locally generated from ϕ through an inverter.

ⓒ Inverting latch:-

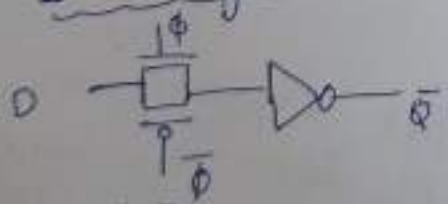


fig ⓒ.

Fig ⓒ. adds an o/p inverter so that the state node X is isolated from noise on the o/p. This create an inverting latch.

ⓓ Buffered input latch:-

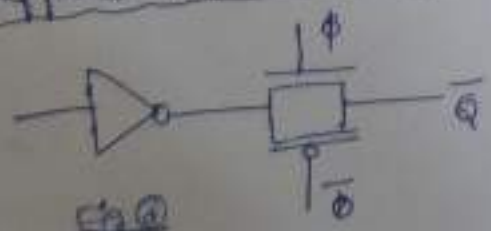


fig ⓓ

Fig ⓓ also behaves as an inverting latch with a buffered i/p but unbuffered o/p.

The inverter followed by a transmission gate is essentially equivalent to a tristate inverter but has a slightly lower logical effort because the o/p is driven by both transistors of the transmission gate in parallel.

Figure (c) & (d) are both fast dynamic latches.

(e) Staticized feedback latches:

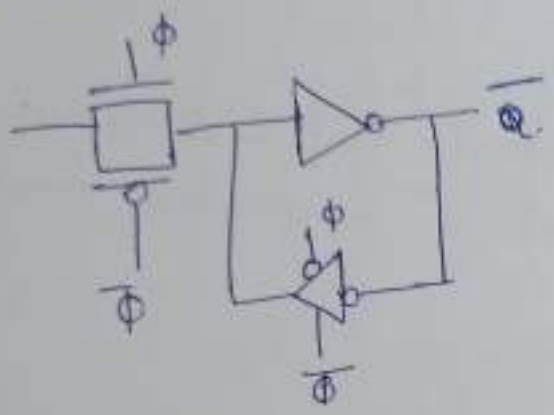


fig (e)

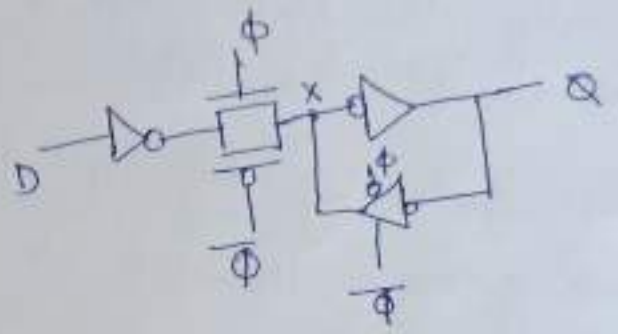


fig (f)

In modern processes, subthreshold leakage is large enough that dynamic nodes retain their values for only a short time, especially at the high temperature & voltage encountered during burn-in-test.

∴ Practical latches need to be staticized, adding feedback to prevent the o/p from floating as shown in fig (e).

When the clock is 1, the i/p transmission gate is ON, the feedback tristate is OFF, & the latch is transparent.

When the clock is 0, the input transmission gate turns OFF, the feedback tristate turns ON, holding X at the correct level.

Figure (f) adds an input inverter so the input is a transistor gate rather than unbuffered diffusion. (3)

Unfortunately, both (e) & (f) reintroduced o/p noise sensitivity. A large noise spike on the o/p can propagate backward through the feedback gates & corrupt the state node X.

(g) Robust transparent latch:-

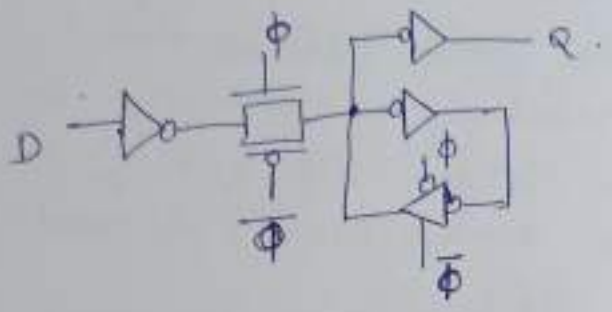


fig (g)

Fig (g) is a robust transparent latch that addresses all of the deficiencies mentioned so far: The latch is static, all nodes swing rail-to-rail, the state node is isolated from o/p noise, & the i/p drives transistor gates rather than diffusion.

It is recommended for all but the most performance-or area-critical designs.

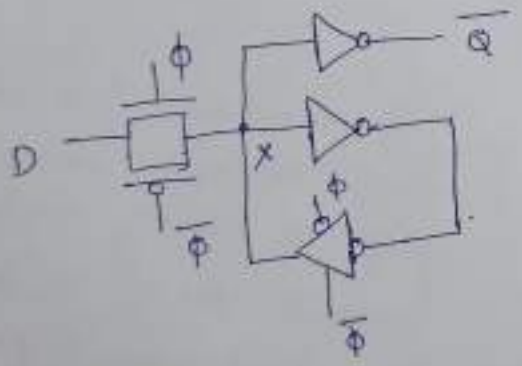


fig (h)

In semicustom datapath applications where i/p noise can be better controlled, the inverting latch of fig (h), may be preferable because it is faster & more compact.

(i) Jamb latch:-

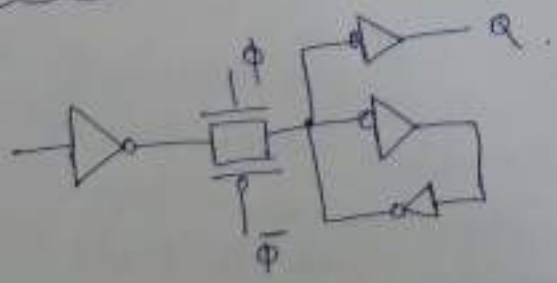


fig (i)

Fig (i) shows the jamb latch, a variation of Fig (g) that reduces the clock load & saves two transistors by using a weak feedback inverter in place of the tristate.

(j)

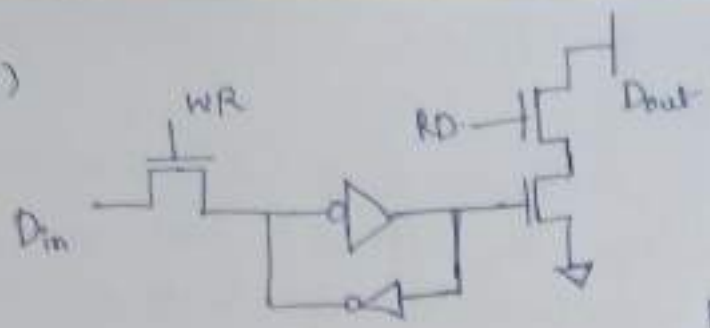


fig (j)

Figure (j) shows another famb latch commonly used in register files & FPGA cells. Many such latches read out onto a single Dout wire & only one latch is enabled at any given time with its RD signal.

With the i/p inverter, the latch can be viewed as far away. With the i/p inverter, the latch can be viewed as a cross b/w the designs shown in (g) and (i). Some latches add one more inverter to provide both true & complementary o/p's.

Clocked CMOS (C^2MOS) :-

The dynamic latch of fig (d) can also be drawn as a clocked tristate, as shown in fig below.

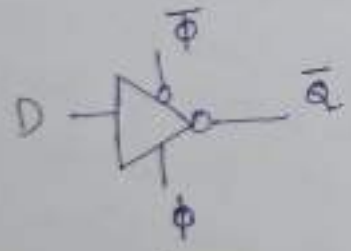


fig (a)

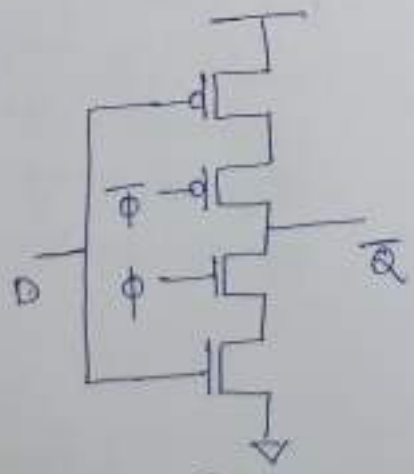


fig (b)

Figures:- C^2MOS Latch

The conventional form using the inverter & transmission gate is slightly faster because the o/p is driven through the nMOS & pMOS working in parallel.

C^2MOS is slightly smaller because it eliminates two contacts. Fig (b) shows another form of the tristate that swaps the data & clock terminals. It is logically equivalent but electrically inferior because toggling D while the latch is opaque can cause charge-sharing noise on the output node.

NOTE :- All of the latches shown so far are transparent while ϕ is high. They can be converted to active-low latches by swap-ping ϕ & $\overline{\phi}$.

Conventional CMOS Flip-Flops:-

Figure (a) below shows a dynamic inverting flip-flop built from a pair of back-to-back dynamic latches. Either the first (or) the last inverter can be removed to reduce delay at the expense of greater noise sensitivity on the unbuffered i/p (or) o/p.

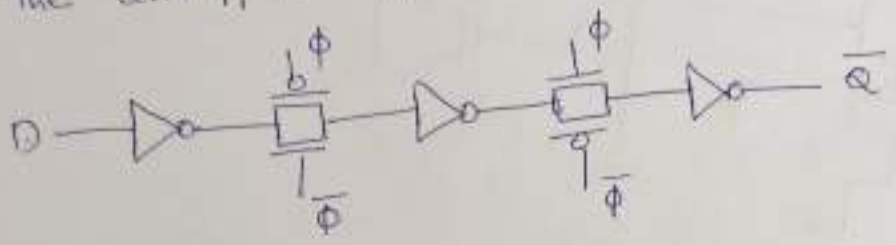


fig (a)

Figure (b) adds feedback & another inverter to produce a non-inverting static flip-flop.

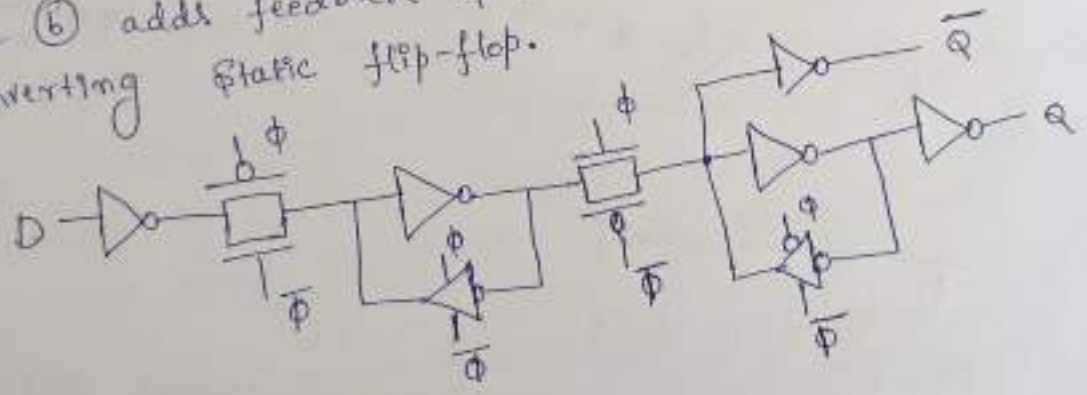


fig (b)

Most standard cell libraries employ this design because it is simple, robust, compact & energy-efficient.

Flip-flops usually take a single clock signal ϕ & locally generate its complement $\overline{\phi}$. If the clock rise/fall time is very slow, it is possible that both the clock & complement will simultaneously be at intermediate voltages, making both latches transparent & increasing the flip flop hold time.

NORA dynamic Flip-flops:-

The flip-flop has a potential internal race condition b/w the two latches. This race can be exacerbated by skew b/w the clock & its complement caused by the delay of the inverter.

Figure below shows a Transmission gate & NORA dynamic flip-flops with a built-in Clock Inverter.

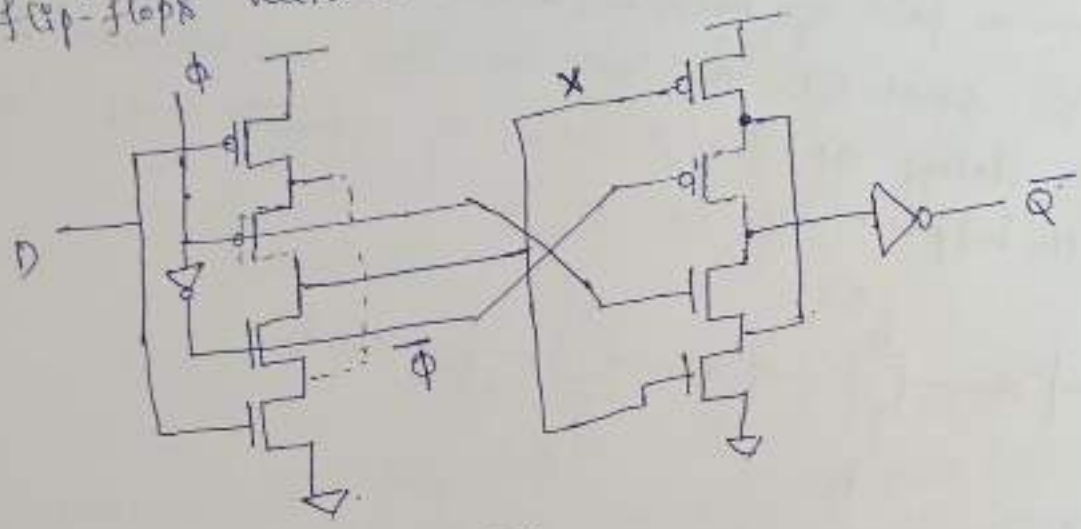


fig (a).

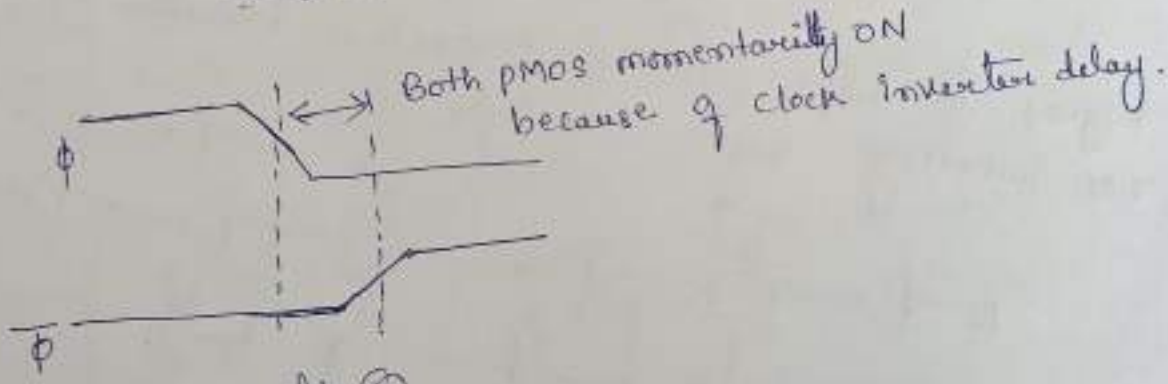


fig (b)

When $\phi = 0$, both the clock & its complement are momentarily low, turning on the clocked pMOS transistors in both transmission gates.

If the skew (i.e, inverter delay) is too large, the data can sneak through both latches on the falling clock edge, leading to incorrect operation.

Figure (c) below shows a C^2MOS dynamic flip-flop built using C^2MOS latches rather than inverters & transmission gates. Because each stage inverts, data passes through the nMOS stack of one latch & the pMOS of the other, so skew that turns on both clocked pMOS transistors is not a hazard.

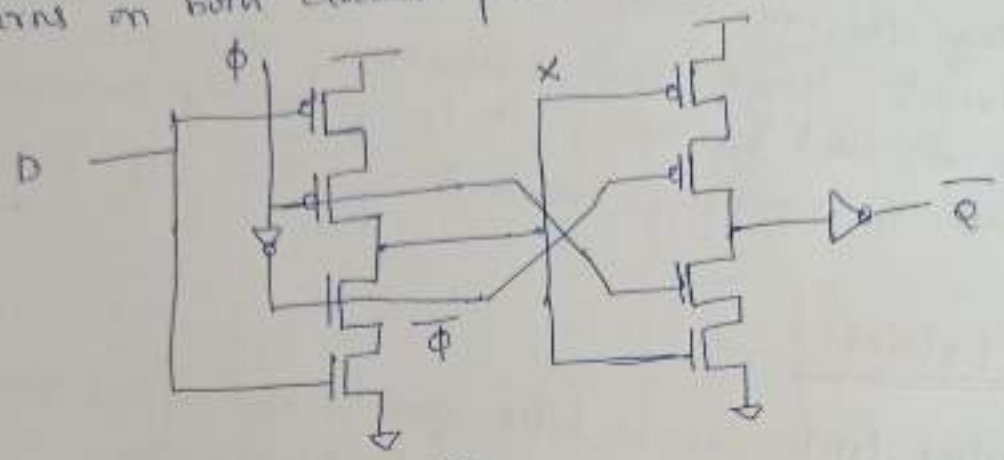


fig (c)

However, the flip-flop is still susceptible to failure from very slow edge rates that turn both transistors partially ON. The same skew advantages apply even when an even no of inverting logic stages are placed b/w the latches, this technique is sometimes called NORA.

In practice, most flip-flop designs carefully control the delay of the clock inverter so the transmission gate design is safe & slightly faster than C^2MOS.

Flip-flop with two-phase non-overlapping

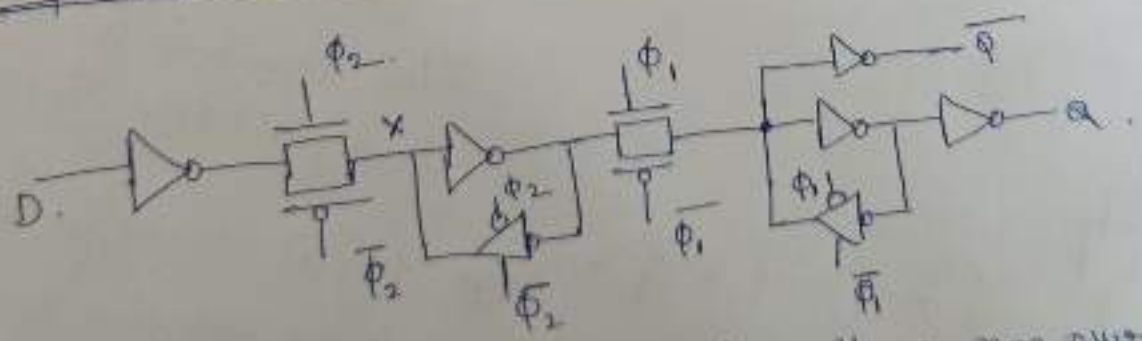


fig:- Flip-flop with two-phase non-overlapping clocks.

Figure above is a flip-flop with two-phase non-overlapping clocks. ~~Instead~~ which use a pair of two-phase non overlapping clocks instead of the clock & its complement. The flip-flop captures its input on the rising edge of ϕ_1. By making the non-overlap large enough, the circuit will work despite large skews. However, the non-overlap time is not used by logic, so it directly increases the setup time & sequencing overhead of the flip-flop.

Pulsed Latches:-

A pulsed latch can be built from a conventional CMOS transparent latch driven by a brief clock pulse.

Figure (a) below shows a simple pulse generator, also called a clock chopper (or) one-shot.

The pulsed latch is faster than a regular flip-flop because it involves a single latch rather than two & because it allows time borrowing.

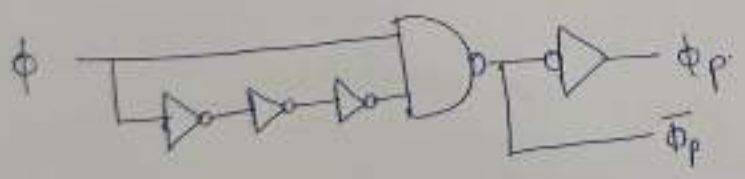


fig (a).

It can also consume less energy, although the pulse generator adds to the energy consumption.

The drawback is the increased hold time.

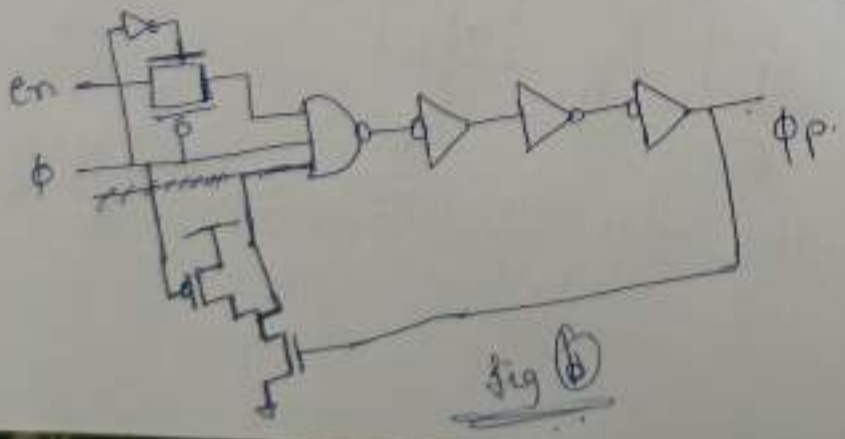


fig (b)

Fig (b) shows another pulse generator used on an NEC RISC processor to produce substantially longer pulses.

Resettable Latches and Flip-Flops:-

* Most practical sequencing elements require a reset signal to enter a known initial state on startup & ensure deterministic behavior.

Figure below shows latches & flip-flops with reset inputs.

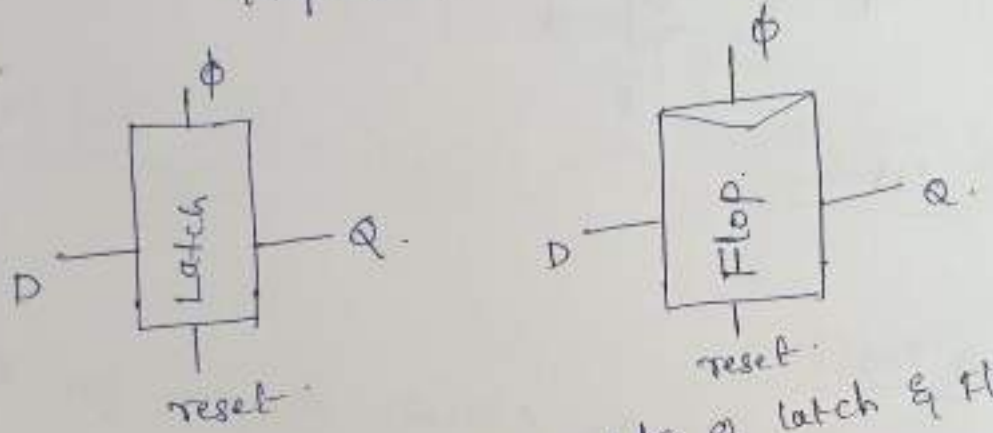


Figure:- Symbols of latch & flop.

* There are two types of reset :-
  1. Synchronous reset.
  2. Asynchronous reset.

* Asynchronous reset forces Q low immediately, while synchronous reset waits for the clock.

* Synchronous reset signals must be stable for a setup & hold time around the clock edge while asynchronous reset is characterized by a propagation delay from reset to output.

* Synchronous reset simply requires ANDing the i/p D with $\overline{reset}$.

* Asynchronous reset requires gating both the data & the feedback to force the reset independent of the clock.

* The tristate NAND gate can be constructed from a NAND gate in series with a clocked transmission gate.

Synchronous Reset

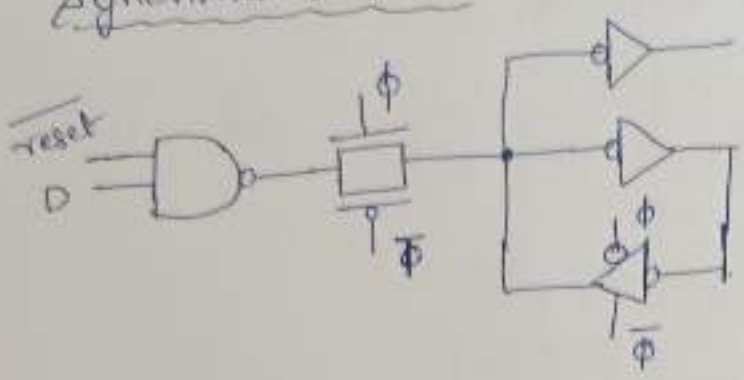


fig:- latch.

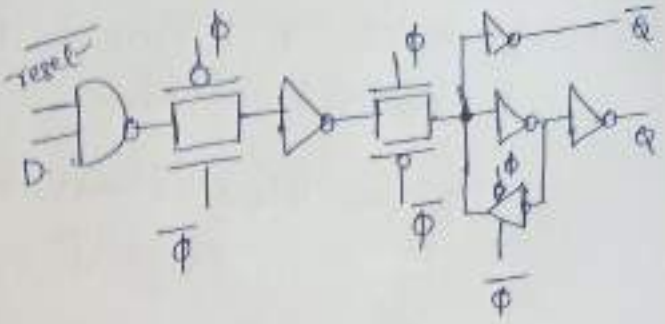


fig:- flip-flop

Asynchronous Reset

Fig below shows the circuit diagram for Asynchronous reset latch & flip flop.

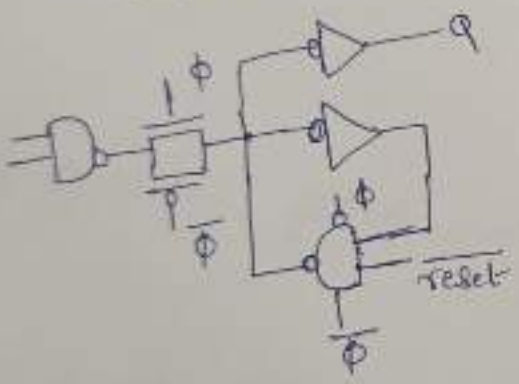


fig:- latch

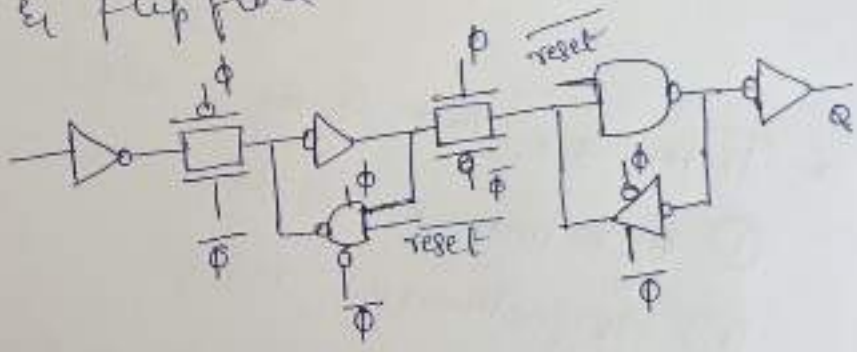


fig:- flipflop

Flip flop with asynchronous Set & reset;-

Settable latches & flip-flops forces the output high instead of low. They are similar to resettable elements. But replace NAND with NOR & reset with set.

Figure below shows a flip-flop combining both asynchronous set & reset.

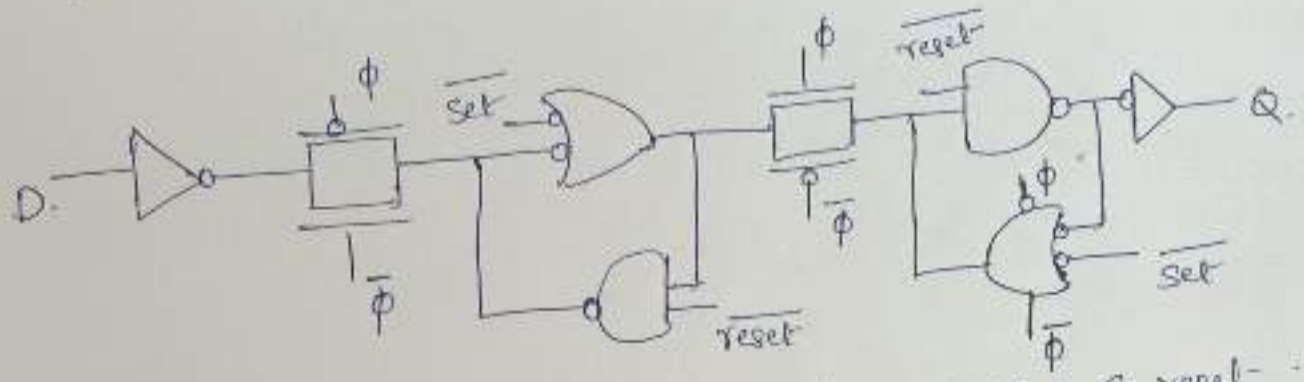


Figure :- Flip-flop with asynchronous set & reset.

Swetha. B.V.
Asst. Prof
ECE Dept

Module - 4 (Part-B)

Dynamic Logic Circuits

Introduction:-

Static logic circuits :-

A typical static logic gate generates its o/p corresponding to the applied i/p voltages after a certain time delay, & it can preserve its output level (or state) as long as the power supply is provided.

Disadvantages

* Require a large number of transistors to implement a function.
* Require large silicon area.
* May cause a considerable time delay.

Dynamic logic circuits :-

Dynamic logic circuits offer several significant advantages over static logic circuits.

The operation of all dynamic logic gates depends on temporary storage of charge in parasitic node capacitances, instead of relying on steady-state circuit behavior.

The capability of temporarily storing a state, i.e., a V+g level, at a capacitive node allows us to implement very simple sequential circuits with memory functions.

Also, the use of common clock signals throughout the system enables us to synchronize the operations of various circuit blocks.

As a result, dynamic circuit techniques lend themselves well to synchronous logic design.

Advantages:-

* Requires smaller Area than the Static logic implementation.
* Consume less power than the Static counterpart.

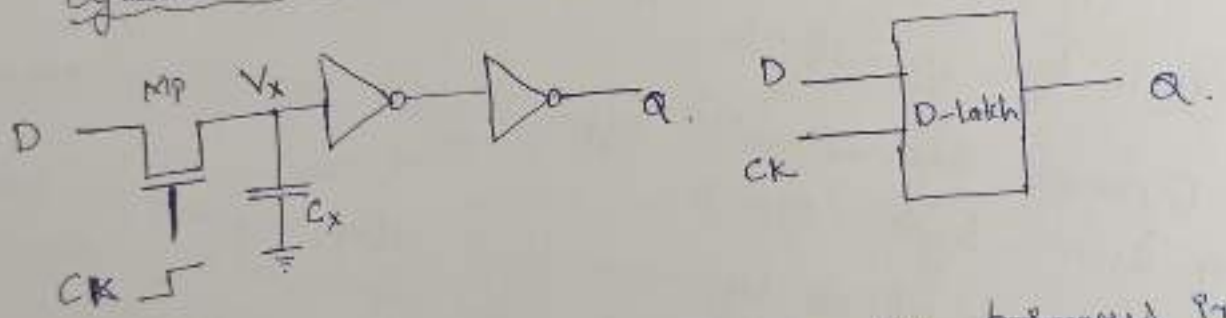
Disadvantages:-

* Dynamic logic circuits requires periodic clock signals in order to control charge refreshing.

Example:-

The following Example presents the operation of a dynamic D-latch circuit, which essentially consists of two inverters in cascade. This simple circuit illustrates most of the basic operational concepts involved in dynamic circuit design.

Dynamic D-Latch circuit



The parasitic i/p capacitance Cx of the primary inverter stage play an important role in the dynamic operation of this circuit.

Operation:-

when CK = 1, MP → turns on.
The capacitor Cx is either charged up, or charged down through the pass transistor MP, depending on the input (D) voltage level. The O/p (Q) assumes the same logic level as the input.

when the CK = 0, Mp → turns OFF.
The capacitor Cx is isolated from the input D.
Since there is no current path from the intermediate node X to either V_{DD} (or) ground, the amount of charge stored in Cx during the previous cycle determines the o/p Vtg level Q.

Basic principles of pass Transistor Circuits:-

The fundamental building block of nMOS dynamic logic circuits, consisting of an nMOS pass transistor driving the gate of another nMOS transistor, is shown in Figure below,

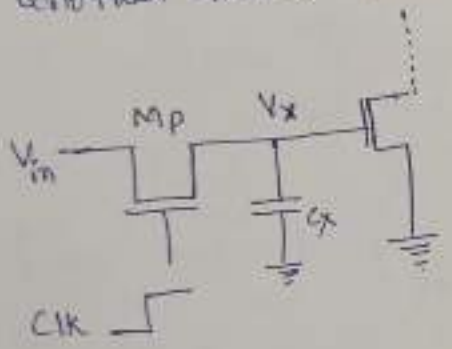


fig:- The basic building block for nMOS dynamic logic, which consists of an nMOS pass transistor driving the gate of another nMOS transistor.

As we know that, the pass transistor Mp is driven by the periodic clock signal & acts as an access switch to either charge up (or) charge down the parasitic capacitance Cx, depending on the input signal Vin.

Logic "1" Transfer:-

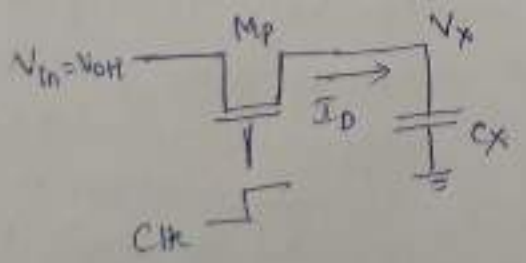


Fig:- Equivalent circuit for the logic "1" transfer event.

⇒ Initially, assume,
$V_x(t=0) = 0V$.

⇒ A logic "1" level is applied to the input terminal;
i.e $V_{in} = V_{OH} = V_{DD}$.

⇒ Mp starts to conduct as soon as clock signal becomes active & operates in saturation region.

$\Rightarrow$ Since $V_{DS} = V_{GS}$, &

$$V_{DS} > V_{GS} - V_{T,n}$$

The pass transistor MP operating in the saturation region starts to charge up the capacitor C_x, thus.

$$C_x \frac{dV_x}{dt} = \frac{K_n}{2}(V_{DD} - V_x - V_{T,n})^2 \quad \text{--- (1)}$$

Integrating, we get

$$\int_0^t dt = \frac{2C_x}{K_n} \int_0^{V_x} \frac{dV_x}{(V_{DD} - V_x - V_{T,n})^2}$$

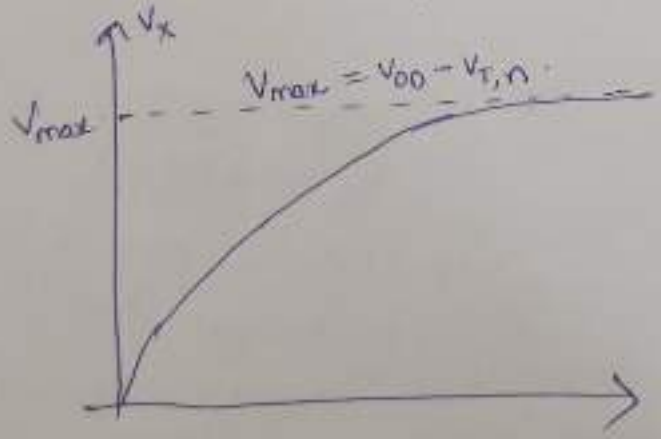
$$= \frac{2C_x}{K_n}\left(\frac{1}{(V_{DD} - V_x - V_{T,n})}\right)\Bigg|_0^{V_x} \quad \text{--- (2)}$$

$$t = \frac{2C_x}{K_n}\left[\left(\frac{1}{V_{DD} - V_x - V_{T,n}}\right) - \left(\frac{1}{V_{DD} - V_{T,n}}\right)\right] \quad \text{--- (3)}$$

This above eqn can be solved for $V_x(t)$,

$$V_x(t) = (V_{DD} - V_{T,n}) \frac{\left[\frac{(K_n(V_{DD} - V_{T,n})}{2C_x}\right] t}{1 + \left[\frac{K_n(V_{DD} - V_{T,n})}{2C_x}\right] t} \quad \text{--- (4)}$$

The graph below shows the variation of the node V_{tg} V_x as a function of time,



The V_{tg} rises from its initial value of 0V & approaches a limit value for target t, but it cannot exceed its limit value of $V_{max} = (V_{DD} - V_{T,n})$.

Fig:- Variation of V_x as a function of time during logic "1" transfer.

Synchronous Dynamic Circuit Techniques:

We will investigate different examples of synchronous dynamic circuits implemented using deplition-~~load~~ load nMOS, Enhancement-load nMOS, and CMOS building blocks.

Dynamic pass Transistor Circuits:

Consider the generalized view of a multi-stage synchronous circuit shown in Figure below,

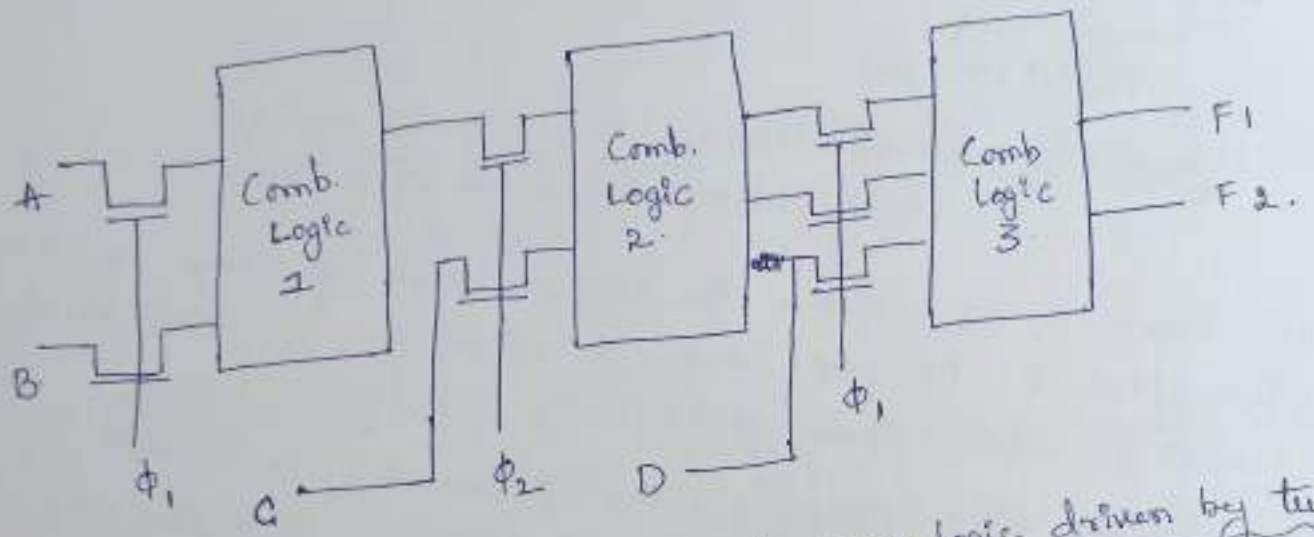


Figure (a): Multi-Stage pass transistor logic driven by two non-overlapping clocks.

The circuit consists of cascaded combinational logic stages, which are interconnected through nMOS pass transistors. All I/P's of each combinational logic block are driven by a single clock signal.

To drive the pass transistors in this system, two non-overlapping clock signals, ϕ_1 & ϕ_2 are used.

The non-overlapping property of the two clock signals guarantees that at any given time point, only one of the two clock signals guarantees that at any given time point, only one of the two clock signals can be active as illustrated in Figure (b)

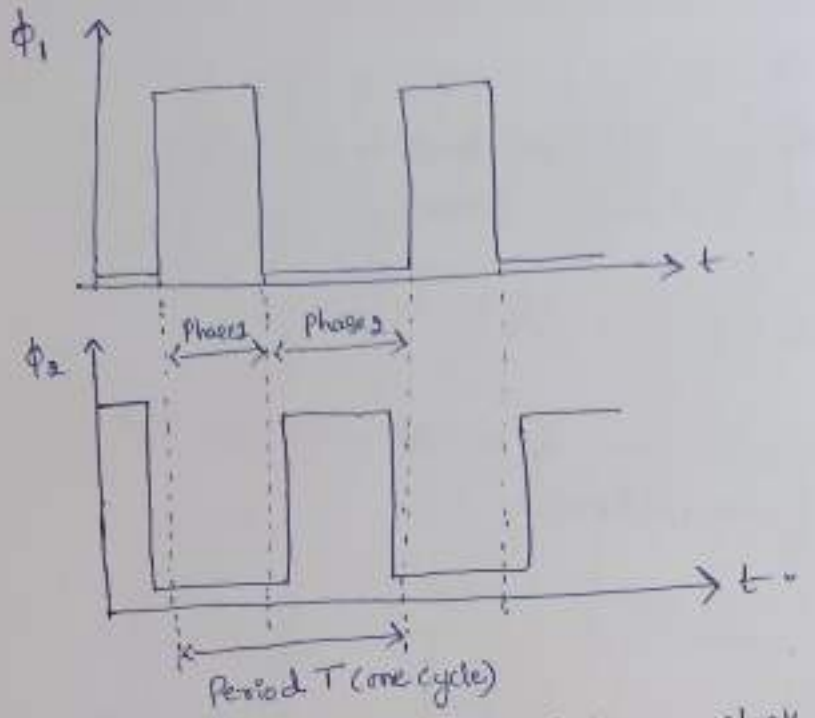


Figure (b):- Non-overlapping clock signals used for two-phase synchronous operation.

When clock ϕ_1 is active, the input levels of stage 1 are applied through the pass transistors, while the i/p capacitances of stage 2 retain their previously set logic levels. During the next phase, when clock ϕ_2 is active, the input levels of stage 2 will be applied through the pass transistors, while the i/p capacitances of stage 1 & stage 3 retain their logic levels.

This allows us to incorporate the simple dynamic memory function at each stage i/p, & at the same time, to facilitate synchronous operation by controlling the signal flow in the circuit using the two periodic clock signals. This timing scheme is called as two-phase clocking.

Depletion-load dynamic Shift Register Circuit

Figure below shows a depletion-load dynamic shift-register circuit, in which the input data are inverted once & transferred, (or) shifted into the next stage during each clock phase.

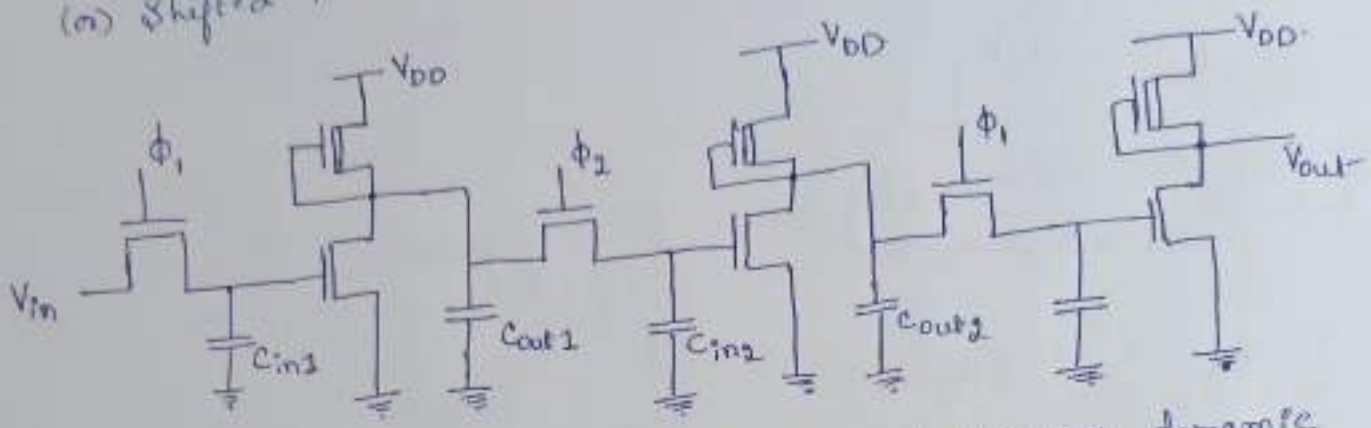


Figure:- Three stages of a depletion-load nMOS dynamic shift register circuit driven with two-phase clocking.

Operation:-

⇒ During the active phase of ϕ_1, the i/p voltage level V_{in} is transferred into the input capacitance C_{in1}.
Thus, the valid o/p voltage level of the first stage is determined as the inverse of the current input during this cycle.

⇒ When ϕ_2 becomes active, during the next phase. The o/p voltage level of the first stage is transferred into the second stage input capacitance C_{in2} & the valid o/p v+g level of the second stage is determined.

⇒ When ϕ_1 becomes active again, the original data bit written into the register during the previous cycle is transferred into the third stage, & the first stage can now accept the next data bit.

Synchronous Complex logic:-

Figure ⓐ & ⓑ below shows a two-stage Circuit Example implemented using depletion-load nMOS Complex logic gates.

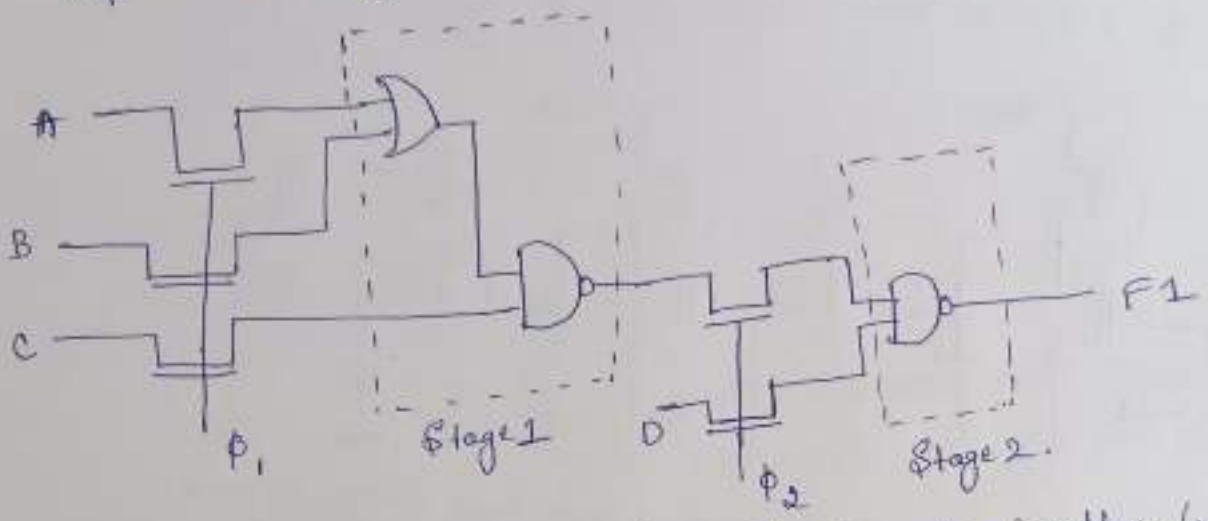


Fig: ⓐ - A two-stage Synchronous Complex logic Circuit Example

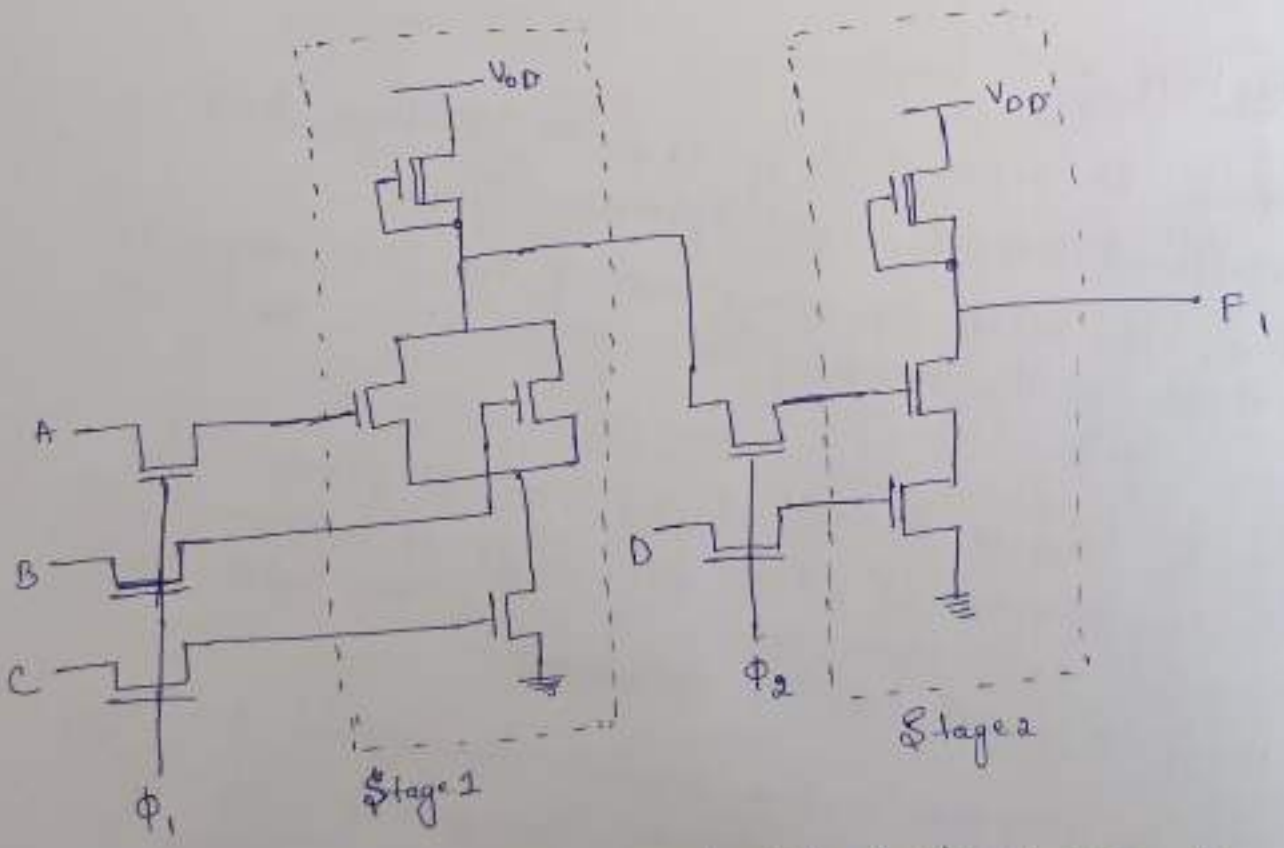


Fig. ⓑ:- Depletion-load nMOS Implementation of Synchronous Complex logic.

In a complex logic circuit as shown in fig (b), the signal propagation delay of each stage may be different. Thus, in order to guarantee that correct logic levels are propagated during each active clock cycle, the half-period length of the clock signal must be longer than the largest single-stage signal propagation delay found in the circuit.

Enhancement-load dynamic shift register (ratioed logic):-

The power dissipation & the silicon area can be reduced significantly by using this dynamic (clocked) load approach.

There are two variants of the dynamic enhancement-load shift register, both of which are driven by two non-overlapping clock signals.

(1) Ratioed dynamic logic.

(2) Ratioless logic.

(1) Figure below shows the first implementation, where in each stage the i/p pass transistor & the load transistor are driven by opposite clock phases, ϕ_1 & ϕ_2.

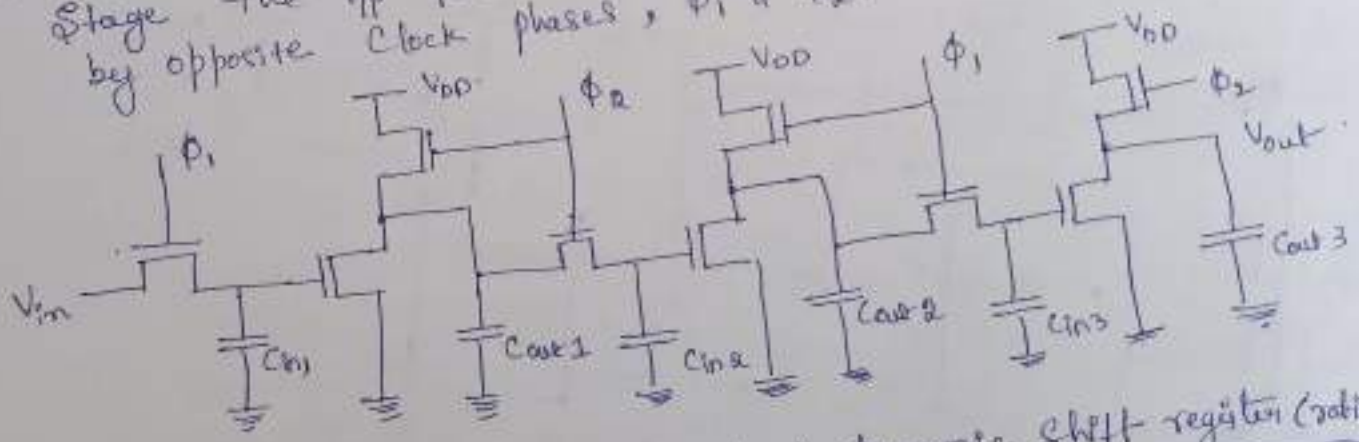


Figure:- Enhancement-load dynamic shift register (ratioed logic)

Operation:-

⇒ When ϕ_1 is active, the input voltage level V_{in} is transferred into the first-stage input capacitance C_{in1} through the pass transistor.

⇒ During the next phase (active ϕ_2), the load transistor is turned on. Since the input logic level is still being preserved in C_{in1},

the O/P of the first-stage attains its valid logic level.

⇒ At the same time, the input-pass transistor of the second stage is also turned on, which allows this newly determined O/P level to be transferred into the I/P capacitance C_{in2} of the second stage.

⇒ When clock ϕ_1 becomes active again, the valid O/P level across C_{out2} is determined, & transferred into C_{in3}. Also, a new I/P level can be accepted into C_{in} during this phase.

⇒ The valid low-output-voltage level V_{OL} of each stage is strictly determined by the driver-to-load ratio, since the O/P pass transistor turns on in phase with the load transistor. Therefore, this circuit arrangement is also called <u>ratioed dynamic logic</u>.

The basic operation principle can obviously be extended to arbitrary complex logic, as shown in fig below.

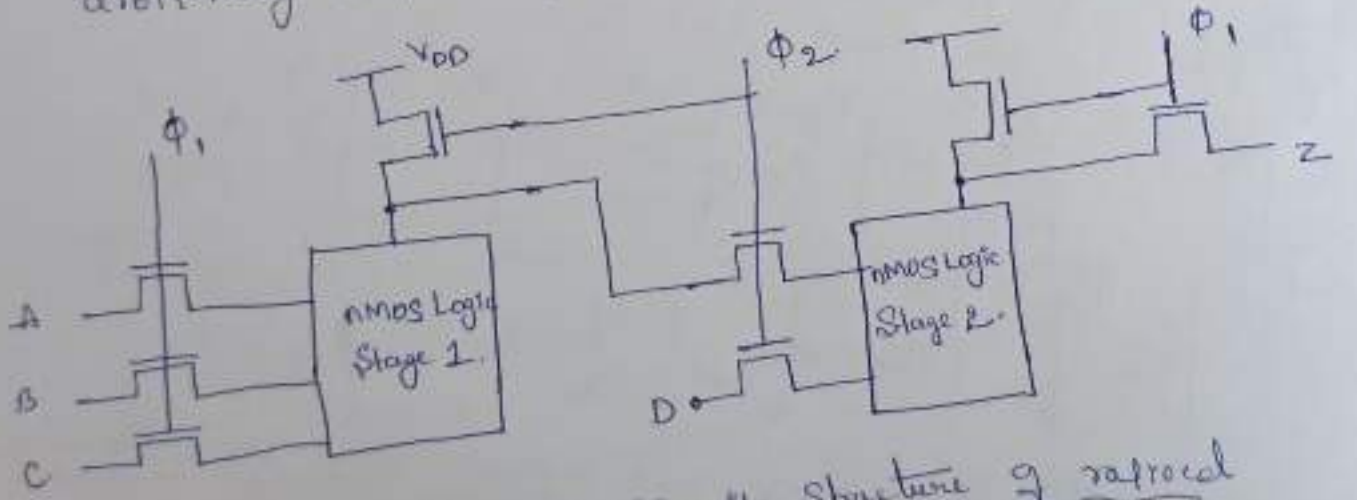


Figure:- General Circuit Structure of ratioed Synchronous dynamic logic.

The overall power consumption of dynamic Enhancement-load logic is generally lower than for depletion-load nMOS logic.

Enhancement-load dynamic shift register (ratioless logic):-

The second variant of dynamic Enhancement-load shift register implementation is shown in fig below, where, in each stage, the input pass transistor & the load transistor are driven by the same clock phase.

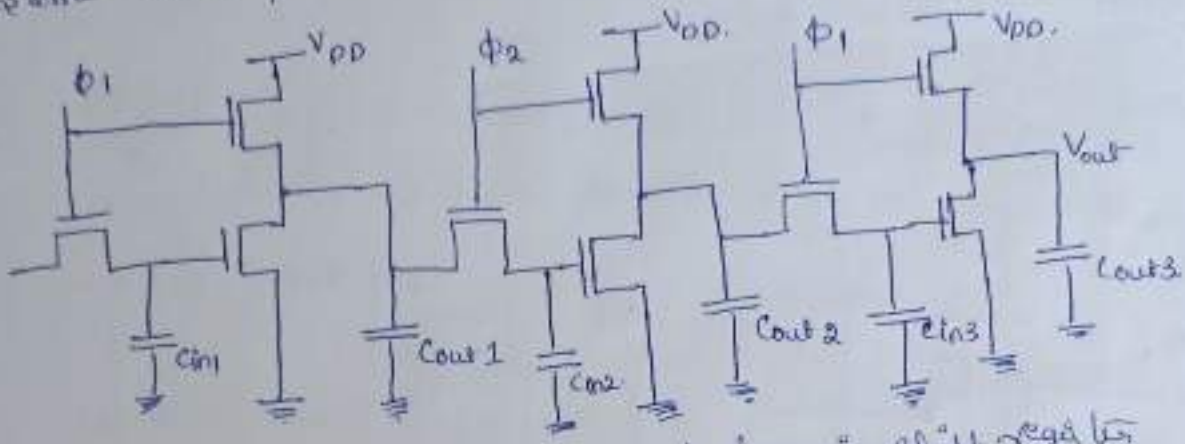


Fig:- Enhancement-load dynamic shift register (ratioless logic).

Operation:-

⇒ When ϕ_1 is active, the i/p voltage level V_{in} is transferred into the first-stage i/p capacitance C_{in1} through the pass transistor. At the same time, the Enhancement-type nMOS load transistor of the first stage inverter is active.

⇒ ∴ The o/p of the first stage inverter attains its valid logic level.

⇒ During the next phase (active ϕ_2), the input pass transistor of the next stage is turned on, & the logic level is transferred onto the next stage.

Case (I):- If the o/p level across C_{out1} is logic high at the end of the active ϕ_1 phase, this V_{tg} level is transferred to C_{in2}.

Case (II):- If the o/p level of the first stage is logic-low at the end of the active ϕ_1 phase, then the o/p capacitor C_{out1} will be completely drained to a voltage of $V_{OL} = 0V$ when ϕ_1 turns off.

when clock ϕ_1 becomes active again, the valid o/p level across C_{out2} is determined & transferred into C_{in3}. Also a new i/p level can be accepted into C_{in1} during this phase, since the valid logic-low level of $V_{OL} = 0V$ can be achieved regardless of the driver-to-load ratio. This circuit arrangement is called ratioless dynamic logic.

The General circuit structure of ratioless synchronous dynamic logic is shown in fig below:

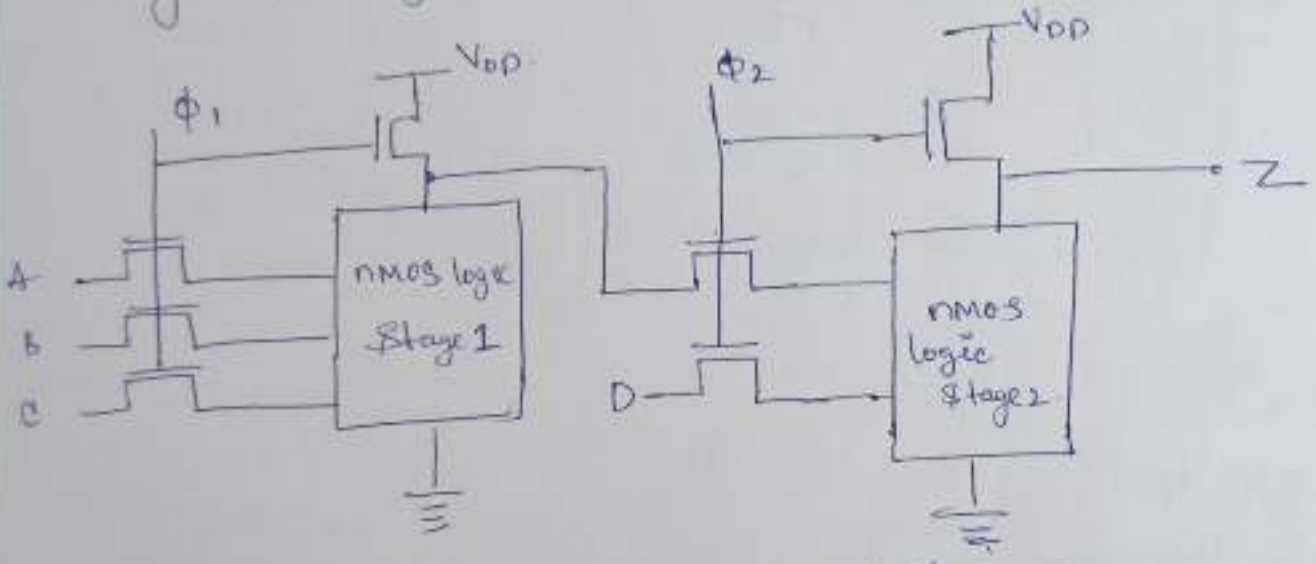


fig:- General circuit structure.

The basic operation principle can be extented to arbitrary complex logic as shown above.

Dynamic CMOS circuit Techniques

CMOS Transmission Gate Logic:-

Static CMOS gates are used for implementing the logic blocks, & CMOS transmission gates are used for transferring the output levels of one stage to the inputs of the next stage as shown in figure (a) below.

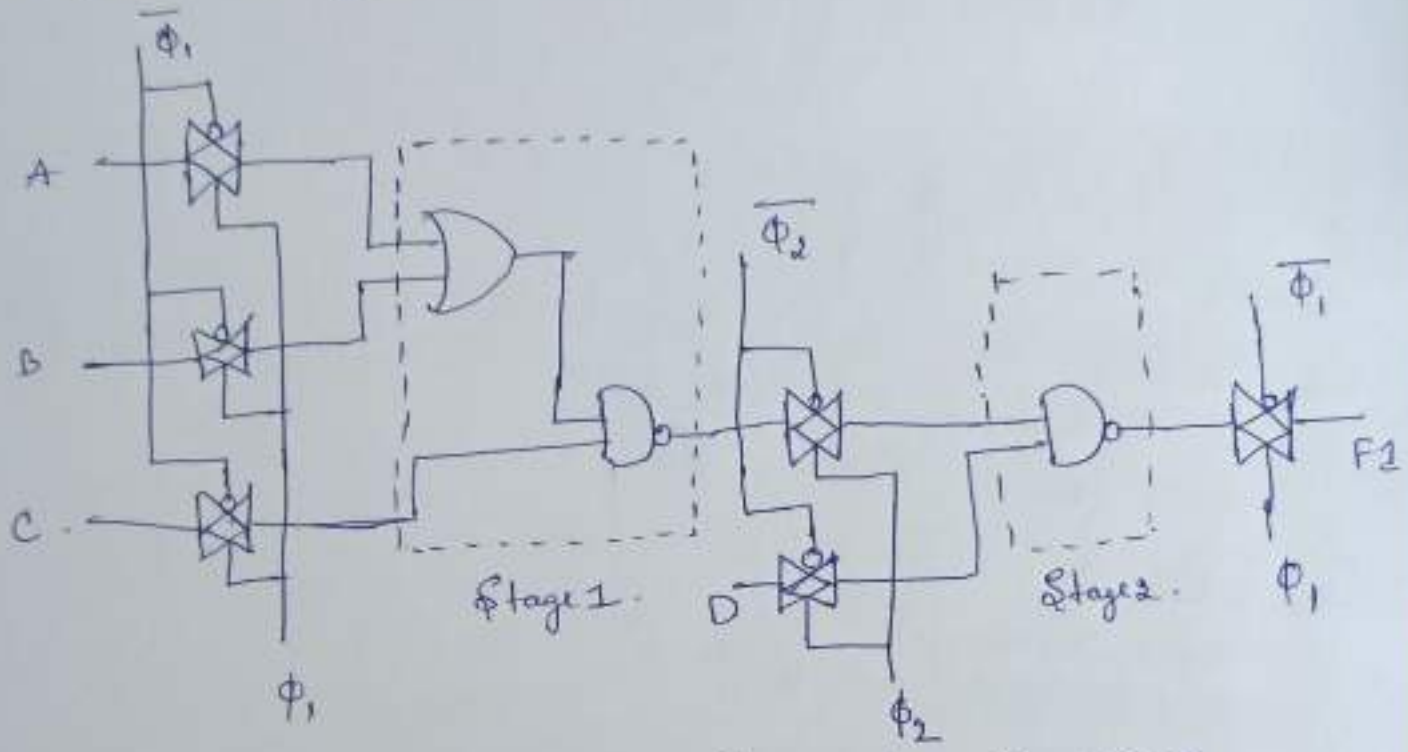


Figure (a):- Typical example of dynamic CMOS Transmission gate logic.

Each transmission gate is actually controlled by the clock signal & its complement.

As a result, two-phase clocking in CMOS transmission gate logic requires that a total of four clock signals are generated & routed throughout the circuit.

To illustrate the basic operation principles, the fundamental building block of a dynamic CMOS transmission gate shift register is shown in Fig (b).

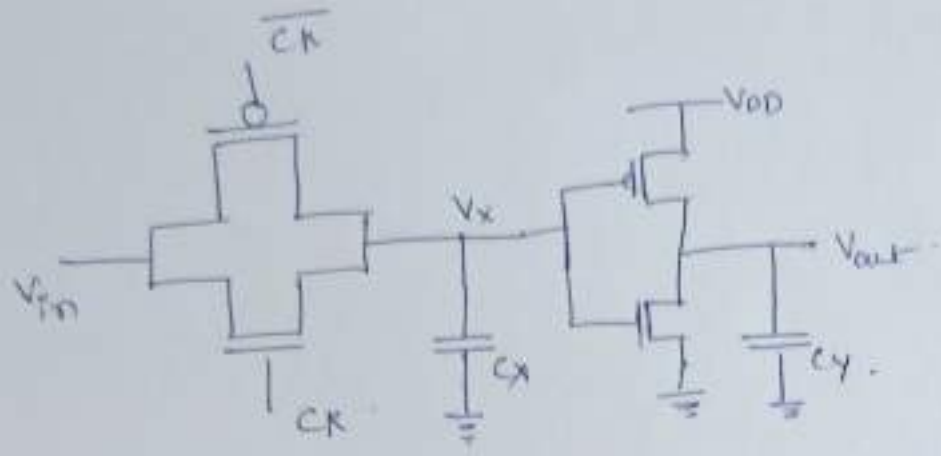


fig ⑥:- Basic building block of a CMOS transmission gate dynamic shift register.

It consists of a CMOS inverter, which is driven by a CMOS transmission gate.

⇒ During the active clock phase (CK = 1), the i/p vtg V_{in} is transferred onto the parasitic input capacitance C_x via the transmission gate.

⇒ When the clock signal becomes inactive, the CMOS Transmission gate turns off & the voltage level across C_x can be preserved until the next cycle.

Single-phase CMOS shift register:-

The single-phase CMOS shift register, which is built by cascading identical unit & by driving each stage alternately with the clock signal & its complement.

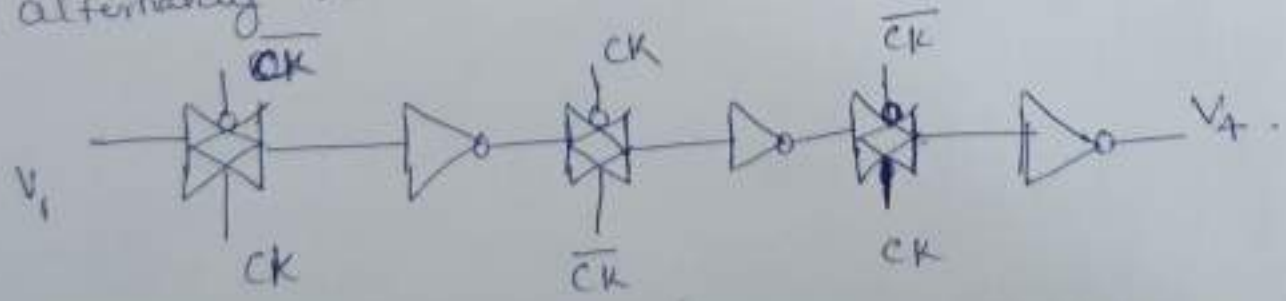


Fig:- Single phase CMOS transmission gate dynamic Shift Register.

Ideally, the transmission gates of the odd-numbered stages would conduct during the active clock phase (when CK = 1), while the transmission gates of the even-numbered stages are off, so that the cascaded inverter stages in the chain are alternately isolated.

Dynamic CMOS Logic (precharge-Evaluate Logic):-

A dynamic CMOS circuit technique allows to significantly reduce the number of transistors used to implement any logic function.

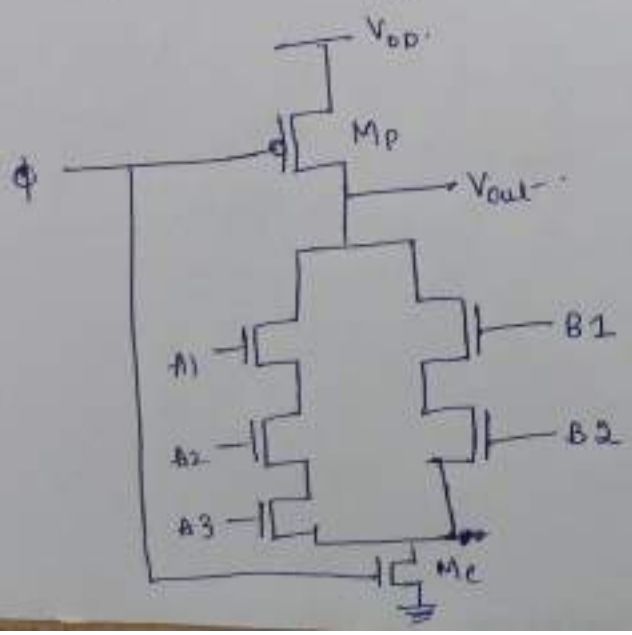
The circuit operation is based on first precharging the output node capacitance & subsequently, evaluating the output level according to the applied inputs.

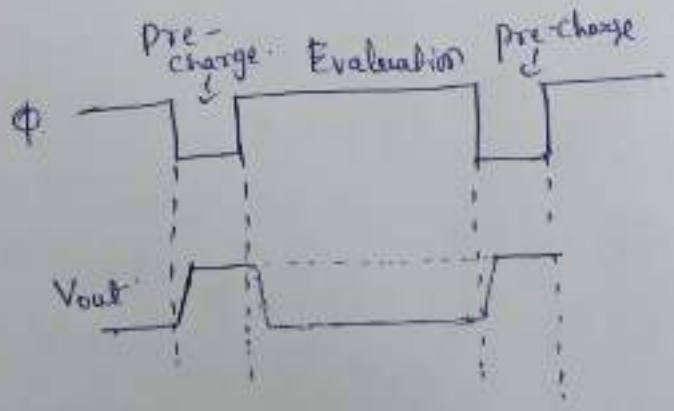
Both of these operations are scheduled by a single clock signal, which drives one nMOS & one pMOS transistor in each dynamic stage.

Example:-

$$F = \overline{(A_1 A_2 A_3 + B_1 B_2)}$$

A dynamic CMOS logic gate implementation of function 'F' is shown in figure below.





Operation:-

Case (i):- when the Clock signal is low (Pre-charge phase).

PMOS, $M_p \rightarrow$ Conducting.

NMOS, $M_e \rightarrow$ OFF.

$$V_{out} = V_{DD}.$$

The input voltages are also applied during this phase, but they have no influence yet upon the o/p level since M_e is turned-off.

Case (ii):- when the clock signal becomes high (Evaluate phase).

$M_p \rightarrow$ turns off.

$M_e \rightarrow$ turns on.

The output node voltage may now remain at the logic-high level (or) drop to a logic low, depending on the input-voltage levels.

Cascading problem in dynamic CMOS logic:-

The operation of the single-stage dynamic CMOS logic gate is quite straight forward. For practical multi-stage applications, the dynamic CMOS gate presents a significant problem.

To examine this fundamental limitation, consider the two-stage cascaded structure shown in Figure below. The output of the first dynamic CMOS stage drives one of the inputs of the second dynamic CMOS stage, which is assumed to be a two-input NAND gate for simplicity.

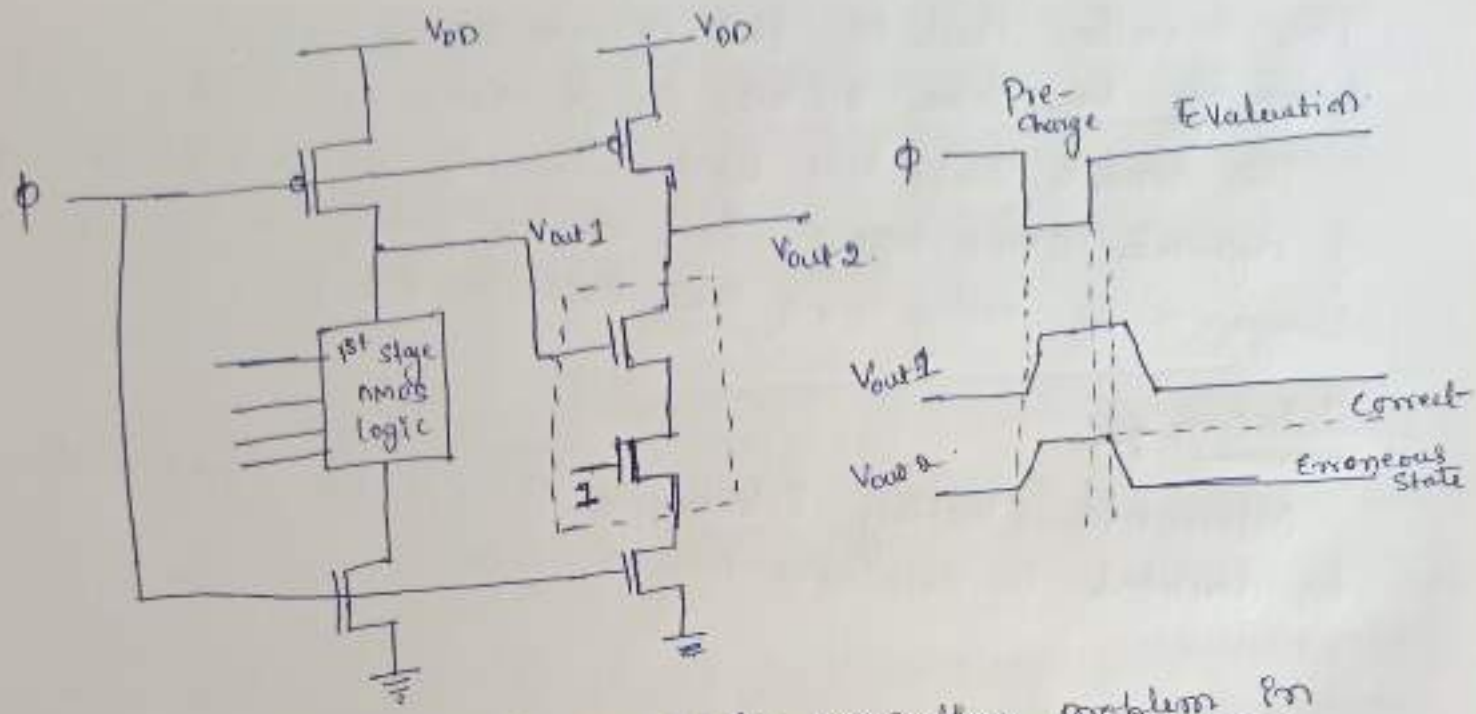


Fig:- Illustration of the cascading problem in dynamic CMOS logic.

=> During the precharge phase, both o/p voltages Vout1 & Vout2 are pulled up by the respective pMOS precharge devices. Also, the external inputs are applied during this phase.

=> The input variables of the first stage are assumed to be such that the o/p Vout1 drop to logic "0" during the Evaluation phase.

=> on the other hand, the external input of the second stage NAND2 gate is assumed to be a logic "1" as shown in Fig.

=> when the Evaluation phase begins, both O/P Vtg's Vout1 & Vout2 are logic high. The O/P of the first stage (Vout1) eventually drops to its correct logic level after a certain time delay. The Evaluation in the second stage is done concurrently, starting with the high value of Vout2 at the beginning of the Evaluation phase, will be erroneously low.

This Example illustrates that dynamic CMOS logic Stages driven by the Same Clock Signal cannot be cascaded directly. This severe limitation undermine all the other advantages of dynamic CMOS logic, Such as low power dissipation, large noise margins, & low transistor Count.

<u>Solution</u> :-

Alternative clocking Schemes & circuit structures must be developed to overcome this problem.

Module - 5

Semiconductor Memories

&

Testing and Verification

Scanned with CamScanner

Introduction

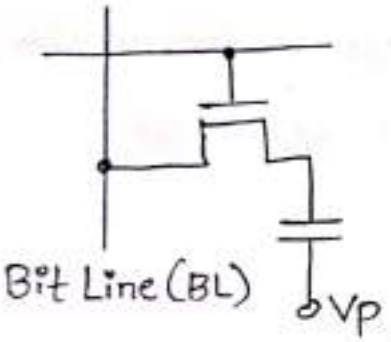
- Semiconductor memory arrays capable of storing large quantities of digital information are essential to all digital systems.

Semiconductor memories

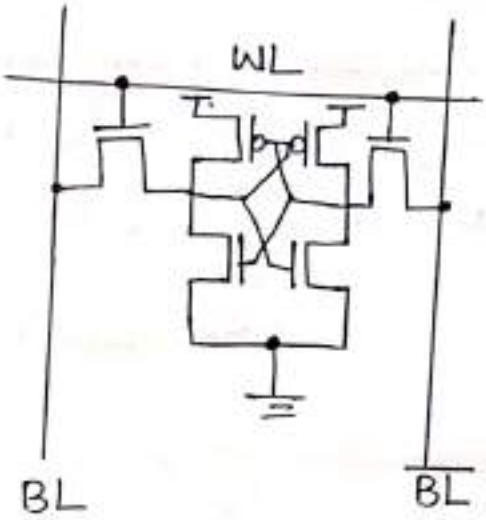
- Read/Write (R/W) Memory or Random access memory (RAM)
  - Dynamic RAM (DRAM)
  - Static RAM (SRAM)
- Read only memory (ROM)
  - * Mask (Fuse) ROM
  - * programmable ROM (PROM)
    - -> Eragable PROM (EPROM)
    - -> Electrically erasable PROM (EEPROM)
    - -> Flash Memory
    - -> Ferroelectric RAM (FRAM)

Equivalent Circuits of Memory cells

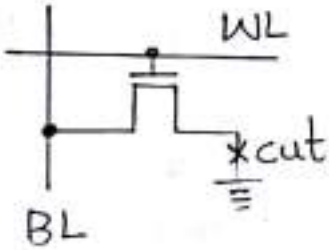
Scanned with CamScanner



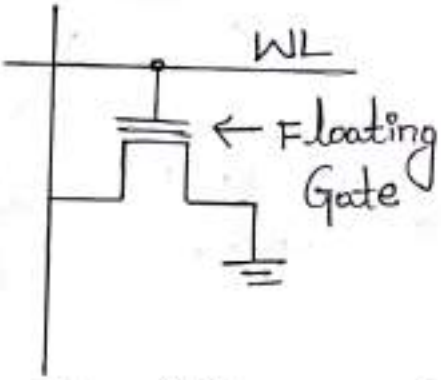
(a) DRAM



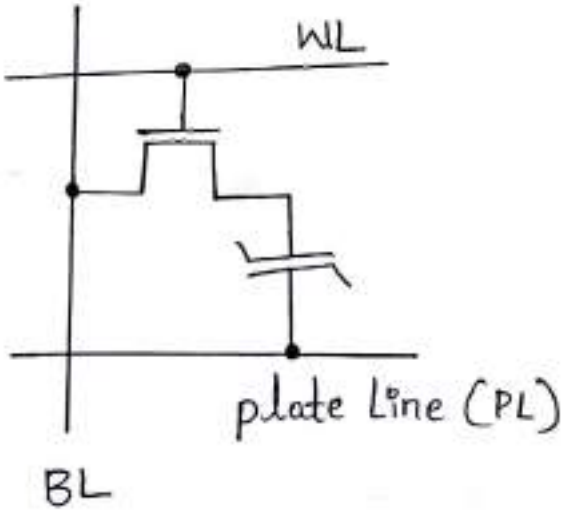
(b) SRAM



(c) Mask(Fuse)



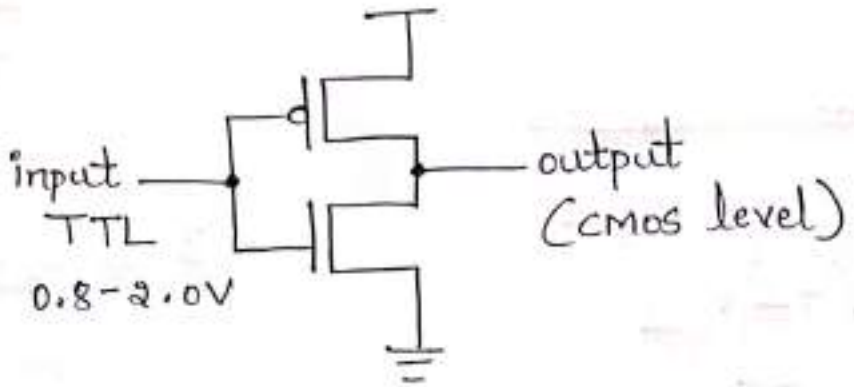
(d) EPROM(EEPROM)



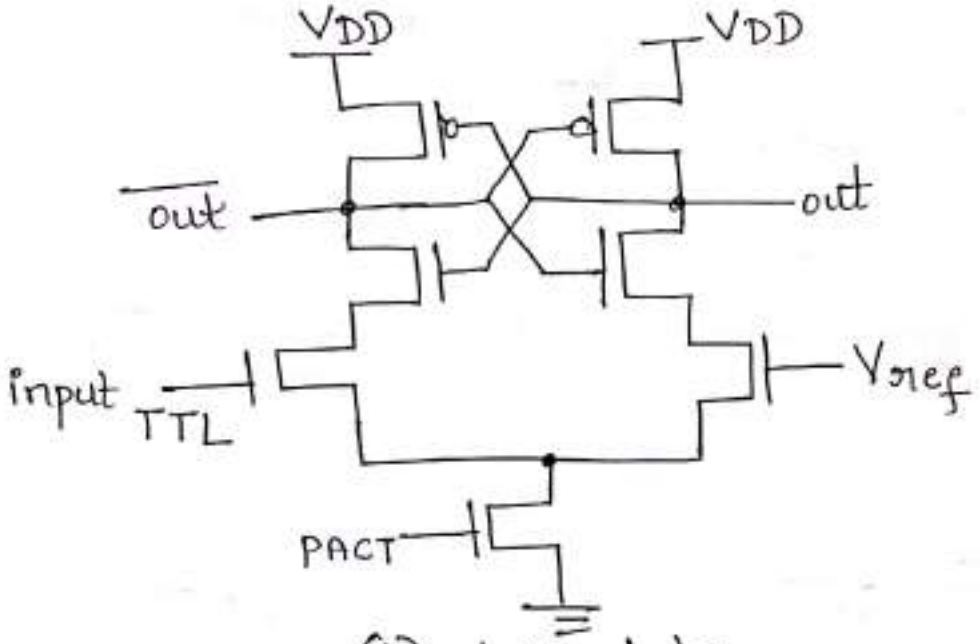
(e) FRAM

Scanned with CamScanner

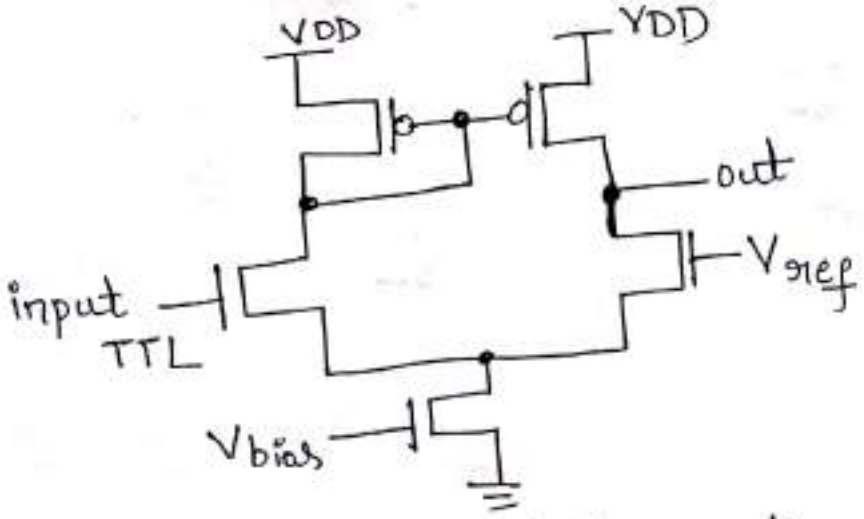
DRAM input/output Circuits



(a) Inverter type



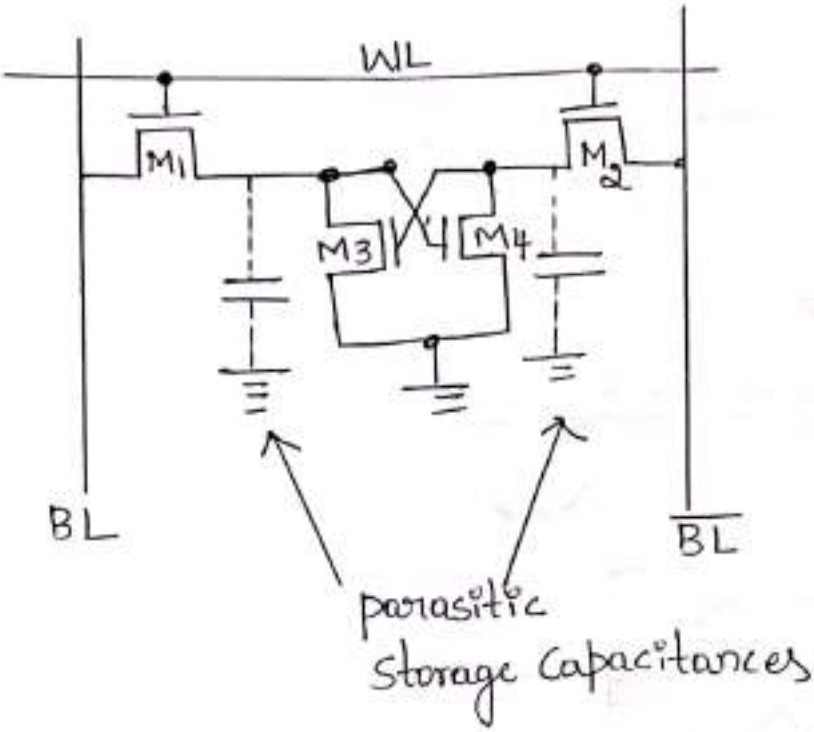
(b) latch type



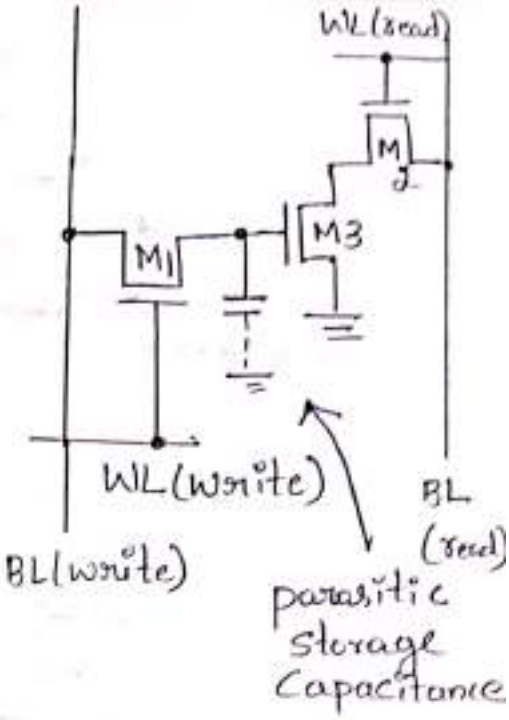
(c) differential amplifier type

Scanned with CamScanner

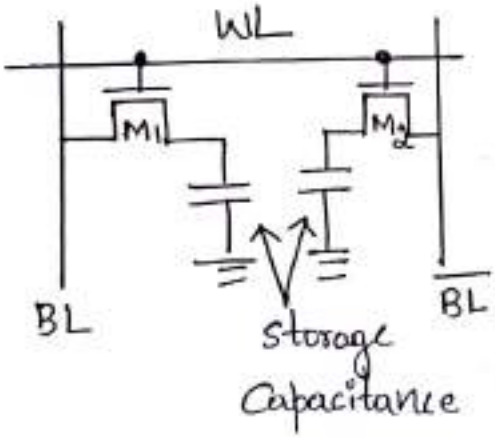
Various Configurations of DRAM cell



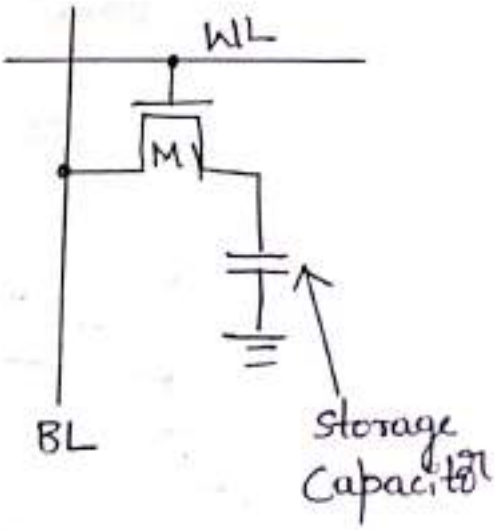
(a) Four Transistor DRAM cell



(b) Three Transistor DRAM Cell



(c) Two Transistor DRAM cell

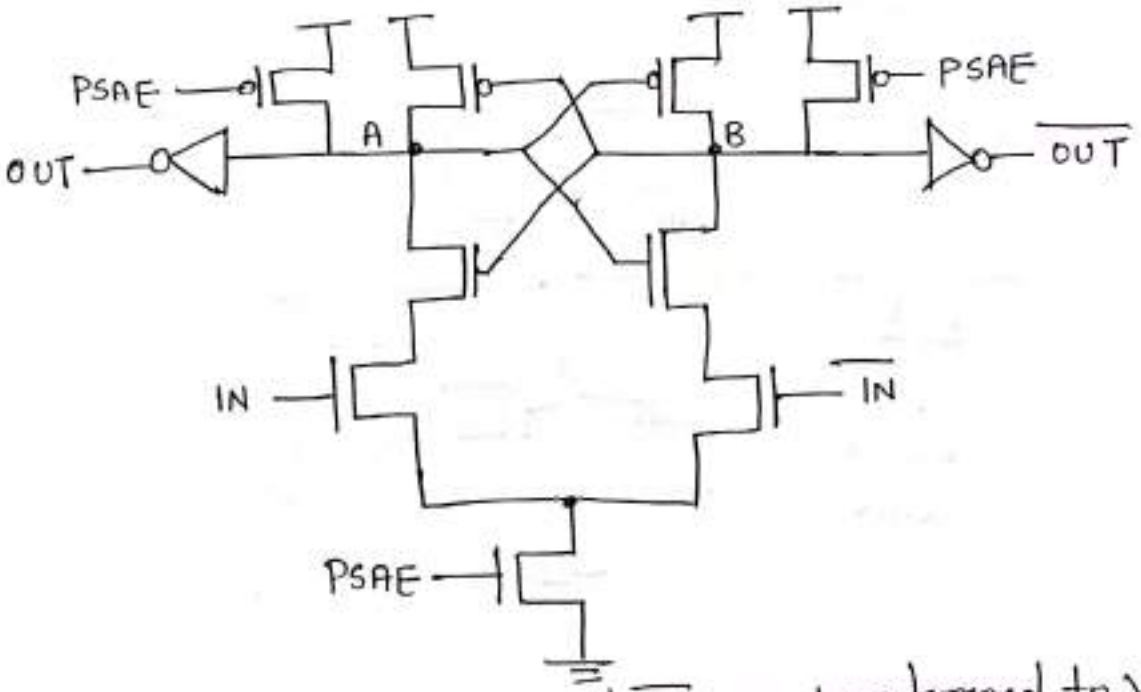


(d) one-transistor DRAM cell

Scanned with CamScanner

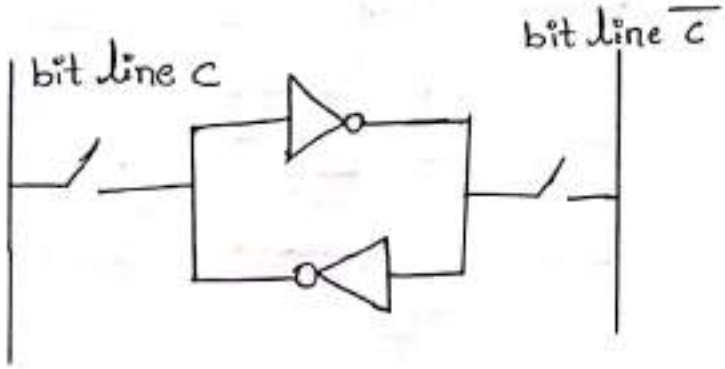
* The inverter type has the simplest structure but its sensitivity to process, temperature and operating Voltage is larger than that of other types.
* The latch type shows the fastest speed and lowest power Consumption.

Full CMOS latch type



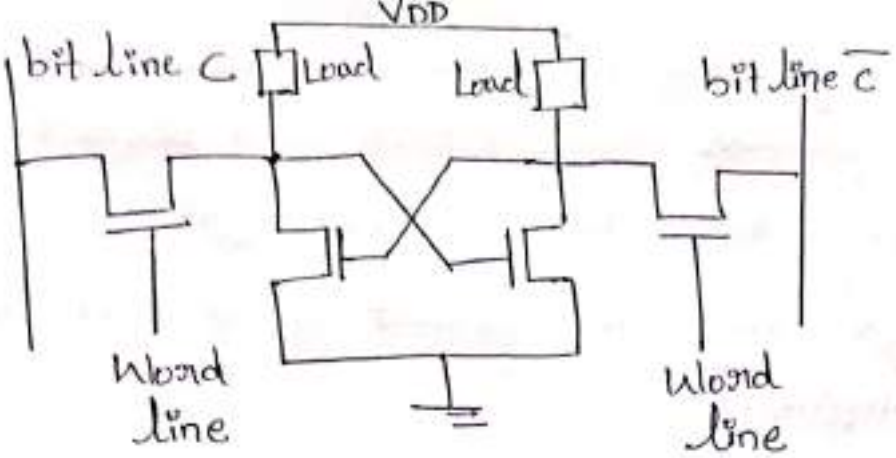
* In the full CMOS type OUT and $\overline{OUT}$ are prechanged to Vss by PSAE.

Various Configurations of SRAM

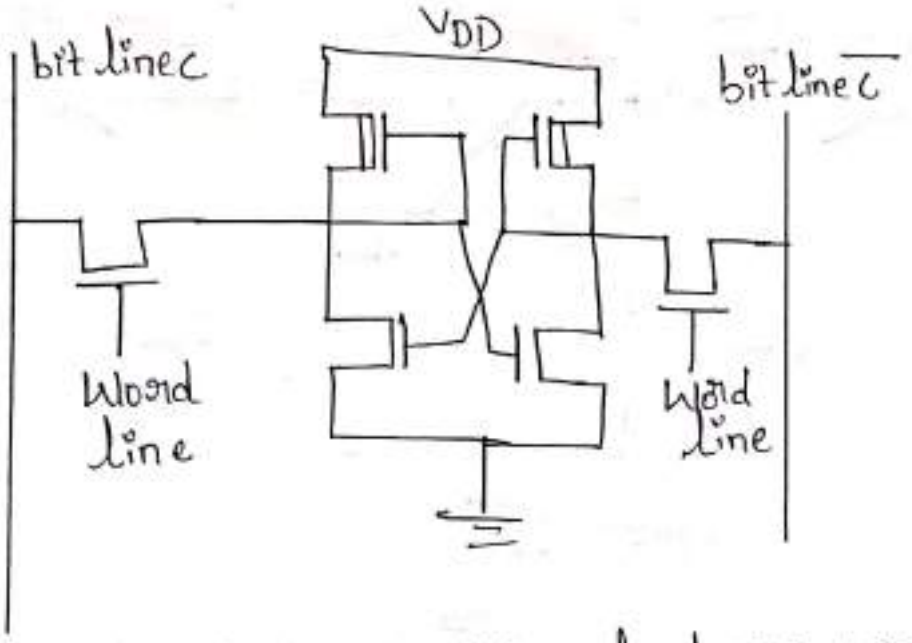


(a) Symbolic representation of two-inverter latch

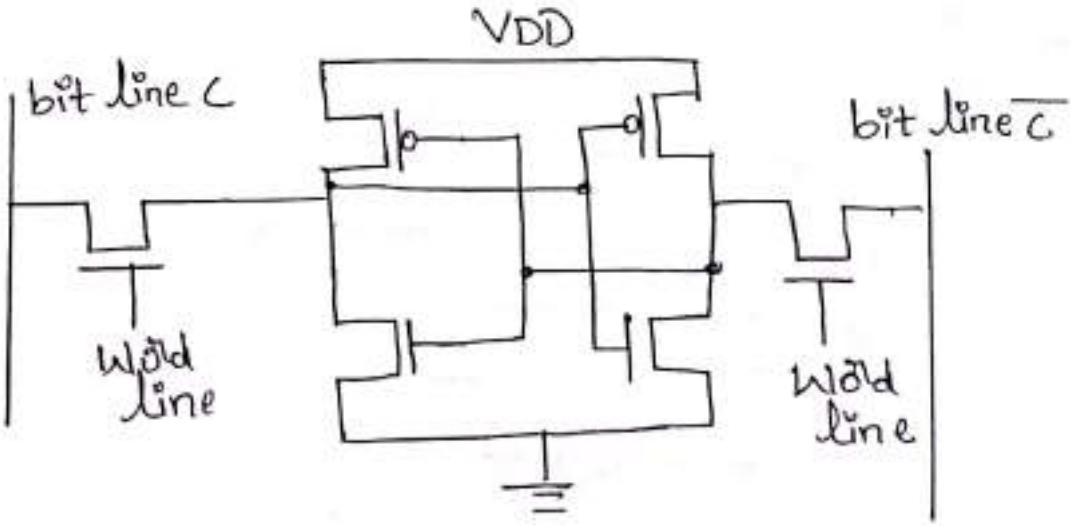
Scanned with CamScanner



(b): Mos Circuit Static RAM cell



(c) : Depletion-load nMOS SRAM cell



(d) Full CMOS SRAM cell

Scanned with CamScanner

Semiconductor Memories

Operation of Three- Transistor DRAM cell

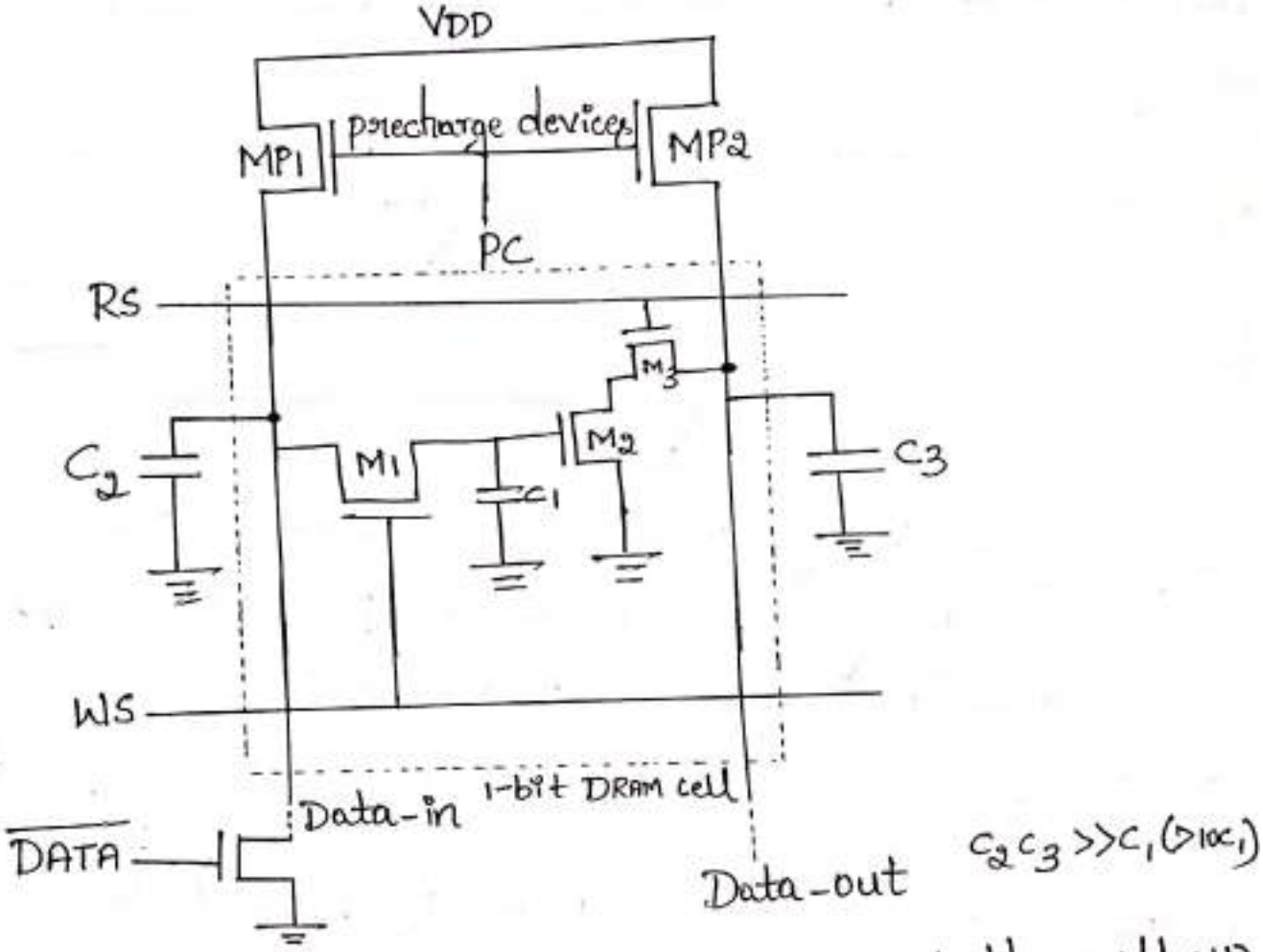


Fig : Three- transistor DRAM cell with the pull-up and read/Write Circuitry

* Here, the binary information is stored in the form of charge in the parasitic node capacitance C_1. The storage transistor M_2 is turned on or off depending on the charge stored in C_1, and the pass transistors M_1 and M_3 act as access switches for data read and Write operations.
* The cell has two separate bit lines for "data read" and "data Write", and the two separate word lines to Control the access transistors.

Scanned with CamScanner

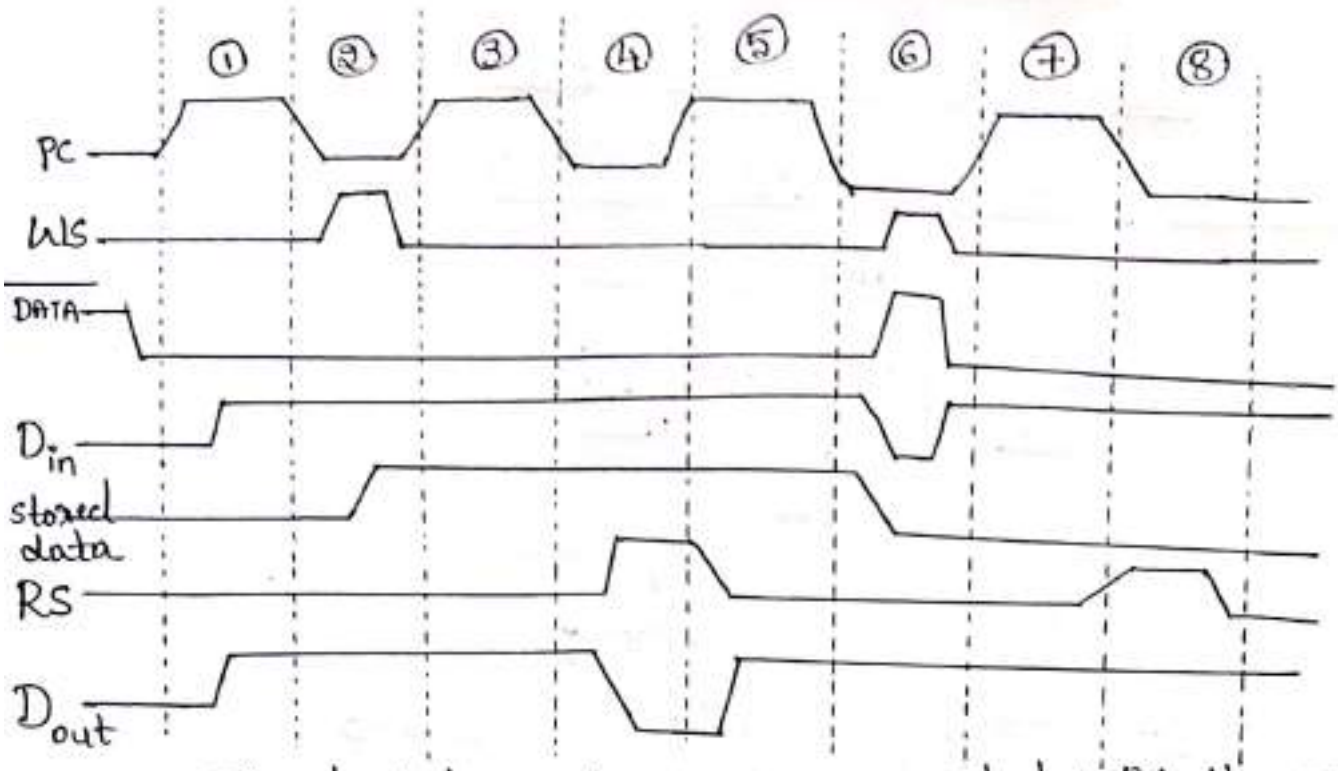
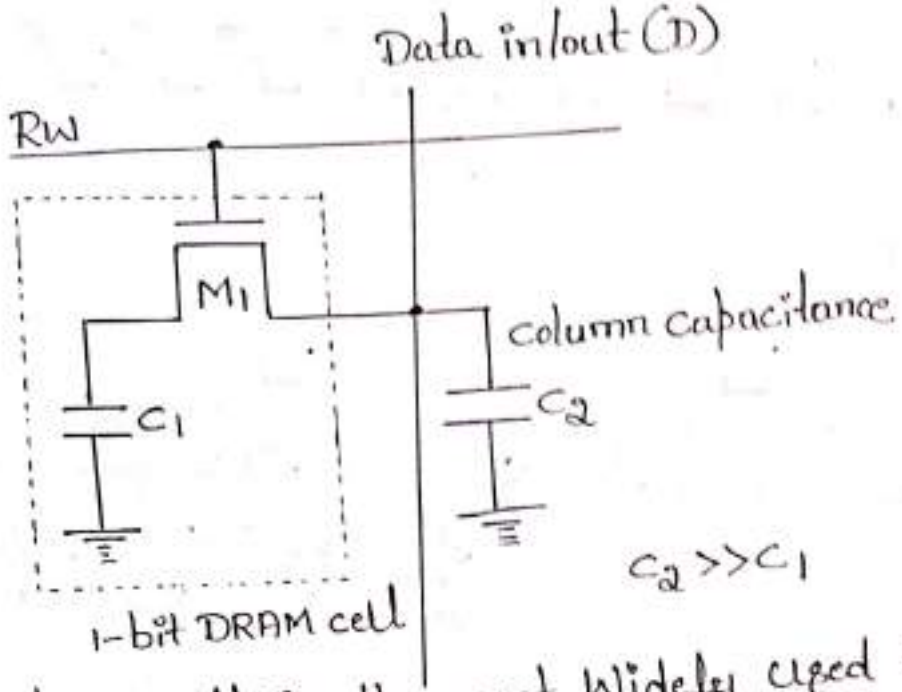


Fig: Typical Voltage Waveforms associated with the 3-T DRAM cell during four Consecutive operations. Write "1", read "1", Write "0", and read "0".

* The operation is based on a two-phase non-overlapping clock scheme.
* The precharge events are driven by ϕ_1, and the "read" and "Write" operations are driven by ϕ_2.
* Every "read" and "Write" operation is preceded by a precharge cycle, which is initiated with PC going high.
* Write "1" operation: $\overline{DATA}=0$, WS=1, RS=0 C_2 & C_1 share charge due to M_1 ON.
* Read "1" operation: $\overline{DATA}=0$, WS=0, RS=1 M_2, M_3 ON $\Rightarrow$ C_3, C_1 discharge through M_2 and M_3.
* Write "0" operation: $\overline{DATA}=1$, WS=1, RS=0 M_2 M_3 ON $\Rightarrow$ C_2 & C_1 discharge to 0.
* Read "0" operation: $\overline{DATA}=1$, WS=0, RS=1 C_3 does not discharge due to M_2 OFF.

Scanned with CamScanner

One- Transistor DRAM cell.



* One- Transistor cell is the most widely used storage structure in the DRAM industry.
* The Circuit diagram of the one-transistor (1-T) DRAM cell Consisting of one explicit storage Capacitor and one access transistor.
* In DRAM architectures, the bit lines are folded and precharged to 1/2 VDD to improve noise-immunity and reduce power-Consumption.
* The operation of one-Transistor DRAM cell are read, Write and Refresh operations.
* Write "1" operation : BL = 1, WL = 1 (M1 ON) => C1 charges to "1".
* Write "0" operation : BL = 0, WL = 1 (M1 ON) => C1 discharges to "0".
* Read operation : Destroys stored charge on C1 => destructive refresh is needed after every data read operation.

Scanned with CamScanner

Synchronous DRAM read Modes

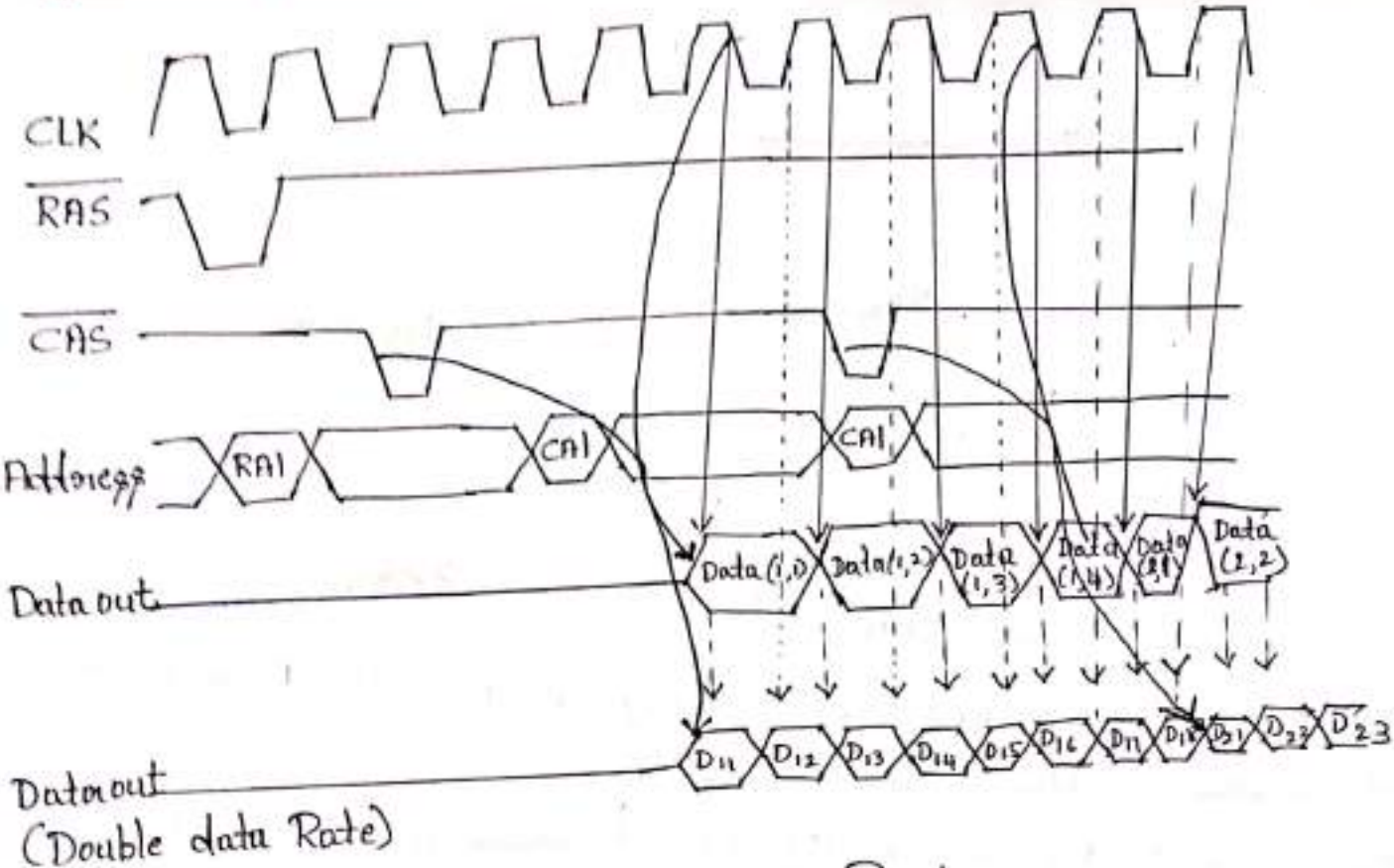


Fig a): Synchronous mode Read

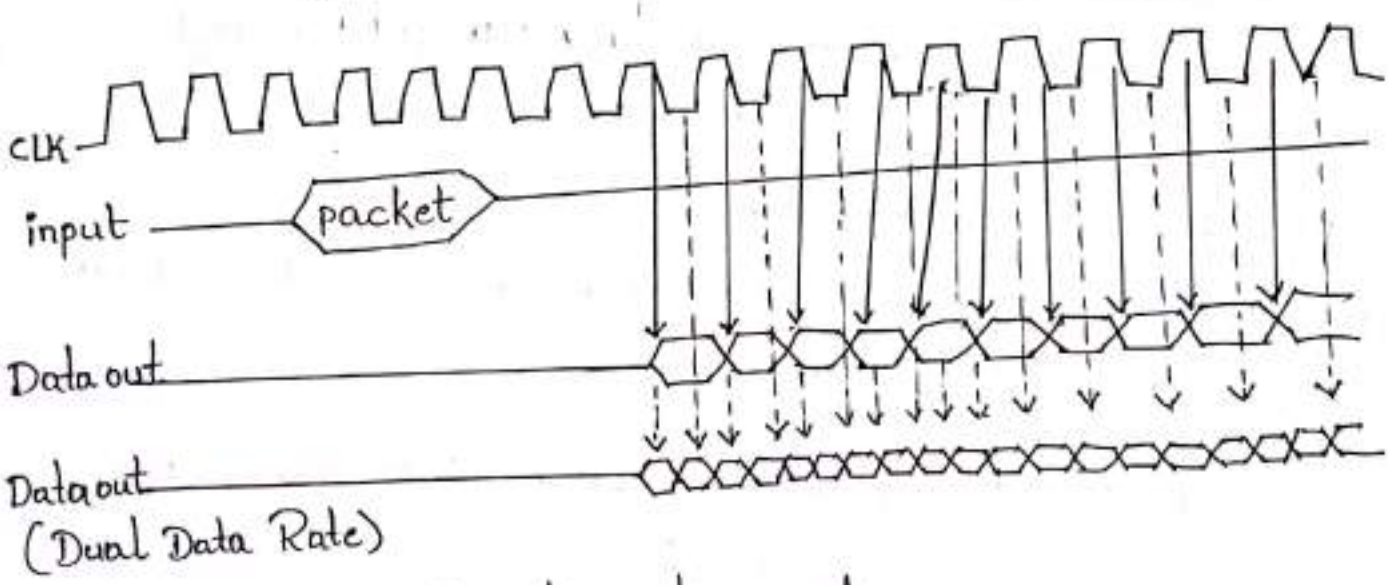


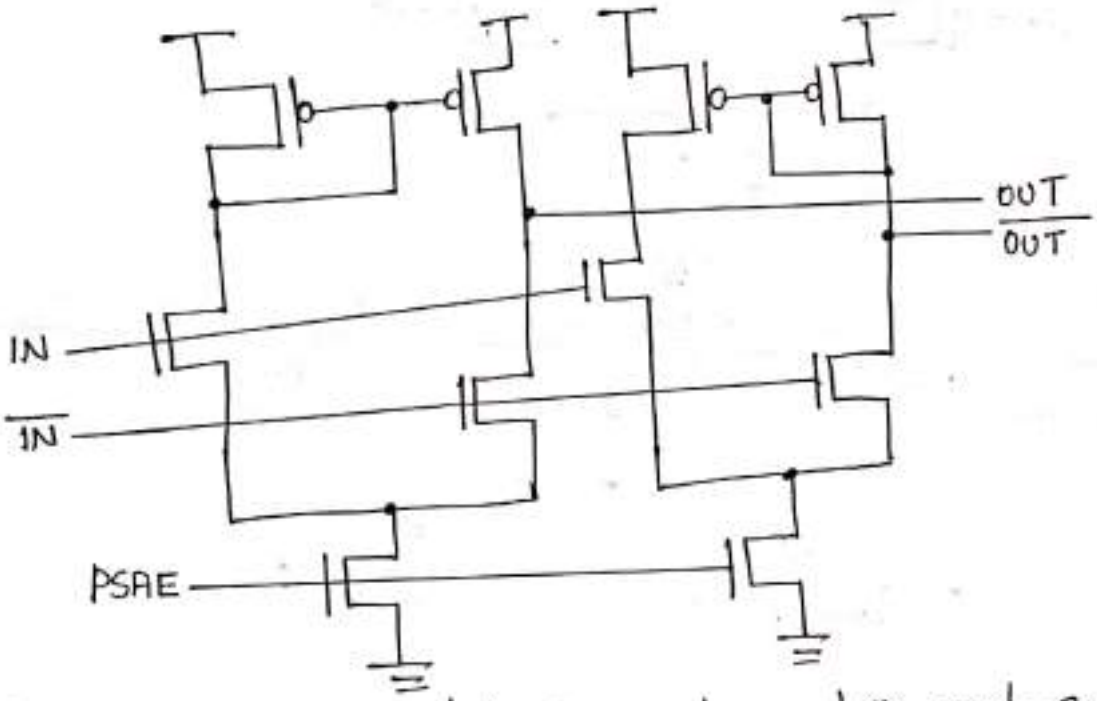
Fig b): Serial mode read

* In the Synchronous access mode, the data read frequency can be improved Very significantly with the Use of the System clock to schedule the operation.
* All DRAM operations are done by the Combination of Control and the System clock (CLK) as shown in the figure.

Scanned with CamScanner

* At the falling edge of CLK, Control Signals and addresses become active. Internal chip operations are identical but pipelined based on the system clock to improve data throughput as shown in the figure.

Differential - output Current Mirror type

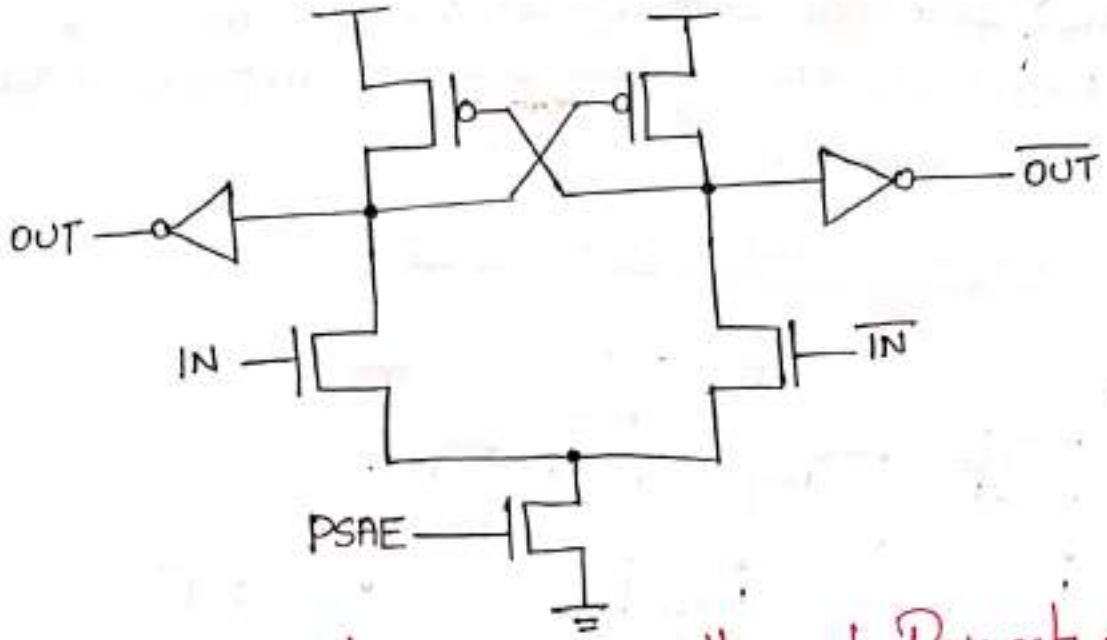


* The Current-mirror type is popular and is good-common mode rejection ratio. But relatively the large area required for large transconductance of input transistors and the large power Consumption can become limiting factors for this Circuit.
* To improve the signal Swing at the output nodes, a differential-output type is preferred shown in the figure.

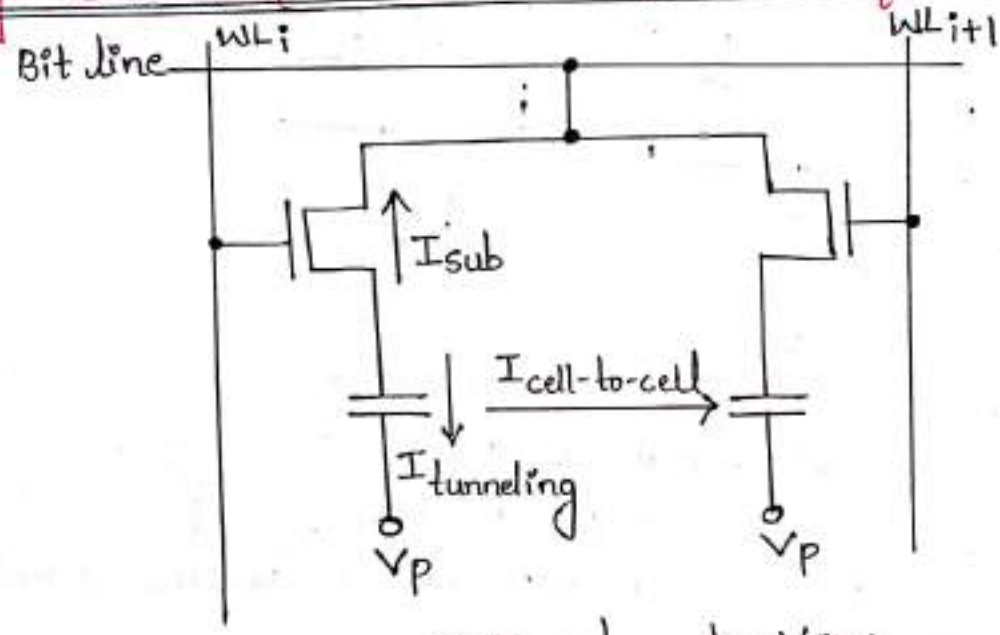
PMOS latch type

* The Cross-Coupled differential amplifier is Used to achieve high speed, low power Consumption and Small area.
* pre-charge Signal (PSAE) is required and output nodes are pre-charged properly before starting a new sensing operation.

Scanned with CamScanner



Leakage Currents in DRAM cells and Refresh operation



Fig(a): Schematic View

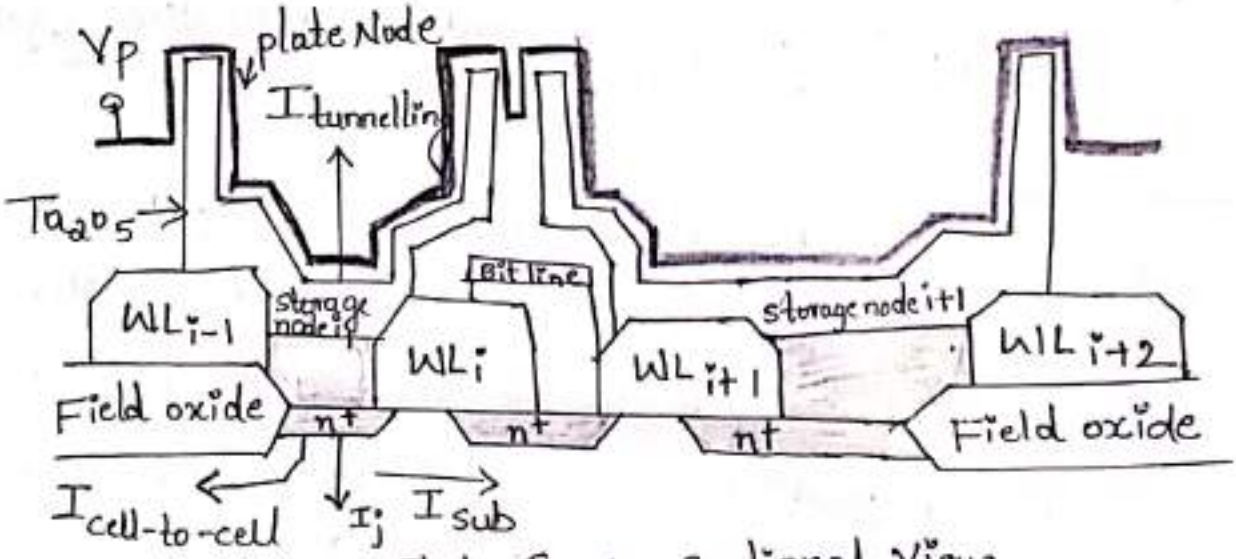


Fig b: Cross-Sectional View

Scanned with CamScanner

* Figure a) and b) shows equivalent schematic and Cross-Sectional views of DRAM cells. For high density, the contact to Connect to the cell access transistors and the bit line is shared by two adjacent cells.
* The total leakage Current is expressed as

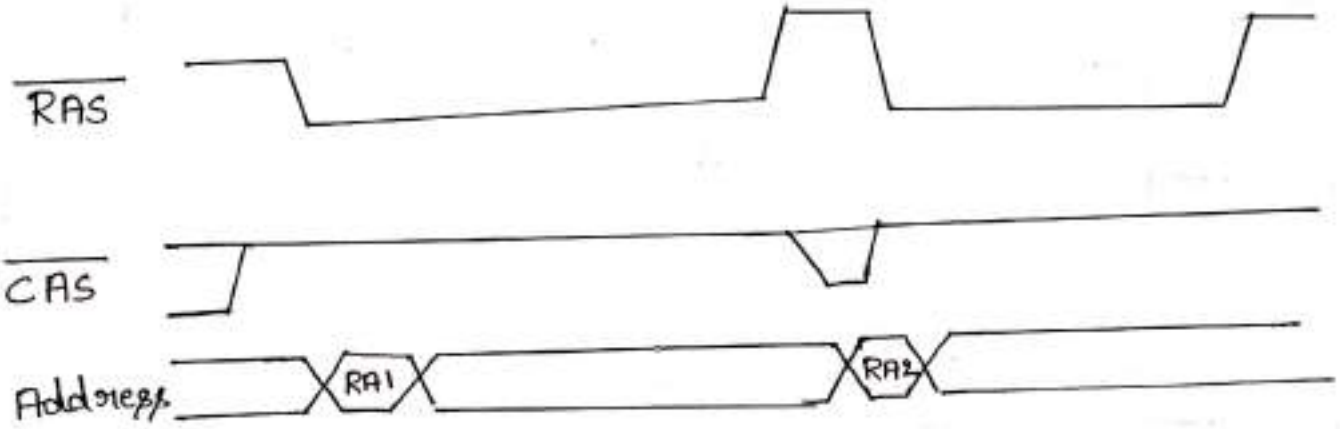
$$I_{leakage} = I_{sub} + I_{tunneling} + I_j + I_{cell\text{-}to\text{-}cell}$$

where I_{sub} is the leakage Current through the cell access transistor.

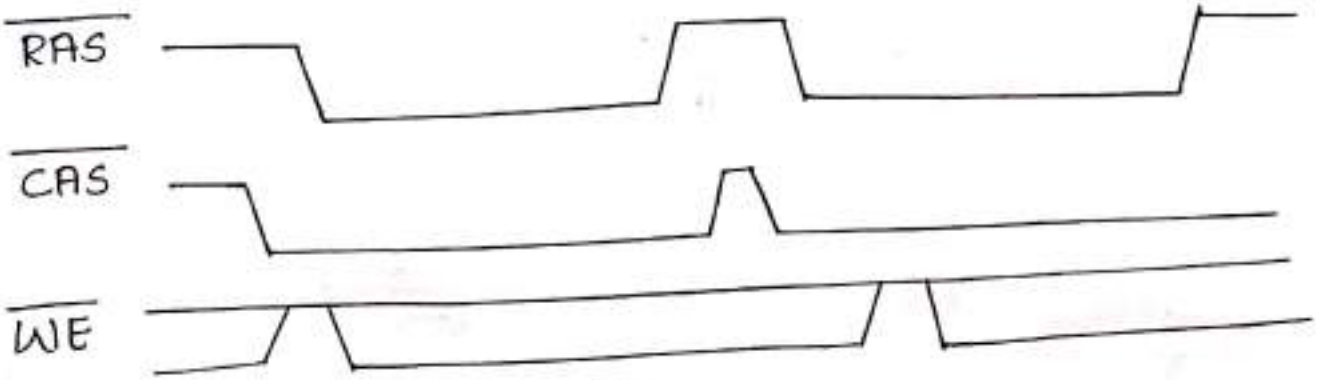
$I_{tunneling}$ Current through di-electric material

I_j is the junction leakage Current at the storage node.

$I_{cell\text{-}to\text{-}cell}$ the leakage current across field oxide.

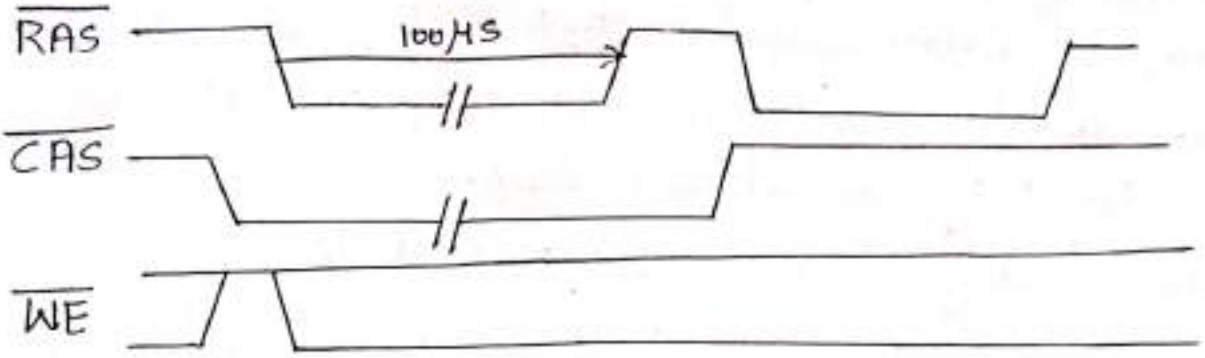


(a)



(b)

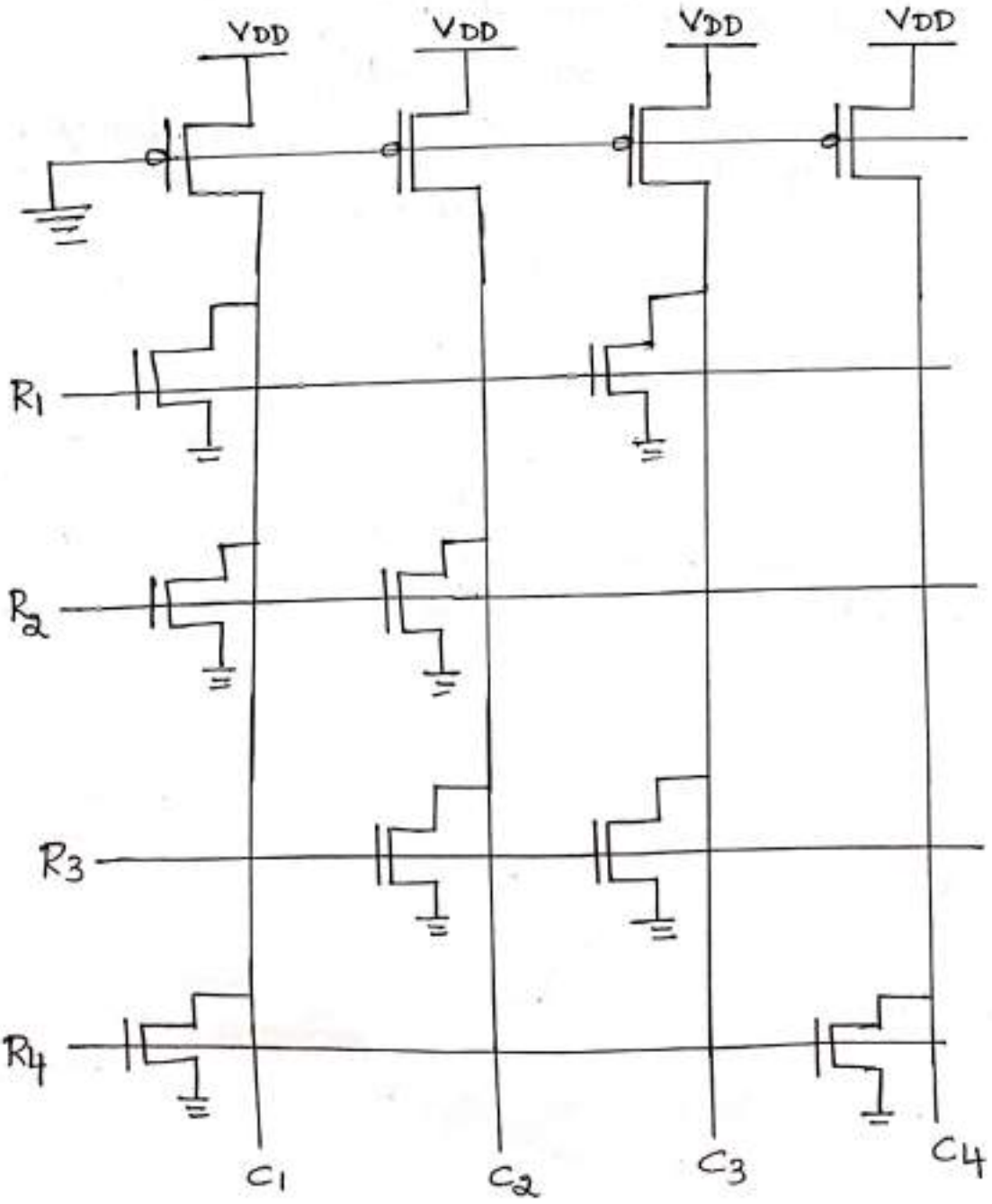
Scanned with CamScanner



(c)

Fig: Timing diagram for Various DRAM refresh operation modes. a) RoR refresh b) CBR refresh c) Self refresh

4-bit X 4-bit NOR-based ROM array



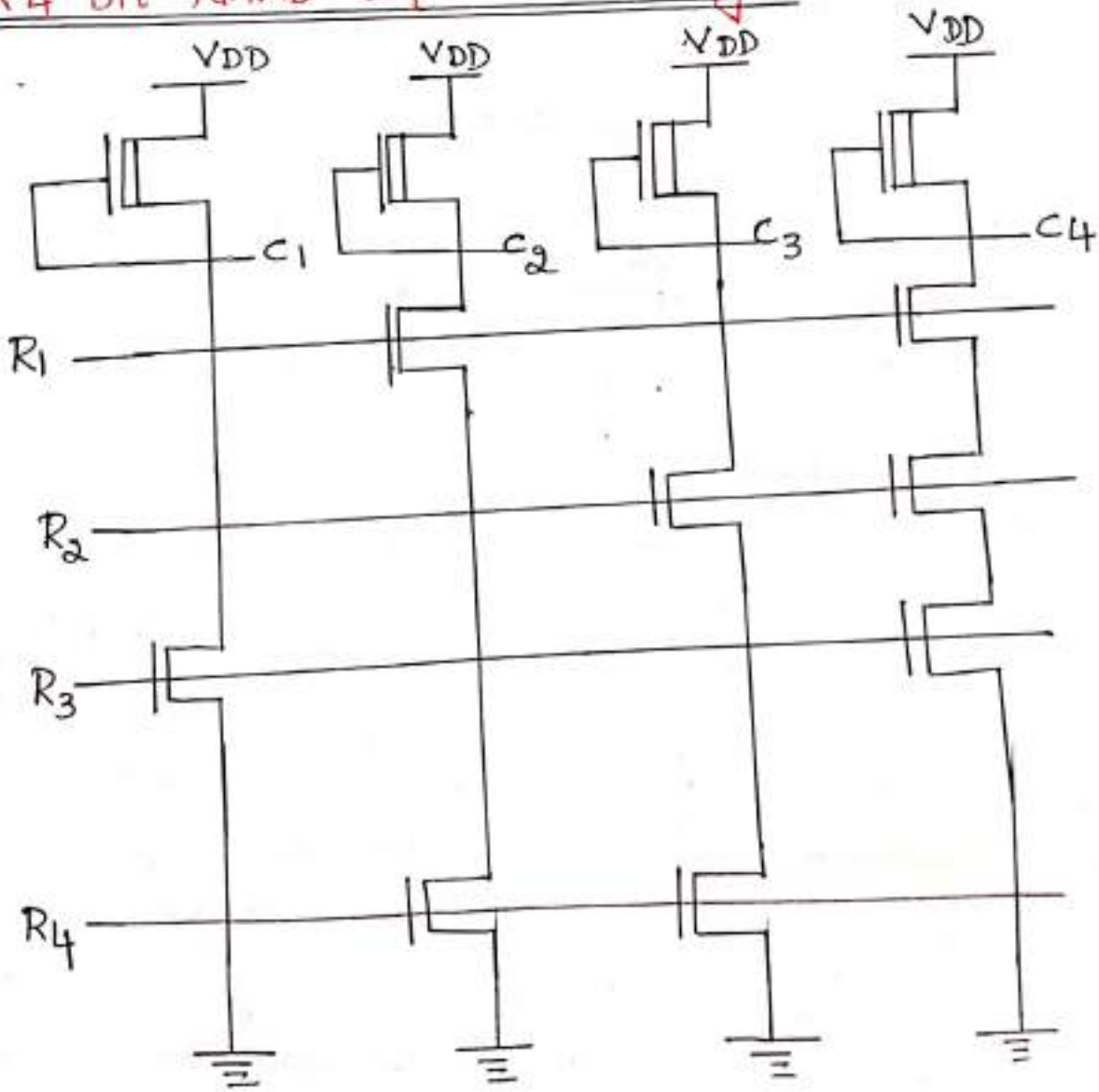
Scanned with CamScanner

Truth table

R_1	R_2	R_3	R_4	C_1	C_2	C_3	C_4
1	0	0	0	0	1	0	1
0	1	0	0	0	0	1	1
0	0	1	0	1	0	0	1
0	0	0	1	0	1	1	0

* Each column consist of a pseudo-nMOS NOR gate driven by some of the row signals.

4bit x 4-bit NAND-based ROM array



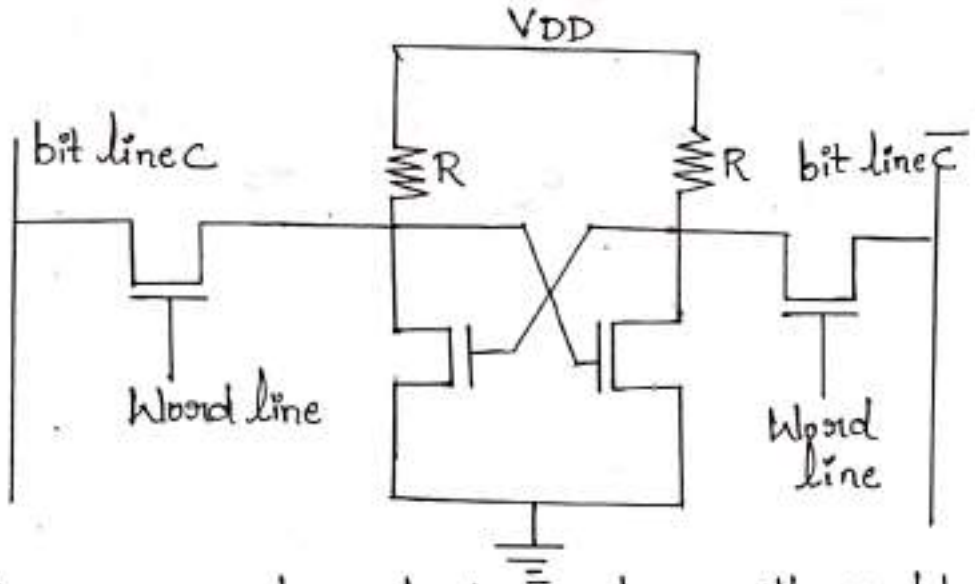
Scanned with CamScanner

Truth table

R_1	R_2	R_3	R_4	C_1	C_2	C_3	C_4
0	1	1	1	0	1	0	1
1	0	1	1	0	0	1	1
1	1	0	1	1	0	0	1
1	1	1	0	0	1	1	0

* Each bit line consists of a depletion-load NAND gate, driven by some of the row signals.

Resistive-Load SRAM cell



* The use of resistive-load inverters with undoped polysilicon in the latch circuitry structure typically results in a significantly more compact cell size, compared with the other alternatives as shown in the above figure.
* If multiple polysilicon layers are available, one layer can be used for the gates of the enhancement-type nMOS transistors, while another can be used for the load resistors and interconnects.
* Basic cross coupled 2 inverters latch with 2 stable opposite

Scanned with CamScanner

points for storing one-bit.

* pass transistors to activated by a row select (RS) signal to enable read/write operators.
* SRAM cell is accessed via two bit (column) lines C and its complement for reliable operation.

Full CMOS SRAM cell

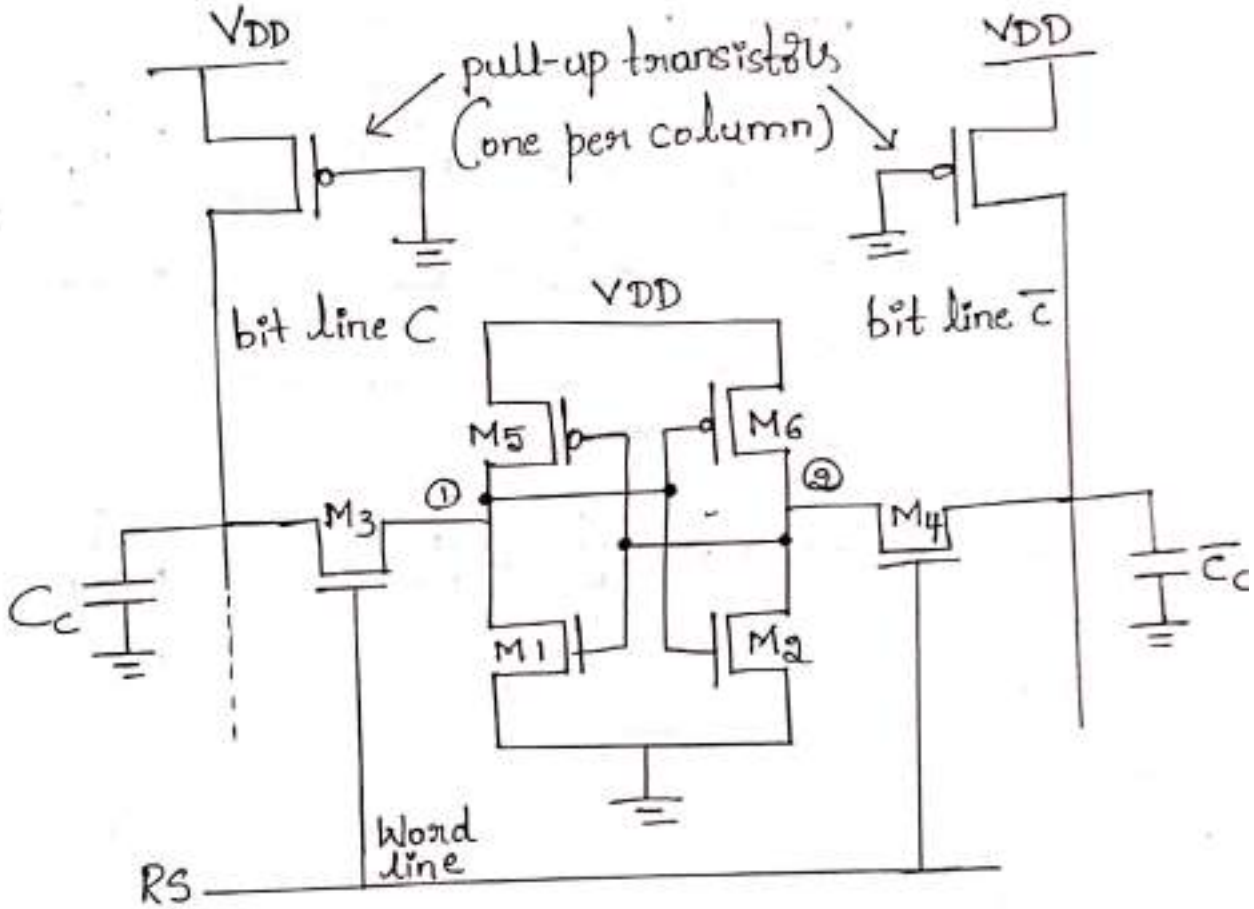


Fig: Circuit topology of the CMOS SRAM cell

* The circuit structure of the full CMOS static RAM cell is shown in the figure along with the PMOS column pull-up transistors on the complementary bit lines.
* The memory cell consists of a simple CMOS latch and two complementary access transistors (M3 and M4).
* The cell will preserve one of its two possible stable states, as long as power supply is available.

Scanned with CamScanner

* The access transistors are turned on whenever a word line is activated for read or write operation, Connecting the cell to the Complementary bit-line Columns.

Random-access Memory array organization

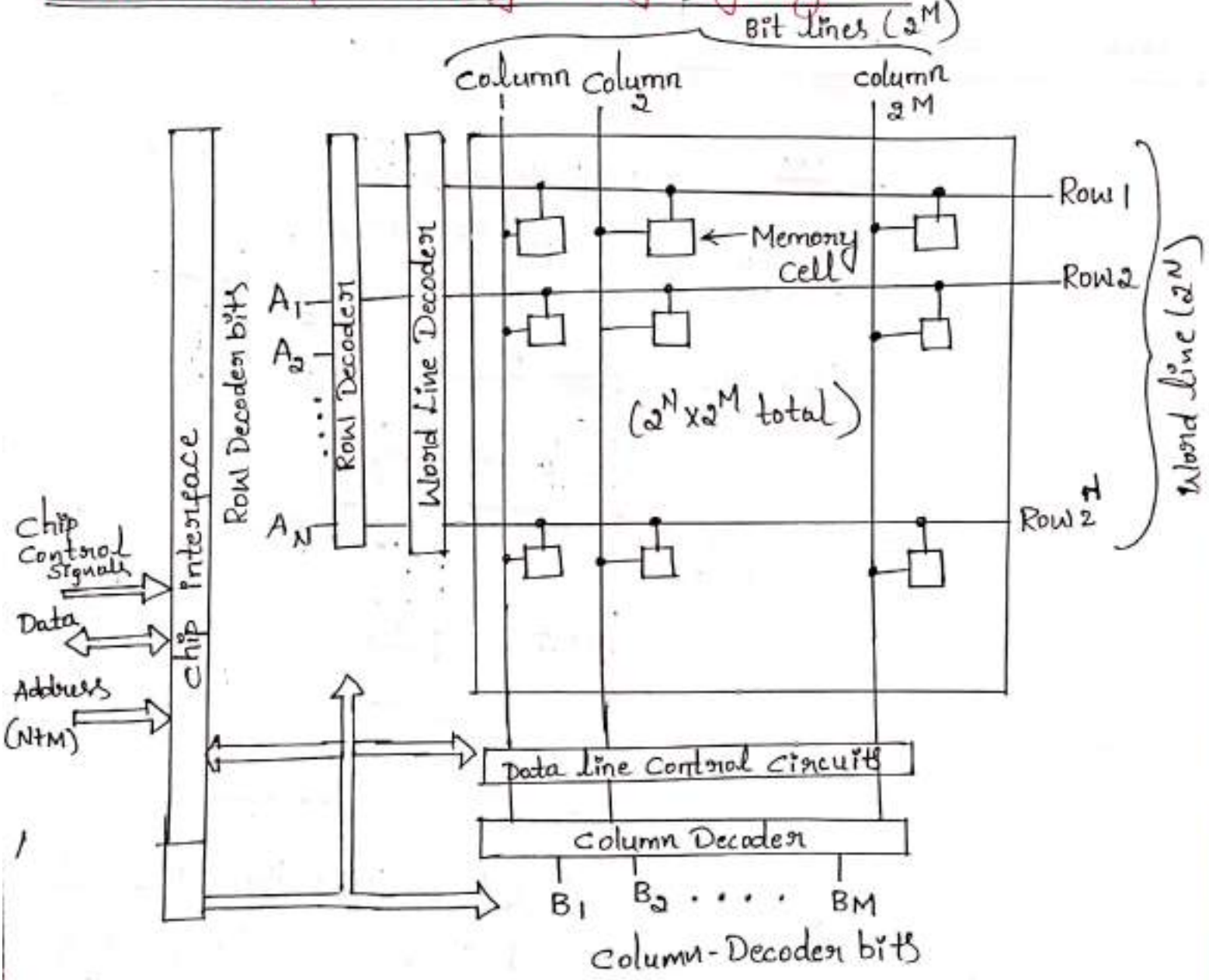


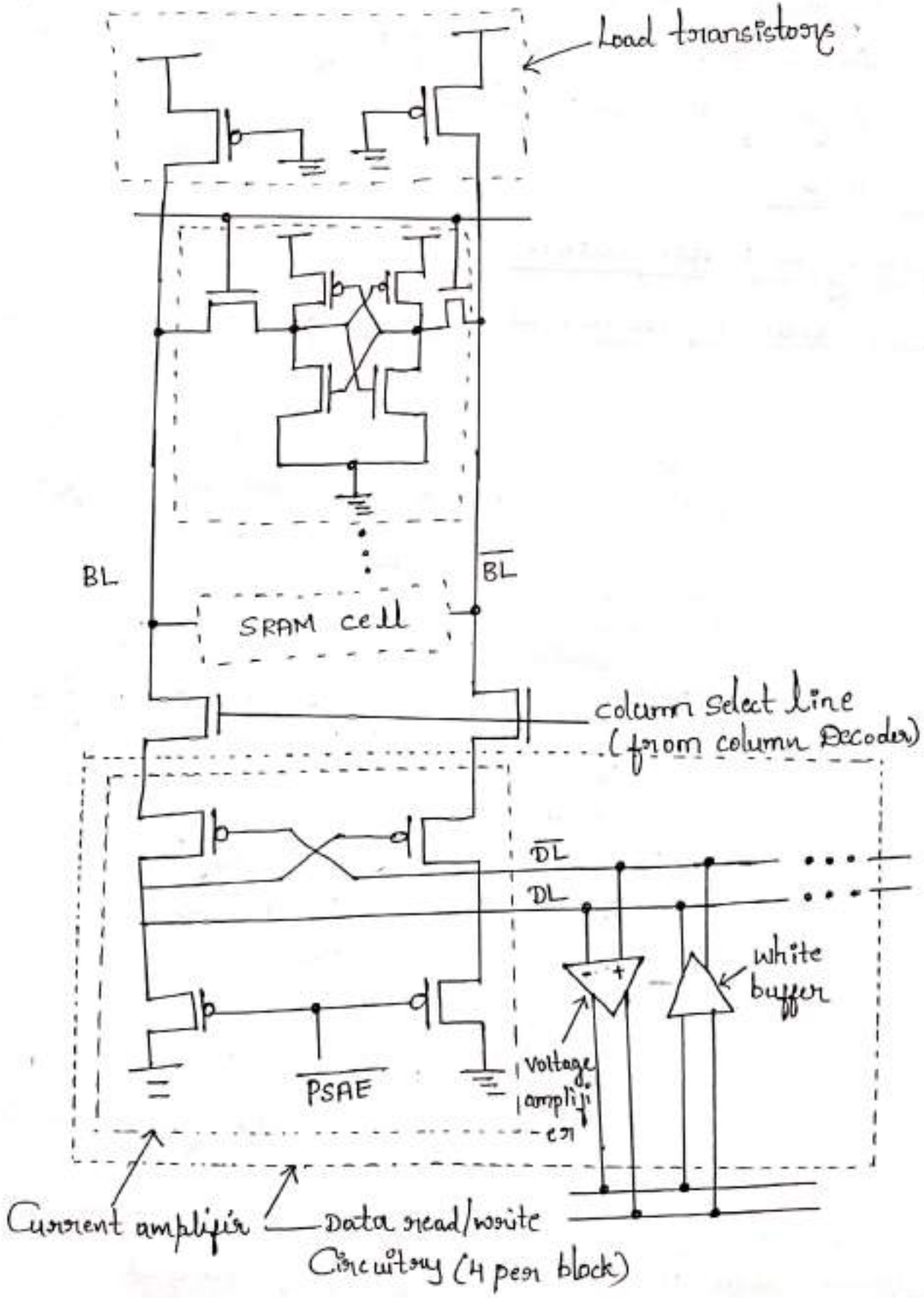
Fig: Conceptual random-access memory array organization

* The data storage structure, or core, Consists of individual memory cells arranged in an array of horizontal rows and Vertical columns.
* Each cell is capable of storing one bit of binary information.
* To access a particular memory cell, i.e., a particular data

Scanned with CamScanner

in this array, the Corresponding bit line must be activated according to the address Coming from the outside of the memory array.

Memory structure of SRAM with read and Write Circuitry



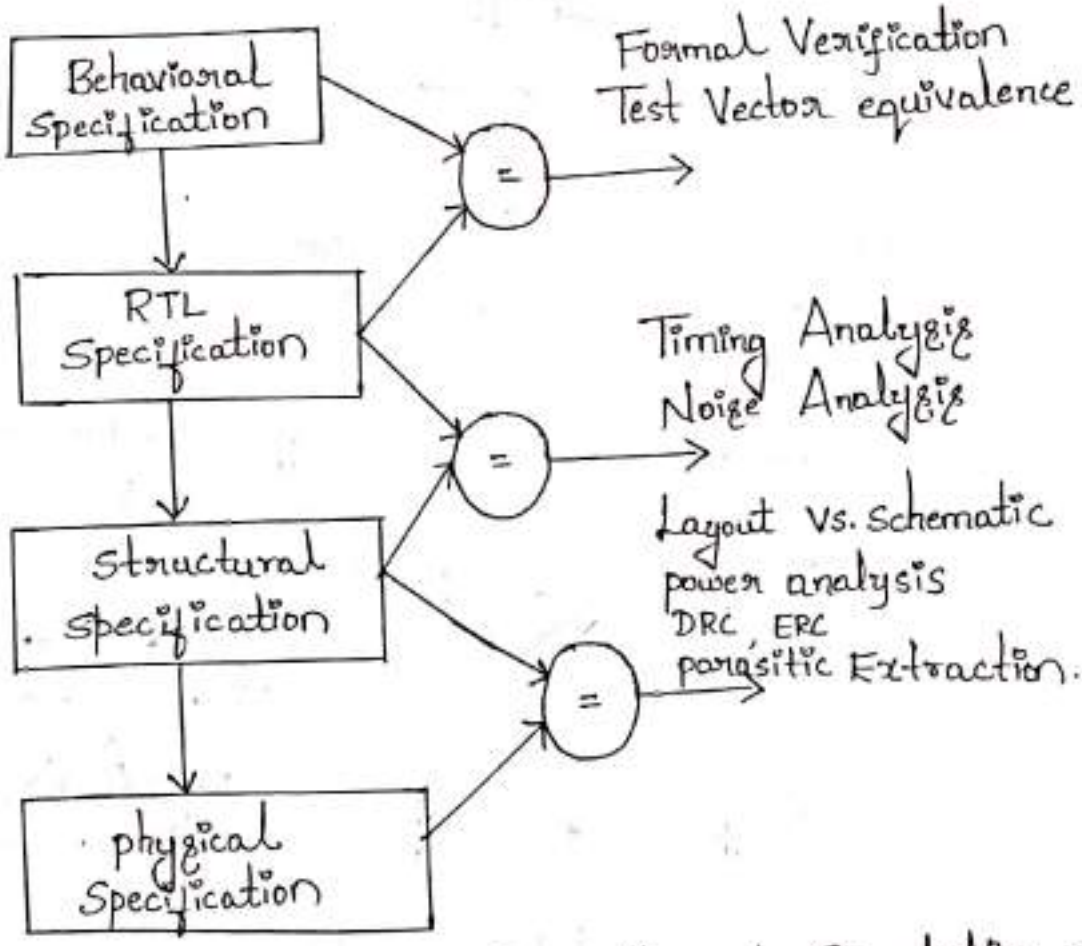
Scanned with CamScanner

* Typically a multi-stage amplifier is used to improve the read speed. A high gain Voltage or Current-mode amplifier, shown in the figure.
* The first stage amplifier and the Voltage mode amplifier to generate CMOS level signals with a large Current driving capability is used as the last stage of the amplifiers.

part - B

Testing and Verification

Functional equivalence at Various levels of abstraction



* We can check functional equivalence through Simulation at Various levels of a design hierarchy.
* If the description is at the RTL level, the behaviour at a system level may be able to be fully Verified.

Scanned with CamScanner

* At each level, you can write small tests to verify the equivalence between the new higher-level functional model and the lower-level gate or functional level.

Debugging :-

Many times when a chip returns from fabrication, the first set of tests are run in lab environment, you can begin by constructing a circuit board that provides the following attributes:

* power for the IC with ability to vary V_{DD} and measure power dissipation.
* Real-world signal connections (i.e., analog and digital inputs and outputs are required).
* clock inputs as required (it is helpful to have a stable variable-frequency clock generator)
* A digital interface to a PC (either serial or parallel ports for slow data or PCI bus for fast data interchanges).

Manufacturing Tests

Whereas verification or functionality tests seek to confirm the function of a chip as a whole, manufacturing tests are used to verify that every gate operates as expected. The need to do this arrises from a number of manufacturing defects that might occur during either chip fabrication or accelerated life testing. Typical defects include the following.

* Layer-to-Layer shorts (e.g., metal-to-metal)
* Discontinuous wires (e.g., metal thing when crossing vertical topology jumps)
* Missing or damaged vias

Scanned with CamScanner

* shorts through the thin gate oxide to the Substrate or Well.

These in turn lead to particular circuit maladies, including the following

* Nodes shorted to power or ground
* Nodes shorted to each other
* Inputs floating / outputs disconnected.

Logic Verification principles

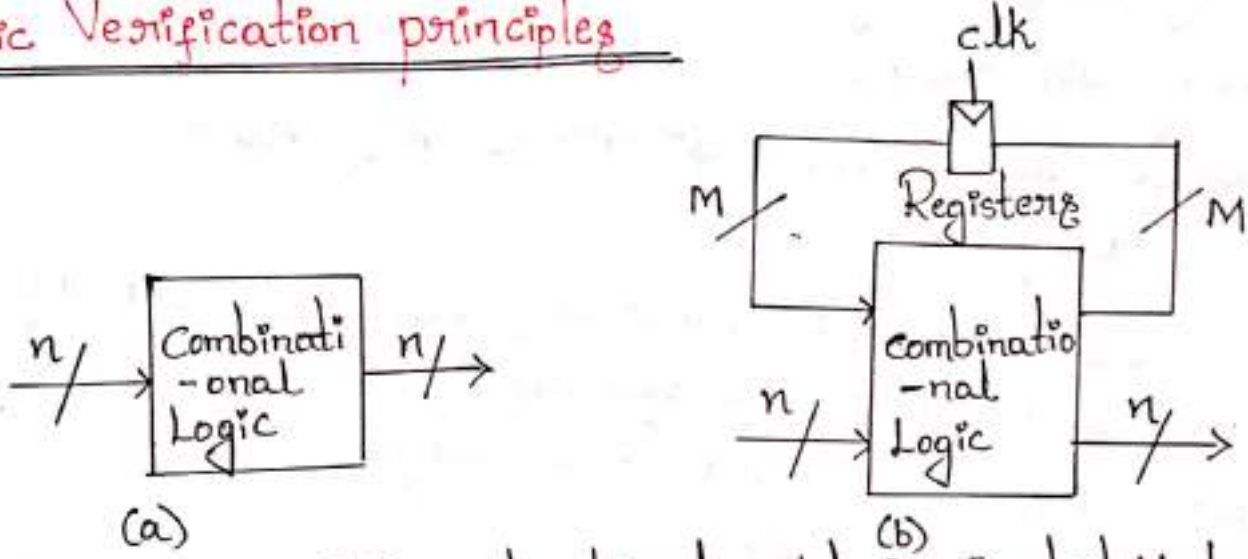


Fig : The combinational explosion in test Vectors

* Figure a) shows a Combinational Circuit with N inputs. To test this Circuit exhaustively, a sequence of 2^N inputs must be applied and observed to fully exercise the Circuit.
* This Combinational Circuit is Converted to a Sequential Circuit with addition of M registers, as shown in fig b). The state of the Circuit is determined by the inputs and the previous state. A minimum of 2^{N+M} test Vectors must be applied to exhaustively test the circuit.
* With LSI, this may be a Network with N=25 and M=50, or 2^{75} pattern, which is approximately 3.8×10^{22}. Assuming one had the patterns and applied them at application rate of 1 μs per pattern, the test time would be over a billion years (10^9).

Scanned with CamScanner

Test Vectors

* Test Vectors are a set of patterns applied to inputs and a set of expected outputs. Both logic Verification and Manufacturing test require a good set of test Vectors.
* The set should be large enough to catch all the logic errors and Manufacturing defects, yet small enough to keep test time reasonable.
* Directed and random Vectors are the most Common types: directed Vectors are selected by an engineer who is knowledge-able about the System.
* The Circuit Could be tested by applying all Combinations of these directed Vectors to the Various inputs.
* Always a large number of random or Semirandom Vectors is a Surprisingly good way to detect more Subtle errors.

Testbenches and Harnesses:

* A Verification test bench or harness is a piece of HDL code that is placed as a Wrapper around a Core piece of HDL to apply and check test Vectors.
* In the simplest test bench, input Vectors are applied to the module under test and at each cycle, the outputs are examined to determine whether they Comply with a predefined expected data set.

Regression Testing

* High-level language scripts are frequently Used when running large testbenches, especially for regression testing.
* Regression testing involves performing a Suite of Simulations to automatically Verify that no functionality has inadvertently changed in a module or set of modules.

Scanned with CamScanner

Version Control

Combined with regression testing is the Use of Versioning that is, the orderly management of different design iterations. Unix/Linux tools Such as CVS or Subversion are Useful for this.

Bug Tracking

* Another important tool to Use during Verification is a bug-tracking System.
* Bug-tracking systems Such as the Unix/Linux based GNATS allow the management of a Wide Variety of bugs.
* In these systems, each bug is entered and the location, nature, and Severity of the bug noted.

Manufacturing Test principles

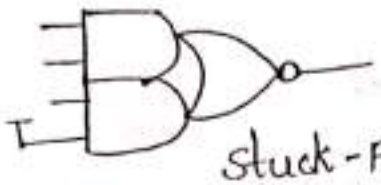
The purpose of manufacturing test is to Screen out most of the defective parts before they are shipped to Customers. Typical Commercial products target a defect rate of 350-1000 defects per million (DPM) chips shipped.

Fault Models - To deal With the existence of good and bad parts, it is necessary to propose a fault model. i,e model for how faults occur and their impact on Circuits. The most popular model is Called the Stuck-At model. The short Circuit/open Circuit model Can be a closer fit into reality, but is harder to incorporate into logic Simulation tools.

Stuck-At faults : In the Stuck-At model, a faulty gate input is modeled as a Stuck at Zero (Stuck-At-0, S-A-0) or stuck at one (Stuck-At-1, S-A-1). This model dates from board-level designs, Where it was determined to be

Scanned with CamScanner

adequate for modeling faults.



Stuck-At-1
SA1 fault.

Stuck-At-0
SA0 fault

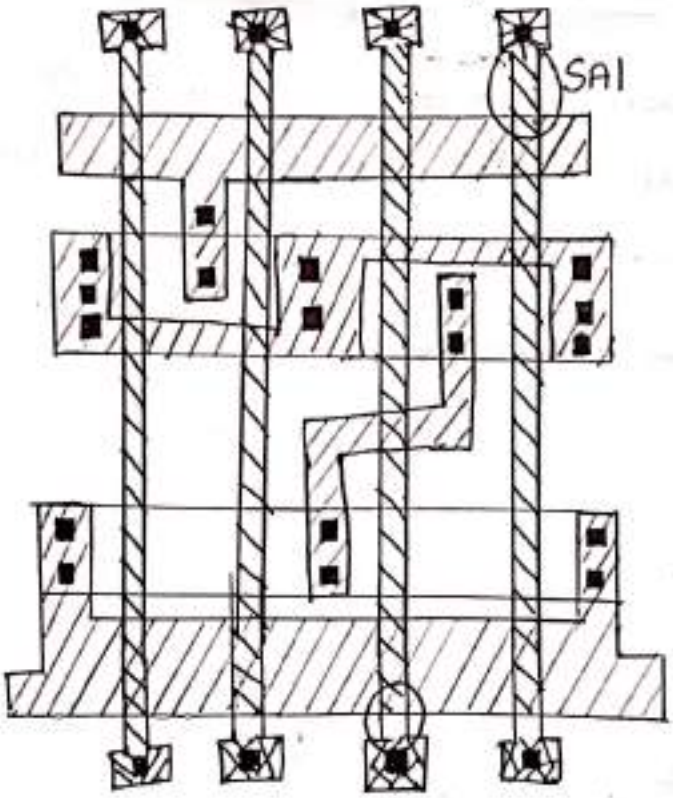


Fig: CMOS stuck At faults

Short-Circuit and open-Circuit Faults - Two bridging or shorted faults are shown in fig. The short S_1 results in an S-A-0 fault at input A, while short S_2 modifies the function of the gate.

Scanned with CamScanner

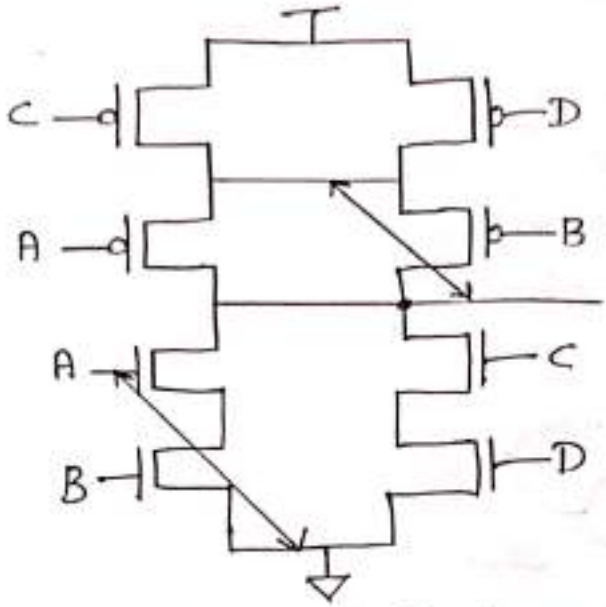
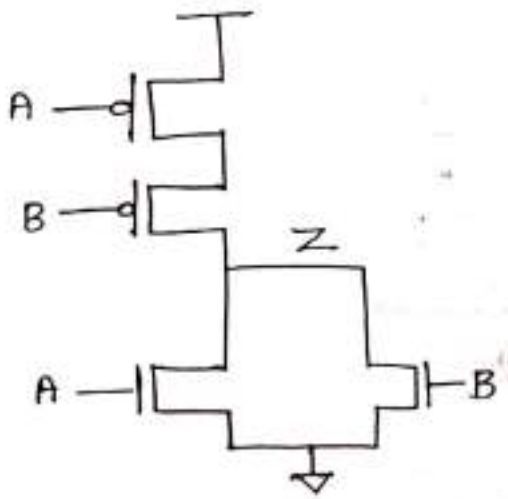


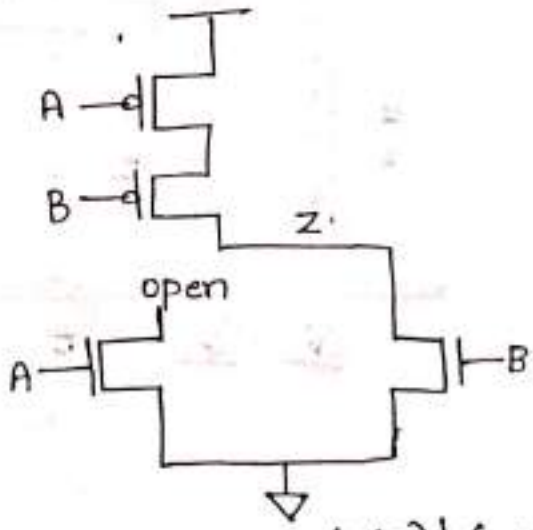
Fig: CMOS Bridging Faults

A particular problem that arrises with CMOS is that it is possible for a fault to Convert a Combinational Circuit into a Sequential Circuit.

$$Z = \overline{A+B} + \overline{B}Z'$$



$Z = \sim(A|B)$



$Z = \sim(A|B) | (\sim B \& Z')$

Scanned with CamScanner

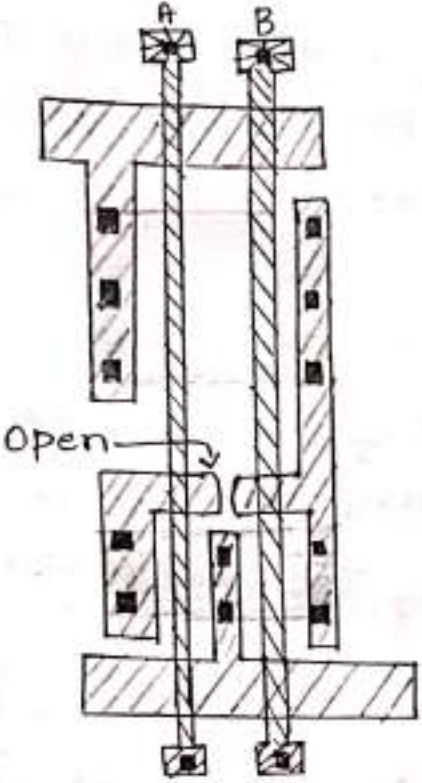


Fig : A CMOS open fault that Causes Sequential faults

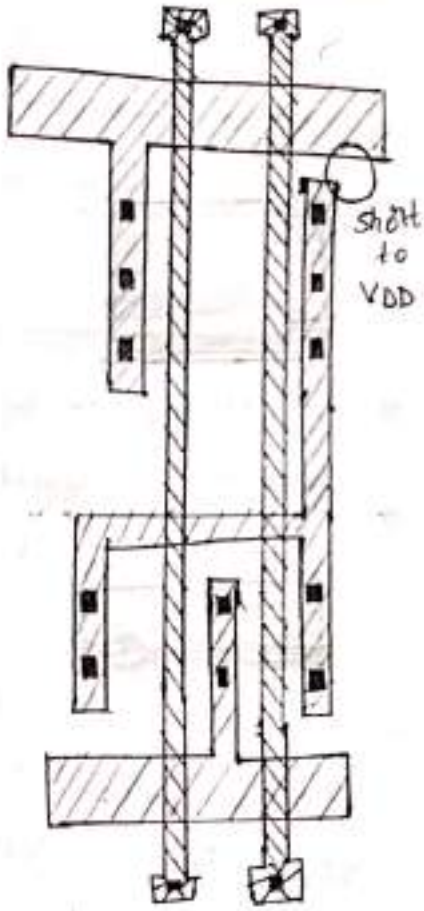


Fig : A defect that Causes static I_{DD} Current

Observability : The observability of a particular circuit node is the degree to which you can observe that node at the outputs of an integrated circuit. Given the limited number of nodes that can be directly observed, it is the aim of good chip designers to have easily observed gate outputs.

Controllability : The Controllability of an internal circuit node within a chip is a measure of the ease of setting the node to a 1 or 0 state. An easily Controllable node would be directly settable via an input pad.

Repeatability : The Repeatability of system is the ability to produce the same outputs given the same inputs. Combinatio-nal logic and Synchronous sequential logic is always repeatable when it is functioning Correctly.

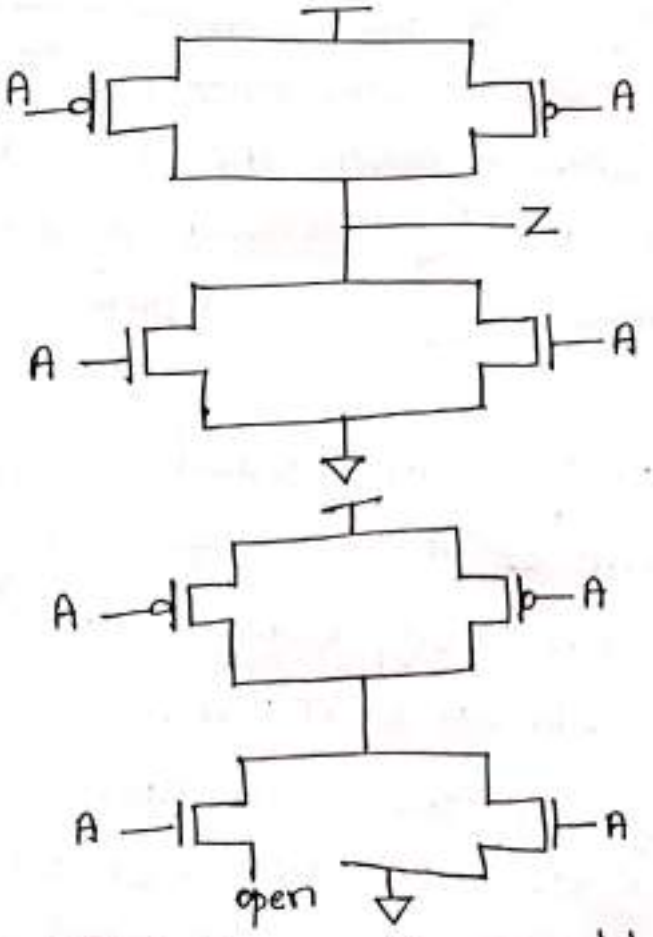
Scanned with CamScanner

Survivability : The Survivability of a System is the ability to Continue function after a fault. For example, error-Correcting codes provide Survivability in the event of Soft errors.

Fault Coverage

* A measure of goodness of a set of test Vectors is the amount of fault Coverage it achieves, That is, for the Vectors applied, what percentage of the chip's internal nodes were checked.
* Each node is taken in Sequence and held to 0 (s-A-0), and the Circuit is simulated with the test Vectors Comparing the chip outputs with a known good machine - a Circuit with no nodes artificially set to 0 (or 1).

Automatic Test pattern Generation (ATPG)



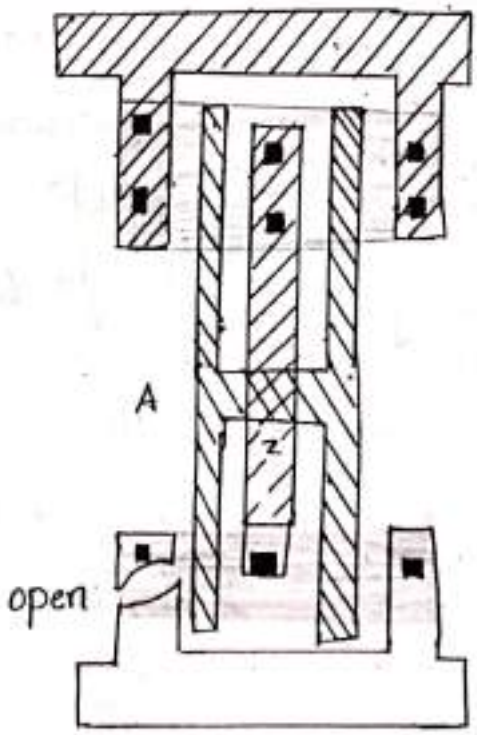


Fig : An example of a delay fault

Scanned with CamScanner

Design for Testability

Design for Testability categorized as follows :-

→ Ad hoc testing

→ Scan-based approaches

→ Built-in Self-Test (BIST)

Ad Hoc Testing

* Ad hoc test techniques, as their name suggests, are collections of ideas aimed at reducing the Combinational explosion of testing. They are only useful for small designs where scan, ATPG, and BIST are not available. The following are Common techniques for ad hoc testing.
  * partitioning large Sequential Circuits.
  * Adding test points
  * Adding Multiplexers
  * providing for easy state reset

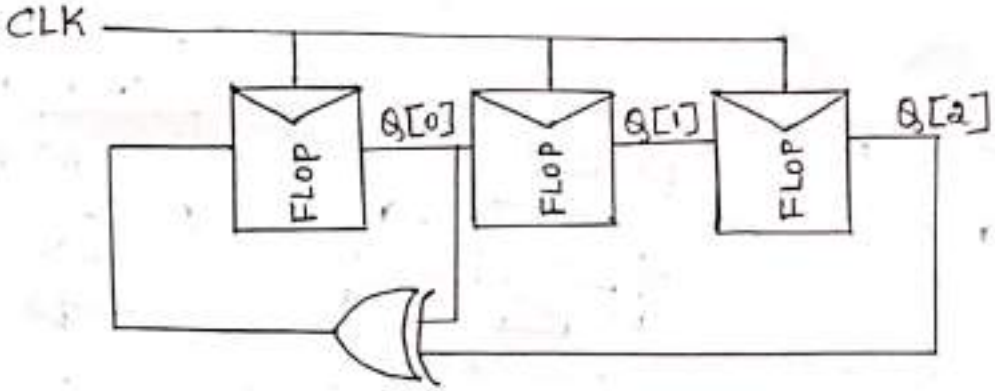
Built in Self Test (BIST)

* These techniques add area to the chip for the test logic, but reduce the test time required and thus can lower the overall system cost.
* One method of testing a module is to use Signature analysis or Cyclic redundancy checking.
* This involves a pseudo-random sequence generator (PRSG) to produce the input signals. A PRSG of length n is Constructed from a linear feedback shift register (LFSR), which intum is made of n flip-flop Connected in serial fashion.
* A Signature analyzer receives successive outputs of a Combinati-onal logic block and produces a syndrome that is

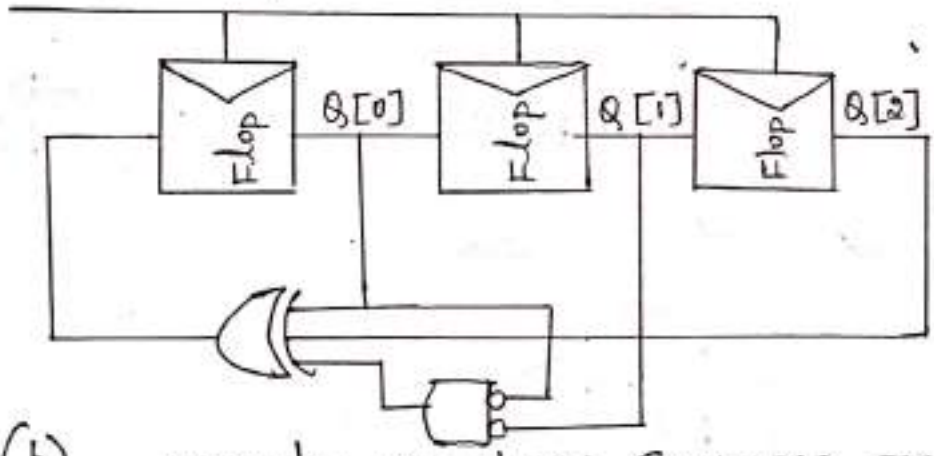
Scanned with CamScanner

function of these outputs.

* The bottom n bits of an n+1 bit LFSR Can be Used to cycle through the all Zeros state Without the delay of the NOR gate.



(a) $f(x) = 1 + x + x^3$



(b) pseudo-random Sequence register

Scan Design

* Modern Scan is based on the Use of Scan registers as shown in the below figure.
* The Scan register is a D flip-flop preceded by a multiplexer.
* When the SCAN Signal is deasserted, the register behaves as a Conventional register, storing data on the D input.
* For a Circuit to load the Scan chain, SCAN is asserted and CLK is pulsed eight times to load the first two ranks of 4-bit registered with Data.

Scanned with CamScanner

* In this Circuit, every input to the Combinational block can be Controlled and every output can be observed.
* In addition, running a random pattern of 1's and 0's through the Scan chain can test the chain itself.

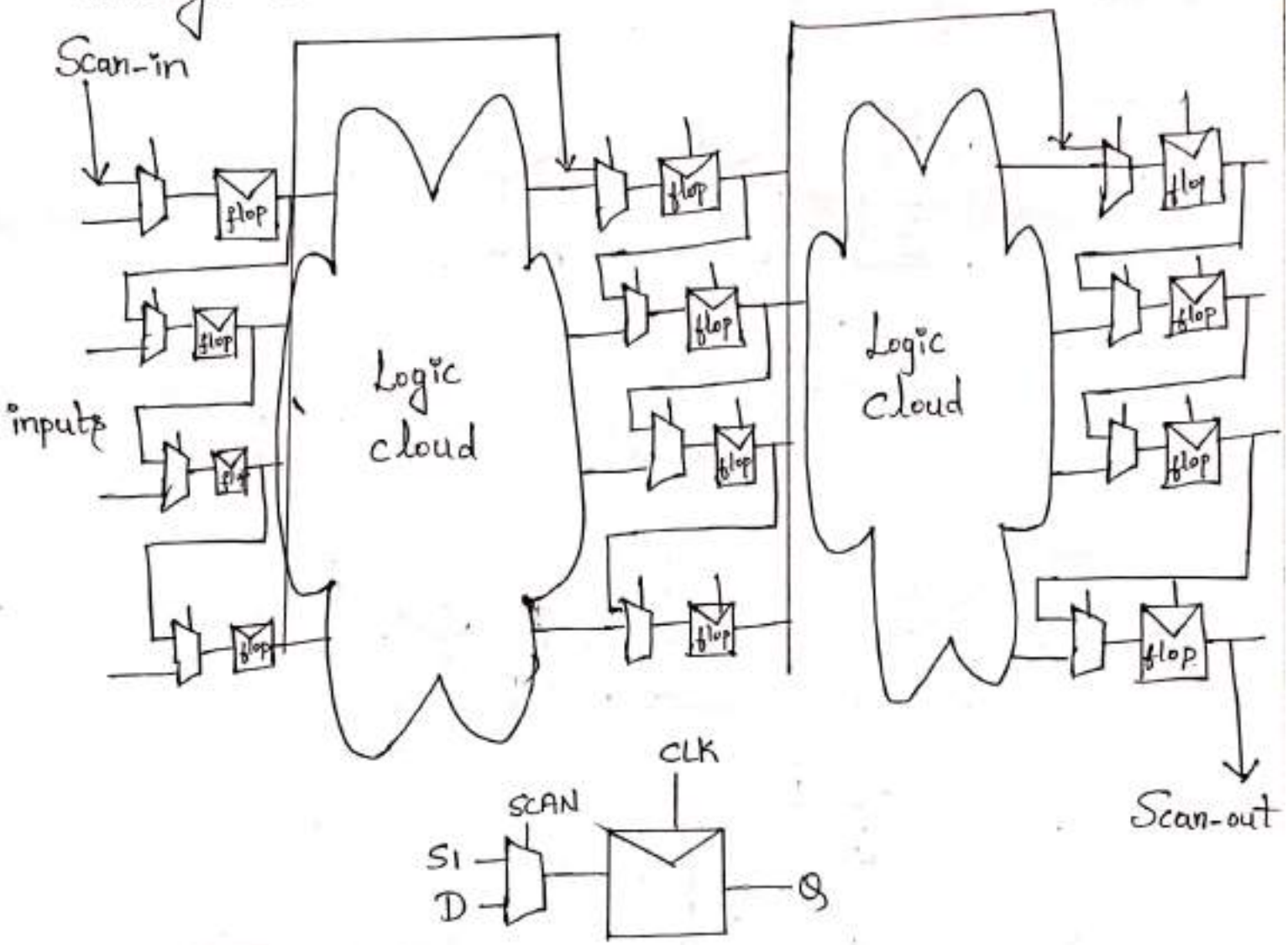


Fig : Scan-based testing

parallel Scan : The figure shows a two-by-two register section. Each register receives a column and row access signal along with a row data line. A global Wire signal (Write) is Connected to all registers. By asserting the row and column access signals in Conjunction with the Write signal, any register can be read or Written in exactly the same method as a Conventional RAM.

Scanned with CamScanner

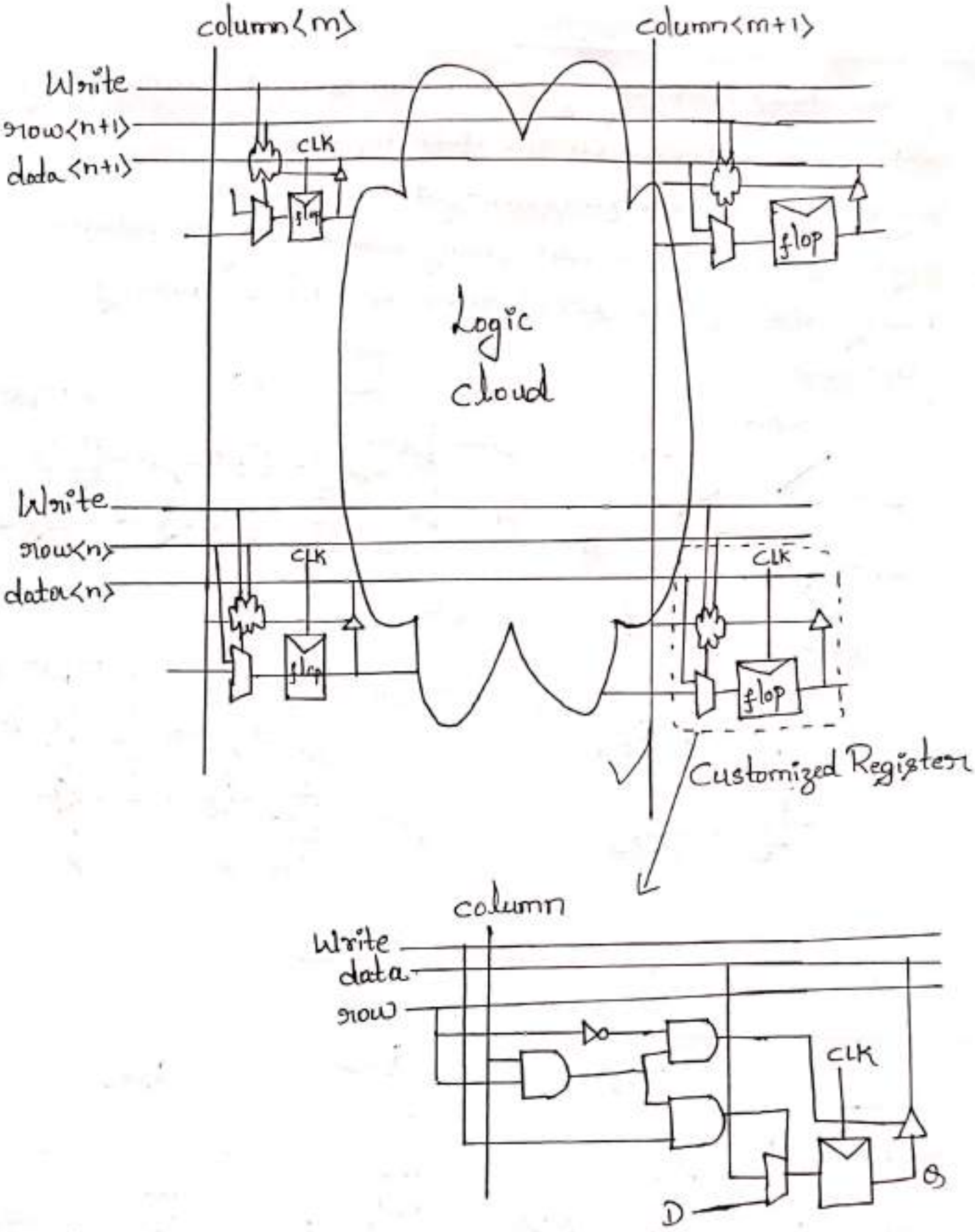


Fig: parallel Scan-basic structure

Scanned with CamScanner

Scannable Register Design

* Fig a) shows ordinary flip-flop can be made Scannable by adding a multiplexer on the data input.
* Fig b) shows transmission-gate multiplexer.
* Fig c) shows a Circuit using clock gating to obtain nearly the same setup time as the ordinary flip-flop.

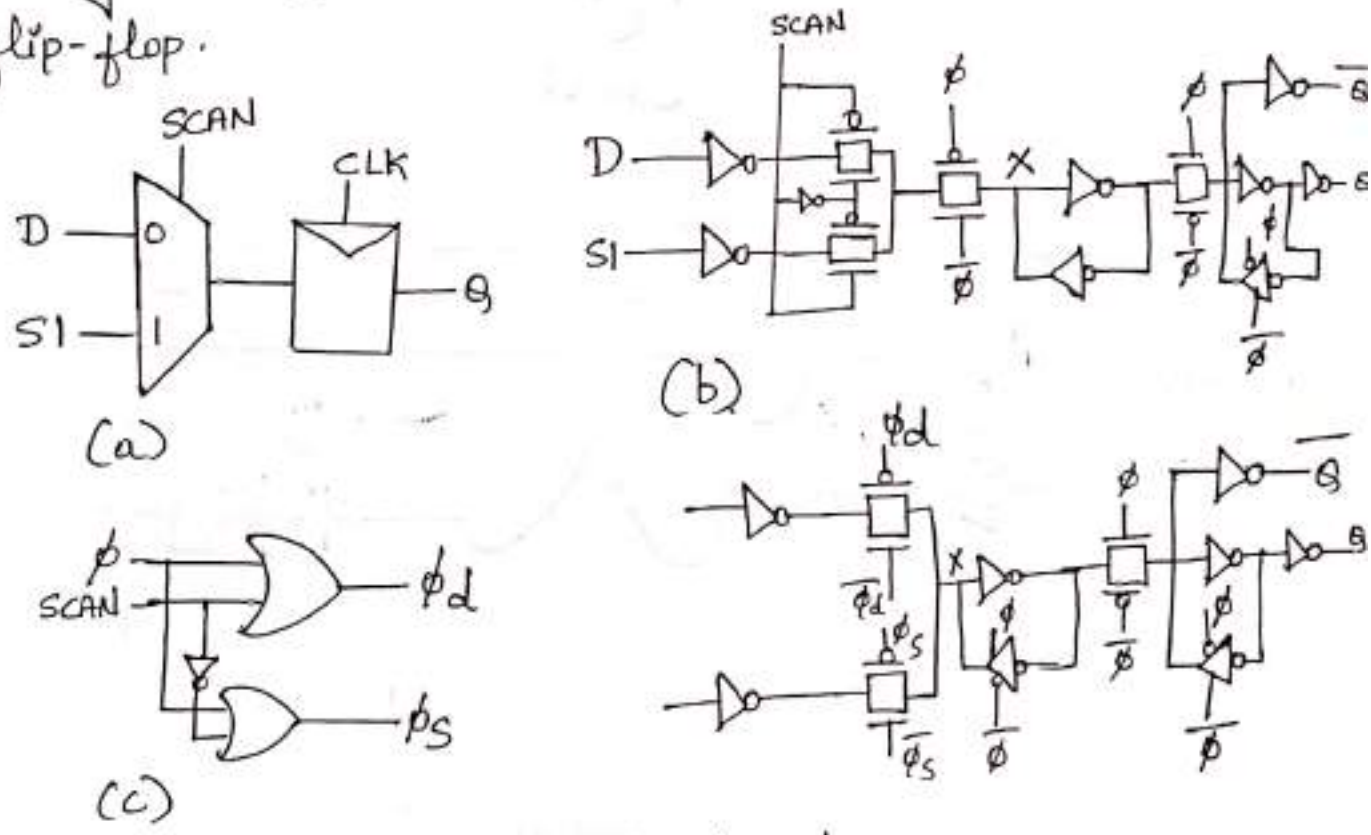
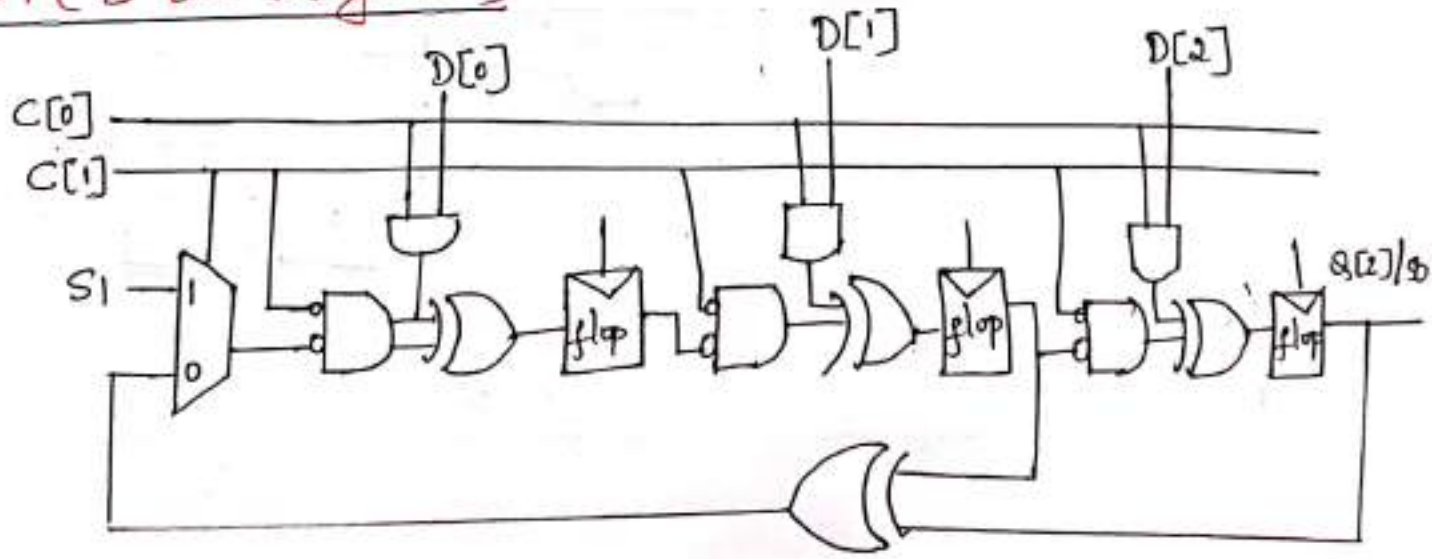


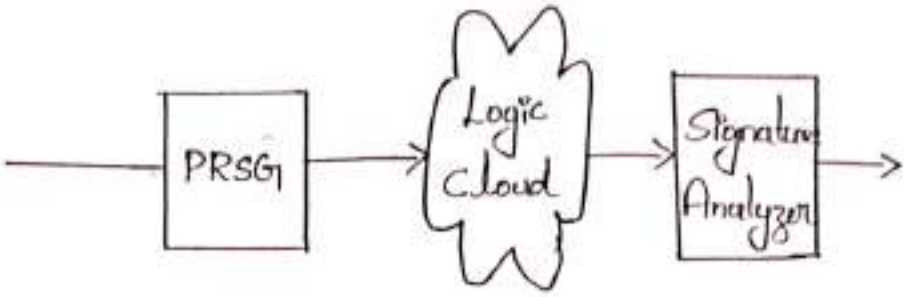
Fig : Scannable flip-flops

BIST (3 bit register)



(a)

Scanned with CamScanner



(b) Use in a System

Mode	C[1]	C[0]
Scan	0	0
Test	0	1
Reset	1	0
Normal	1	1

Other on-chip Test strategies

* On chip speeds are usually so high that directly observing internal behaviour for testing can be difficult or impossible.
* Analog/digital Converter testing requires real-time access to the digital output of the ADC.
* If both ADC and DACs are present, a loopback strategy can be employed.
* Both analog and digital signals can loop back.

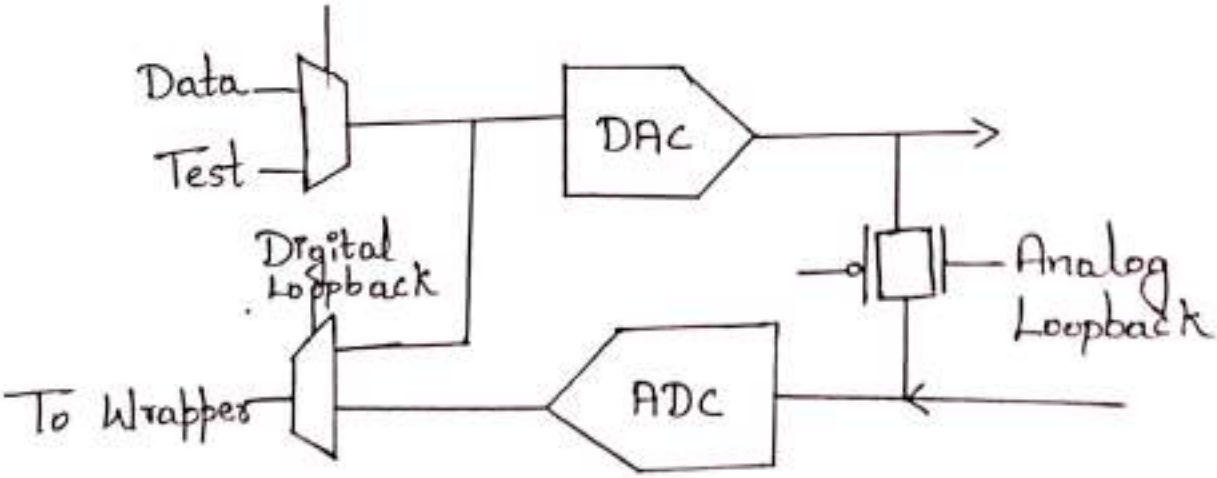


Fig : Analog and digital Loop back

Scanned with CamScanner

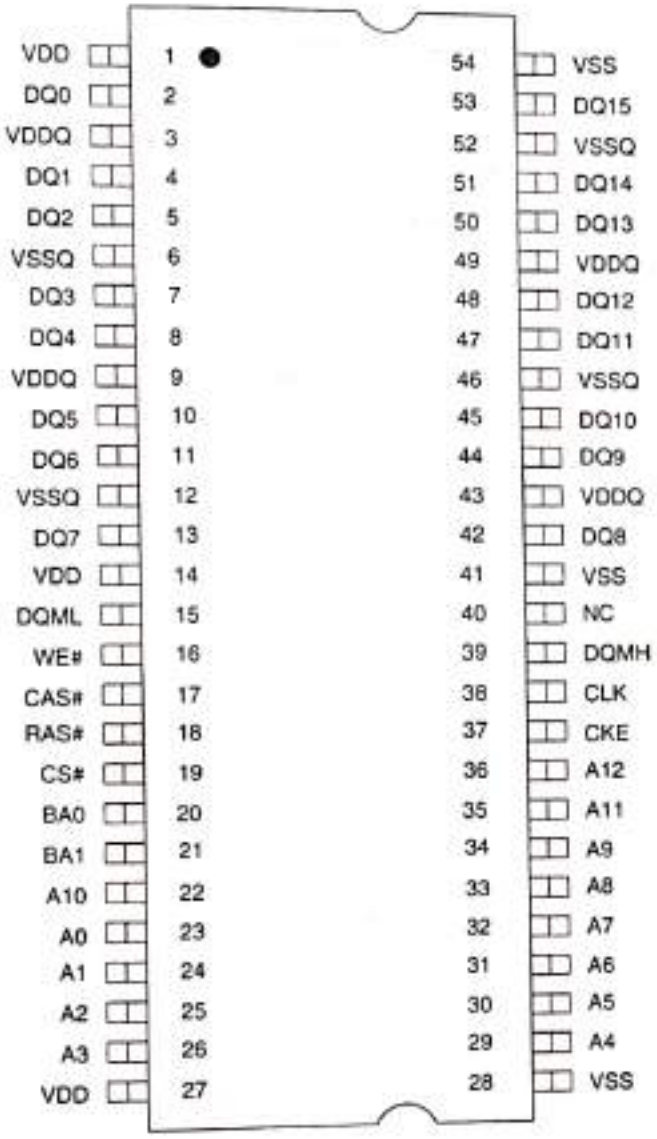


Figure 10.5 Pin assignment of 256-Mb synchronous DRAM. (The # symbol indicates signal is active "low".)

address multiplexing, which reduces the package size. Row and column addresses are captured by $\overline{RAS}$ and $\overline{CAS}$ signals.

Historical Evolution of DRAM Cell

As the continuing trend for high-density memories favors smaller memory cell sizes, the *dynamic* RAM cell with a simple structure has become a popular choice, where binary data are stored as charge in a capacitor and the presence or absence of stored charge determines the value of the stored bit. Note that the data stored as charge in a capacitor cannot be retained indefinitely, because the leakage currents eventually remove or modify the stored data. Thus, all DRAM cells require a periodic refreshing of the stored data, so that unwanted modifications due to leakage are prevented

Scanned with CamScanner