

LECTURE NOTES

ON

Basic VLSI Design (20A04606)

III B. Tech II Semester (R20)

Prepared by

Mr R.NARAYANA RAO, M.Tech
Assistant Professor



DEPARTMENT OF ELECTRICAL AND ELECTRONICS ENGINEERING

VEMU INSTITUTE OF TECHNOLOGY

NEAR PAKALA, CHITTOOR-517112, A.P

(Approved by AICTE, New Delhi & Affiliated to JNTUA, Anantapuramu)



JAWAHARLAL NEHRU TECHNOLOGICAL UNIVERSITY ANANTAPUR
B.Tech III-I Sem **L T P C**
3 0 0 3

(20A04606) BASIC VLSI DESIGN

Course Objectives:

- Understand the fundamental aspects of circuits in silicon
- Relate to VLSI design processes and design rules

Course Outcomes:

- Identify the CMOS layout levels, and the design layers used in the process sequence.
- Describe the general steps required for processing of CMOS integrated circuits.
- Design static CMOS combinational and sequential logic at the transistor level.
- Demonstrate different logic styles such as complementary CMOS logic, pass-transistor Logic, dynamic logic, etc.
- Interpret the need for testability and testing methods in VLSI.

UNIT I

Moore's law, speed power performance, nMOS fabrication, CMOS fabrication: n-well, pwell processes, BiCMOS, Comparison of bipolar and CMOS. Basic Electrical Properties of MOS And BiCMOS Circuits: Drain to source current versus voltage characteristics, threshold voltage, trans conductance.

UNIT II

Basic Electrical Properties of MOS And BiCMOS Circuits: nMOS inverter, Determination of pull up to pull down ratio: nMOS inverter driven through one or more pass transistors, alternative forms of pull up, CMOS inverter, BiCMOS inverters, latch up. Basic Circuit Concepts: Sheet resistance, area capacitance calculation, Delay unit, inverter delay, estimation of CMOS inverter delay, super buffers, BiCMOS drivers.

UNIT III

MOS and BiCMOS Circuit Design Processes: MOS layers, stick diagrams, nMOS design style, CMOS design style Design rules and layout & Scaling of MOS Circuits: λ - based design rules, scaling factors for device parameters

UNIT IV

Subsystem Design and Layout-1: Switch logic pass transistor, Gate logic inverter, NAND gates, NOR gates, pseudo nMOS, Dynamic CMOS Examples of structured design: Parity generator, Bus arbitration, multiplexers, logic function block, code converter.

UNIT V

Subsystem Design and Layout-2: Clocked sequential circuits, dynamic shift registers, bus lines, General considerations, 4-bit arithmetic processes, 4-bit shifter, Regularity Definition & Computation Practical aspects and testability: Some thoughts of performance, optimization and CAD tools for design and simulation.

Textbooks:

1. "Basic VLSI Design", Douglas A Pucknell, Kamran Eshraghian, 3 rd Edition, Prentice Hall of India publication, 2005.

References:

1. "CMOS Digital Integrated Circuits, Analysis And Design", Sung – Mo (Steve) Kang, Yusuf Leblebici, Tata McGraw Hill, 3 rd Edition, 2003.
2. VLSI Technology", S.M. Sze, 2nd edition, Tata McGraw Hill, 2003

UNIT-I

Introduction to VLSI Technology

Introduction:

The invention of the transistor by William B. Shockley, Walter H. Brattain and John Bardeen of Bell Telephone Laboratories drastically changed the electronics industry and paved the way for the development of the Integrated Circuit (IC) technology. The first IC was designed by Jack Kilby at Texas Instruments at the beginning of 1960 and since that time there have already been four generations of ICs .Viz SSI (small scale integration), MSI (medium scale integration), LSI (large scale integration), and VLSI (very large scale integration). Now we are ready to see the emergence of the fifth generation, ULSI (ultra large scale integration) which is characterized by complexities in excess of 3 million devices on a single IC chip. Further miniaturization is still to come and more revolutionary advances in the application of this technology must inevitably occur.

Over the past several years, Silicon CMOS technology has become the dominant fabrication process for relatively high performance and cost effective VLSI circuits. The revolutionary nature of this development is understood by the rapid growth in which the number of transistors integrated in circuits on a single chip.

METAL-OXIDE-SEMICONDUCTOR (MOS) AND RELATED VLSI TECHNOLOGY:

The MOS technology is considered as one of the very important and promising technologies in the VLSI design process. The circuit designs are realized based on PMOS, NMOS, CMOS and BiCMOS devices.

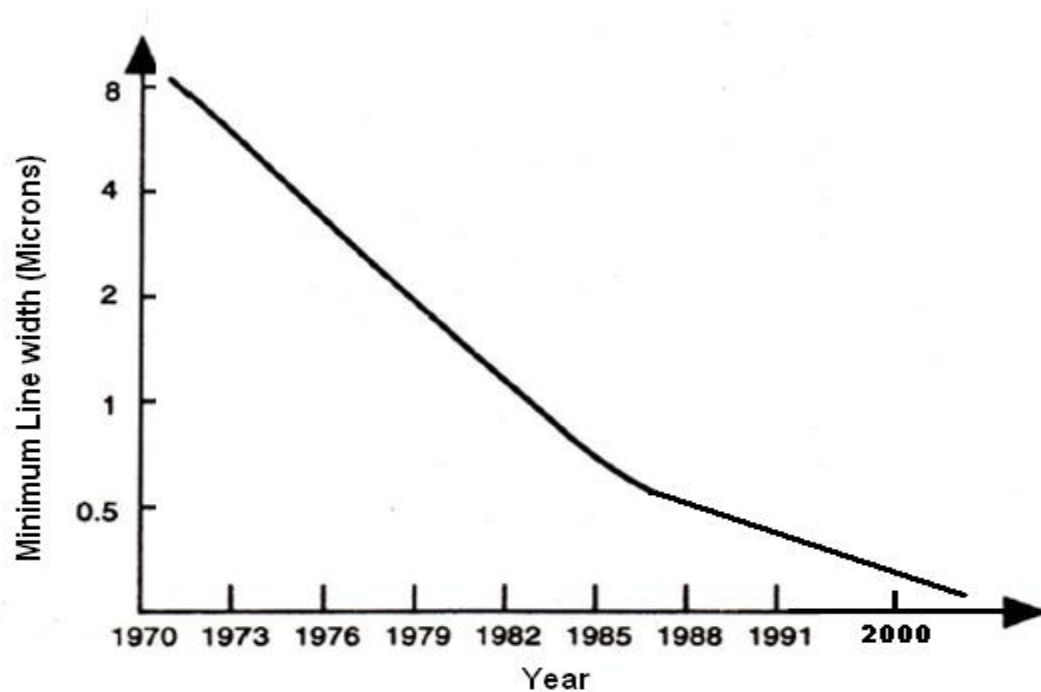
The PMOS devices are based on the p-channel MOS transistors. Specifically, the PMOS channel is part of a n-type substrate lying between two heavily doped p+ wells beneath the source and drain electrodes. Generally speaking, a PMOS transistor is only constructed in consort with an NMOS transistor.

The NMOS technology and design processes provide an excellent background for other technologies. In particular, some familiarity with NMOS allows a relatively easy transition to CMOS technology and design.

The techniques employed in NMOS technology for logic design are similar to GaAs technology.. Therefore, understanding the basics of NMOS design will help in the layout of GaAs circuits

In addition to VLSI technology, the VLSI design processes also provides a new degree of freedom for designers which helps for the significant developments. With the rapid advances in technology the size of the ICs is shrinking and the integration density is increasing.

The minimum line width of commercial products over the years is shown in the graph below.

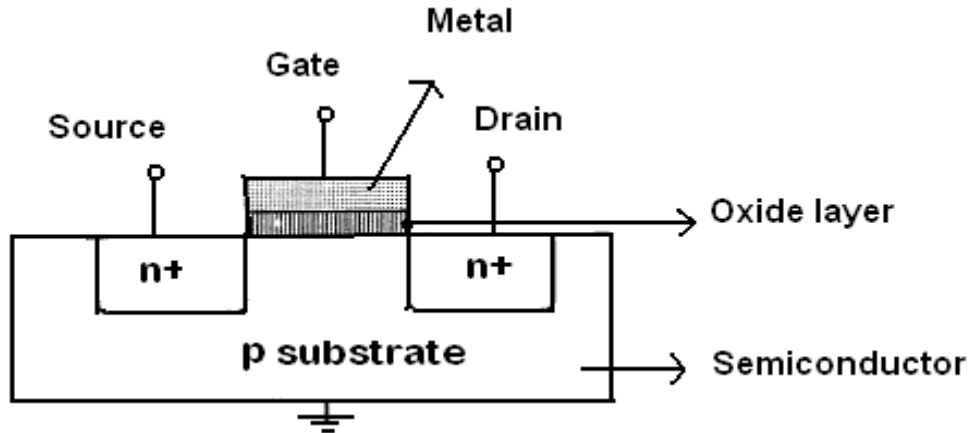


The graph shows a significant decrease in the size of the chip in recent years which implicitly indicates the advancements in the VLSI technology.

BASIC MOS TRANSISTORS:

The MOS Transistor means, Metal-Oxide-Semiconductor Field Effect Transistor which is the most basic element in the design of a large scale integrated circuits(IC).

These transistors are formed as a "sandwich" consisting of a semiconductor layer, usually a slice, or wafer, from a single crystal of silicon; a layer of silicon dioxide (the oxide) and a layer of metal. These layers are patterned in a manner which permits transistors to be formed in the semiconductor material (the "substrate"); a diagram showing a MOSFET is shown below in Figure .



Silicon dioxide is a very good insulator, so a very thin layer, typically only a few hundred molecules thick, is used. In fact, the transistors which are used do not use metal for their gate regions, but instead use polycrystalline silicon (poly). Polysilicon gate FET's have replaced virtually all of the older devices using metal gates in large scale integrated circuits. (Both metal and polysilicon FET's are sometimes referred to as IGFET's (insulated gate field effect transistors), since the silicon dioxide under the gate is an insulator.

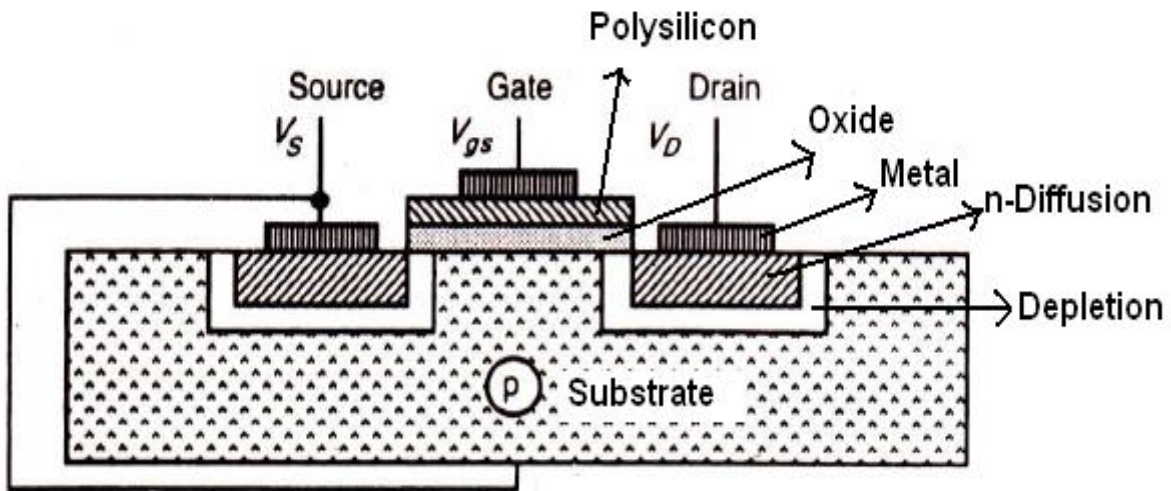
MOS Transistors are classified as n-MOS, p-MOS and c-MOS Transistors based on the fabrication.

NMOS devices are formed in a p-type substrate of moderate doping level. The source and drain regions are formed by diffusing n-type impurities through suitable masks into these areas to give the desired n-impurity concentration and give rise to depletion regions which extend mainly in the more lightly doped p-region. Thus, source and drain are isolated from one another by two diodes. Connections to the source and drain are made by a deposited metal layer. In order to make a useful device, there must be the capability for establishing and controlling a current between source and drain, and this is commonly achieved in one of two ways, giving rise to the enhancement mode and depletion mode transistors.

Enhancement Mode Transistors:

In an enhancement mode device a polysilicon gate is deposited on a layer of insulation over the region between source and drain. In the diagram below channel is not established and the device is in a non-conducting condition, i.e $V_D = V_S = V_{GS} = 0$. If this gate is connected to a suitable positive voltage with respect to the source, then the electric field established between the gate and

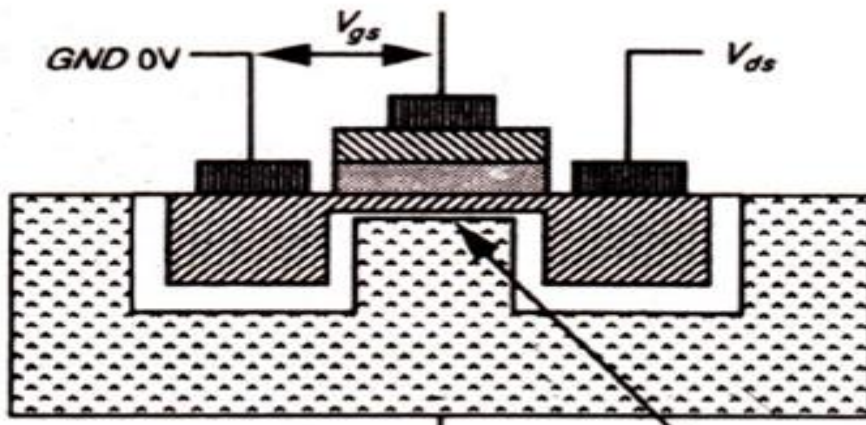
the substrate gives rise to a charge inversion region in the substrate under the gate insulation and a conducting path or channel is formed between source and drain.



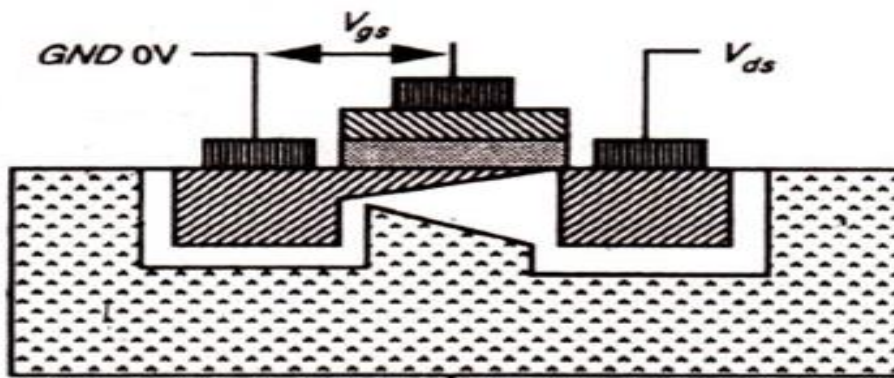
ENHANCEMENT MODE TRANSISTOR ACTION :

To understand the enhancement mechanism, let us consider the enhancement mode device. In order to establish the channel, a minimum voltage level called threshold voltage (V_t) must be established between gate and source. Fig. (a) Shows the existing situation where a channel is established but no current flowing between source and drain ($V_{ds} = 0$).

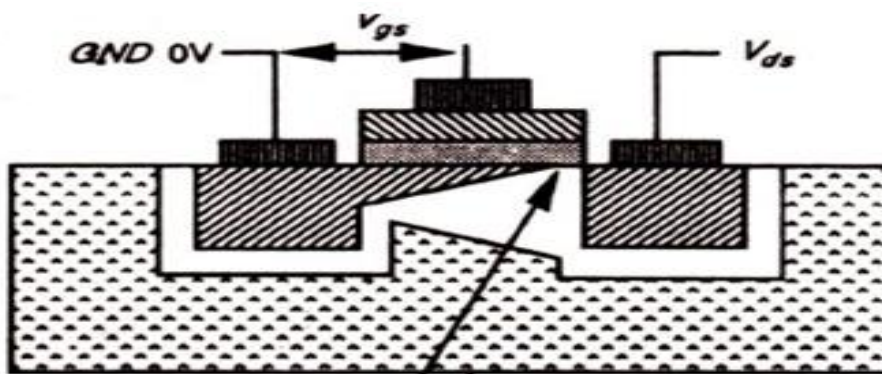
Let us now consider the conditions when current flows in the channel by applying a voltage V_{ds} between drain and source. The IR drop = V_{ds} along the channel. This develops a voltage between gate and channel varying with distance along the channel with the voltage being a maximum of V_{gs} at the source end. Since the effective gate voltage is $V_g = V_{gs} - V_t$, (no current flows when $V_{gs} < V_t$) there will be voltage available to invert the channel at the drain end so long as $V_{gs} - V_t \sim V_{ds}$. The limiting condition comes when $V_{ds} = V_{gs} - V_t$. For all voltages $V_{ds} < V_{gs} - V_t$, the device is in the non-saturated region of operation which is the condition shown in Fig. (b) below.



(a) $V_{gs} > V_t$
 $V_{ds} = 0V$



(b) $V_{gs} > V_t$
 $V_{ds} < V_{gs} - V_t$

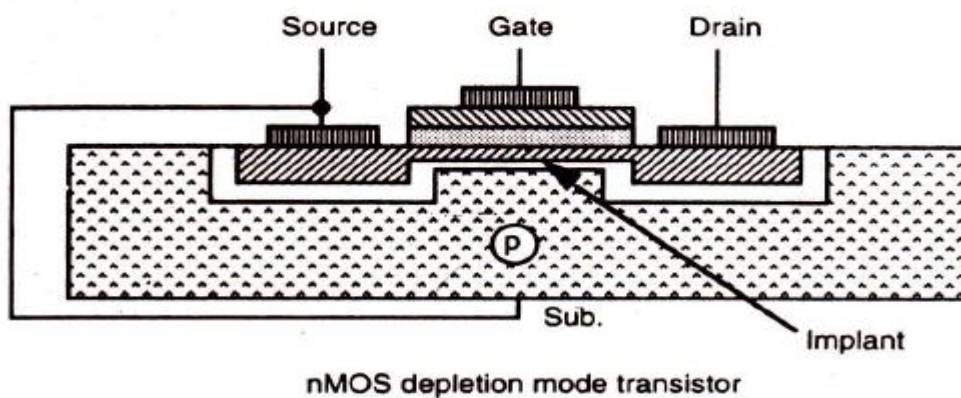


(c) $V_{gs} > V_t$
 $V_{ds} > V_{gs} - V_t$

Let us now consider the situation when V_{ds} is increased to a level greater than $V_{gs} - V_t$. In this case, an IR drop equal to $V_{gs} - V_t$ occurs over less than the whole length of the channel such that, near the drain, there is insufficient electric field available to give rise to an inversion layer to create the channel. The channel is, therefore, 'pinched off' as shown in Fig. (c). Diffusion current completes the path from source to drain in this case, causing the channel to exhibit a high resistance and behave as a constant current source. This region, known as saturation, is characterized by almost constant current for increase of V_{ds} above $V_{ds} = V_{gs} - V_t$. In all cases, the channel will cease to exist and no current will flow when $V_{gs} < V_t$. Typically, for enhancement mode devices, $V_t = 1$ volt for $V_{DD} = 5$ V or, in general terms, $V_t = 0.2 V_{DD}$.

DEPLETION MODE TRANSISTOR ACTION

N-MOS Depletion mode mosfets are built with P-type silicon substrates, and P-channel versions are built on N-type substrates. In both cases they include a thin gate oxide formed between the source and drain regions. A conductive channel is deliberately formed below the gate oxide layer and between the source and drain by using ion-implantation. By implanting the correct ion polarity in the channel region during fabrication determines the polarity of the threshold voltage (i.e. $-V_t$ for an N channel transistor, or $+V_t$ for an P-channel transistor). The actual concentration of ions in the substrate-to-channel region is used to adjust the threshold voltage (V_t) to the desired value. Depletion-mode devices are a little more difficult to manufacture and their characteristics harder to control than enhancement types, which do not require ion implantation. In depletion mode devices the channel is established, due to the implant, even when $V_{gs} = 0$, and to cause the channel to cease a negative voltage V_{td} must be applied between gate and source.



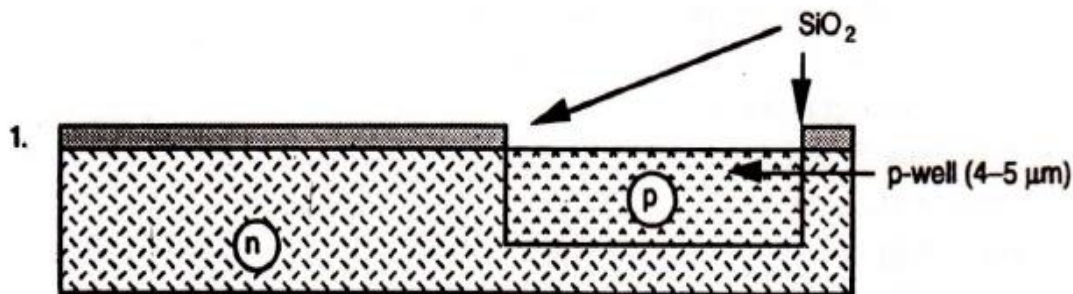
V_{td} is typically $< -0.8 V_{DD}$, depending on the implant and substrate bias, but, threshold voltage differences apart, the action is similar to that of the enhancement mode transistor.

CMOS FABRICATION :

CMOS fabrication is performed based on various methods , including the p-well, the n-well, the twin-tub, and the silicon-on-insulator processes .Among these methods the p-well process is widely used in practice and the n-well process is also popular, particularly as it is an easy retrofit to existing NMOS lines.

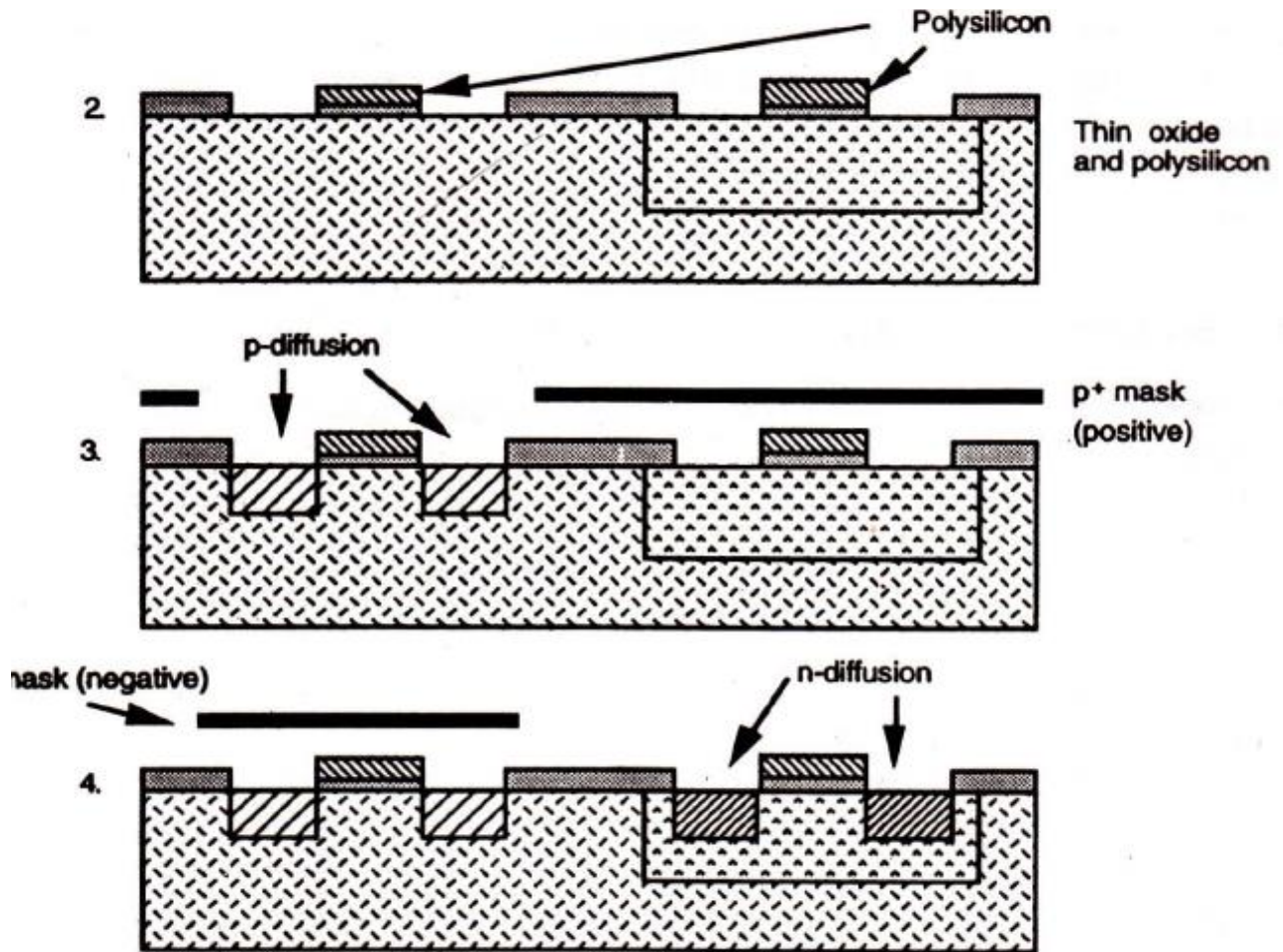
(i) The p-well Process:

The p-well structure consists of an n-type substrate in which p-devices may be formed by suitable masking and diffusion and, in order to accommodate n-type devices, a deep p-well is diffused into the n-type substrate as shown in the Fig.below.



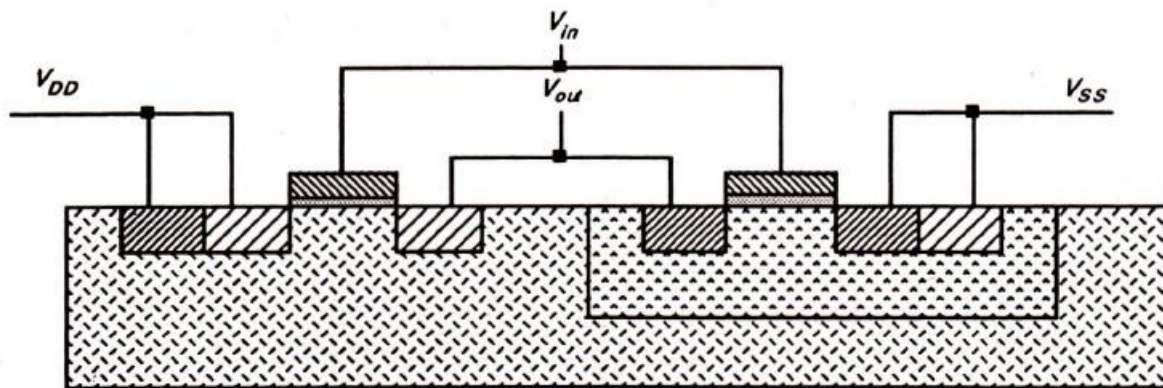
This diffusion should be carried out with special care since the p-well doping concentration and depth will affect the threshold voltages as well as the breakdown voltages of the n-transistors. To achieve low threshold voltages (0.6 to 1.0 V) either deep-well diffusion or high-well resistivity is required. However, deep wells require larger spacing between the n- and p-type transistors and wires due to lateral diffusion and therefore a larger chip area. The p-wells Act as substrates for the n-devices within the parent n-substrate, and, the two areas are electrically isolated.

Except this in all other respects- like masking, patterning, and diffusion-the process is similar to NMOS fabrication.



P-well fabrication process(Figs 1,2,3 & 4)

The diagram below shows the CMOS p-well inverter showing V_{DD} and V_{SS} substrate connections



The n-well Process : Though the p-well process is widely used in C-MOS fabrication the n-well fabrication is also very popular because of the lower substrate bias effects on transistor threshold voltage and also lower parasitic capacitances associated with source and drain regions.

The typical n-well fabrication steps are shown in the diagram below.

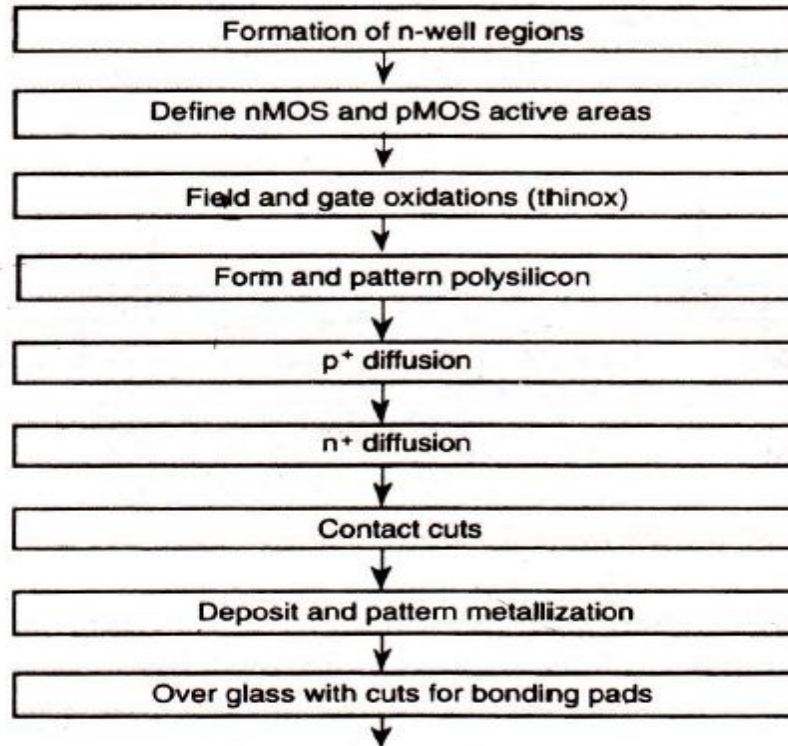
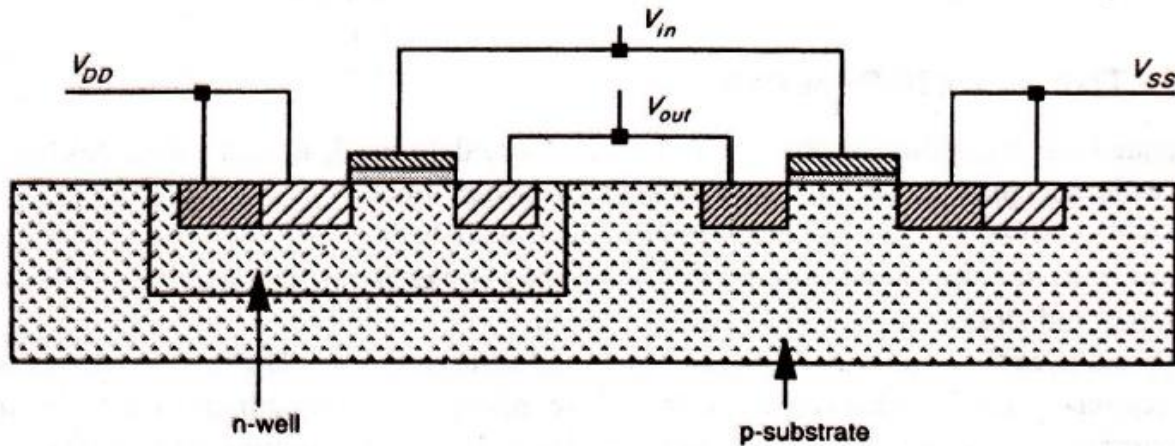


Fig. n-well fabrication steps

The first mask defines the n-well regions. This is followed by a low dose phosphorus implant driven in by a high temperature diffusion step to form the n-wells. The well depth is optimized to ensure against-substrate top+ diffusion breakdown without compromising then-well to n+ mask separation. The next steps are to define the devices and diffusion paths, grow field oxide, deposit and pattern the poly silicon, carry out the diffusions, make contact cuts, and finally metalize as before. It will be seen that an n+ mask and its complement may be used to define the n- and p-diffusion regions respectively. These same masks also include the V_{DD} and V_{SS} contacts (respectively). It should be noted that, alternatively, we could have used a p+ mask and its complement since the n + and p + masks are generally complementary.

The diagram below shows the Cross-sectional view of n-well CMOS Inverter.



Due to the differences in charge carrier mobilities, the n-well process creates non-optimum p-channel characteristics. However, in many CMOS designs (such as domino-logic and dynamic logic structures), this is relatively unimportant since they contain a preponderance of n-channel devices. Thus then-channel transistors are mainly those used to form logic elements, providing speed and high density of elements.

However, a factor of the n-well process is that the performance of the already poorly performing p-transistor is even further degraded. Modern process lines have come to grips with these problems, and good device performance may be achieved for both p-well and n-well fabrication.

BICMOS Technology:

A Bi-CMOS circuit of both bipolar junction transistors and MOS transistors on a single substrate. The driving capability of MOS transistors is less because of limited current sourcing and sinking capabilities of the transistors. To drive large capacitive loads Bi-CMOS technology is used. As this technology combines Bipolar and CMOS transistors in a single integrated circuit, it has the advantages of both bipolar and CMOS transistors. Bi-CMOS is able to achieve VLSI circuits with speed-power-density performance previously not possible with either technology individually. The diagram given below shows the cross section of the Bi-CMOS process which uses an NPN transistor

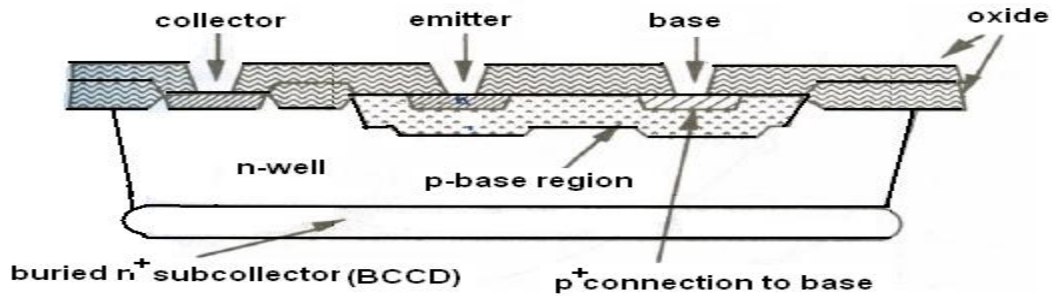
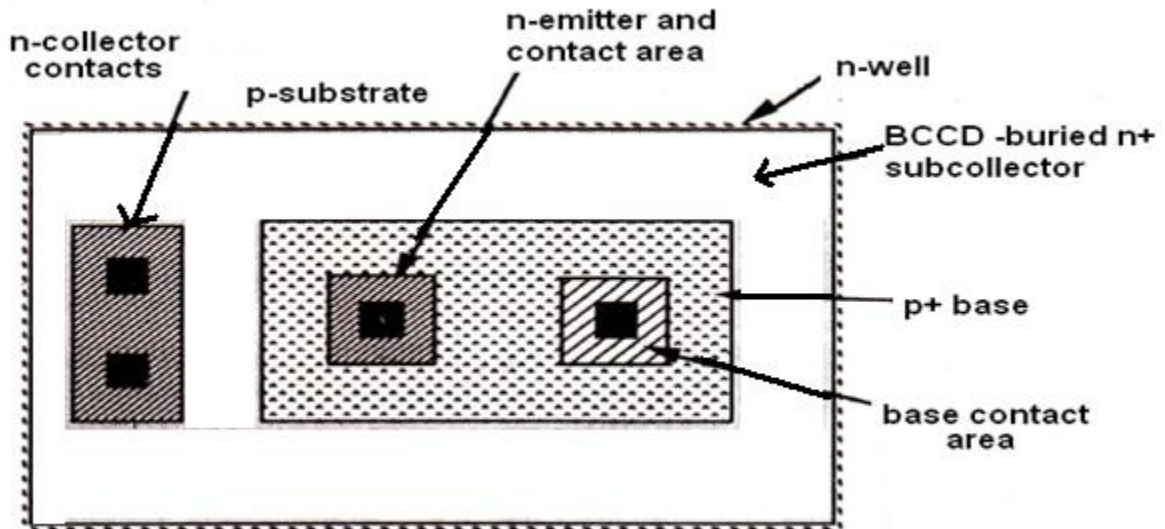


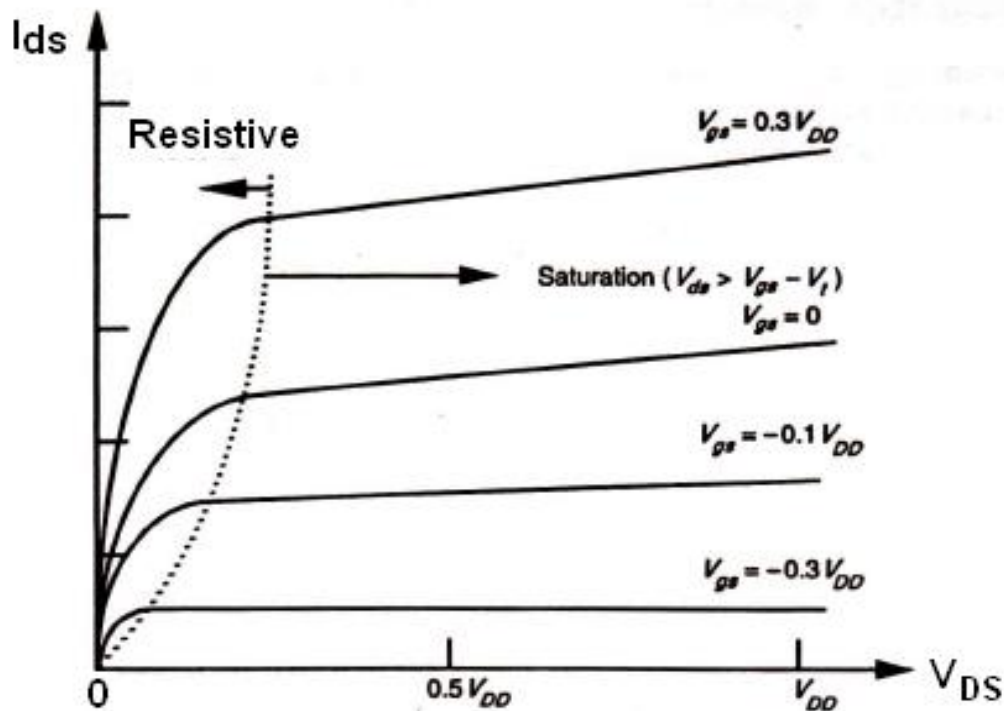
Fig. Cross section of Bi-CMOS process

The lay-out view of Bic-MOS transistor is shown in the figure below. The fabrication of Bi-CMOS is similar to CMOS but with certain additional process steps and additional masks are considered. They are (i) the p+ base region; (ii) n+ collector area; and (iii) the buried sub collector (BCCD).



I_{DS} - V_{DS} characteristics of MOS Transistor:

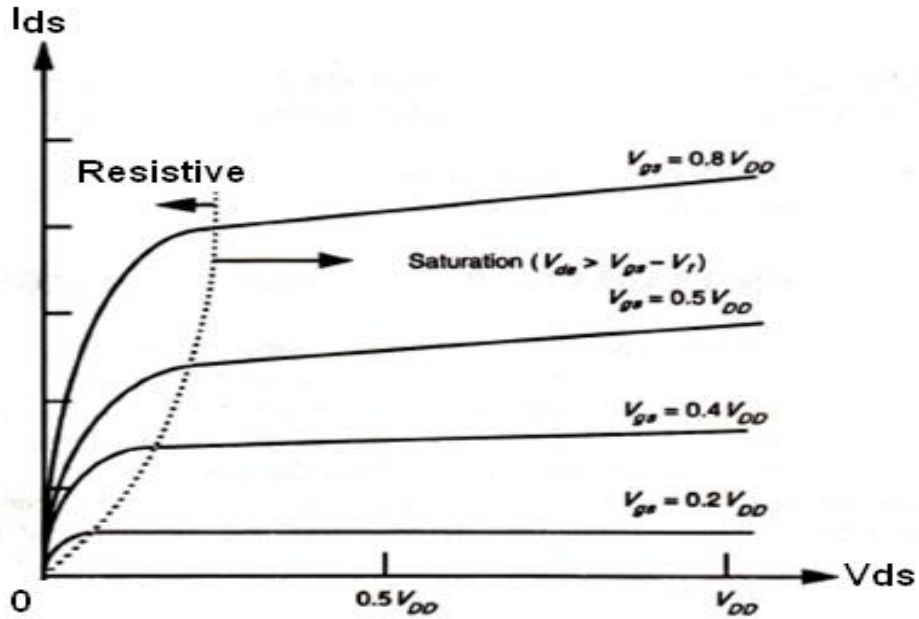
The graph below shows the I_D Vs V_{DS} characteristics of an n- MOS transistor for several values of V_{GS} . It is clear that there are two conduction states when the device is ON. The saturated state and the non-saturated state. The saturated curve is the flat portion and defines the saturation region. For $V_{GS} < V_{DS} + V_{th}$, the NMOS device is conducting and I_D is independent of V_{DS} . For $V_{GS} > V_{DS} + V_{th}$, the transistor is in the non-saturation region and the curve is a half parabola. When the transistor is OFF ($V_{GS} < V_{th}$), then I_D is zero for any V_{DS} value.



(a) Depletion mode device

The boundary of the saturation/non-saturation bias states is a point seen for each curve in the graph as the intersection of the straight line of the saturated region with the quadratic curve of the non-saturated region. This intersection point occurs at the channel pinch off voltage called V_{DSAT} . The diamond symbol marks the pinch-off voltage V_{DSAT} for each value of V_{GS} . V_{DSAT} is defined as the minimum drain-source voltage that is required to keep the transistor in saturation for a given V_{GS} . In the non-saturated state, the drain current initially increases almost linearly from the origin before bending in a parabolic response. Thus the name ohmic or linear for the non-saturated region.

The drain current in saturation is virtually independent of V_{DS} and the transistor acts as a current source. This is because there is no carrier inversion at the drain region of the channel. Carriers are pulled into the high electric field of the drain/substrate pn junction and ejected out of the drain terminal.



(b). Enhance mode device

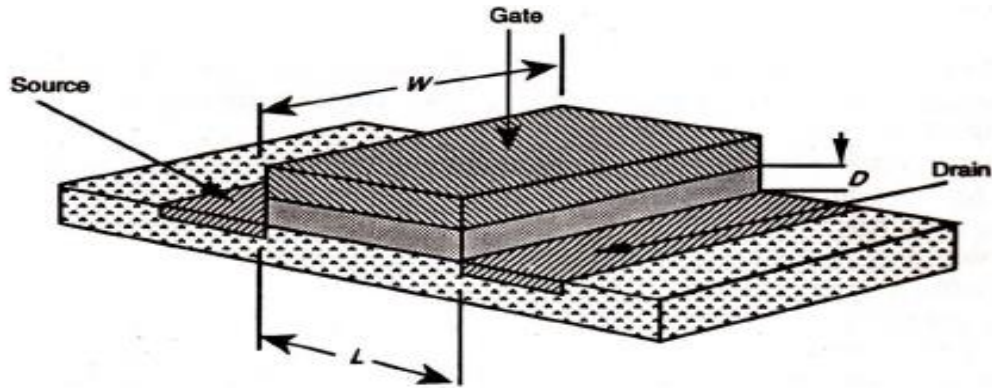
Drain-to-Source Current I_{DS} versus Voltage V_{DS} Relationships:

The working of a MOS transistor is based on the principle that the use of a voltage on the gate induce a charge in the channel between source and drain, which may then be caused to move from source to drain under the influence of an electric field created by voltage V_{DS} applied between drain and source. Since the charge induced is dependent on the gate to source voltage V_{GS} then I_{DS} is dependent on both V_{GS} and V_{DS} .

Let us consider the diagram below in which electrons will flow source to drain .So, the drain current is given by

$$I_{DS} = -I_{SD} = \frac{\text{Charge induced in channel (} Q_c \text{)}}{\text{Electron transit time}(\tau)}$$

Where the transit time is given by $\tau_{sd} = \frac{\text{Length of the channel}}{\text{Velocity (} v \text{)}}$



But velocity $v = \mu E_{ds}$

Where μ =electron or hole mobility and E_{ds} = Electric field

Also , $E_{ds} = V_{ds}/L$

So, $v = \mu \cdot V_{ds}/L$

And $\tau_{ds} = L^2 / \mu \cdot V_{ds}$

The typical values of μ at room temperature are given below.

$$\mu_n \approx 650 \text{ cm}^2/\text{V sec (surface)}$$

$$\mu_p \approx 240 \text{ cm}^2/\text{V sec (surface)}$$

The Non-saturated Region:

Let us consider the I_d vs V_d relationships in the non-saturated region .The charge induced in the channel due to due to the voltage difference between the gate and the channel, V_{gs} (assuming substrate connected to source). The voltage along the channel varies linearly with distance X from the source due to the IR drop in the channel .In the non-saturated state the average value is $V_{ds}/2$. Also the effective gate voltage $V_g = V_{gs} - V_t$ where V_t , is the threshold voltage needed to invert the charge under the gate and establish the channel.

Hence the induced charge is $Q_c = E_g \epsilon_{ins} \epsilon_o W \cdot L$

Where

E_g = average electric field gate to channel

ϵ_{ins} = relative permittivity of insulation between gate and channel

ϵ_0 = permittivity of free space.

So, we can write that

$$E_g = \frac{\left((V_{gs} - V_t) - \frac{V_{ds}}{2} \right)}{D}$$

Here D is the thickness of the oxide layer. Thus

$$Q_c = \frac{WL\epsilon_{ins}\epsilon_0}{D} \left((V_{gs} - V_t) - \frac{V_{ds}}{2} \right)$$

So, by combining the above two equations ,we get

$$I_{ds} = \frac{\epsilon_{ins}\epsilon_0\mu}{D} \frac{W}{L} \left((V_{gs} - V_t) - \frac{V_{ds}}{2} \right) V_{ds}$$

Or the above equation can be written as

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

In the non-saturated or resistive region where $V_{ds} < V_{gs} - V_t$ and

$$K = \frac{\epsilon_{ins}\epsilon_0\mu}{D}$$

Generally ,a constant β is defined as

$$\beta = K \frac{W}{L}$$

So that ,the expression for drain –source current will become

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

The gate /channel capacitance is

$$C_g = \frac{\epsilon_{ins} \epsilon_0 W L}{D} \text{ (parallel plate)}$$

Hence we can write another alternative form for the drain current as

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

Some time it is also convenient to use gate –capacitance per unit area, C_g

So, the drain current is

$$I_{ds} = C_0 \mu \frac{W}{L} \left((V_{gs} - V_t) V_{ds} - \frac{V_{ds}^2}{2} \right)$$

This is the relation between drain current and drain-source voltage in non-saturated region.

The Saturated Region

Saturation begins when $V_{ds} = V_{gs} - V_t$, since at this point the IR drop in the channel equals the effective gate to channel voltage at the drain and we may assume that the current remains fairly constant as V_{ds} increases further. Thus

$$I_{ds} = K \frac{W}{L} \frac{(V_{gs} - V_t)^2}{2}$$

Or we can also write that

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

Or it can also be written as

$$I_{ds} = \frac{C_g \mu}{2L^2} (V_{gs} - V_t)^2$$

Or

$$I_{ds} = C_0 \mu \frac{W}{2L} (V_{gs} - V_t)^2$$

The expressions derived above for I_{ds} hold for both enhancement and depletion mode devices.

Here the threshold voltage for the NMOS depletion mode device (denoted as V_{td}) is negative.

MOS Transistor Threshold Voltage V_t :

The gate structure of a MOS transistor consists, of charges stored in the dielectric layers and in the surface to surface interfaces as well as in the substrate itself. Switching an enhancement mode MOS transistor from the off to the on state consists in applying sufficient gate voltage to neutralize these charges and enable the underlying silicon to undergo an inversion due to the electric field from the gate. Switching a depletion mode NMOS transistor from the on to the off state consists in applying enough voltage to the gate to add to the stored charge and invert the 'n' implant region to 'p'.

The threshold voltage V_t may be expressed as:

$$V_t = \phi_{ms} \frac{Q_D - Q_{SS}}{C_o} + 2\phi_{fn}$$

Where Q_D = the charge per unit area in the depletion layer below the oxide

Q_{SS} = charge density at Si: SiO_2 interface

C_o = Capacitance per unit area.

Φ_{ns} = work function difference between gate and Si

Φ_{fn} = Fermi level potential between inverted surface and bulk Si

For polynomial gate and silicon substrate, the value of Φ_{ns} is negative but negligible and the magnitude and sign of V_t are thus determined by balancing the other terms in the equation.

To evaluate the V_t the other terms are determined as below.

$$Q_D = \sqrt{2\epsilon_0\epsilon_{Si}qN(2\phi_{fn} + V_{SB})} \text{ coulomb/m}^2$$

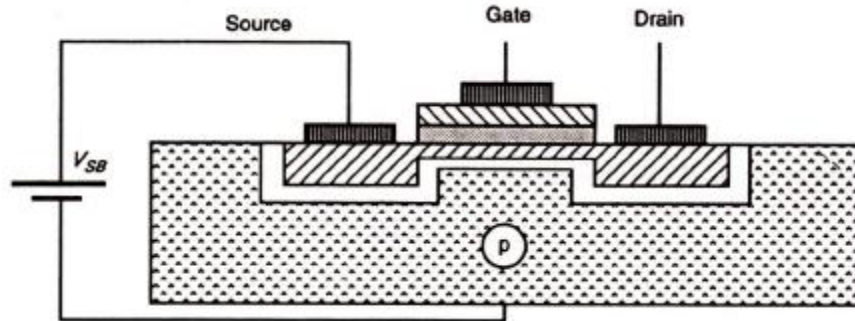
$$\phi_{fn} = \frac{kT}{q} \ln \frac{N}{n_i} \text{ volts}$$

$$Q_{SS} = (1.5 \text{ to } 8) \times 10^{-8} \text{ coulomb/m}^2$$

Body Effect :

Generally while studying the MOS transistors it is treated as a three terminal device. But ,the body of the transistor is also an implicit terminal which helps to understand the characteristics of the transistor. Considering the body of the MOS transistor as a terminal is known as the body effect.

The potential difference between the source and the body (V_{sb}) affects the threshold voltage of the transistor. In many situations, this Body Effect is relatively insignificant, so we can (unless **otherwise** stated) ignore the Body Effect. But it is not always insignificant, in some cases it can have a tremendous impact on MOSFET circuit performance.



Body effect - NMOS device

Increasing V_{sb} causes the channel to be depleted of charge carriers and thus the threshold voltage is raised. Change in V_t is given by $\delta v_t = \gamma \cdot (V_{sb})^{1/2}$ where γ is a constant which depends on substrate doping so that the more lightly doped the substrate, the smaller will be the body effect. The threshold voltage can be written as

$$V_t = V_t(0) + \left(\frac{D}{\epsilon_{ins} \epsilon_0} \right) \sqrt{2 \epsilon_0 \epsilon_{si} Q_N \cdot (V_{sb})^{1/2}}$$

Where $V_t(0)$ is the threshold voltage for $V_{sd} = 0$

For n-MOS depletion mode transistors, the body voltage values at different V_{DD} voltages are given below.

$$V_{SB} = 0 \text{ V} ; V_{sd} = -0.7V_{DD} (= - 3.5 \text{ V for } V_{DD} = +5\text{V})$$

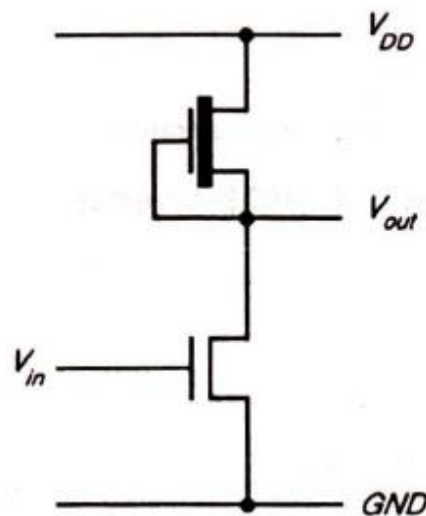
$$V_{SB} = 5 \text{ V} ; V_{sd} = -0.6V_{DD} (= - 3.0 \text{ V for } V_{DD} = +5\text{V})$$

The NMOS INVERTER :

An inverter circuit is a very important circuit for producing a complete range of logic circuits. This is needed for restoring logic levels, for Nand and Nor gates, and for sequential and memory circuits of various forms .

A simple inverter circuit can be constructed using a transistor with source connected to ground and a load resistor of connected from the drain to the positive supply rail V_{DD} . The output is taken from the drain and the input applied between gate and ground .

But, during the fabrication resistors are not conveniently produced on the silicon substrate and even small values of resistors occupy excessively large areas .Hence some other form of load resistance is used. A more convenient way to solve this problem is to use a depletion mode transistor as the load, as shown in Fig. Below.



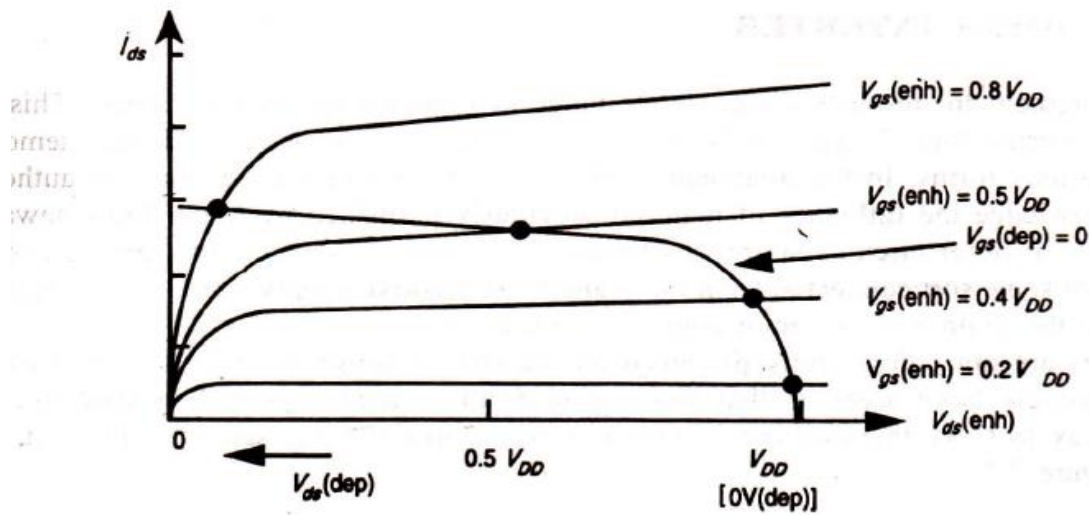
The salient features of the n-MOS inverter are

- For the depletion mode transistor, the gate is connected to the source so it is always on .
- In this configuration the depletion mode device is called the pull-up (P.U) and the enhancement mode device the pull-down (P.D) transistor.
- With no current drawn from the output, the currents I_{ds} for both transistors must be equal.

NMOS Inverter transfer characteristic.

The transfer characteristic is drawn by taking V_{ds} on x-axis and I_{ds} on Y-axis for both enhancement and depletion mode transistors. So, to obtain the inverter transfer characteristic for $V_{gs} = 0$ depletion mode characteristic curve is superimposed on the family of curves for the

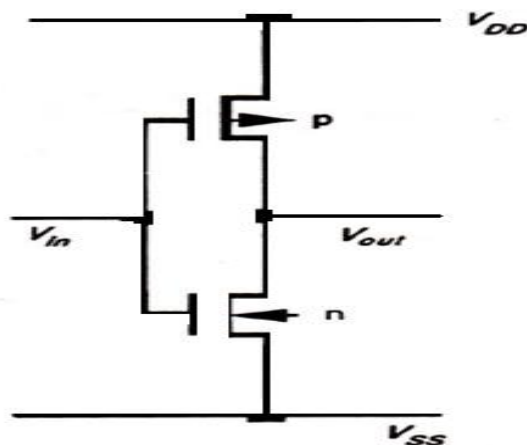
enhancement mode device and from the graph it can be seen that , maximum voltage across the enhancement mode device corresponds to minimum voltage across the depletion mode transistor.



From the graph it is clear that as $V_{in}(=V_{gs} \text{ p.d. Transistor})$ exceeds the Pulldown threshold voltage current begins to flow. The output voltage V_{out} thus decreases and the subsequent increases in V_{in} will cause the Pull down transistor to come out of saturation and become resistive.

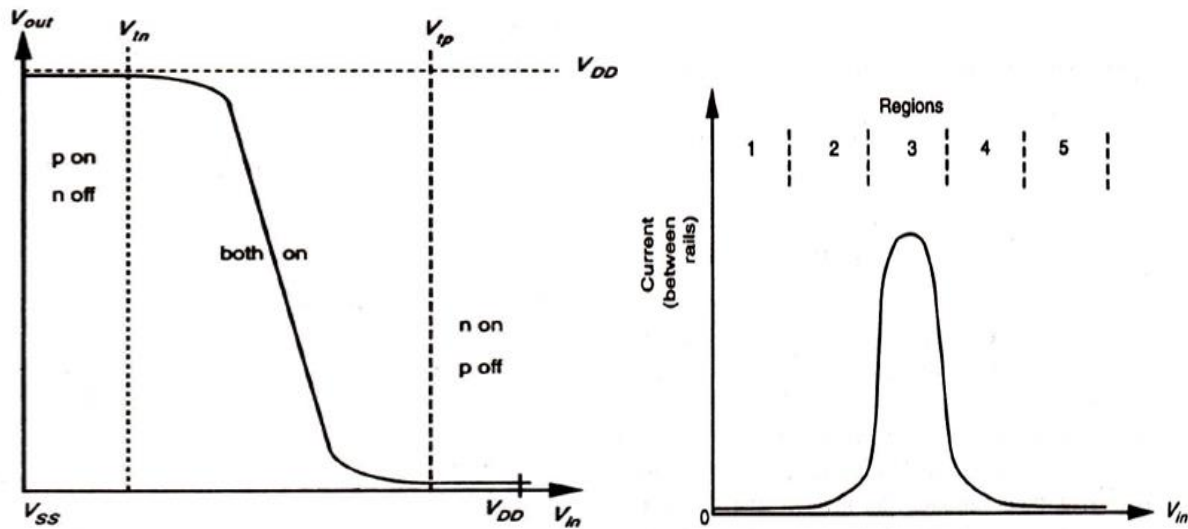
CMOS Inverter :

The inverter is the very important part of all digital designs. Once its operation and properties are clearly understood, Complex structures like NAND gates, adders, multipliers, and microprocessors can also be easily done. The electrical behavior of these complex circuits can be almost completely derived by extrapolating the results obtained for inverters. As shown in the diagram below the CMOS transistor is designed using p-MOS and n-MOS transistors.



In the inverter circuit ,if the input is high .the lower n-MOS device closes to discharge the capacitive load .Similarly ,if the input is low,the top p-MOS device is turned on to charge the capacitive load .At no time both the devices are on ,which prevents the DC current flowing from positive power supply to ground. Qualitatively this circuit acts like the switching circuit, since the p-channel transistor has exactly the opposite characteristics of the n-channel transistor. In the transition region both transistors are saturated and the circuit operates with a large voltage gain. The C-MOS transfer characteristic is shown in the below graph.

Considering the static conditions first, it may be seen that in region 1 for which $V_{in} = \text{logic } 0$, we have the p-transistor fully turned on while the n-transistor is fully turned off. Thus no current flows through the inverter and the output is directly connected to V_{DD} through the p-transistor.



Hence the output voltage is logic 1 . In region 5 , $V_{in} = \text{logic } 1$ and the n-transistor is fully on while the p-transistor is fully off. So, no current flows and a logic 0 appears at the output.

In region 2 the input voltage has increased to a level which just exceeds the threshold voltage of the n-transistor. The n-transistor conducts and has a large voltage between source and drain; so it is in saturation. The p-transistor is also conducting but with only a small voltage across it, it operates in the unsaturated resistive region. A small current now flows through the inverter from V_{DD} to V_{SS} . If we wish to analyze the behavior in this region, we equate the p-device resistive region current with the n-device saturation current and thus obtain the voltage and current relationships.

Region 4 is similar to region 2 but with the roles of the p- and n-transistors reversed. However, the current magnitudes in regions 2 and 4 are small and most of the energy consumed in switching from one state to the other is due to the larger current which flows in region 3.

Region 3 is the region in which the inverter exhibits gain and in which both transistors are in saturation.

The currents in each device must be the same, since the transistors are in series. So, we can write that

$$I_{dsp} = -I_{dsn}$$

where

$$I_{dsp} = \frac{\beta_p}{2} (V_{in} - V_{DD} - V_{tp})^2$$

and

$$I_{dsn} = \frac{\beta_n}{2} (V_{in} - V_{tn})^2$$

Since both transistors are in saturation, they act as current sources so that the equivalent circuit in this region is two current sources in series between V_{DD} and V_{SS} with the output voltage coming from their common point. The region is inherently unstable in consequence and the changeover from one logic level to the other is rapid.

Determination of Pull-up to Pull-Down Ratio ($Z_{p.u}/Z_{p.d}$) For an NMOS Inverter driven by another NMOS Inverter :

Let us consider the arrangement shown in Fig.(a). In which an inverter is driven from the output of another similar inverter. Consider the depletion mode transistor for which $V_{gs} = 0$ under all conditions, and also assume that in order to cascade inverters without degradation the condition

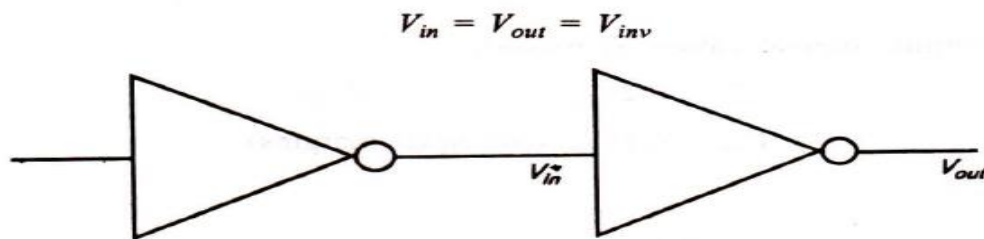


Fig.(a). Inverter driven by another inverter.

For equal margins around the inverter threshold, we set $V_{inv} = 0.5V_{DD}$. At this point both transistors are in saturation and we can write that

$$I_{ds} = K \frac{W}{L} \frac{(V_{gs} - V_t)^2}{2}$$

In the depletion mode $I_{ds} = K \frac{W_{p.u.}}{L_{p.u.}} \frac{(-V_{td})^2}{2}$ since $V_{gs} = 0$

and in the enhancement mode

$$I_{ds} = K \frac{W_{p.d.}}{L_{p.d.}} \frac{(V_{inv} - V_t)^2}{2} \text{ since } V_{gs} = V_{inv}$$

Equating (since currents are the same) we have

$$\frac{W_{p.d.}}{L_{p.d.}} (V_{inv} - V_t)^2 = \frac{W_{p.u.}}{L_{p.u.}} (-V_{td})^2$$

Where $W_{p.d.}$, $L_{p.d.}$, $W_{p.u.}$ And $L_{p.u.}$ are the widths and lengths of the pull-down and pull-up transistors respectively.

So, we can write that

$$Z_{p.d.} = \frac{L_{p.d.}}{W_{p.d.}}; Z_{p.u.} = \frac{L_{p.u.}}{W_{p.u.}}$$

we have

$$\frac{1}{Z_{p.d.}} (V_{inv} - V_t)^2 = \frac{1}{Z_{p.u.}} (-V_{td})^2$$

whence

$$V_{inv} = V_t - \frac{V_{td}}{\sqrt{Z_{p.u.}/Z_{p.d.}}}$$

The typical, values for V_t , V_{inv} and V_{td} are

$$V_t = 0.2V_{DD}; V_{td} = -0.6V_{DD}$$

$$V_{inv} = 0.5V_{DD} \text{ (for equal margins)}$$

Substituting these values in the above equation, we get

$$0.5 = 0.2 + \frac{0.6}{\sqrt{Z_{p.u.}/Z_{p.d.}}}$$

Here

$$\sqrt{Z_{p.u.}/Z_{p.d.}} = 2$$

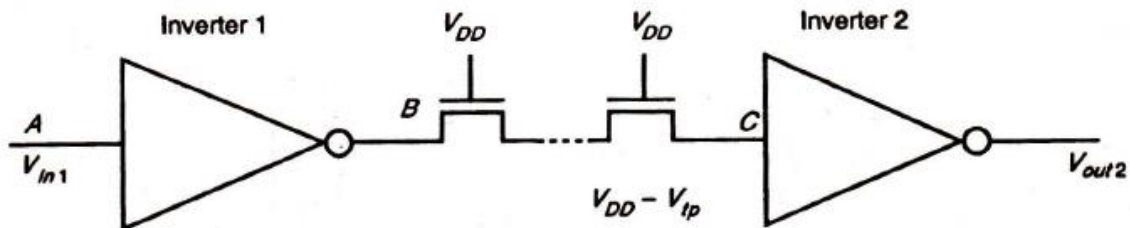
So, we get

$$Z_{p.u.}/Z_{p.d.} = 4/1$$

This is the ratio for pull-up to pull down ratio for an inverter directly driven by another inverter.

Pull-Up to Pull-Down ratio for an NMOS Inverter driven through one or more Pass Transistors

Let us consider an arrangement in which the input to inverter 2 comes from the output of inverter 1 but passes through one or more NMOS transistors as shown in Fig. Below (These transistors are called pass transistors).



The connection of pass transistors in series will degrade the logic 1 level / into inverter 2 so that the output will not be a proper logic 0 level. The critical condition is , when point A is at 0 volts and B is thus at V_{DD} . But the voltage into inverter 2 at point C is now reduced from V_{DD} by the threshold voltage of the series pass transistor. With all pass transistor gates connected to V_{DD} there is a loss of V_{tp} , however many are connected in series, since no static current flows through them and there can be no voltage drop in the channels. Therefore, the input voltage to inverter 2 is

$$V_{in2} = V_{DD} - V_{tp}$$

Where V_{tp} = threshold voltage for a pass transistor.

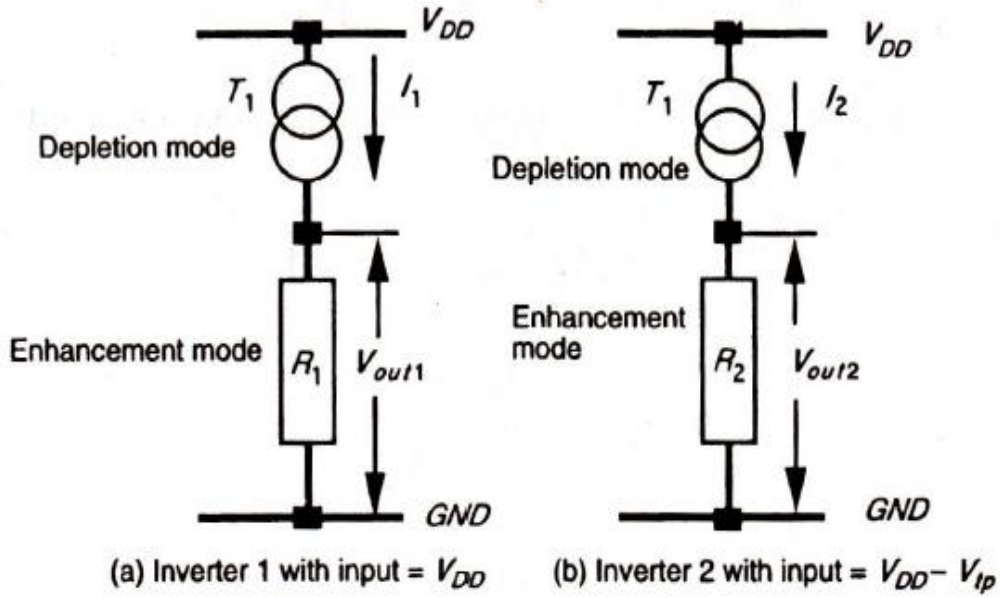
Let us consider the inverter 1 shown in Fig.(a) with input = V_{DD} . If the input is at V_{DD} , then the pull-down transistor T2 is conducting but with a low voltage across it; therefore, it is in its resistive region represented by R_1 in Fig.(a) below. Meanwhile, the pull up transistor T1 is in saturation and is represented as a current source.

For the pull down transistor

$$R_1 = \frac{V_{ds1}}{I_{ds}} = \frac{1}{K} \frac{L_{p.d.1}}{W_{p.d.1}} \left(\frac{1}{V_{DD} - V_t - \frac{V_{ds1}}{2}} \right)$$

$$I_{ds} = K \frac{W_{p.d.1}}{L_{p.d.1}} \left((V_{DD} - V_t) V_{ds1} - \frac{V_{ds1}^2}{2} \right)$$

Since V_{ds} is small, $V_{ds}/2$ can be neglected in the above expression.



So,

$$R_1 \doteq \frac{1}{K} Z_{p.d.1} \left(\frac{1}{V_{DD} - V_t} \right)$$

Now, for depletion mode pull-up transistor in saturation with $V_{gs} = 0$

$$I_1 = I_{ds} = K \frac{W_{p.u.1}}{L_{p.u.1}} \frac{(-V_{td})^2}{2}$$

The product

$$I_1 R_1 = V_{out1}$$

So,

$$V_{out1} = I_1 R_1 = \frac{Z_{p.d.1}}{Z_{p.u.1}} \left(\frac{1}{V_{DD} - V_t} \right) \frac{(V_{td})^2}{2}$$

Let us now consider the inverter 2 Fig.b .when input = $V_{DD} - V_{tp}$.

$$R_2 \doteq \frac{1}{K} Z_{p.d.2} \frac{1}{((V_{DD} - V_{tp}) - V_t)}$$

$$I_2 = K \frac{1}{Z_{p.u.2}} \frac{(-V_{td})^2}{2}$$

Whence,

$$V_{out2} = I_2 R_2 = \frac{Z_{p.d.2}}{Z_{p.u.2}} \left(\frac{1}{V_{DD} - V_{tp} - V_t} \right) \frac{(-V_{td})^2}{2}$$

If inverter 2 is to have the same output voltage under these conditions then $V_{out1} = V_{out2}$. That is

$$I_1 R_1 = I_2 R_2 \quad , \quad \text{therefore}$$

$$\frac{Z_{p.u.2}}{Z_{p.d.2}} = \frac{Z_{p.u.1}}{Z_{p.d.1}} \frac{(V_{DD} - V_t)}{(V_{DD} - V_{tp} - V_t)}$$

Considering the typical values

$$V_t = 0.2V_{DD}$$

$$V_{tp} = 0.3V_{DD}^*$$

$$\frac{Z_{p.u.2}}{Z_{p.d.2}} = \frac{Z_{p.u.1}}{Z_{p.d.1}} \frac{0.8}{0.2}$$

Therefore

$$\frac{Z_{p.u.2}}{Z_{p.d.2}} \doteq 2 \frac{Z_{p.u.1}}{Z_{p.d.1}} = \frac{8}{1}$$

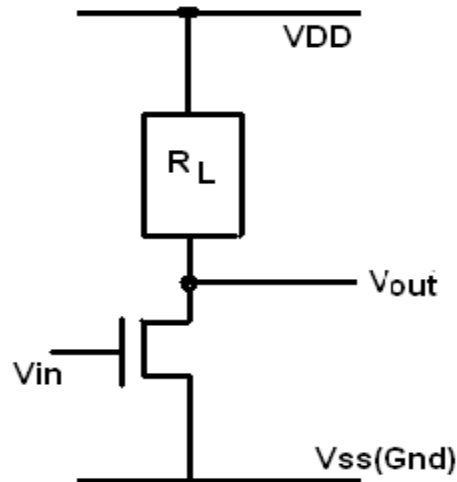
From the above theory it is clear that, for an n-MOS transistor

- (i). An inverter driven directly from the output of another should have a $Z_{p,u}/Z_{p,d}$ Ratio Of $\geq 4/1$.
- (ii). An inverter driven through one or more pass transistors should have a $Z_{p,u}/Z_{p,d}$ ratio of $\geq 8/1$

ALTERNATIVE FORMS OF PULL -UP

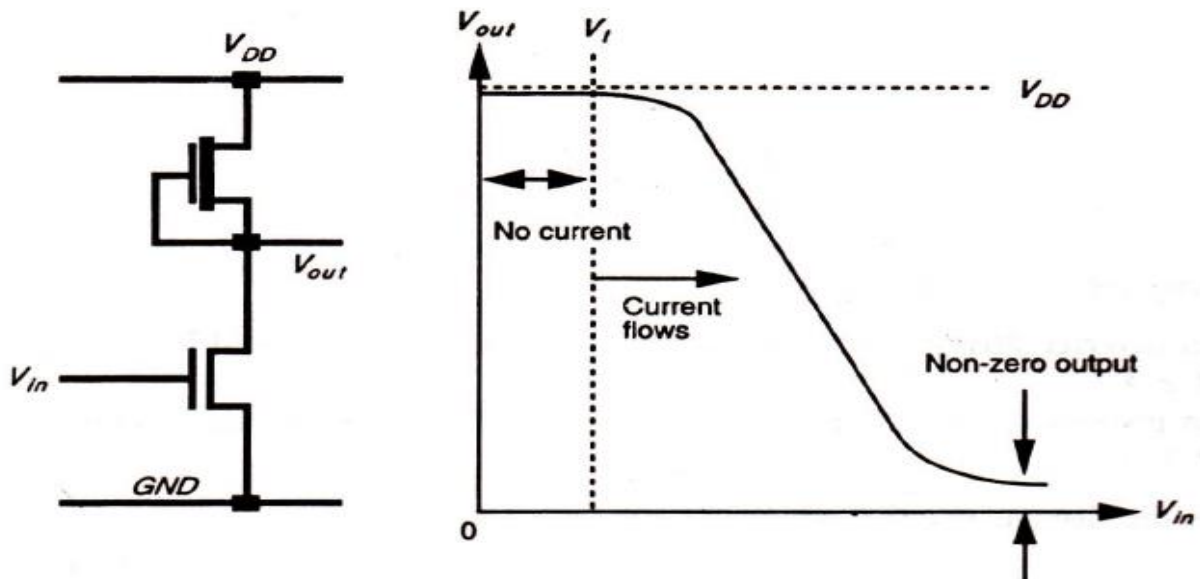
Generally the inverter circuit will have a depletion mode pull-up transistor as its load. But there are also other configurations. Let us consider four such arrangements.

(i). **Load resistance R_L** : This arrangement consists of a load resistor as a pull-up as shown in the diagram below. But it is not widely used because of the large space requirements of resistors produced in a silicon substrate.



2. **NMOS depletion mode transistor pull-up** : This arrangement consists of a depletion mode transistor as pull-up. The arrangement and the transfer characteristic are shown below. In this type of arrangement we observe

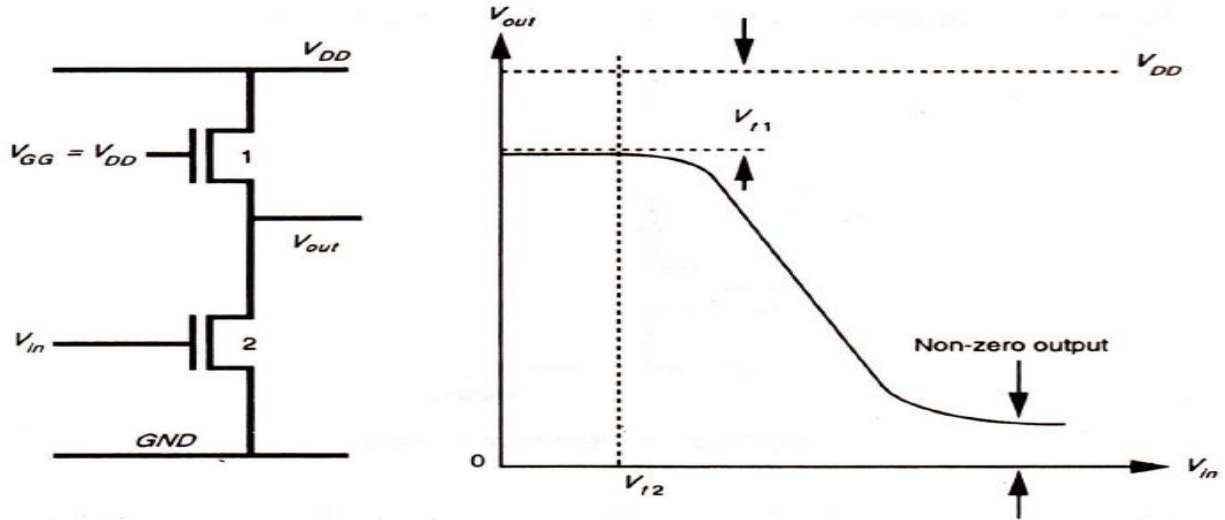
- (a) Dissipation is high, since rail to rail current flows when $V_{in} = \text{logical } 1$.
- (b) Switching of output from 1 to 0 begins when V_{in} exceeds V_t of pull-down device.



NMOS depletion mode transistor pull-up and transfer characteristic

(c) When switching the output from 1 to 0, the pull-up device is non-saturated initially and this presents lower resistance through which to charge capacitive loads .

3. NMOS enhancement mode pull-up : This arrangement consists of a n-MOS enhancement mode transistor as pull-up. The arrangement and the transfer characteristic are shown below.

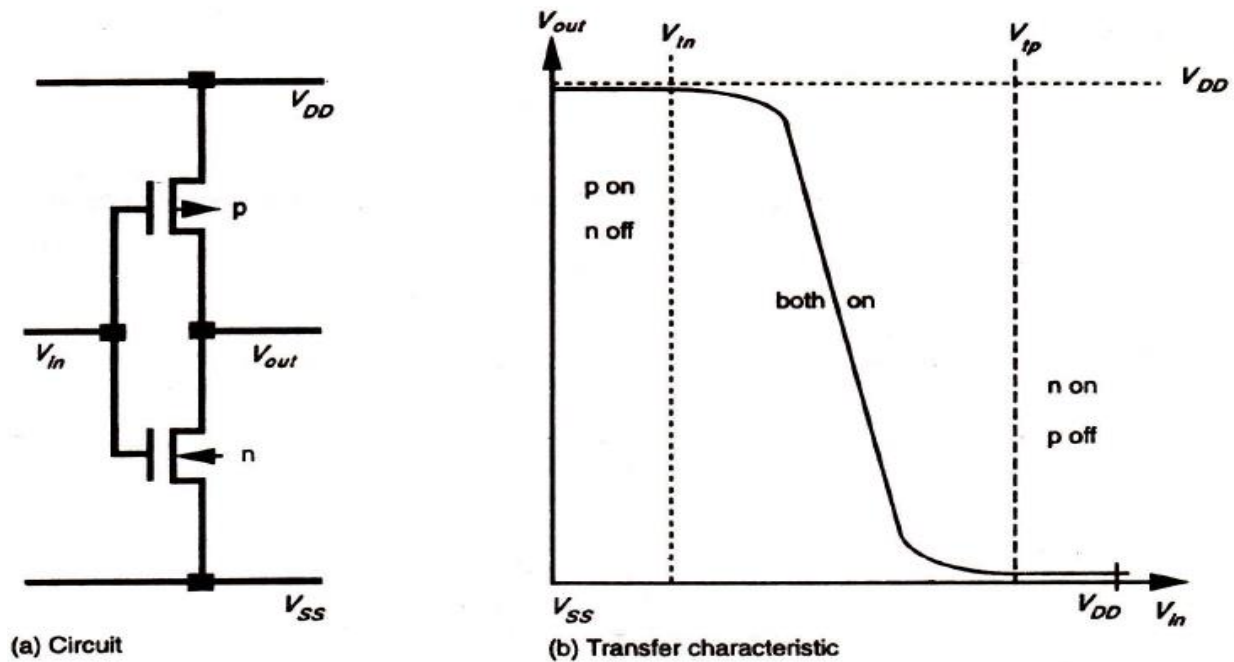


NMOS enhancement mode pull-up and transfer characteristic

The important features of this arrangement are

- (a) Dissipation is high since current flows when $V_{in} = \text{logical 1}$ (V_{GG} is returned to V_{DD}).
- (b) V_{out} can never reach V_{DD} (logical 1) if $V_{GG} = V_{DD}$ as is normally the case.
- (c) V_{GG} may be derived from a switching source, for example, one phase of a clock, so that Dissipation can be greatly reduced.
- (d) If V_{GG} is higher than V_{DD} then an extra supply rail is required.

4. Complementary transistor pull-up (CMOS) : This arrangement consists of a C-MOS arrangement as pull-up. The arrangement and the transfer characteristic are shown below

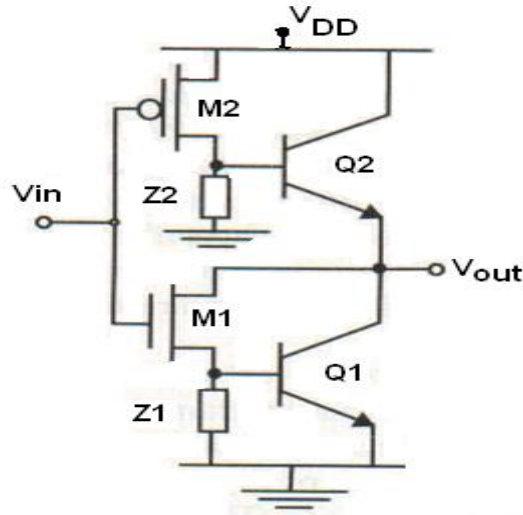


The salient features of this arrangement are

- (a) No current flows either for logical 0 or for logical 1 inputs.
- (b) Full logical 1 and 0 levels are presented at the output.
- (c) For devices of similar dimensions the p-channel is slower than the n-channel device.

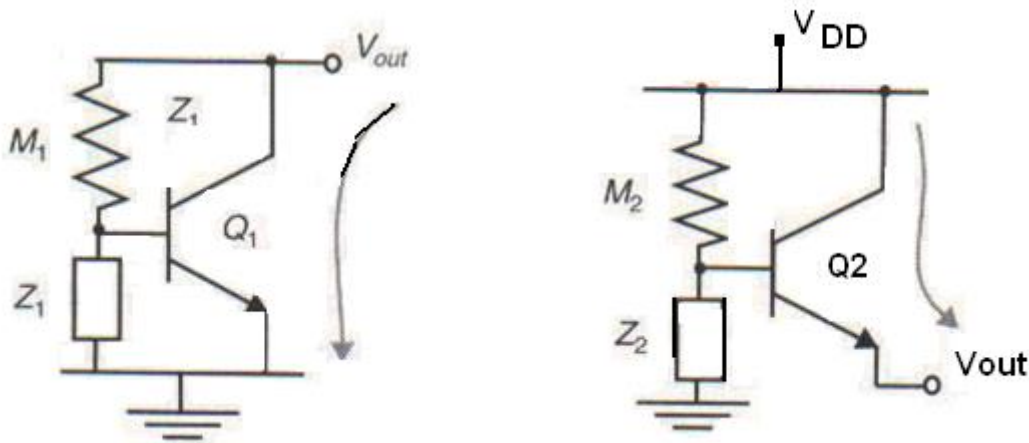
THE BiCMOS INVERTER :

A BiCMOS inverter, consists of a PMOS and NMOS transistor (M2 and M1), two NPN bipolar junction transistors, (Q2 and Q1), and two impedances which act as loads (Z2 and Z1) as shown in the circuit below.



When input, V_{in} , is high (V_{DD}), the NMOS transistor (M_1), turns on, causing Q_1 to conduct, while M_2 and Q_2 are off, as shown in figure (b). Hence, a low (GND) voltage is translated to the output V_{out} . On the other hand, when the input is low, the M_2 and Q_2 turns on, while m_1 and Q_1 turns off, resulting to a high output level at the output as shown in Fig.(b).

In steady-state operation, Q_1 and Q_2 never turns on or off simultaneously, resulting to a lower power consumption. This leads to a push-pull bipolar output stage. Transistors m_1 and M_2 , on the other hand, works as a phase-splitter, which results to a higher input impedance.



The impedances Z_2 and Z_1 are used to bias the base-emitter junction of the bipolar transistor and to ensure that base charge is removed when the transistors turn off. For example when the input voltage makes a high-to-low transition, M_1 turns off first. To turn off Q_1 , the base charge must be removed, which can be achieved by Z_1 . With this effect, transition time reduces. However, there

exists a short time when both Q1 and Q2 are on, making a direct path from the supply (V_{DD}) to the ground. This results to a current spike that is large and has a detrimental effect on both the noise and power consumption, which makes the turning off of the bipolar transistor fast .

Comparison of BiCMOS and C-MOS technologies

The BiCMOS gates perform in the same manner as the CMOS inverter in terms of power consumption, because both gates display almost no static power consumption.

When comparing BiCMOS and CMOS in driving small capacitive loads, their performance are comparable, however, making BiCMOS consume more power than CMOS. On the other hand, driving larger capacitive loads makes BiCMOS in the advantage of consuming less power than CMOS, because the construction of CMOS inverter chains are needed to drive large capacitance loads, which is not needed in BiCMOS.

The BiCMOS inverter exhibits a substantial speed advantage over CMOS inverters, especially when driving large capacitive loads. This is due to the bipolar transistor's capability of effectively multiplying its current.

For very low capacitive loads, the CMOS gate is faster than its BiCMOS counterpart due to small values of C_{int} . This makes BiCMOS ineffective when it comes to the implementation of internal gates for logic structures such as alus, where associated load capacitances are small.

BiCMOS devices have speed degradation in the low supply voltage region and also BiCMOS is having greater manufacturing complexity than CMOS.

Basic circuit concepts

Simple MOS capacitance Model:-

- The gate terminal of MOS transistor offers a considerable capacitance also its capacitance is necessary to attract charge to invert the channel. A high gate capacitance is desirable to obtain high I_{DS} .
- The gate capacitance is a parallel plate capacitance with a gate on top and channel on bottom and thin oxide dielectric is between the plates. The gate capacitance is given by

$$C_g = C_o W \cdot L$$

where, C_o is capacitance per unit area of gate oxide
 W is channel width
 L is length of channel

- When, the transistor is ON, the channel extends from source to drain when transistor is unsaturated.
- In order to achieve higher speed & lower power consumption, the transistor length is kept minimum.
- Thus taking this minimum L as a constant for particular process, the capacitance is defined as

$$C_g = C_{\text{permicron}} W$$

where, $C_{\text{permicron}} = C_o L = \frac{\epsilon_o}{t_o} \cdot L$

t_o is oxide thickness

ϵ_o is permittivity of free space $= (8.85 \times 10^{-14} \text{ F/cm})$

→ The source and drain also have capacitances. These capacitances do not affect the operation of device but they affect the circuit performance and are called as parasitic capacitances.

→ The parasitic capacitance arise from reverse biased p-n junction and are called as diffusion capacitance. The size of these junction are dependent on

- Area and perimeter of source and drain diffusion
- The depth of diffusion
- The doping levels
- Voltage applied.

Parasitic capacitance :-

→ The parasitic capacitance associated with a MOSFET is shown in below figure.

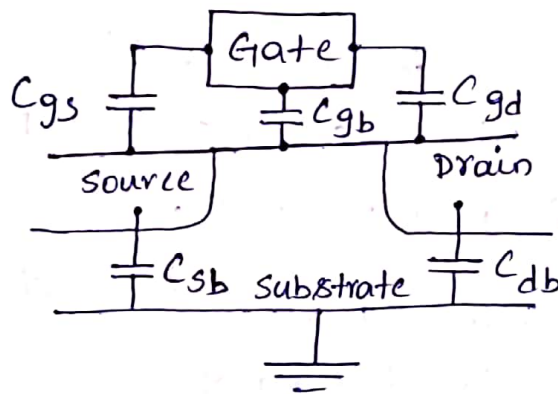


Fig:- Parasitic capacitance associated with a MOSFET

→ Various parasitic capacitances are:

- C_{gs} represents gate to source capacitance
- C_{gb} represents gate to substrate capacitance
- C_{gd} represents gate to drain capacitance
- C_{sb} represents source to substrate capacitance

C_{db} represents drain to substrate capacitance. (2)

→ C_{sb} and C_{db} are known as junction capacitance or diffusion capacitance arises from the depletion charge between the source and oppositely doped substrate. Similarly between drain and substrate. The capacitance varies with source or drain voltage.

→ The total capacitance seen from gate terminal of CMOS transistor is

$$C_g = C_{gs} + C_{gb} + C_{gd}$$

Detailed MOS Gate capacitance model:-

→ The MOS gate located above the channel overlapping, the source and drain diffusion areas. Hence gate capacitance has two components

- i) Intrinsic capacitance (over the channel)
- ii) Overlap capacitance (to the source, drain and body).

→ The intrinsic capacitance is approximated as simple parallel plate capacitance and overlap (bottom plate) capacitance depends on mode of operation (cut-off, linear and saturation) of transistor.

1. Cut-off: when transistor is OFF i.e. $V_{gs} = 0$, the charge on the gate is matched with opposite charge from the body (as channel is not inverted), this is called gate-to-body capacitance (C_{gb}).

when V_{gs} is below threshold, a depletion region forms at the surface. Because of this bottom plate moves downward from oxide and reduces the capacitance.

2. Linear: when $V_{gs} > V_t$, the channel inverts and again serves as a good conductive bottom plate. But the channel is connected to the source and drain rather than the body. For low values of the channel charge is shared between source and drain

$$\text{i.e., } C_{gs} = C_{gd} = C_0/2.$$

→ When V_{gs} is increased, the region near drain becomes less inverted, therefore more fraction of capacitance is attributed to source compared to drain.

3. Saturation: when $V_{ds} > (V_{gs} - V_t)$, the channel is saturated and channel pinches off. All capacitance is attributed to source. Due to pinch off, capacitance reduces to $C_{gs} = \frac{2}{3} C_0$.

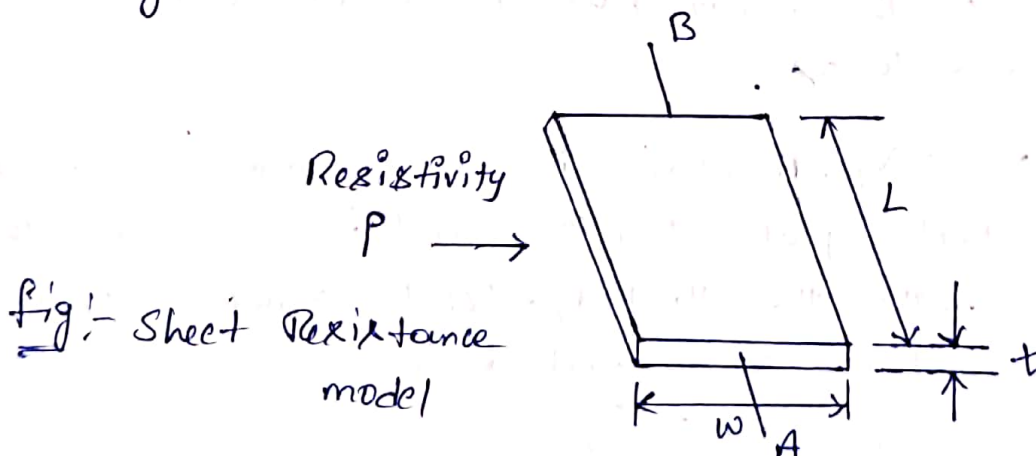
→ The intrinsic MOS gate capacitances in these three regions are summarized in below table.

Parameter	Cut-off region	Linear region	Saturation region
C_{gb}	C_0	0	0
C_{gs}	0	$C_0/2$	$\frac{2}{3} C_0$
C_{gd}	0	$C_0/2$	0
$C_g = C_{gs} + C_{gd} + C_{gb}$	C_0	C_0	$\frac{2}{3} C_0$

Table: Intrinsic MOS gate capacitance

Sheet Resistance R_s :-

→ consider a uniform slab of conducting material of resistivity P , of width ' w ', thickness ' t ' and length between faces ' L '. The arrangement is shown in below figure.



→ with reference to above figure, consider the resistance R_{AB} between two opposite faces. (3)

$$R_{AB} = \frac{\rho L}{A} \text{ ohm}$$

where, A = cross-section area

Thus
$$R_{AB} = \frac{\rho L}{tW} \text{ ohm}$$

Now, consider the case in which $L=W$, that is, a square of resistive material, then

$$R_{AB} = \frac{\rho}{t} = R_s$$

where, R_s = sheet resistance (ρ/t) ohm per square

Thus
$$R_s = \frac{\rho}{t} \text{ ohm/square}$$

→ Note that R_s is completely independent of area of the square. For example, a $1 \mu\text{m}$ per side square slab of material has exactly the same resistance as a 1cm per side square slab of the same material if the thickness is same.

→ Thus the actual values associated with the layers in a MOS circuit depend on the thickness of the layer and the resistivity of the material forming the layer.

→ For the metal and polysilicon layers, the thickness of a layer is easily predicted and the resistivity of the material is known.

→ For the diffusion layer, the depth of the diffusion regions contribute toward the effective thickness while the impurity concentration profile determines the resistivity.

For the MOS processes considered here, typical values of sheet resistance are given in Table.

→ Typical sheet resistances R_s of MOS layers for $5\mu\text{m}$, $2\mu\text{m}$ and $1.2\mu\text{m}$ technologies.

Layer	R_s Ohm per square		
	$5\mu\text{m}$	orbit $2\mu\text{m}$	orbit $1.2\mu\text{m}$
Metal	0.03	0.04	0.04
Diffusion	10 → 50	20 → 45	20 → 45
silicide	2 → 4	—	—
Polysilicon	15 → 100	15 → 30	15 → 30
n-transistor channel	10^4	2×10^4	2×10^4
p-transistor channel	2.5×10^4	4.5×10^4	4.5×10^4

Sheet Resistance concept applied to MOS Transistors & Inverters :-

→ Consider the transistor structures shown in below figure and note that the diagrams distinguish the actual diffusion (active) regions from the channel regions.

→ The simple n-type pass transistor shown in figure (a) has a channel length $L = 2\lambda$ and a channel width $w = 2\lambda$. The channel is therefore square and channel resistance is given as

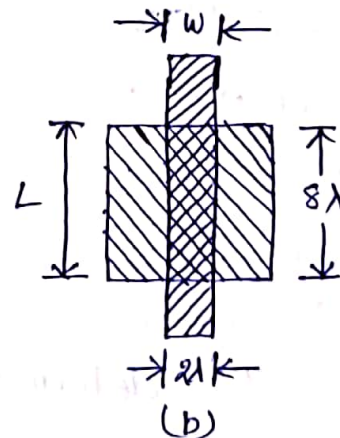
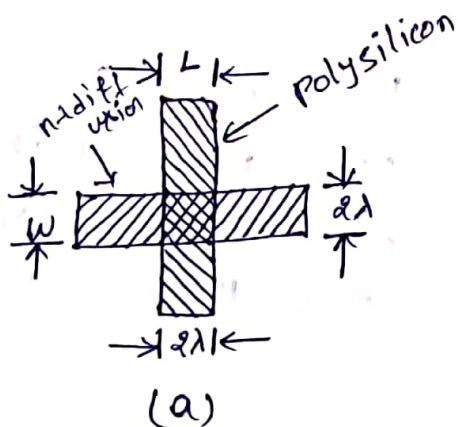


Fig:- Resistance calculation for transistor channels

$$R = 1 \text{ square} \times R_s \frac{\text{ohm}}{\text{square}} = R_s = 10^4 \text{ ohm} = 10 \text{ k}\Omega \quad (4)$$

→ The length to width ratio, denoted with Z is 1:1 in the above case.

→ The transistor structure of figure (b) has a channel length $L = 8\lambda$ and width $w = 2\lambda$. Therefore,

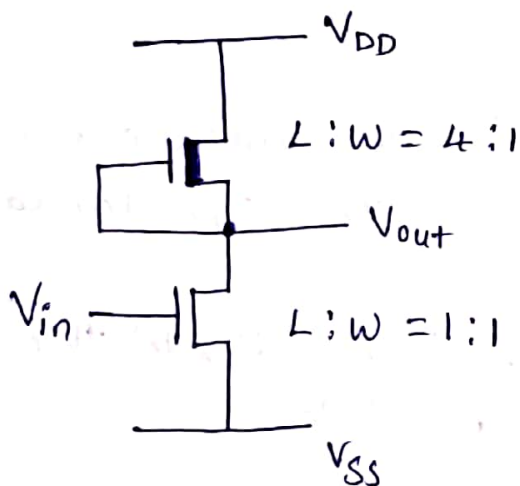
$$Z = \frac{L}{w} = \frac{8}{2} = 4$$

Thus, channel resistance

$$R = Z R_s = 4 \times 10^4 \text{ ohm} = 40 \text{ k}\Omega$$

→ ON Resistance calculation for Inverters:

→ consider an NMOS inverter circuit as shown in below figure (a)



Pull-up transistor

$$L_{pu} : w_{pu} = 4 : 1 ; Z_{pu} = 4$$

$$R_{on} = Z_{pu} \times R_{sn}$$

$$= 4 \times 10^4 = 40 \text{ k}\Omega$$

Pull-down transistor

$$L_{pd} : w_{pd} = 1 : 1 ; Z_{pd} = 1$$

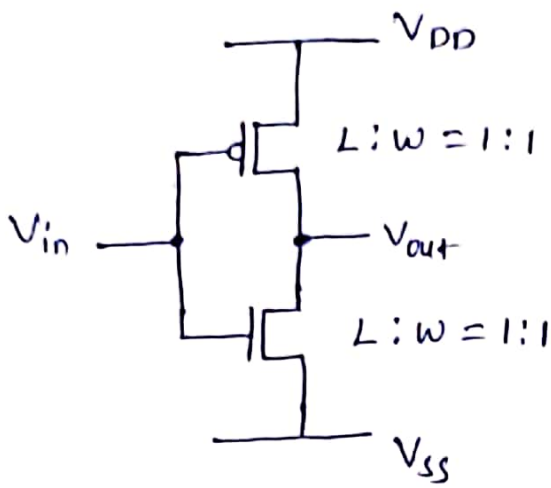
$$R_{on} = Z_{pd} \times R_{sn}$$

$$= 1 \times 10^4 = 10 \text{ k}\Omega$$

$$\therefore \text{Total on Resistance} = 40 \text{ k} + 10 \text{ k} = 50 \text{ k}\Omega$$

$$Z_{pu} : Z_{pd} = 4 : 1$$

→ consider a CMOS Inverter as shown in below figure (b)



Pull-up transistor

$$L_{pu} : W_{pu} = 1 : 1 ; Z_{pu} = 1$$

$$R_{on} = Z_{pu} \times R_{sp}$$

$$= 1 \times 2.5 \times 10^4 = 25 \text{ k}\Omega$$

Pull-down transistor

$$L_{pd} : W_{pd} = 1 : 1 ; Z_{pd} = 1$$

$$R_{on} = Z_{pd} \times R_{sn}$$

$$= 1 \times 10^4 = 10 \text{ k}\Omega$$

\therefore Total on Resistance, $R_{on} = 25 \text{ k} + 10 \text{ k} = 35 \text{ k}\Omega$

$$Z_{pu} : Z_{pd} = 1 : 1$$

Area capacitances of Layers :-

→ The conducting layers are separated from the substrate and each other by insulating (dielectric) layers, and these separated layers are called as area capacitances.

→ For any layer, knowing the dielectric (silicon dioxide) thickness, we can calculate area capacitance as follows

$$C = \frac{\epsilon_0 \epsilon_{ins} A}{D} \text{ farads}$$

where D = thickness of silicon dioxide

A = Area of plates

ϵ_{ins} = relative permittivity of $\text{SiO}_2 = 4.0$

ϵ_0 = Free space permittivity = $8.85 \times 10^{-14} \text{ F/cm}$

$$C = \frac{\epsilon_0 \epsilon_{ins}}{D} \left(\frac{PF}{\mu m^2} \right) \times A (\mu m^2)$$

This capacitance is always expressed as some constant value

i.e., $K \times 10^{-4} \frac{PF}{\mu m^2}$

→ The below table shows the typical area capacitance values for MOS circuits.

capacitance	Value in $PF \times 10^{-4} / \mu m^2$ (Relative values in brackets)		
	5 μm	2 μm	1.2 μm
Gate to channel	4 (1.0)	8 (1.0)	16 (1.0)
Diffusion (active)	1 (0.25)	1.75 (0.22)	3.75 (0.23)
Polysilicon to substrate	0.4 (0.1)	0.6 (0.075)	0.6 (0.038)
metal-1 to substrate	0.3 (0.075)	0.33 (0.04)	0.33 (0.02)
metal-2 to substrate	0.2 (0.05)	0.17 (0.02)	0.17 (0.01)
metal-2 to metal-1	0.4 (0.1)	0.5 (0.06)	0.5 (0.03)
metal-2 to polysilicon	0.3 (0.075)	0.3 (0.038)	0.3 (0.018)

Table: Typical area capacitance values for MOS circuits

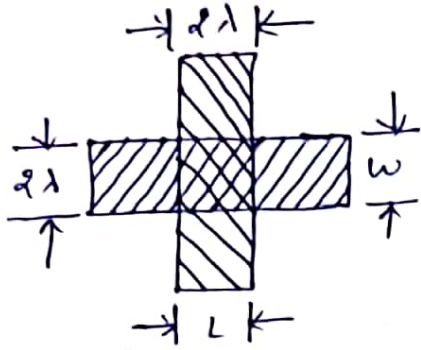
Standard unit of capacitance C_g :-

→ The standard unit of capacitance is denoted with C_g , & is defined as the gate-to-channel capacitance of a MOS transistor

having $w=L$ = Standard of square. $C_g = \frac{\epsilon_0 \epsilon_{ins}}{D} \times A$

C_g may be evaluated for any MOS process. For example, for $5\mu\text{m}$ MOS circuit.

$$\begin{aligned} \therefore A &= w \times L \\ &= 2\lambda \times 2\lambda \\ &= (2\lambda)^2 \end{aligned}$$



$$C_g = 4 \times 10^{-4} \text{ PF}/\mu\text{m}^2 \times 25 \mu\text{m}^2 = (5\mu\text{m})^2 = 25 \mu\text{m}^2$$

$$C_g = 100 \times 10^{-4} \text{ PF}$$

$$= 10^{-2} \text{ PF (or)}$$

$$= 10^{-2} \text{ PF (or)}$$

$$C_g = .01 \text{ pF}$$

→ It is the standard unit capacitance value for $5\mu\text{m}$ technology.

→ For $2\mu\text{m}$ MOS circuit

$$C_g = 8 \times 10^{-4} \text{ PF}/\mu\text{m}^2 \times 4 \mu\text{m}^2$$

$$= 32 \times 10^{-4} \text{ pF}$$

$$C_g = .0032 \text{ pF}$$

$$\begin{aligned} A &= w \times L \\ &= 2\lambda \times 2\lambda \\ &= (2\lambda)^2 \\ &= (2\mu\text{m})^2 \\ &= 4 \mu\text{m}^2 \end{aligned}$$

→ For $1.2\mu\text{m}$ MOS circuit

$$C_g = 16 \times 10^{-4} \text{ PF}/\mu\text{m}^2 \times 1.44 \mu\text{m}^2$$

$$C_g = 0.0023 \text{ pF}$$

$$\begin{aligned} A &= w \times L \\ &= 2\lambda \times 2\lambda \\ &= (2\lambda)^2 \\ &= (1.2\mu\text{m})^2 \\ &= 1.44 \mu\text{m}^2 \end{aligned}$$

Delay unit τ :- Delay unit is the time taken for output to reach 63% of final value from initial value. This is given by

$$\text{Time constant } \tau = [1 R_s (\text{n-channel}) \times 1 \text{ } \square C_g] \text{ seconds}$$

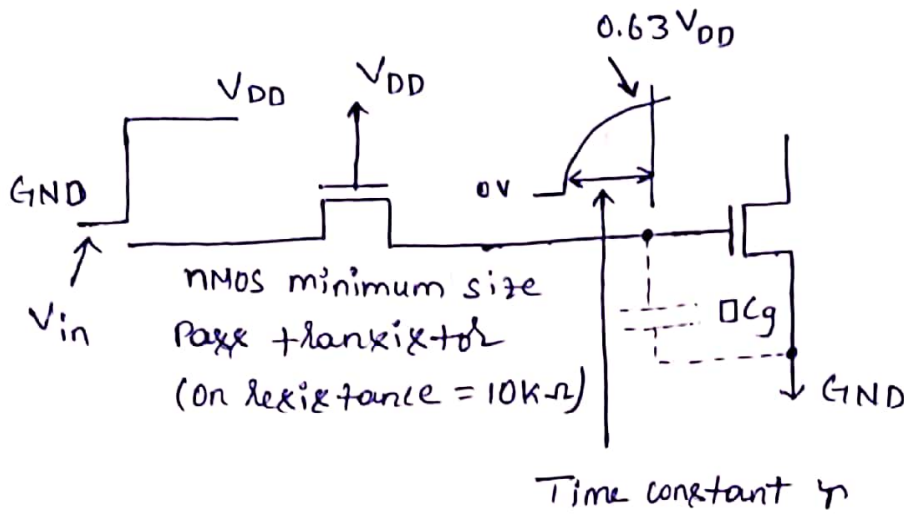


Fig:- Model for derivation of τ .

→ This delay time is depending on technology, for example for 5 μm technology,

$$\tau = 10^4 \Omega \times 0.01 \text{ pF} = 0.1 \text{ nsec}$$

for 2 μm technology,

$$\tau = 2 \times 10^4 \Omega \times 0.0032 \text{ pF} = 0.064 \text{ nsec}$$

for 1.2 μm technology,

$$\tau = 2 \times 10^4 \Omega \times 0.0023 \text{ pF} = 0.046 \text{ nsec}$$

Transit Time τ_{sd} :- Transit time is generally given by channel length per velocity.

$$\tau_{sd} = \frac{\text{Channel Length } (L)}{\text{velocity } (v)}$$

This velocity is drift velocity which is given by

$$V_{\text{drift}} = \mu_n E_{\text{ds}}$$

Now we know $E_{\text{ds}} = \frac{V_{\text{ds}}}{L}$

$$V_{\text{drift}} = \mu_n \frac{V_{\text{ds}}}{L}$$

Then $\tau_{\text{sd}} = \frac{L}{\mu_n \frac{V_{\text{ds}}}{L}} = \frac{L^2}{\mu_n V_{\text{ds}}}$

→ Note that V_{ds} varies as C_g charges from 0 volts to 63% of V_{DD} in period τ . So that an appropriate value for V_{ds} is the average value = 3 Volts.

→ However, circuit wiring and parasitic capacitances have an effect on propagation of signals. Hence in practice, the figure for τ is often increased by a factor of two or three & then used for design.

→ The safe figures recommended for use of different technologies are as follows:

→ for 5 μm MOS technology, $\tau = 0.3 \text{ nsec}$

→ for 2 μm MOS technology, $\tau = 0.2 \text{ nsec}$

→ for 1.2 μm MOS technology, $\tau = 0.1 \text{ nsec}$

Inverter Delay :-

→ consider a single basic nMOS inverter. The Z_{pu} to Z_{pd} ratio of this device is 4:1. To achieve this ratio,

$$4 R_{pu} = R_{pd}$$

→ If the R_{pd} is by a minimum size transistor, then its channel

$$R_s \text{ is } 10 \text{ k}\Omega. \text{ Then } R_{pu} = 4 R_s = 40 \text{ k}\Omega$$

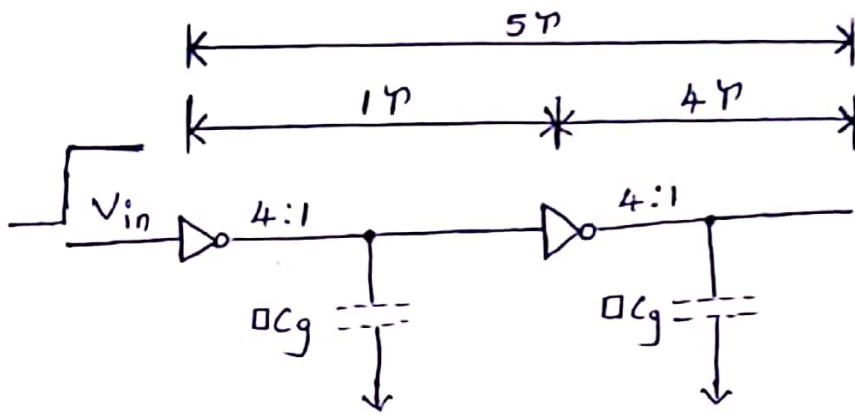


Fig:- NMOS inverter pair delay

→ Thus there is an asymmetry in R_{pu} & R_{pd} . Hence delay associated with inverter will depend on whether it is being turned-off or on.

→ However, conditions are different with a pair of cascaded inverters. As shown in above figure, here the delay over the inverter pair is constant irrespective of the type of logic level transition of the input of the first inverter.

→ This is because transition at input of second inverter is opposite to that at the input of first inverter.

→ Note that the delay in turning on is τ while the corresponding delay in turning-off is 4τ . Taking $\tau = 0.3 \text{ nsec}$ & making no extra provisions for wiring capacitance, the overall delay comes out to $\tau + 4\tau = 5\tau$.

→ In terms of pull-up & pull-down impedances, the delay through a pair of identical NMOS inverters can be generalized as

$$T_d = (1 + Z_{nu}/Z_{pd})^2$$

→ As per this formula, also, delay for pair of inverters having 4:1 ratio comes to 5τ.

CMOS Inverter Delay:-

→ In case of CMOS inverters, the ratio rule of nMOS inverter is not applicable.

→ However there is a natural asymmetry in the R_S between the pull-up and pull-down, this is because the pull-up device is a p-type and the pull-down device is of n-type.

→ Estimation of delay associated with a pair of inverters is shown in below figure. Since the input to a CMOS inverter is connected to the gate of both the transistors, the gate capacitance is double that of the comparable nMOS inverter.

→ The below figure also indicates the considerations made for the asymmetric channel resistances.

→ This resistance asymmetry can be avoided by increasing the width of the p-device channel by a factor of two or three.

→ However this also increases the gate input capacitance of the p-transistor by the same factor.

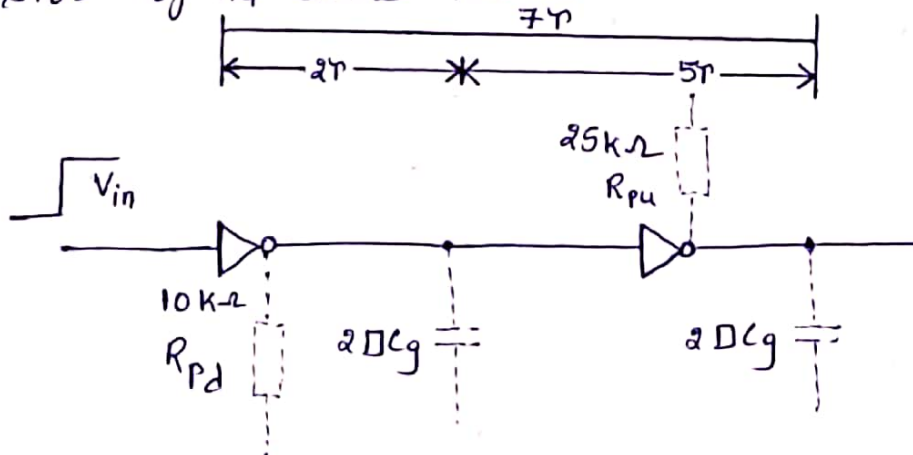


Fig:- Minimum size CMOS inverter pair delay

$$R_n = 10k\Omega$$

$$R_p = 25k\Omega$$

$$R_p = 2.5 R_n$$

$$R_{nmos} = R_s = R_n$$

$$\tau_1 = R_n \cdot 2 \cdot 0.69 C_g$$

$$\tau_2 = R_p \cdot 2 \cdot 0.69 C_g$$

$$\tau_d = \tau_1 + \tau_2$$

$$\tau_d = R_n (2 \cdot 0.69 C_g) + 2.5 R_n (2 \cdot 0.69 C_g)$$

$$= R_n \cdot 2 \cdot 0.69 C_g + 5 R_n \cdot 0.69 C_g$$

$$= 7 R_s \cdot 0.69 C_g$$

$$[\because \tau = R_s \cdot 0.69 C_g]$$

$$\tau_d = 7 \tau$$

Estimation of CMOS Inverter delay:-

→ The delay associated with the CMOS inverter can be estimated by splitting the output transitions into fall time τ_f & rise-time τ_r corresponding to the charging and discharging of the capacitive load C_L .

Estimation of Rise-time:-

→ The pull-up p device drives the capacitive load & can be assumed to be in saturation for the entire charging period of the load capacitor C_L .

→ The equivalent circuit for this condition is shown in below figure.

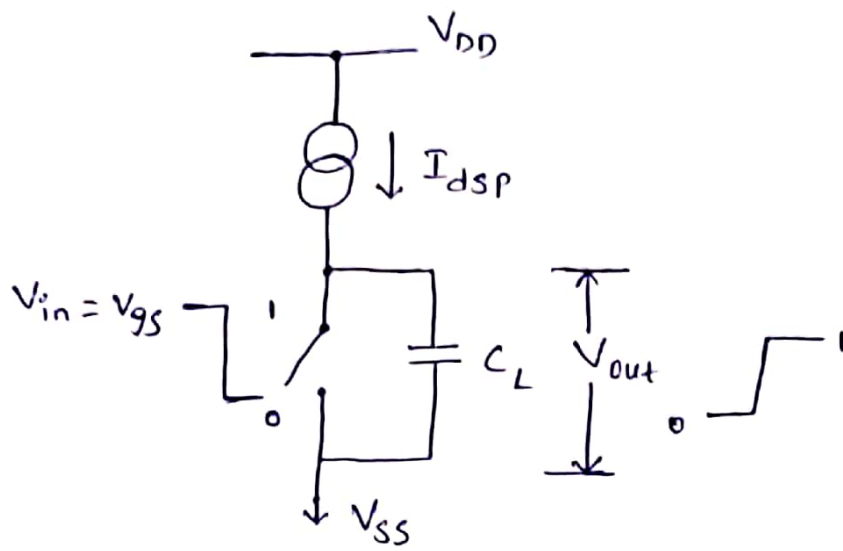


Fig.:- Rise-time model

The saturation current for the p-transistor is given by

$$I_{dsp} = \frac{\beta_p (V_{gs} - |V_{tp}|)^2}{2}$$

This current charges C_L & since its magnitude is constant, we have

$$V_{out} = \frac{I_{dsp} t}{C_L} \quad [\because t = \text{Time}]$$

Substitute I_{dsp} in above equation

$$V_{out} = \frac{\beta_p (V_{gs} - |V_{tp}|)^2 t}{2 C_L}$$

$$t = \frac{2 C_L V_{out}}{\beta_p (V_{gs} - |V_{tp}|)^2}$$

Now assume that $t = T_r$ when $V_{out} = +V_{DD}$ so that

$$T_r = \frac{2 C_L V_{DD}}{\beta_p (V_{gs} - |V_{tp}|)^2}$$

$$T_r = \frac{2C_L V_{DD}}{\beta_P (-V_{DD} - |V_{tp}|)^2}$$

$$\begin{aligned} \therefore V_{gs} &= V_g - V_s \quad (9) \\ &= 0 - V_s \\ V_{gs} &= -V_{DD} \end{aligned}$$

with $|V_{tp}| = -0.2V_{DD}$, then

$$T_r = \frac{2C_L V_{DD}}{\beta_P (-V_{DD} + 0.2V_{DD})^2}$$

$$T_r = \frac{2C_L V_{DD}}{\beta_P (-0.8V_{DD})^2}$$

$$T_r = \frac{2C_L V_{DD}}{\beta_P (-0.8)^2 V_{DD}^2}$$

$$T_r = \frac{2C_L}{\beta_P (0.64) V_{DD}}$$

$$\left[\therefore 2/0.64 = 3.126 \right]$$

$$T_r = \frac{3C_L}{\beta_P V_{DD}} \rightarrow (1)$$

Estimation of Fall-time :-

→ Fall-time is associated with the discharging of C_L through the pull-down n-type device. The equivalent circuit model for fall-time estimation is shown in below figure, it shows a constant discharge current.

Making similar assumptions we may write for fall-time

$$T_f = \frac{3C_L}{\beta_n V_{DD}} \rightarrow (2)$$

From $\tau_{au} \textcircled{1}$ & $\tau_{au} \textcircled{2}$ we may write

$$\frac{\tau_r}{\tau_f} \approx \frac{\beta_n}{\beta_p}$$

$$\left[\begin{aligned} \tau_r &= \frac{3C_L}{\mu_n \beta_n} \\ \tau_f &= \frac{3C_L}{\mu_p \beta_p} \\ &= \beta_n / \beta_p \end{aligned} \right]$$

→ However, μ_n & μ_p are not same and $\mu_n = 2.5 \mu_p$ because of which $\beta_n = 2.5 \beta_p$. This shows that rise time is slower by a factor of 2.5 when both n & p-devices are minimum size devices.

→ Keeping the channel length minimum, symmetrical operation can be achieved by making $w_p = 2.5 w_n$ where w_p is channel width of p-device & w_n that of n-device.

→ However, with the other geometries as per the minimum size lambda-based rules, this would result in the inverter having an input capacitance of $2.5 C_g$ for p-device plus $1 C_g$ for n-device giving a total capacitance of $3.5 C_g$.

→ The analytical models used above for the estimation of rise & fall-times are enough to get good results. But, they do not consider certain factors affecting the rise & fall-times such as

1. τ_r & τ_f are proportional to C_L .
2. τ_r & τ_f are proportional to $1/V_{DD}$.
3. For equal n & p-transistor geometries, $\tau_r = 2.5 \tau_f$.

Driving Large capacitance loads :-

→ whenever output signals are required to propagate from the chip to off-chip destinations, the output experiences a

comparatively large capacitive load.

→ Such off chip capacitances may be several orders higher than on chip $\square C_g$ values & typically the off-chip load

$$C_L \geq 10^4 \square C_g.$$

→ Such capacitances must clearly be driven through low resistances to avoid excessively long delays.

Cascaded Inverters as Drivers :-

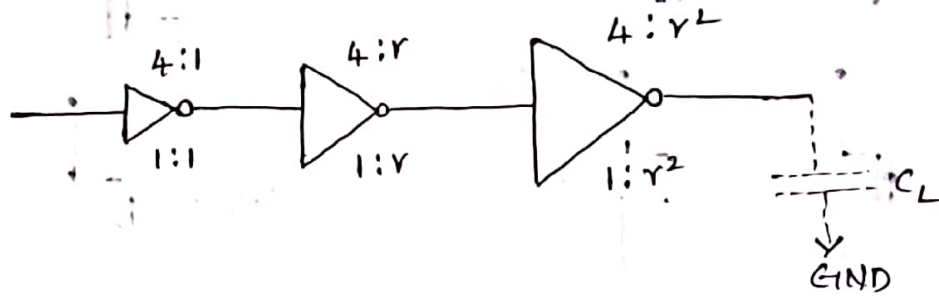


Fig:- Driving Large capacitance loads

→ Inverters which drives large capacitive loads must have low pull up and pull-down resistance.

→ For MOS circuits low resistance values for Z_{pu} & Z_{pd} means low L:w ratio.

→ Since channel length cannot be reduced below the minimum feature size, & the width 'w' must be made wide. This in turn increases the chip area and capacitance at input and slow down the rate of change of voltage at the input.

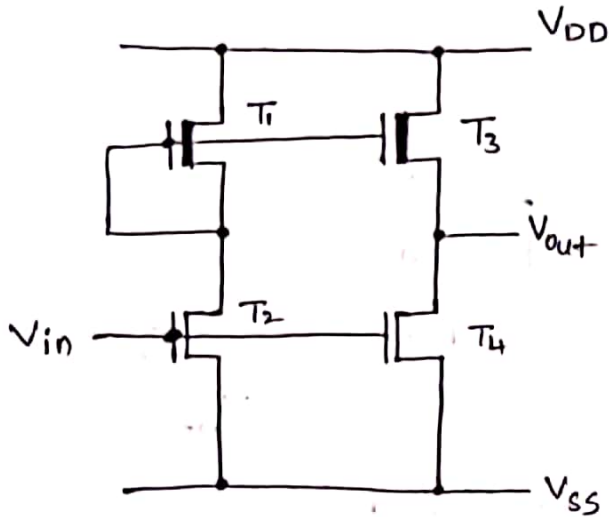
→ The remedy is to use N cascaded inverters, each one of which is larger than the preceding stage by a width factor 'r' as shown in above figure. The rate at which width factor 'r' increases, the number of inverters N decreases but delay per stage increases.

Delay per stage = τ_p for ΔV_{in} } ΔV_{in} indicates logic 0 to 1
 = $4\tau_p$ for ∇V_{in} } ∇V_{in} indicates logic 1 to 0

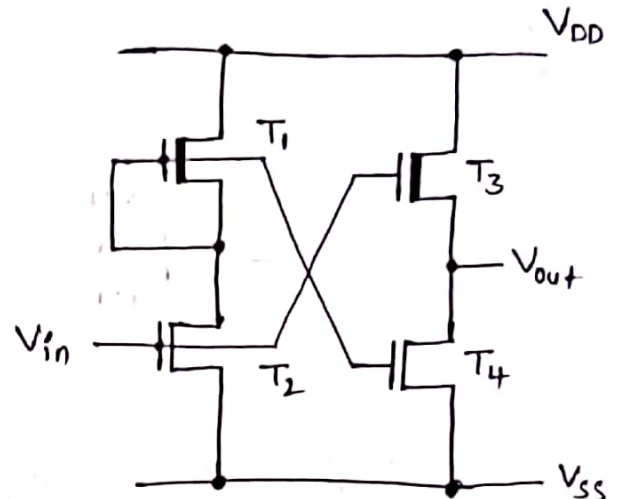
\therefore Total delay per nMOS pair is $5\tau_p$.

\therefore Similarly delay per CMOS pair is $7\tau_p$.

Super Buffers :-



fig(a):- Inverting type nMOS super buffer



fig(b):- Non-Inverting type nMOS Super buffer

- The asymmetry of the conventional inverter is undesirable to drive large capacitive loads due to delay problem.
- A common approach is to use nMOS super buffers as shown in above fig (a) & (b).
- An inverting type super buffer is shown in fig (a). Consider a positive going (0 to 1) transition at the input V_{in} turns on the inverter formed by T_1 & T_2 .
- With a small delay, the gate of T_3 is pulled down to 0 volts.
- Thus device T_3 is cut-off while T_4 is connected to V_{in} , it is turned on and the output is pulled down very fast.

→ Now consider the opposite transition of V_{in} (1 to 0), V_{in} drops to '0' volts, The gate of device T_3 is allowed to rise to V_{DD} very quickly, simultaneously the low V_{in} turns-off T_4 very fast.

→ This makes T_3 to conduct with its gate voltage approximating V_{DD} . Now, $I_{DS} \propto V_{GS}$, the effective V_{GS} increases the current and thereby reduces the delay in charging at the load capacitance of the output.

Bicmos Drivers :-

→ since the Bicmos technology is enriched with bipolar transistors, it is convenient to use bipolar transistor drivers as the output stage of inverter & logic gate circuits.

→ Bipolar transistors have far superior characteristics, especially the transconductance g_m & the current area I/A characteristics, as compared to those of MOS transistors.

→ Bicmos devices have high current drive capabilities in spite of occupying smaller areas in silicon.

→ In the bipolar transistors, there is an exponential dependence of the collector current I_c on the base to emitter voltage V_{be} . Hence the bipolar transistors can be operated with much smaller input voltage swings than MOS transistors and still switch larger currents.

→ This better switching performance is offset by the fact that a small amount of charge is required to be moved during switching.

→ Another consideration in bipolar devices is that of its temperature effect on input voltage V_{be} .

→ The switching performance of a bipolar transistor driving a capacitive load can be analyzed to begin with the help of the equivalent circuit given in below figure.

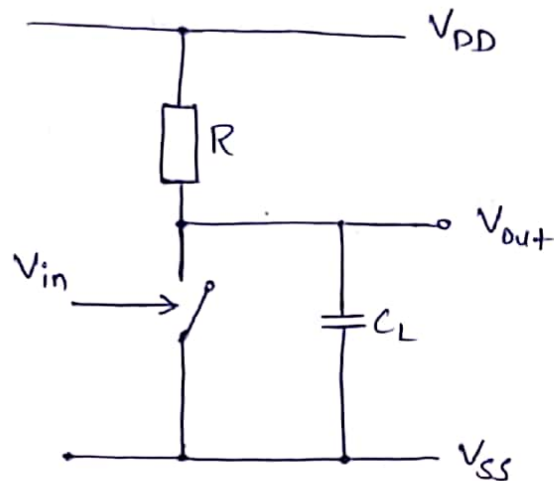


Fig:- Driving ability of bipolar transistor

→ The time Δt required to charge the output voltage V_{out} by an amount equal to the input voltage V_{in} is

$$\Delta t = \frac{C_L}{g_m}$$

→ where C_L is the load capacitance and g_m the transconductance of the bipolar transistor.

→ The value of Δt is small because the transconductance of the bipolar transistor is relatively higher.

→ A more detailed analysis of the delay due to the bipolar transistor reveals that, it is made up of due to two main components i.e., T_{in} & T_L .

→ The time T_{in} is that time required to first charge the base emitter junction of the bipolar (npn) transistor.

- T_{max} time is typically 8ns for the BiCMOS transistor-based driver. The time T_{in} for the CMOS driver required to charge the input gate capacitance is 1ns. T_{in} in case of GaAs driver is around 50-100 ps. Thus the T_{in} for bipolar transistors is the highest in comparison.
- The time T_L is the time required to charge the output load capacitance C_L and equals $(V/I_D)(1/h_{fe})C_L$. This is less for the bipolar driver by a factor of h_{fe} as compared to MOS drivers.
- The parameter h_{fe} is the gain of the bipolar transistor. Thus T_L for the bipolar transistor is less. This compensates for the higher value of T_{in} .
- Another significant aspect while considering delay is the collector resistance R_C through which the charging current for C_L flows.
- Hence a high value of R_C results into a longer propagation delay.
- The effect of value of R_C on the delay can be understood from below figure which shows typical delay values at two values of C_L as a function of R_C .

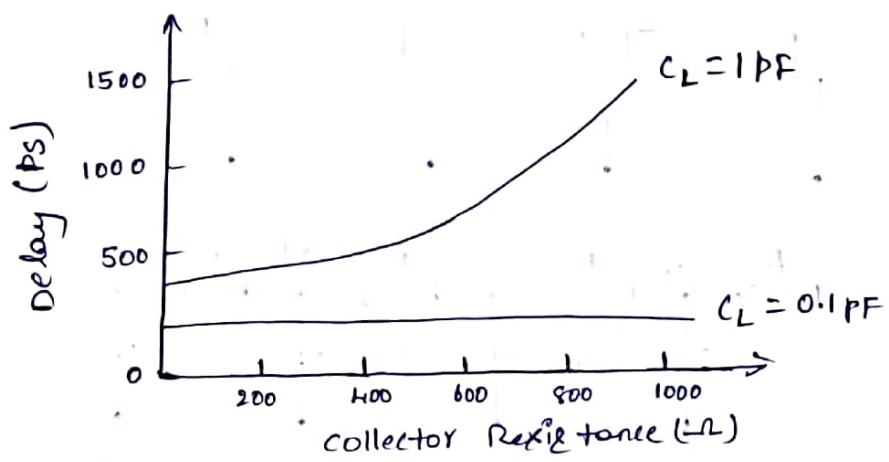


Fig:- Gate delay as a function of collector resistance

→ The use of the buried subcollector (BCSD) region in BiCMOS fabrication is to keep R_c as low as possible.

→ All these desired features are incorporated into the bipolar devices under the BiCMOS fabrication process. The device thus have high β , high g_m , high h_{fe} & low R_c .

→ The presence of such efficient & advantageous devices on chip offers a great deal of scope and freedom to the VLSI designer.

Propagation Delay:-

Delay associated with Pass Transistors:-

→ The pass transistors can be used as parallel or series combinations of switches in logic arrays.

→ In such designs, logic signals are quite frequently required to pass through a number of pass transistors connected in series.

→ consider a chain of four pass transistors connected in series, the gate of each is connected to V_{DD} (logic-1) as shown in below figure.

→ since the gates are all connected to logic-1, the input signal at V_{in} propagates to the output.

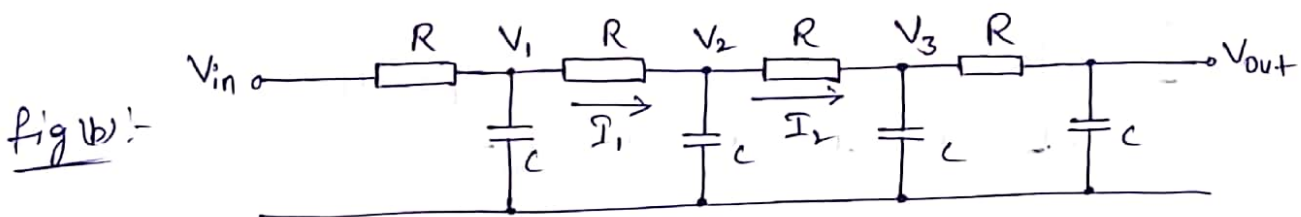
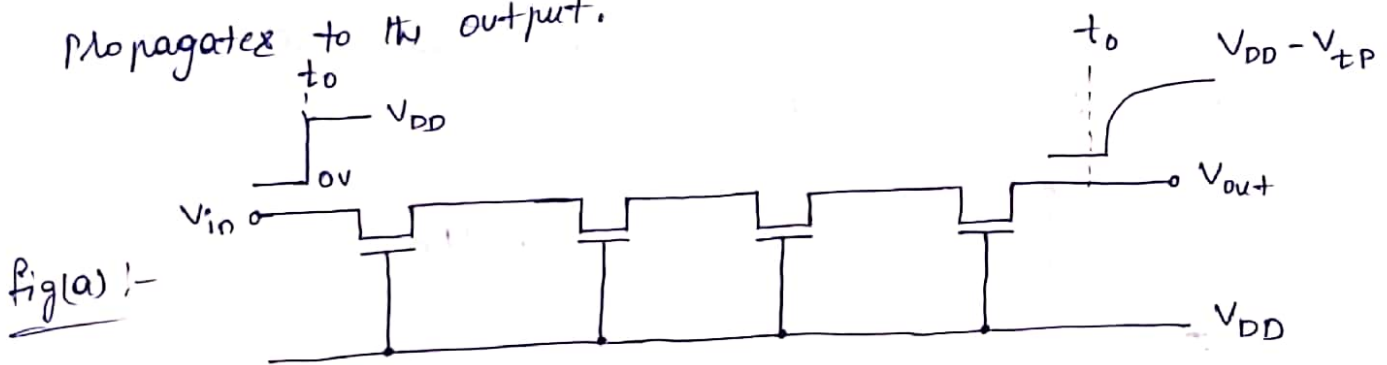


fig:- Propagation delay in pass transistor chain

→ In order to investigate the delay involved in propagation of the signal, we model this circuit by the one shown in above figure.

→ This network consists of those parameters associated with the pass transistors that contribute to the delay.

→ Writing KCL at node V_2 , we get

$$C \frac{dV_2}{dt} = I_1 - I_2 = \frac{(V_1 - V_2) - (V_2 - V_3)}{R}$$

→ Assume that there are a large number of pass transistors in series. Then, in the limit, equation reduces to

$$RC \frac{dV}{dt} = \frac{d^2V}{dx^2}$$

where

R = Resistance per unit length

C = capacitance per unit length

x = distance along network from input.

→ The propagation time taken by a signal to propagate a distance ' x ' is directly proportional to x^2 .

→ Let us define variables ' r ' & ' c ' such that $R = rR_s$ and $C = c \square C_g$ are lumped network elements of R & C . Then the total network elements can be given as

$$R_{total} = nrR_s$$

$$C_{total} = nc \square C_g$$

→ where ' r ' is the relative resistance per section interms of R_s & ' c ' is the relative capacitance per section interms of $\square C_g$.

→ Then the total delay T_d for n sections is given by

$$T_d = n^2 cr(\tau)$$

→ Thus as 'n' increases the total delay rapidly increases & it is recommended that not more than four pass transistors be connected in series.

→ If, due to some reasons, more pass transistors are required to be connected in series, a buffer is required to be inserted between each group of four pass transistors.

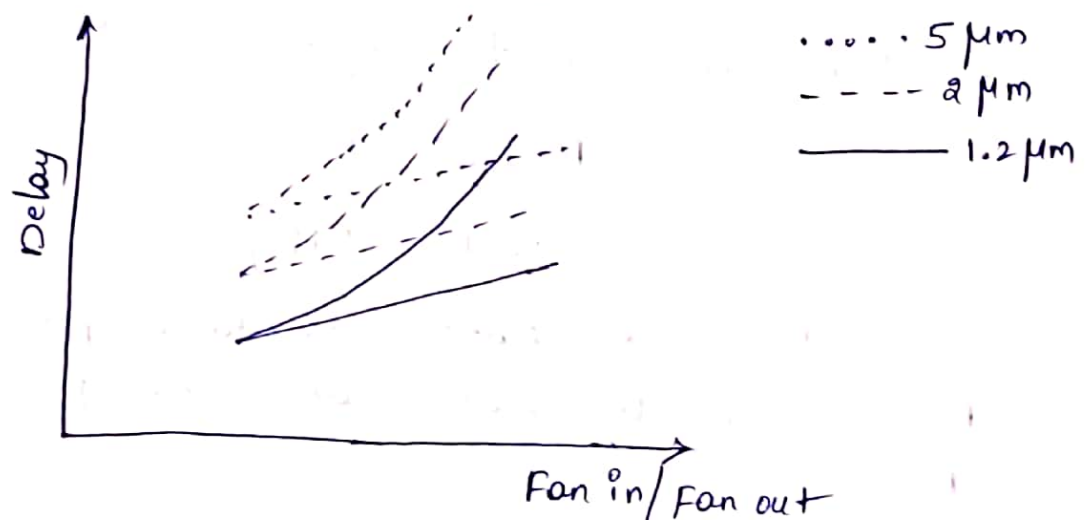
Fan-in and Fan-out characteristics :-

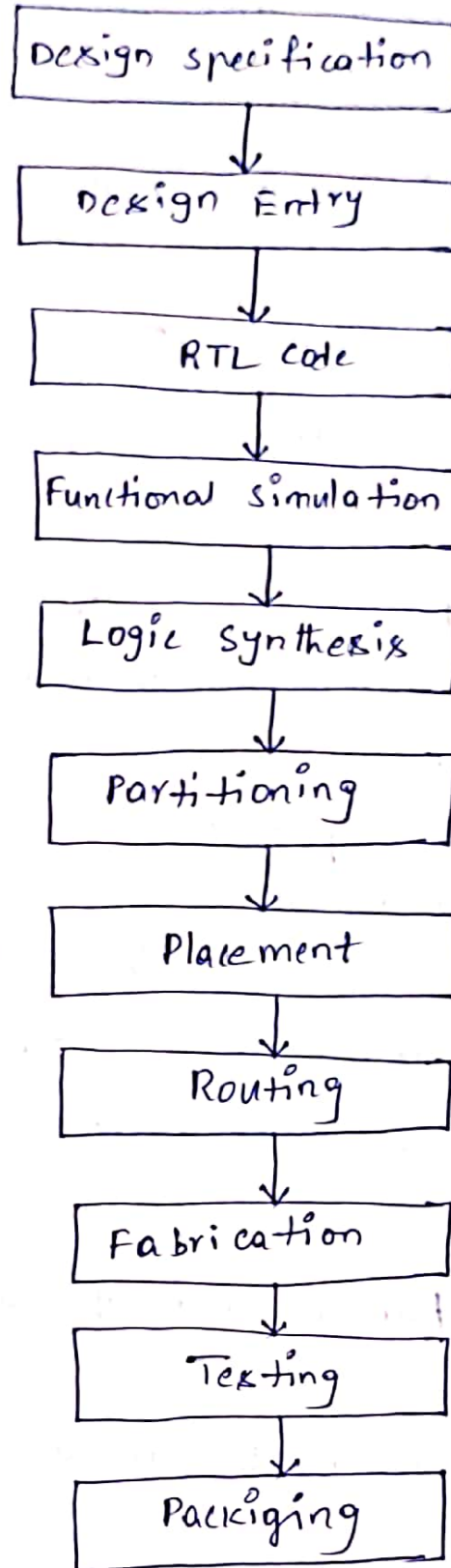
→ Two additional factors that influence the operational speed of gate are

1. Fan-in (Number of inputs)
2. Fan-out (Number of outputs)

→ The delay associated with fan-in & fan-out for three technologies is illustrated in below figure.

→ Number of inputs increases (fan-in) then delay increases linearly. Number of output increases (fan-out) then delay increases exponentially.



VLSI Design Flow:-

1. Design specification:- It gives complete information about design. Specification allows each engineer to understand the entire design. The complete design can be represented as a black box, which includes inputs, outputs and the relation between inputs & outputs.

2. Design Entry :- User can write a software code for the design using VHDL or Verilog HDL languages. It is also called as RTL code for the design.
3. Functional Simulation :- It is the process where logic in the design is checked before user implements it in a device. It is the functional verification of the design.
4. Logic Synthesis :- Here logic synthesis tool is used which produces netlist file of the design.
5. Partitioning :- Partitioning is the process of dividing a large and complex system into smaller modules.
6. Floor Planning :- The main function of the floorplanning is to estimate the required chip area that will be used for each standard cell module of the design.
7. Placement :- The main function of the placement is placing the modules at the selected area in chip.
8. Routing :- Routing is the interconnection between the sub-circuits (or) modules.
9. Fabrication :- Fabrication is the hardware implementation of the design.
10. Testing :- Testing is the verification of the functionality of the hardware chip.


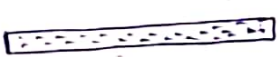



Stick diagrams :-

- Stick diagrams are used to convey layer information through the use of colour code.
- To describe logic networks, it is convenient to use stick diagrams which help to visualize the function as well as the

topology of the network.

- Always VLSI design aims to translate circuit concepts onto silicon.
- Stick diagrams are a means of capturing topography & layer information using simple diagrams.
- Stick diagrams convey layer information through colour codes (or monochrome encoding).
- Stick diagrams act as an interface between symbolic circuit and the actual layout.
- It shows relative placement of components
- Goes one step closer to the layout
- It helps to plan the layout and routing.
- It does not show the exact placement of components
- It does not show the transistor sizes, wire lengths, wire widths, tub boundaries and any other low level details such as parasitics.

Rules for stick encoding:-

- | | | | |
|--|---|---|---------------|
| 1. n-diffusion }
p-diffusion } | - |  | (empty stick) |
| 2. Poly silicon | - |  | (dots stick) |
| 3. Metal | - |  | (dark stick) |
| 4. Contact cut
(electrical contact) | - |  | (dark circle) |
| 5. Implant | - |  | (square box) |

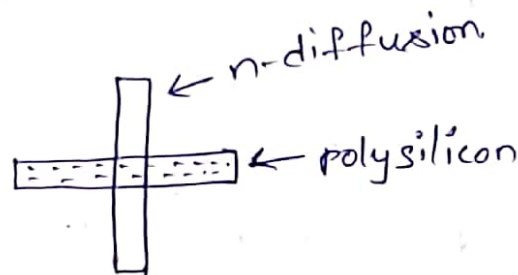
→ When two or more sticks of same type cross or touch each other, that represent electrical contact (i.e., electrical contact is there, we need not represent with dark circle).

→ When two or more sticks of different type cross or touch each other there is no electrical contact between those sticks.

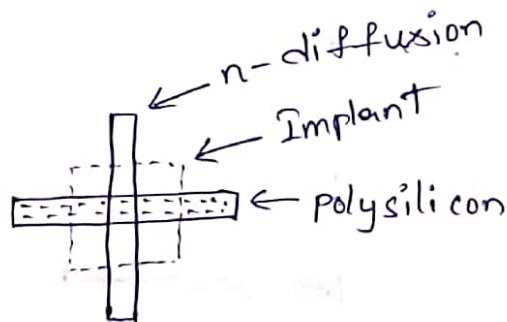
→ If electrical contact is needed we have to draw dark circle at the contact point of those sticks.

Transistor formation with sticks:-

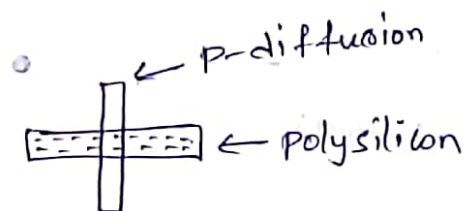
→ When polysilicon crosses the n-diffusion. Enhancement NMOS transistor is formed.



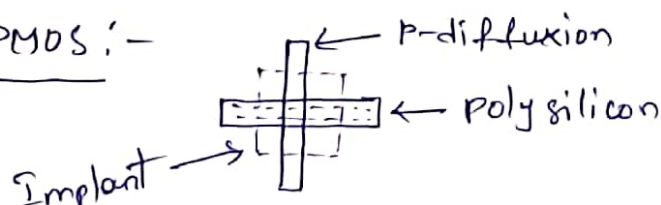
→ For depletion NMOS transistor, we have to draw Implant on the cross point of NMOS enhancement transistor.



Enhancement mode PMOS:-



Depletion mode PMOS:-

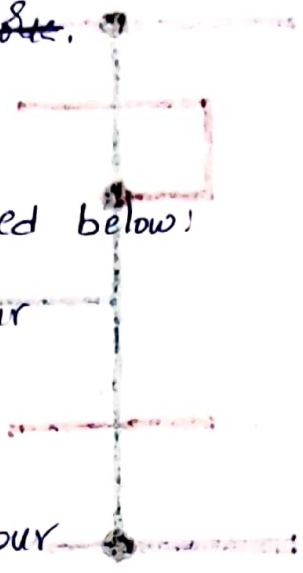


~~the topology of the network.~~

NMOS Design style:-

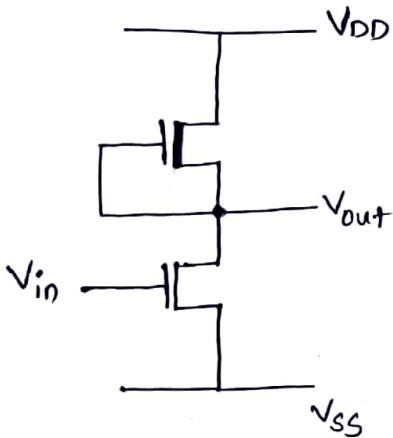
NMOS Design Layers are listed below:

1. N-diffusion - Green colour
2. Polysilicon - Red colour
3. Metal - Blue colour
4. Contact cut - Black colour
5. Implant - Yellow colour
6. Buried contact cut - Brown colour
7. when polysilicon cross over diffusion it forms MOS transistor.

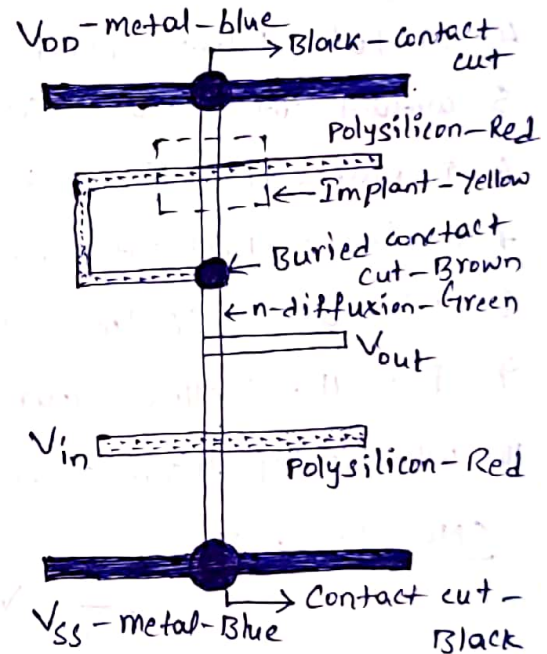


NMOS Inverter stick diagram:-

NMOS Inverter circuit



V_{in}	V_{out}
0	1
1	0



NMOS Inverter stick diagram

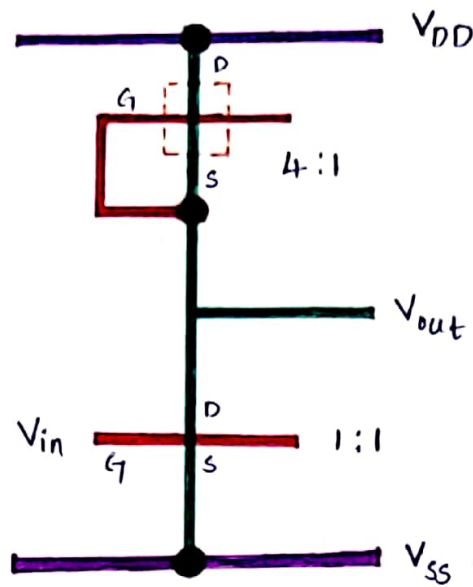


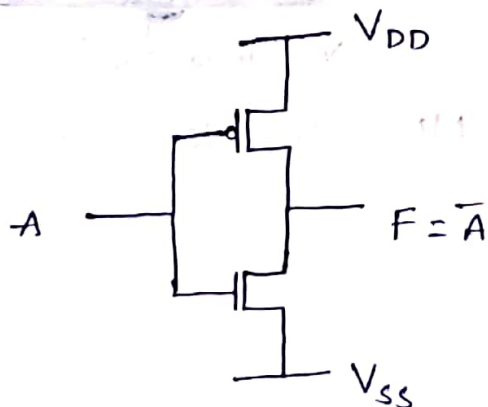
fig:- nMOS Inverter stick diagram

CMOS Design style:-

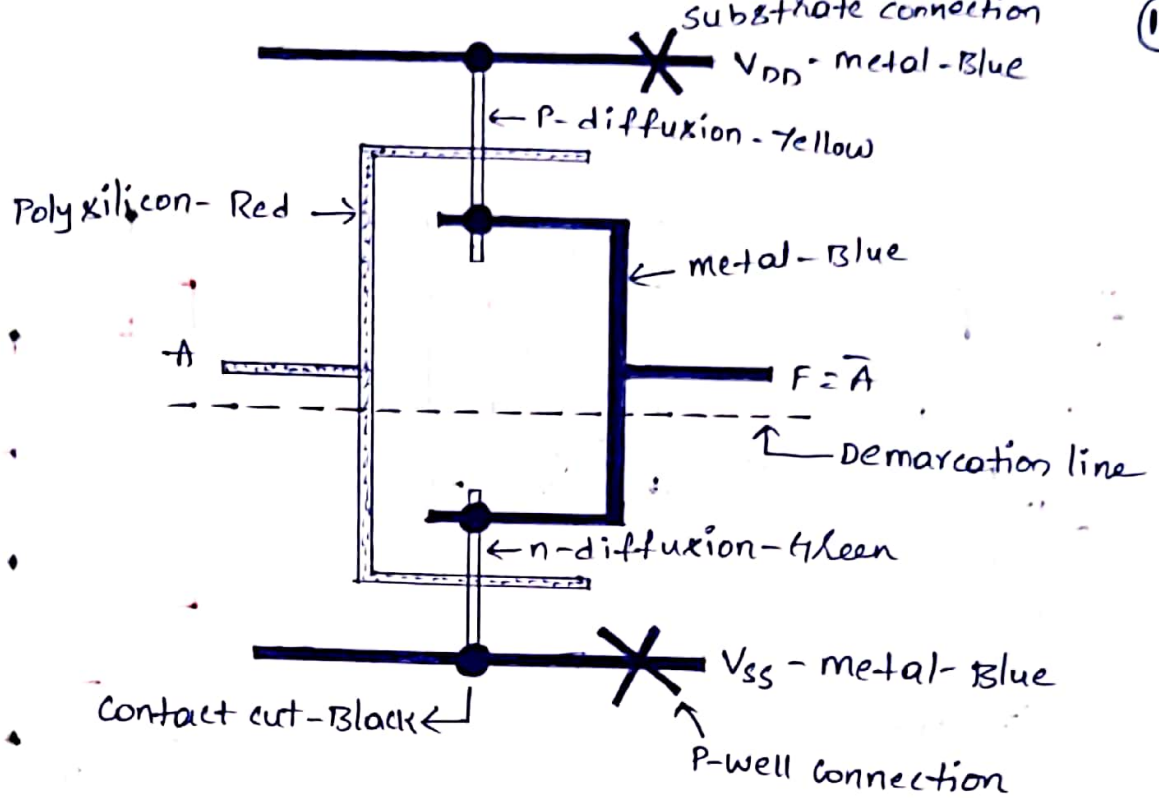
→ CMOS design layers are listed below!

1. N-diffusion - green colour
2. P-diffusion - yellow colour
3. Polysilicon - Red colour
4. Metal - Blue colour
5. contact cut - black colour
6. Demarcation line - brown colour
7. Buried contact cut - brown colour
8. P⁺-mask - yellow colour
9. P-well - yellow colour
10. when polysilicon cross over diffusion it forms MOS transistor.

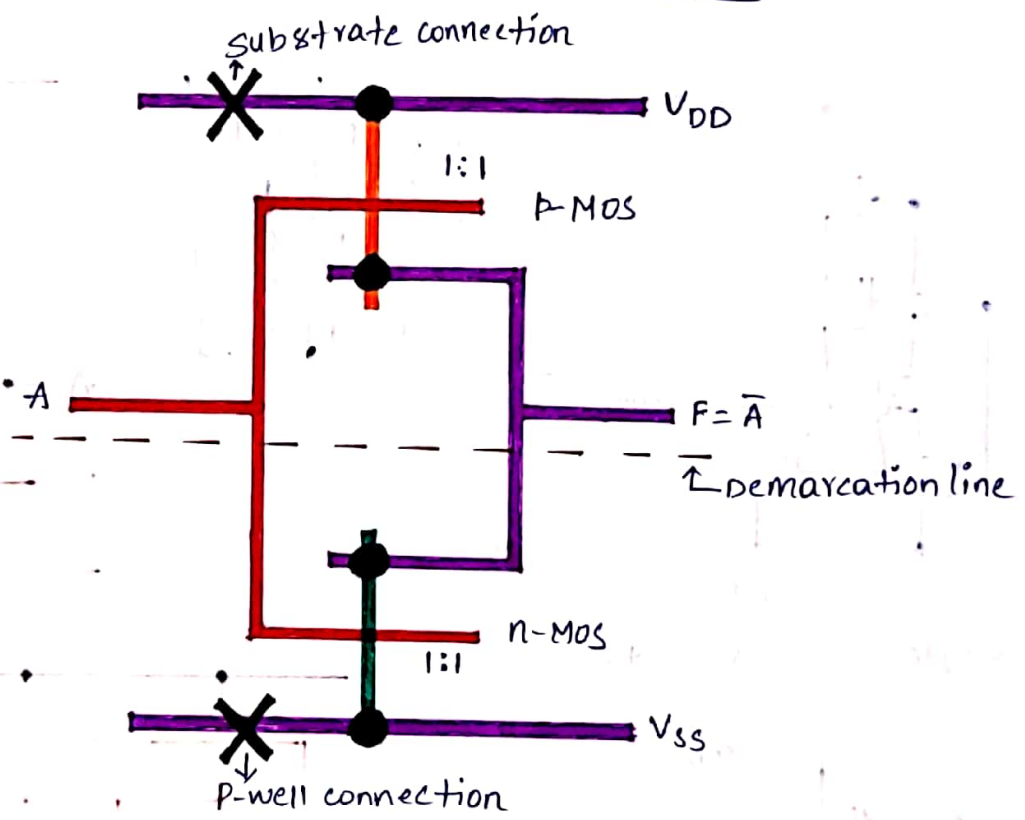
CMOS Inverter:-



A	$F = \bar{A}$
0	1
1	0



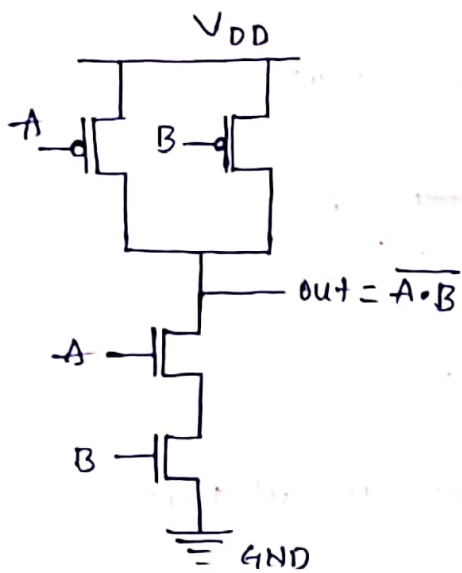
P-well CMOS Inverter stick diagram



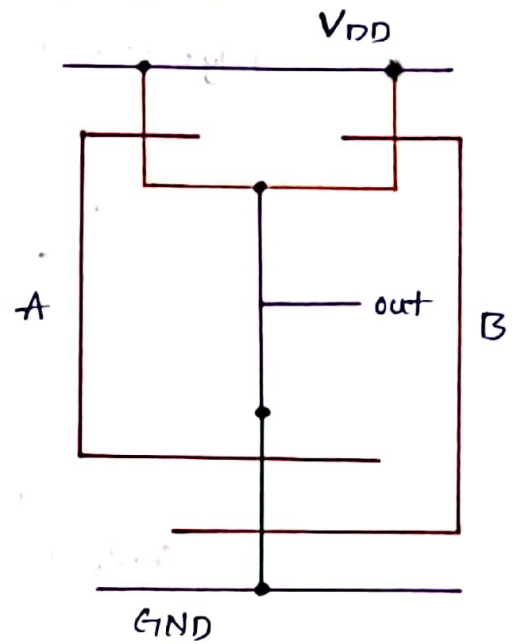
P-well CMOS Inverter stick diagram

Examples of stick diagrams :-

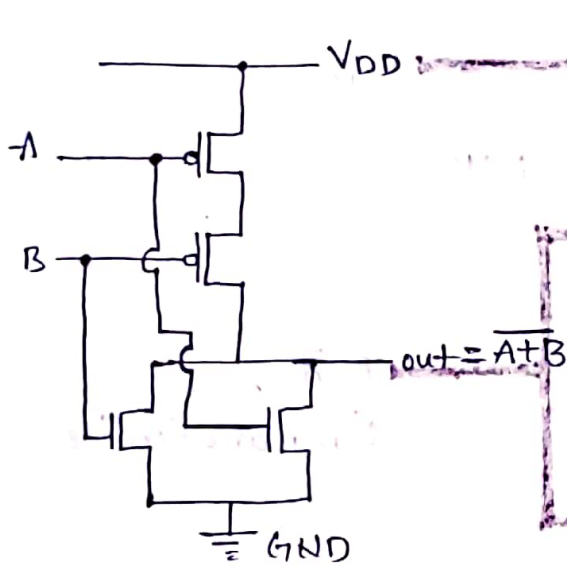
NAND gate can be realized using stick diagrams :-



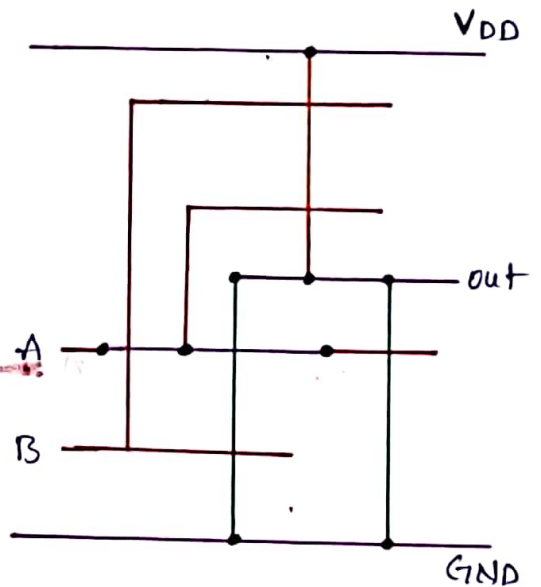
A	B	$\overline{A \cdot B}$
0	0	1
0	1	1
1	0	1
1	1	0



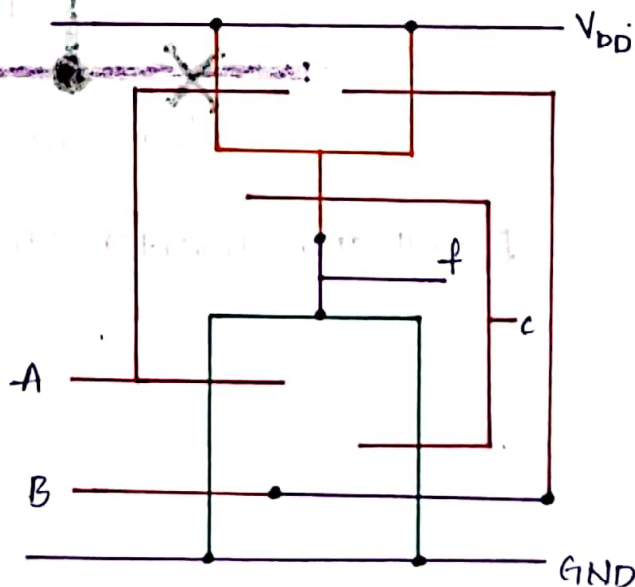
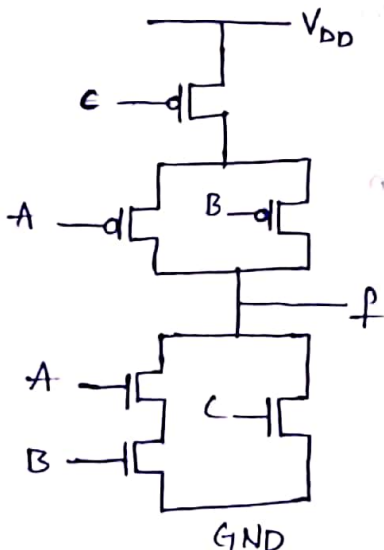
NOR gate can be realized using stick diagrams :-



A	B	$\overline{A + B}$
0	0	1
0	1	0
1	0	0
1	1	0



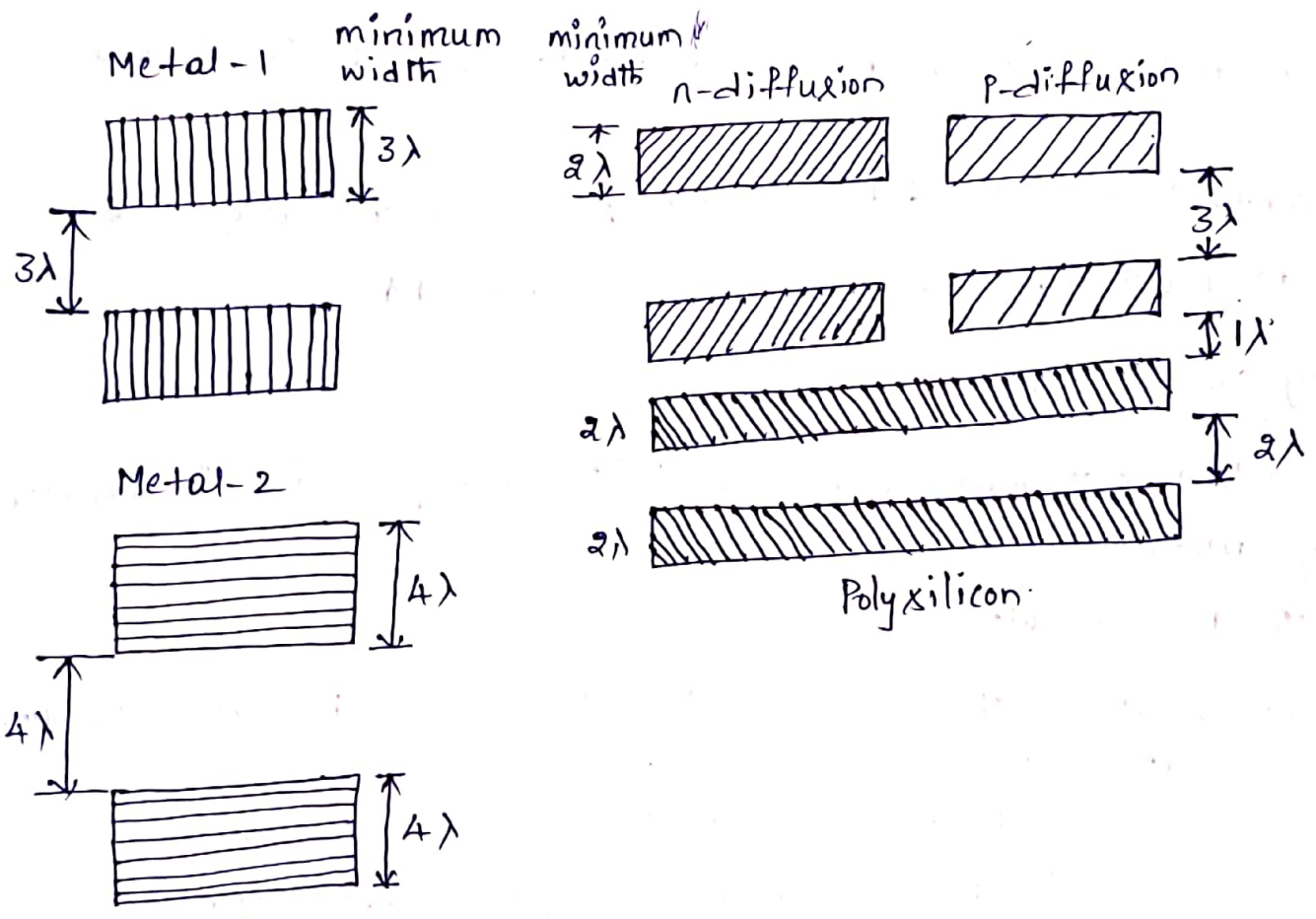
Example: $f = (A \cdot B) + C$



Lambda Based Design Rules :-

Rules for wires :-

1. The width of the polysilicon & diffusion is 2λ
2. The minimum separation between two diffusion layers is 3λ
3. The minimum separation between diffusion and polysilicon is 1λ
4. The minimum separation between two polysilicon layers is 2λ
5. The width of metal-1 layer is 3λ
6. The minimum separation between two metal-1 layers is 3λ
7. The width of metal-2 layer is 4λ
8. The minimum separation between two metal-2 layers is 4λ



Rules for Transistors :-

1. Polysilicon cross over diffusion creates transistor
2. The size of the implant is $6\lambda \times 6\lambda$ around the transistor
3. The extension of polysilicon from diffusion is minimum 2λ
4. The extension of diffusion from polysilicon is minimum 2λ

Rules for contact cuts:-

1. Contact cut dimensions should be $2\lambda \times 2\lambda$
2. Overlapping should be $4\lambda \times 4\lambda$. It should be surrounded the contact cut with 1λ in all the directions.
3. Contact cut between metal & polysilicon should be $2\lambda \times 2\lambda$ cut centered on $4\lambda \times 4\lambda$ superimposed areas of layers to be joined in all cases.
4. The minimum separation between one contact cut to other contact cut is 2λ .

Buried contact cut :-

1. This contact cut is used when polysilicon and diffusion were connected.
2. The contact cut should be extend upto 2λ in diffusion side. Remaining all sides the extension should be 1λ .

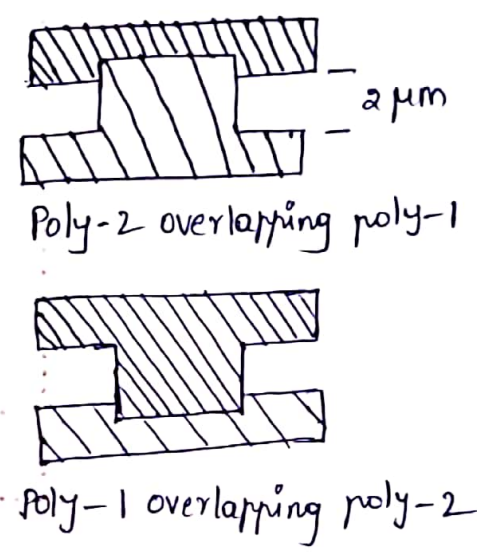
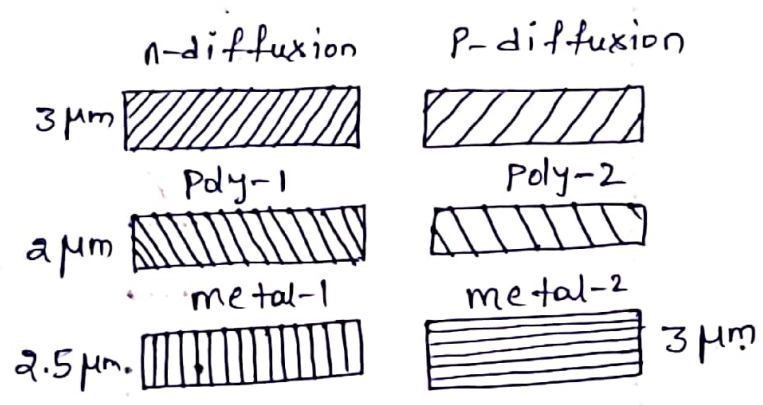
Butting Contact cut:-

1. This contact cut was used when diffusion & polysilicon is connected using metal.

2 μ m CMOS Design Rules:-

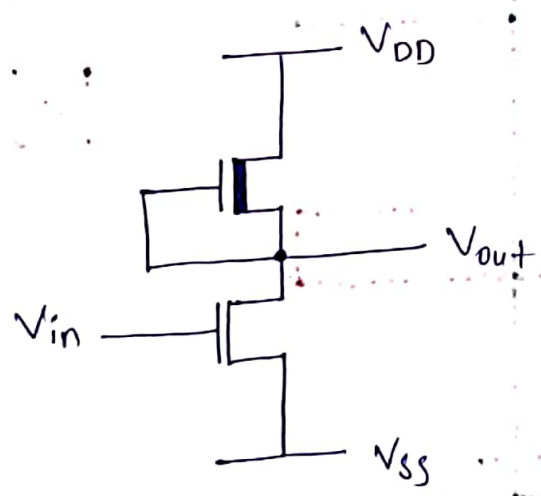
1. Minimum width of the diffusion (P-diffusion & n-diffusion) is $3\mu\text{m}$.
2. Minimum width of the polysilicon (polysilicon-1 & 2) is $2\mu\text{m}$.
3. Minimum separation between two diffusion layers is $2.5\mu\text{m}$
4. Minimum separation between diffusion & polysilicon-1 is $1\mu\text{m}$
5. Minimum separation between diffusion & polysilicon-2 is $1.5\mu\text{m}$.
6. Minimum separation between polysilicon-1 & polysilicon-2 is $2.5\mu\text{m}$.
7. Minimum separation between polysilicon-2 & polysilicon-2 is $3\mu\text{m}$.
8. Minimum separation between polysilicon-1 & polysilicon-2 is $2\mu\text{m}$.
9. Minimum width of the metal-1 layer is $2.5\mu\text{m}$.
10. Minimum width of the metal-2 layer is $3\mu\text{m}$.

- 11. Minimum separation between metal-1 & metal-1 is $2.5\mu\text{m}$.
- 12. Minimum separation between metal-2 & metal-2 is $2.5\mu\text{m}$.
- 13. Minimum extension of diffusion over polysilicon-1 is $2.5\mu\text{m}$.
- 14. Minimum extension of polysilicon-1 over diffusion is $1.5\mu\text{m}$.
- 15. Minimum extension of diffusion over polysilicon-2 is $2.5\mu\text{m}$.
- 16. Minimum extension of polysilicon-2 over diffusion is $2\mu\text{m}$.
- 17. Minimum overlapping of polysilicon-1 with polysilicon-2 is $1.5\mu\text{m}$.
un-overlapped area width also minimum $1.5\mu\text{m}$.



Layout Diagram:-

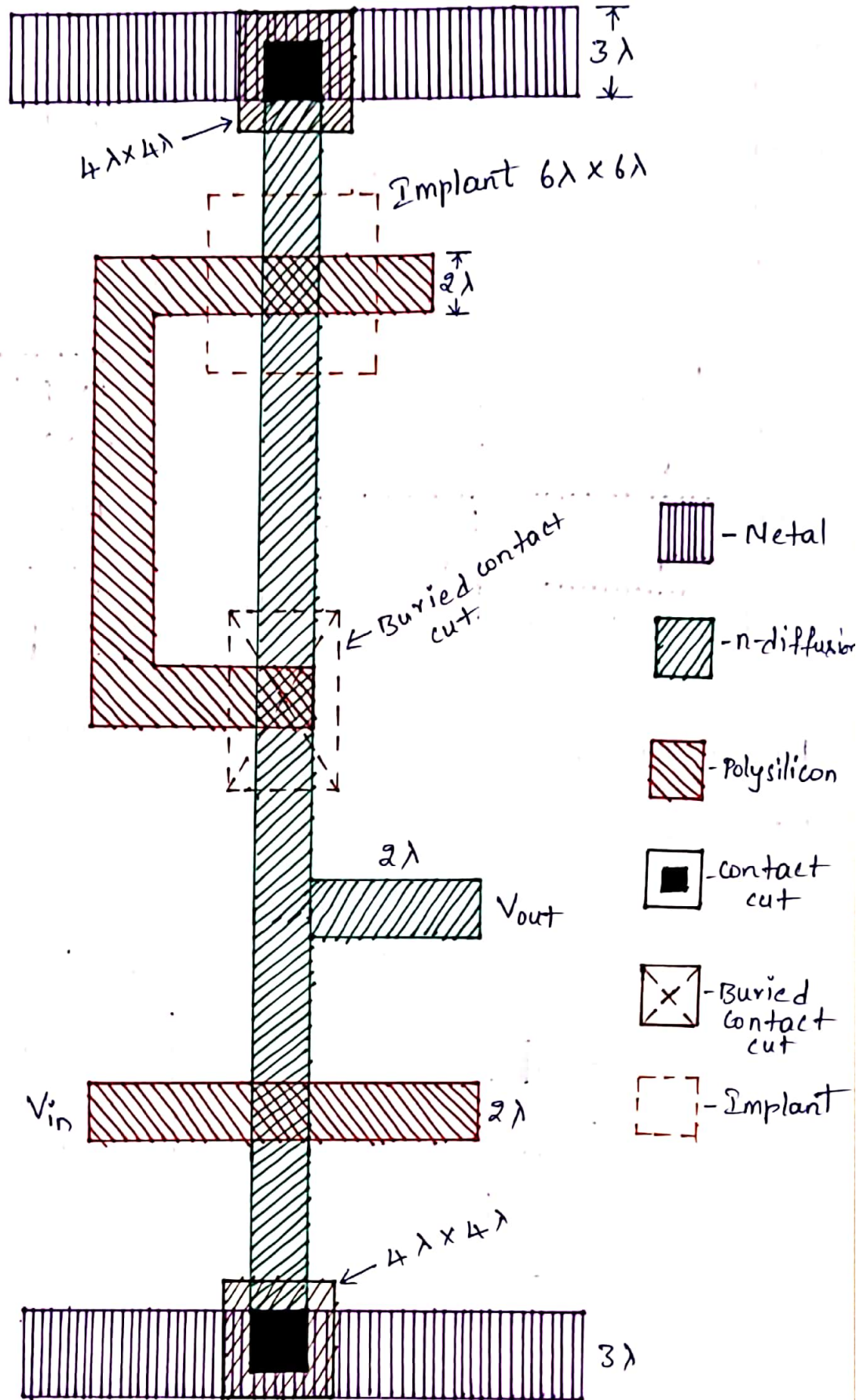
NMOS Inverter circuit



V_{in}	V_{out}
0	1
1	0

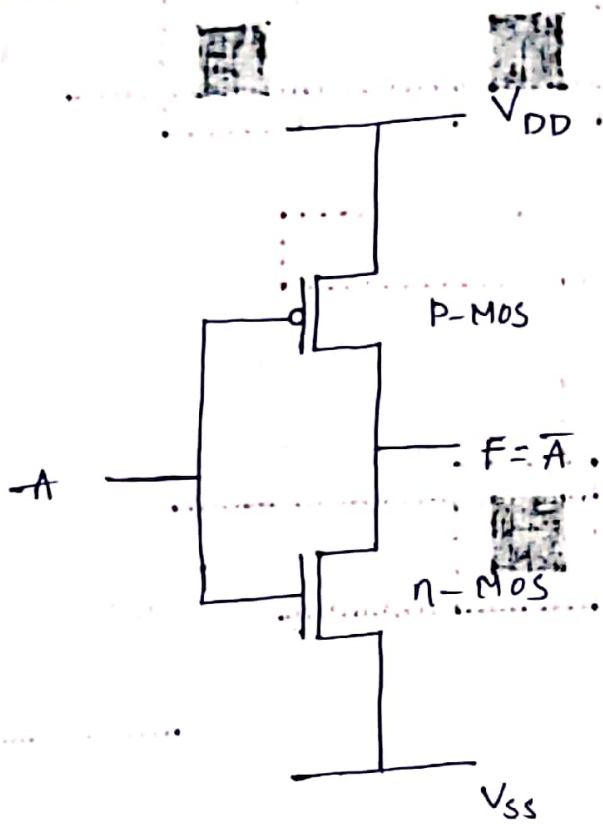
NMOS layout diagram :-

scale : $\frac{1}{2} \text{ cm} = 1\lambda$



CMOS Inverter

CMOS Inverter circuit :-

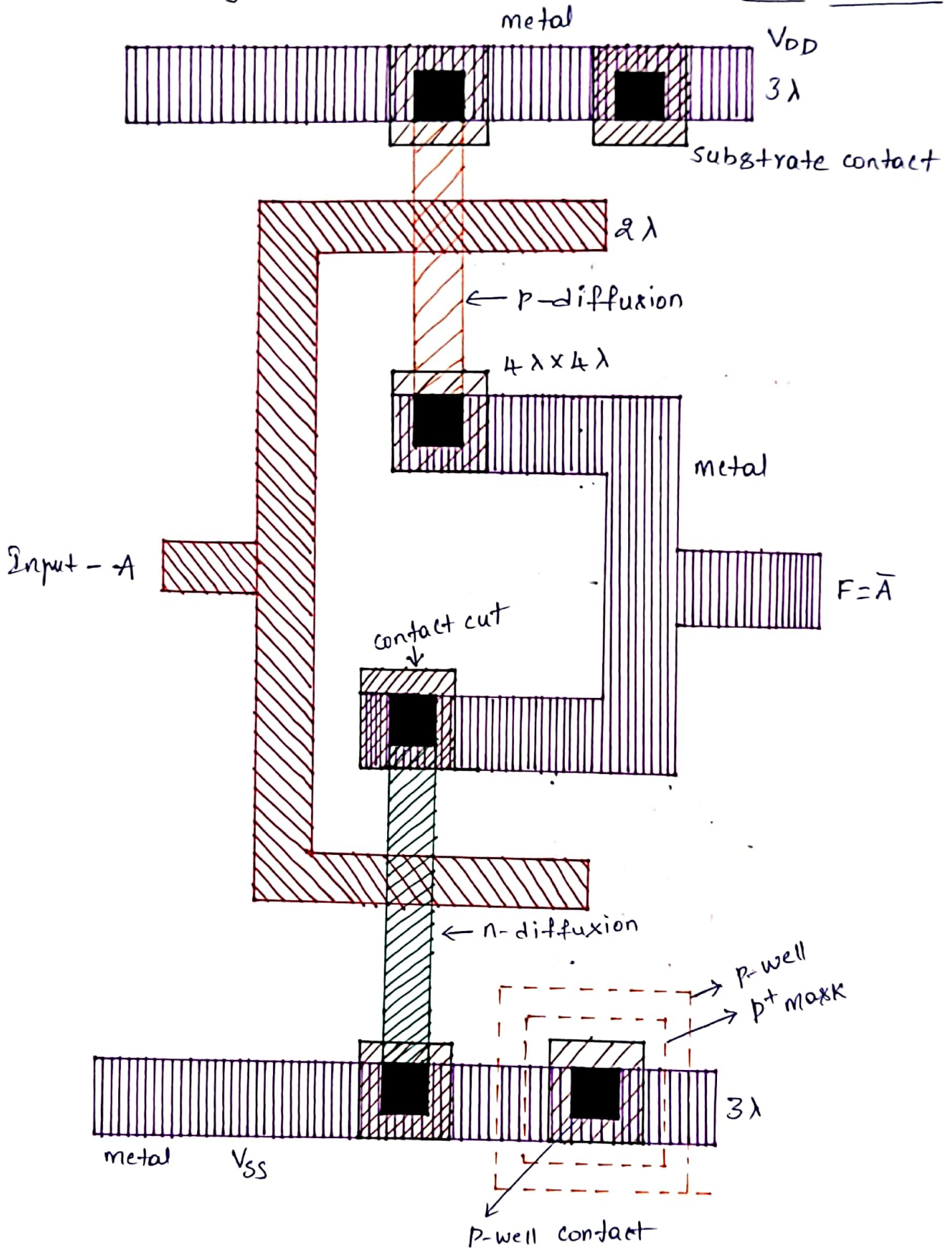


Truth table

A	$F = \bar{A}$
0	1
1	0

CMOS Inverter layout diagram:-

scale: $\frac{1}{2} \text{cm} = 1\lambda$



Scaling of MOS Transistors:-

- VLSI fabrication Technology is still in process of evolution which is leading to smaller line widths and feature size and to higher packing density of circuitry of chip.
- The scaling down of feature size generally leads to improved performance and it is important to understand the effects of scaling.
- Microelectronic technology may be characterized in terms of several indicators, or figure of merit:

The common figure of merits are:

- * Minimum feature size
- * Number of gates on one chip
- * Power dissipation
- * Maximum operational frequency
- * Die size
- * Production cost

→ Many of these figure of merit can be improved by shrinking or reducing the dimensions of transistors, interconnections and the separation between features, and by adjusting the doping levels & supply voltage.

→ so scaling is therefore an important factor, & it is essential for the designer to understand the implementation and the effects of scaling.

Scaling models and scaling factors:-

The most commonly used models are the constant electric field scaling model and constant voltage scaling model. Recently, a combined voltage and dimension scaling model has been presented.

The below figure indicates the device dimensions and doping levels

which are associated with the scaling of a transistor.

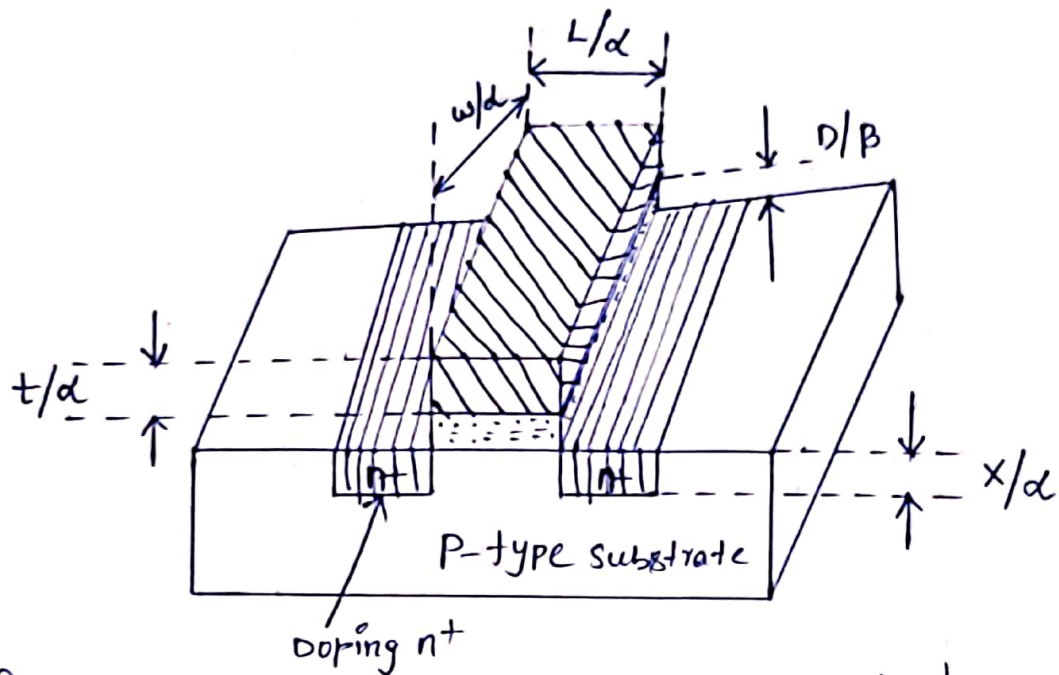


Fig:- Scaled nMOS transistor (PMOS similar)

→ In order to explain the three models, two scaling factors $1/\alpha$ & $1/\beta$ are used. $1/\beta$ is the scaling factor for supply voltage V_{DD} and gate oxide thickness D , and $1/\alpha$ is used for all other linear dimensions, both vertical and horizontal to the chip surface. For the constant field model $\beta = \alpha$ and for constant voltage model $\beta = 1$ is applied.

Scaling factors for device parameters:-

In this section, simple derivations and calculations reveal the effects of scaling.

Gate area (A_g):-

$$A_g = L \cdot W$$

where L & w are the channel length and width. Both are scaled by $1/\alpha$. Thus A_g is scaled by $1/\alpha^2$.

Gate capacitance per unit Area C_0 (or) C_{ox} :-

$$C_0 = \frac{\epsilon_{ox}}{D}$$

where ϵ_{ox} is the permittivity of the gate oxide ($= \epsilon_{ins} \cdot \epsilon_0$) & D is the gate oxide thickness which is scaled by $1/\beta$

Thus C_0 is scaled by $\frac{1}{1/\beta} = \beta$

Gate capacitance (C_g) :-

$$C_g = C_0 \cdot W \cdot L$$

Thus C_g is scaled by $\beta \cdot \frac{1}{\alpha^2} = \beta/\alpha^2$

Parasitic capacitance (C_x) :-

C_x is proportional to $\frac{A_x}{d}$

where 'd' is the depletion width around source and drain which is scaled by $1/\alpha$, and A_x is the area of the depletion region around source or drain which is scaled by $1/\alpha^2$

Thus C_x is scaled by $\frac{1/\alpha^2}{1/\alpha} = 1/\alpha$

Carrier density in channel (Q_{on}) :-

$$Q_{on} = C_0 \cdot V_{gs}$$

where Q_{on} is the average charge per unit area in the channel in the 'on' state. C_0 is scaled by β & V_{gs} is scaled by $1/\beta$,

Thus Q_{on} is scaled by $\beta \cdot \frac{1}{\beta} = 1$

channel Resistance (R_{on}) :-

$$R_{on} = \frac{L}{W} \cdot \frac{1}{Q_{on} \mu}$$

where μ is the carrier mobility in the channel and is assumed constant.

$$\text{Thus } R_{on} \text{ is scaled by } \frac{\frac{1}{\alpha}}{\frac{1}{\alpha}} \cdot \frac{1}{1} = 1$$

Gate delay (T_d):-

T_d is proportional to $R_{on} \cdot C_g$

$$\text{Thus } T_d \text{ is scaled by } 1 \cdot \beta/\alpha^2 = \beta/\alpha^2$$

Maximum operating frequency (f_0):-

$$f_0 = \frac{\omega}{L} \cdot \frac{M C_0 V_{DD}}{C_g}$$

or, f_0 is inversely proportional to delay T_d .

$$\text{Thus } f_0 \text{ is scaled by } \frac{1}{\beta/\alpha^2} = \alpha^2/\beta$$

Saturation current (I_{dss}):-

$$I_{dss} = \frac{C_0 M}{2} \cdot \frac{\omega}{L} (V_{gs} - V_t)^2$$

Both V_{gs} and V_t are scaled by $1/\beta$, we have

$$I_{dss} \text{ is scaled by } \beta (1/\beta)^2 = 1/\beta$$

current density (J):-

$$J = \frac{I_{dss}}{A}$$

where 'A' is the cross-sectional area of the channel in the 'on' state which is scaled by $1/\alpha^2$

$$\text{Thus } J \text{ is scaled by } \frac{1/\beta}{1/\alpha^2} = \alpha^2/\beta.$$

switching energy per gate (Eg):-

$$E_g = \frac{1}{2} C_g (V_{DD})^2$$

Thus Eg is scaled by $\frac{\beta}{\alpha^2} \cdot \frac{1}{\beta^2} = \frac{1}{\alpha^2 \beta}$

Power dissipation per Gate (Pg):-

Pg consists of two components such that

$$P_g = P_{gs} + P_{gd}$$

Static component $P_{gs} = \frac{(V_{DD})^2}{R_{on}}$

Dynamic component $P_{gd} = E_g \cdot f_0$

P_{gs} is scaled by $\frac{(\frac{1}{\beta})^2}{1} = \frac{1}{\beta^2}$

P_{gd} is scaled by $\frac{1}{\alpha^2 \beta} \cdot \frac{\alpha^2}{\beta} = \frac{1}{\beta^2}$

$\therefore P_g$ is scaled by $\frac{1}{\beta^2} + \frac{1}{\beta^2} = \frac{2}{\beta^2} = \frac{1}{\beta^2}$

Power dissipation per unit area (Pa):-

$$P_a = \frac{P_g}{A_g}$$

Thus Pa is scaled by $\frac{\frac{1}{\beta^2}}{\frac{1}{\alpha^2}} = \frac{\alpha^2}{\beta^2}$

Power speed Product (PT):-

$$P_T = P_g \cdot T_d$$

Thus PT is scaled by $\frac{1}{\beta^2} \cdot \frac{\beta}{\alpha^2} = \frac{1}{\alpha^2 \beta}$

Limitations of scaling :-

Although scaling down have many desirable effects, some of the effects may cause problems which become severe to prevent further miniaturization.

LECTURE NOTES

3.1. Regions of operations of MOSFET

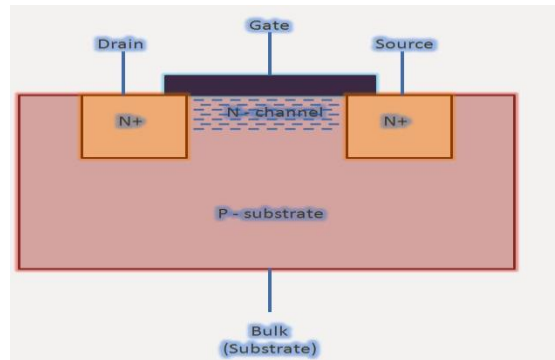


Figure 3.1: MOS transistor

In a MOS device, the current flows on formation of channel of carriers between source and drain terminals. For this, voltage at gate terminal needs to be such that it attracts carriers of appropriate type towards itself. When sufficient carriers are attracted towards gate, channel is said to be formed. A current, then, flows between source and drain terminals depending upon the voltage levels of these terminals. The voltage level of substrate also impacts the magnitude of current as it also determines the level of carriers in the channel.

For an N-MOS device, the channel is formed by electrons. So, to attract electrons, gate voltage must be greater than source voltage. For the formation of channel, the difference between V_G and V_S ($V_G - V_S$) must be greater than V_{th} (threshold voltage of the MOS).

Threshold voltage is defined as the minimum difference in gate-to-source voltage needed for the formation of channel in a MOS device. For NMOS, V_{th} is positive as for channel formation gate needs to be at higher voltage as explained above. Similarly, for PMOS, V_{th} is negative as gate needs to be at lower voltage than source for channel to be formed.

On increasing gate voltage beyond threshold voltage, current through MOS increases with increasing gate voltage. Also, if we increase drain voltage keeping gate voltage constant, current increases till a particular drain voltage. After voltage does not affect the current. Depending upon the relative voltages of its terminals, MOS is said to operate in either of the cut-off, linear or saturation region.

- **Cut off region** – A MOS device is said to be operating when the gate-to-source voltage is less than V_{th} . Thus, for MOS to be in cut-off region, the necessary condition is –
 $0 < V_{GS} < V_{th}$ - for NMOS
 $0 > V_{GS} > V_{th}$ - for PMOS (as threshold voltage of PMOS is negative)

Cut-off region is also known as sub-threshold region. In this region, the dependence of current on gate voltage is exponential. The magnitude of current flowing through MOS in cut-off region is negligible as the channel is not present. The conduction happening in this region is known as sub-threshold conduction.

- **Linear or non saturation region** – For an NMOS, as gate voltage increases beyond threshold voltage, channel is formed between source and drain terminals. Now, if there is voltage difference between source and drain, current will flow. The magnitude of current increases linearly with increasing drain voltage till a particular drain voltage determined by the following relations –

$$V_{GS} \geq V_{th}$$

$$V_{DS} < V_{GS} - V_{th}$$

The current is, then, represented as a linear function of gate-to-source and drain-to-source voltages. That is why, MOS is said to be operating in linear region. The linear region voltage-current relation is given as follows:

$$I_d(\text{Linear}) = \mu C_{ox} W/L (V_{gs} - V_{th} - V_{ds}/2) V_{ds}.$$

Similarly, for P-MOS transistor, condition for P-MOS to be in linear region is represented as:

$$V_{GS} < V_{th}$$

OR

$$V_{SG} > |V_{th}|$$

And $V_{DS} > V_{GS} + V_{th}$

OR

$$V_{SD} < V_{SG} - |V_{th}|$$

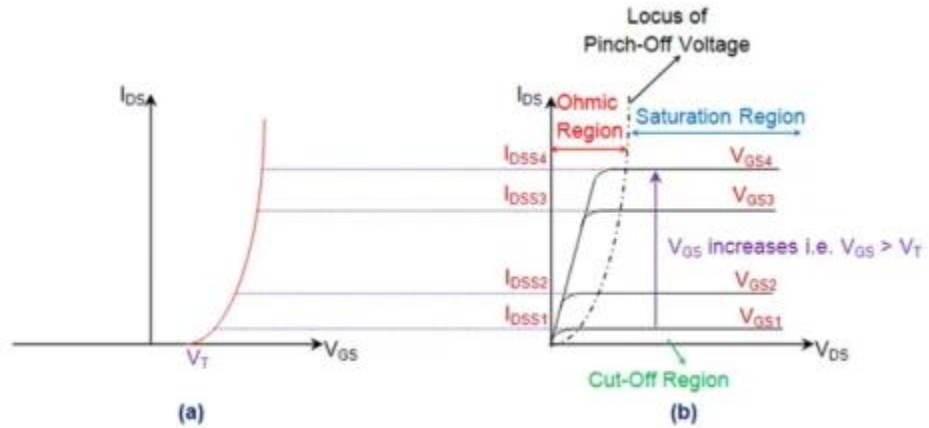
- **Saturation Region** – For an NMOS, at a particular gate and source voltage, there is a particular level of voltage for drain, beyond which, increasing drain voltage seems to have no effect on current. When a MOS operates in this region, it is said to be in saturation. The condition is given as:

- $V_{GS} \geq V_{th}$

- $V_{DS} > V_{GS} - V_{th}$

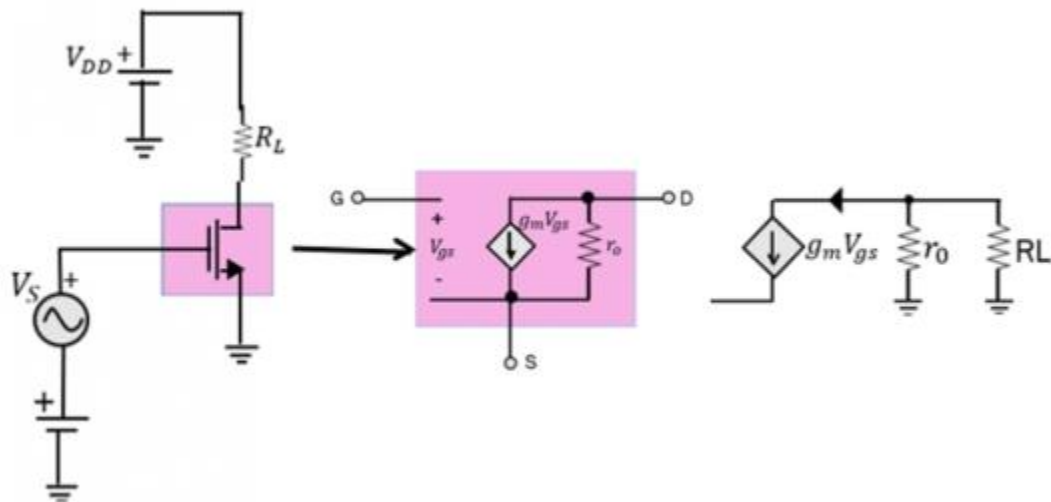
- The current, now, is a function only of gate and source voltages:

- $I_d(\text{saturation}) = \mu C_{ox} W/L (V_{gs} - V_{th})^2$



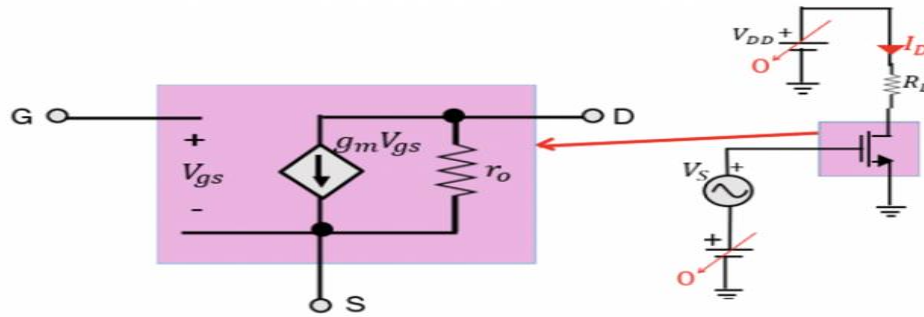
• **Figure 3.2: MOS Transistor Characteristics**

• **3.2 Modeling of Transistor**



• **Figure 3.3: MOS Transistor Model**

• In this circuit, the V_{gs} is the input signal applied between gate and source terminal, and we know that the change in drain current is linearly proportional to V_{gs} . In this model, if you consider the effect of channel and modulation, then there will also be an output resistance (r_o). If it is for a long length channel, then, as read in the [Early Voltage section in the MOS transistor](#) for a long length channel, the curve slope is almost constant in the saturation region, λ is very low, sometimes considered 0. Therefore, under the small-signal approximation, the MOS transistor can be replaced by the small-signal model.



• **Figure 3.4: Small signal Model of MOS Transistor Model**

- In the small-signal model, there is an output resistance r_0 and the current source is $g_m V_{gs}$, so if we can find the Transconductance (g_m), we can find the value of current in this circuit. Output resistance r_0 is the fluctuation of drain-source voltage to current. For the long channel transistor, the V_A (early voltage) is high, and as per the equation, r_0 is directly proportional to length; therefore, r_0 is high for the long channel transistor. Here, I_D is the current bias.

$$r_0 = \frac{\partial V_{ds}}{\partial I_{ds}} = \frac{V_A}{I_D}$$

While doing the AC analysis, all the DC voltage in the circuit is assumed 0 and vice versa. The analysis for DC and AC is done separately. When we talk about gain, it is AC gain, and DC is just to set the operation point. So as you see, when the small circuit model is replaced for the transistor, we get the below circuit arrangement:

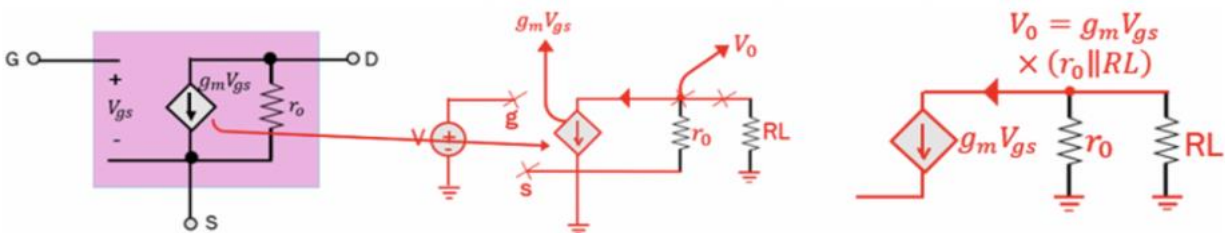


Figure 3.5: Small signal Model of MOS Transistor Model

AC analysis for the given circuit considering DC is 0 & V_{DD} is grounded. V_S in a small signal model is placed between gate and source terminal. When input signal V_S is very low, the MOS transistor can be replaced by the small-signal model. The flow of current is clockwise and is $g_m V_{GS}$, and V_0 is connected to

load resistance R_L . R_0 and R_L are in a parallel arrangement. Therefore, gain here will be $g_m V_{GS} \cdot (R_L || r_0)$ and this value is more than 1, and this shows that the output voltage will be amplified.

The transistor in this model works as an amplifier. If V_S is 1mV AC and good gain, then output from point V_0 can be up to 10mV. The work is mostly in the saturation region due to the reason of having high output resistance. The small-signal model of the MOS transistor is useful as an amplifier. It is easy to analyze the circuits using small-signal models.

In summary, so far, we have read that using the MOS Transistor as an amplifier should be operated in the saturation region. In this region, the transistor acts as a voltage-controlled current source, and the drain current is a function of V_{GS} . The relationship between the V_{GS} and drain current is non-linear.

For a set V_{GS} , the amount of change in the drain current is dependent on the bias point, and it is defined as the Transconductance of the MOS transistor. If the biasing point changes, then for the same change in the V_{GS} , there will be more change in the drain current I_D . As we read [from the small signal analysis](#) the Transconductance is equal to change in the drain current to change in V_{GS} . For a given change in V_{GS} , the difference in the drain current graphically is the slope of the curve at the operating point.

$$\text{Gain} = \frac{V_0}{V_S} = g_m \cdot (R_L || r_0) > 1$$

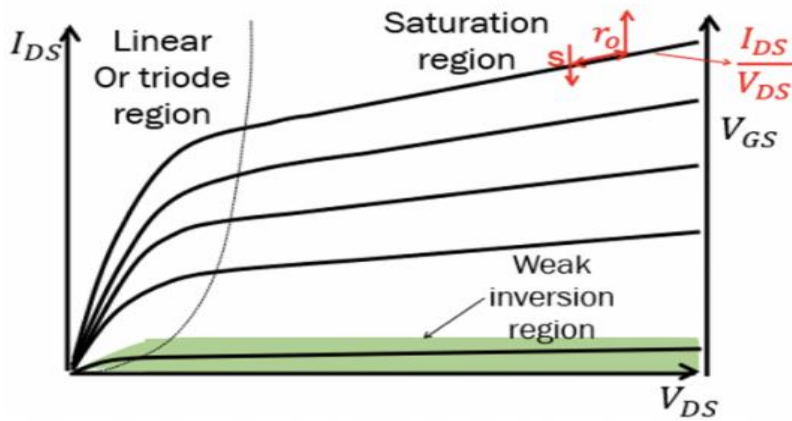


Figure 3.6: Characteristics of MOS transistor

In the graph, you see that the slope of the curve is very low, it is equal to I_{DS}/V_{DS} , which is the inverse of r_o from equation (1). Hence, when the slope is high, r_o is low, and vice versa. In the saturation region, the output resistance of the transistor is increased as the slope is low, and resistance is much higher for a long length channel. In linear regions, the slope is more, and hence the output resistance is less. Mostly the work done in the transistor is in the saturation region as we have higher output resistance.

3.3 Body Bias Effect

Body effect refers to the change in the threshold voltage of the device when there is a difference between substrate (body) and source voltages. Body bias is usually the lowest voltage in the chip (in case of p-substrate).

Let's look at the NMOS given below.

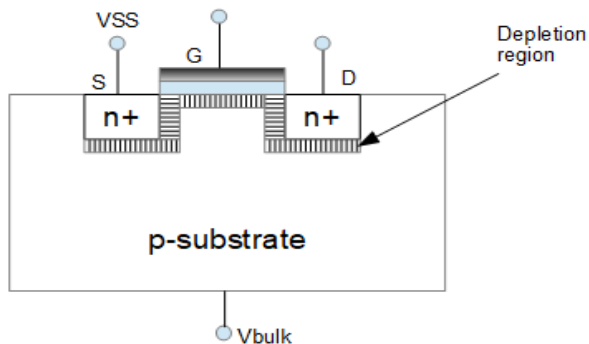


Figure 3.7: MOS transistor

Here source (V_s) is connected to VSS and the substrate is biased with voltage V_{bulk} . There are two pn-junctions formed due to drain-substrate and source-substrate. There will form a depletion region at these junctions.

However, if we were to connect V_{bulk} to a voltage lower than VSS (Source voltage), there is an increased flow of carriers between these source-bulk junction thereby increasing the width of the depletion region. This in turn increases the minimum gate voltage needed to achieve channel inversion.

3.4 Biasing Styles

MOSFET Bias Circuits

E-MOSFET Bias Configuration

- Since for E-MOSFET the value of V_{GS} should be larger than the threshold value $V_{GS(th)}$, so zero biasing cannot be used.
- In below figure you can see 2 configurations of E-MOSFET biasing.

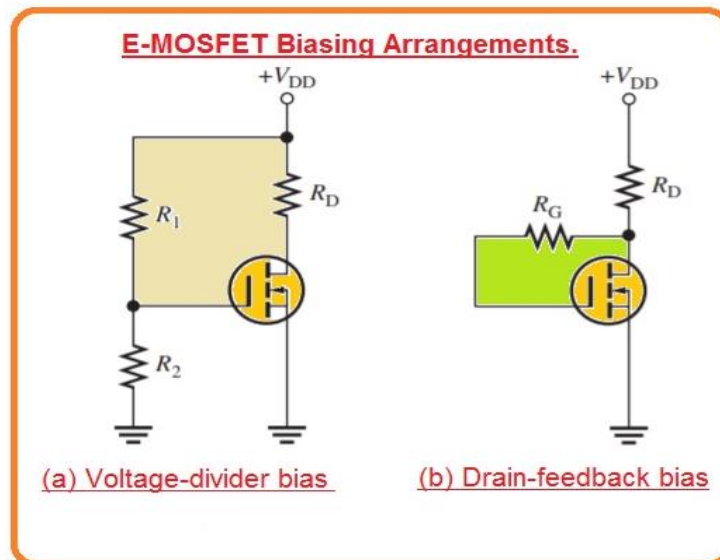


Figure 3.8: E MOSFET Biasing Arrangements

- In these configurations n channels, MOSFET is used.

- Two biasing configurations are shown first is voltage divider and second one drain feedback bias.
- The benefits of the use of this 2 configuration is to make more positive at the gate than by the source through amount $V_{GS(th)}$.
- The equation for voltage divider bias configuration is given here.

$$V_{GS} = [R_2 / (R_1 + R_2)] V_{DD}$$

$$V_{DS} = V_{DD} - I_{DRD}$$

- In this equation I_D will be equal to $K(V_{GS} - V_{GS(th)})^2$ we have discussed this expression in last tutorial about MOSFET.
- In drain feedback bias circuitry shown in figure the gate current is very less and so there will be no voltage drop about the resistance R_G . So $V_{GS} = V_{DS}$.

D-MOSFET Bias Configuration

- As we know that D-MOSFET can operate with both positive and negative values of V_{GS} voltage.
- The basic method of biasing is to make $V_{GS} = 0$ so ac voltage at gate changes the gate to source voltage over this zero voltage biasing point.
- Zero bias configuration for MOSFET is shown in below figure.
- As V_{GS} is zero and $I_D = I_{DSS}$ as denoted. The drain to source voltage will be.

$$V_{DS} = V_{DD} - I_{DSS} R_D$$

- The main function of R_G is to sustain an ac voltage input by separating it from ground as shown in figure denoted as b.

- As there is no dc gate current there will be no effect of R_G on zero gate to source bias.

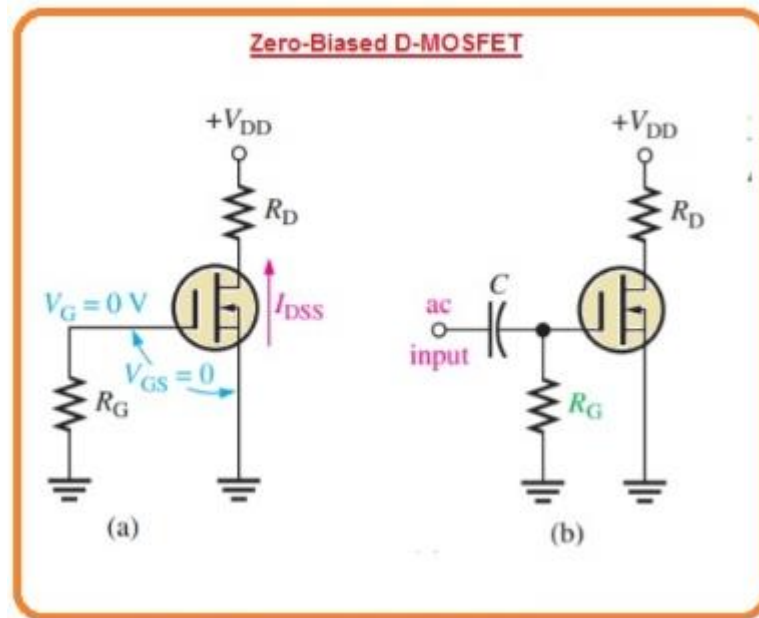
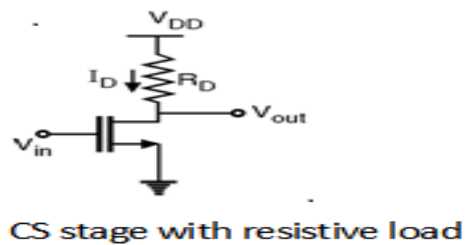


Figure 3.9: Zero Biased D MOSFET

3.5 Common Source Amplifier

Figure below shows the common source amplifier circuit. In this circuit the MOSFET converts variations in the gate-source voltage into a small signal drain current which passes through a resistive load and generates the amplified voltage across the load resistor.



Now from above Figure,

$$I_D = \frac{1}{2} \mu_n C_{ox} \frac{W}{L} (V_{GS} - V_{TH})^2$$

$$\text{i.e. } I_D = \frac{1}{2} \mu_n C_{ox} \frac{W}{L} (V_{in} - V_{TH})^2$$

But by KVL,

$$V_{DD} - I_D R_D = V_{out}$$

$$\therefore V_{out} = V_{DD} - \frac{1}{2} \mu_n C_{ox} \frac{W}{L} (V_{in} - V_{TH})^2 R_D$$

Differentiating this equation with respect to V_{in}

$$\frac{dV_{out}}{dV_{in}} = -\mu_n C_{ox} \frac{W}{L} (V_{in} - V_{TH}) R_D$$

Hence, The voltage gain $A_v = -g_m R_D$ $\left[\because g_m = \mu_n C_{ox} \frac{W}{L} (V_{in} - V_{TH}) \right]$

$$V_{in} - V_{GS} = 0$$

$$\therefore V_{in} = V_{GS}$$

Also

$$V_{out} + g_m V_{GS} R_D = 0$$

$$\therefore V_{out} = -g_m V_{GS} R_D$$

$$\text{or } V_{out} = -g_m V_{in} R_D$$

Hence, The voltage gain $A_v = \frac{V_{out}}{V_{in}} = -g_m R_D$

As the gate terminal of MOSFET draws a zero current we can say that the common source amplifier provides a current gain of infinity.

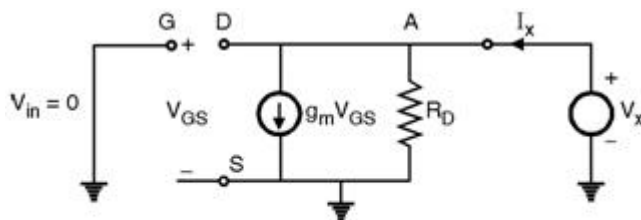
$$\therefore A_i = \infty$$

Also, because of zero gate current the input impedance of CS amplifier is also infinite.

$$\therefore R_{in} = \infty$$

In order to calculate the output impedance R_{out} consider the circuit shown in

Figure below.



Output impedance of CS

By applying KCL at point 'A'

We get

$$g_m V_{GS} + \frac{V_x - 0}{R_D} = I_x$$

But $V_{GS} = 0$,

$$\therefore \frac{V_x}{R_D} = I_x$$

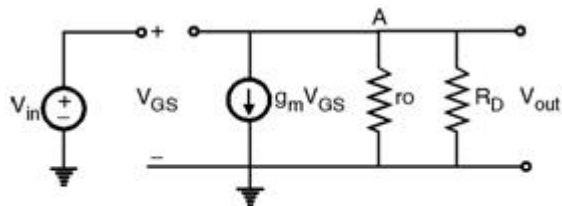
$$\therefore R_{out} = \frac{V_x}{I_x} = R_D$$

$$\therefore R_{out} = R_D$$

\therefore The output impedance of common source amplifier is,

$$R_{out} = R_D$$

If we consider the non Ideal effect such as channel length modulation in the CS amplifier then the small signal model includes one more resistor i.e. r_o as shown in figure below



CS amplifier with CLM

By applying KVL

We get

$$V_{in} - V_{GS} = 0$$

i.e. $V_{in} = V_{GS}$

By applying KCL at node A

We get,

$$g_m V_{GS} + \frac{V_{out} - 0}{r_o} + \frac{V_{out} - 0}{R_D} = 0$$

$$\therefore g_m V_{in} = - \left[\frac{V_{out}}{r_o} + \frac{V_{out}}{R_D} \right]$$

$$\therefore g_m V_{in} = -V_{out} \left(\frac{1}{r_o} + \frac{1}{R_D} \right)$$

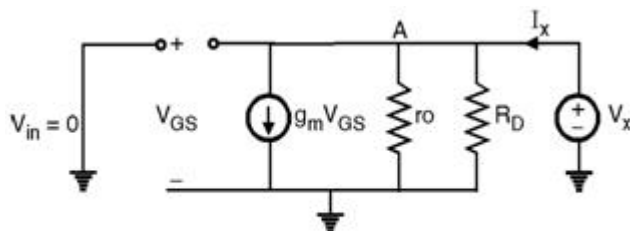
$$\therefore \frac{V_{out}}{V_{in}} = A_v = \frac{-g_m}{\left(\frac{1}{r_o} + \frac{1}{R_D} \right)}$$

$$\therefore A_v = -g_m (r_o \parallel R_D)$$

\therefore The voltage gain of CS amplifier with CLM is $-g_m (r_o \parallel R_D)$

The current gain and input impedance will not be affected by CLM and these are $A_i = \hat{A}_\#$ and $R_{in} = \hat{A}_\#$. But the output impedance is affected because of CLM.

In order to calculate the output impedance of CS amplifier with CLM consider the circuit as shown in Figure below.



Output impedance of CS stage with CLM

By applying KCL at node 'A'

We get, $g_m V_{GS} + \lambda I_D = I_x$

$$\lambda V_x = I_x [V_{GS} = V_{in} = 0]$$

$$\lambda R_{out} = (r_o \parallel R_D)$$

Thus, the output impedance of the CS amplifier with CLM is $r_o \parallel R_D$.

In order to derive the voltage gain of CS amplifier with CLM using I-V characteristics consider the drain current equation with CLM as :

$$I_{DS} = m_n C_{ox} (V_{GS} - V_{TH})^2 (1 + \lambda V_{DS})$$

where λ is channel length modulation coefficient.

From Fig. 8.9.1 (a)

$$V_{DD} - I_D R_D = V_{out}$$

$$\lambda V_{out} = V_{DD} - m_n C_{ox} (V_{in} - V_{TH})^2 (1 + \lambda V_{out})$$

Differentiating this equation with respect to V_{in} .

By product rule of differentiation :

$$= - m_n C_{ox} (V_{in} - V_{TH})^2$$

$$- m_n C_{ox} (1 + \lambda V_{out}) \hat{A}'^2 (V_{in} - V_{TH})$$

$$A_n = - m_n C_{ox} (V_{in} - V_{TH})^2 A_n - g_m R_D$$

By approximating I_D as :

$$I_D = m_n C_{ox} (V_{in} - V_{TH})^2$$

$$\text{We get, } A_n = - g_m R_D - R_D \tilde{A} - \lambda \tilde{A} I_D \tilde{A} - A_n$$

$$\lambda A_n (1 + R_D \tilde{A} - \lambda I_D) = - g_m R_D$$

$$\lambda A_n =$$

As we know that r_o is the linear resistor given as :

$$r_o =$$

To obtain the value of this resistor differentiating I_D with respect to V_{DS} ,

$$\frac{\partial I_D}{\partial V_{DS}} = m_n C_{ox} (V_{GS} - V_{TH})^2 (1 + \lambda V_{DS})$$

$$= m_n C_{ox} (V_{GS} - V_{TH})^2 (\lambda)$$

\ Approximating I_D we get,

$$\lambda = \frac{\partial I_D}{\partial V_{DS}}$$

$$\lambda A_n = =$$

$$\text{i.e. } A_n = -g_m (r_o \parallel R_D)$$

which is same as the voltage gain derived using small signal model.

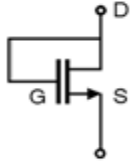
Thus, the voltage gain of CS amplifier is depends upon the transconductance g_m , the linear resistor r_o and load. In order to increase the gain we have to increase the g_m . Inturn we have to increase the ratio.

Hence the gain of amplifier is increases with increasing 'W' and decreasing 'L'. The r_o resistance is appears in shunt with R_D because of this the effect of r_o (i.e. channel length modulation) decreases the voltage gain of amplifier on the other hand the effect of parallel combination of r_o and R_D decreases the output impedance (R_{out}) which is the beneficial effect.

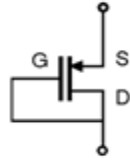
In order to increase the gain of the amplifier along with g_m another important factor is the load impedance connected at the output. To have larger gain load impedance should be larger. The two choices of load impedance of CS stages are :

- 1) Current source load
- 2) Diode connected load.

3.6 CS Amplifier with Active Load :



Diode connected NMOS

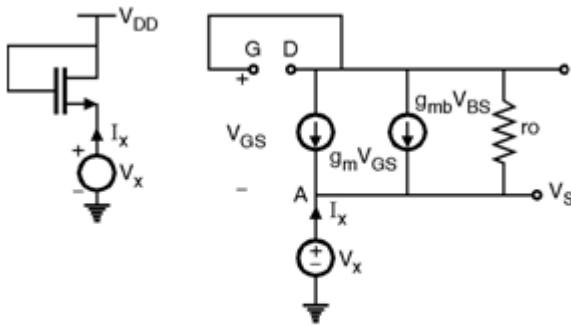


Diode connected PMOS

In CMOS technology it is difficult to fabricate resistors with tightly controlled values of physical size. Hence the load resistor R_D is replaced by the MOS transistor.

To find the load resistance of the diode connected NMOS load consider the circuit shown in

Figure below and its corresponding small signal equivalent circuit.



Diode connected NMOS load

By applying KVL,

$$\text{We have, } V_{GS} = V_G - V_S = -V_x$$

$$V_{BS} = V_B - V_S = -V_x \text{ [Body is connected to ground]}$$

By applying KCL at Node A,

We get,

$$I_x + g_m V_{GS} + g_{mb} V_{BS} + = 0$$

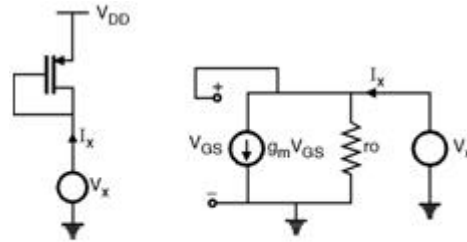
$$\text{i.e. } I_x = g_m V_x + g_{mb} V_x + = V_x$$

=

$$\text{i.e. } = = || r_o$$

\ The load resistance of NMOS diode connected load is given by $| | r_o$

For PMOS diode connected load consider Figure shown below, which shows the circuit to calculate load resistance and its small signal equivalent circuit.



PMOS diode connected load

By applying KVL,.

We have, $V_{GS} = V_G - V_S = - V_x$

$V_{BS} = V_B - V_S = - V_x$ [Body is connected to ground]

By applying KCL at Node A,

We get,

$$I_x + g_m V_{GS} + g_{mb} V_{BS} + = 0$$

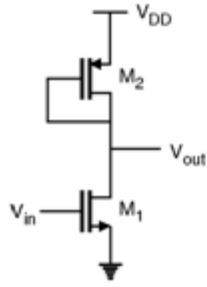
$$\text{i.e. } I_x = g_m V_x + g_{mb} V_x + = V_x$$

=

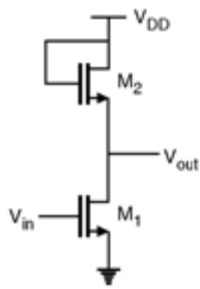
$$\text{i.e. } = = | | r_o$$

The load resistance of PMOS diode connected load is given as $| | r_o$.

Now consider the CS amplifier with diode connected load shown in figure below.



CS amplifier with PMOS diode connected load



CS amplifier with NMOS diode connected load

Here the gain of the amplifier is given by replacing the R_D with the corresponding load resistance of NMOS and PMOS diode connected loads.

\ For NMOS diode connected load.

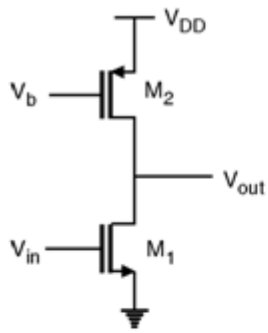
$$A_n = -g_{m1}$$

For PMOS diode connected load,

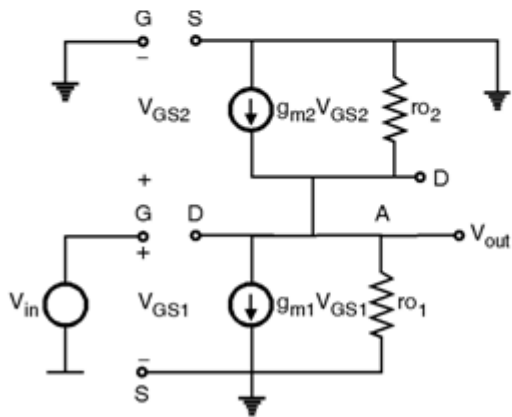
$$A_n = -g_{m1}$$

CS Amplifier with Current Source Load :

Figure below shows the circuit diagram of CS amplifier with current source load. In this a current source is made by using a PMOS transistor operated in saturation mode by using a Gate bias V_b . The small signal model of this circuit is also shown in Figure below

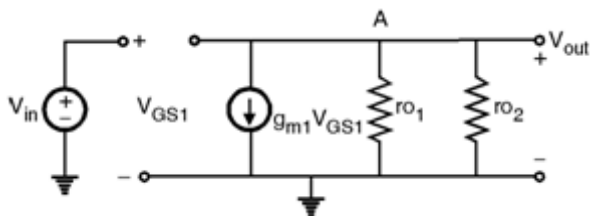


CS amplifier with current source load



small signal model CS amplifier with current source load

The equivalent circuit can be drawn as shown in figure below



By applying KVL, $V_{in} = V_{GS1}$

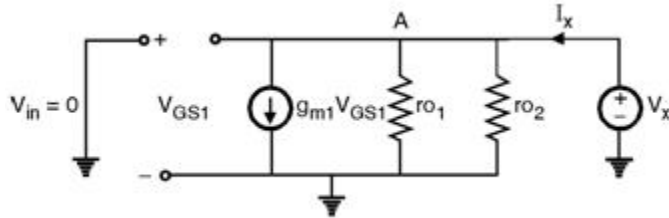
By applying KCL at node A

$$g_{m1} V_{GS1} + \frac{V_{out}}{r_{o2}} = 0$$

$$\therefore g_{m1} V_{in} = - \frac{V_{out}}{r_{o2}}$$

$$A_v = A_n = -g_{m1} (r_{o1} \parallel r_{o2})$$

In order to derive the output impedance consider the circuit shown in Figure below.



By applying KCL at node A,

$$g_{m1} V_{GS1} + I_x = 0$$

$$V_x = I_x [r_{o1} \parallel r_{o2}]$$

$$R_{out} = \frac{V_x}{I_x}$$

$$R_{out} = r_{o1} \parallel r_{o2}$$

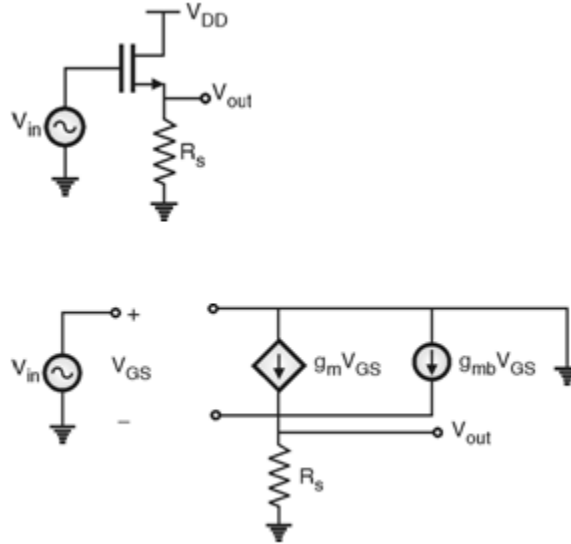
Hence the output impedance of current source load of CS amplifier is $r_{o1} \parallel r_{o2}$.

3.8 Common Drain Amplifier (Source Follower) :

Figure below shows the source follower circuit in which drain terminal of the device is common. In this circuit the drain terminal is directly connected to V_{DD} .

In CS amplifier analysis we have seen that in order to achieve the high voltage gain the load impedance should be as high as possible. Therefore for low impedance load the buffer must be placed after the amplifier to drive the load with negligible loss of the signal level. The source follower thus worked as a buffer stage. The source follower is also called as the common drain amplifier.

In this circuit, the signal at the gate is sensed and drives the load at the source which allows the source potential to follow the gate voltage. The small signal equivalent circuit of the source follower is shown in Figure below.



by analysing the circuit, the voltage gain is given by,

=

From this equation it can be seen that, as g_m increases the A_v approaches to $= 1$.

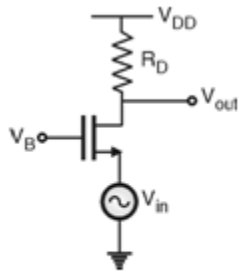
Therefore eventually A_v approaches to unity.

Further, the source follower exhibit a high input impedance and a moderate output impedance. The drawback of source follower are nonlinearity due to body effect and poor driving capability of the input signal.

3.9 Common Gate Amplifier :

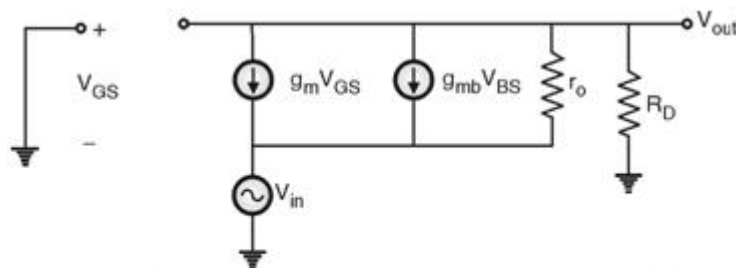
In common source amplifier and source follower circuits, the input signal is applied to the gate of a MOSFET. It is also possible to apply the input signal to the source terminal by keeping common gate terminal. This type of amplifier is called as common gate amplifier.

Figure below shows the CG amplifier in which the input signal is sensed at the source terminal and the output is produced at the drain terminal. The gate terminal is connected to V_B i.e. dc potential which will maintain the proper operating conditions.



Common gate amplifier

Figure below shows the small signal equivalent circuit of the CG amplifier.



small signal equivalent circuit of the CG amplifier

By analyzing the small signal equivalent circuit, the voltage gain of CG amplifier is given by,

$$A_v = g_m R_D$$

The important point is the gain is positive, further the input impedance is given by which shows that the input impedance of common gate amplifier is relatively low. Furthermore, the input impedance of of common gate stage is relatively low only if the load resistance connected to the drain is small.

3.10 CURRENT SOURCE AND SINK

Current source and current sink are two terms that are important to understand, when you choose the type of digital input or output module you want to use for your PLC system. Current source or current sink can only direct current in one direction, which means that the circuit will not operate if you connect it the wrong way. A PLC I/O circuit needs one terminal for current to enter, and another for current to exit. Two terminals are associated with every I/O point.

Source current is the ability of the digital output/input port to supply current. Sink current is the ability of the port to receive current.

When you have a simple circuit where a digital input connects to a digital output, you need three components; a voltage source, a ground, and a load. A sourcing digital I/O provides the voltage, a sinking I/O the ground, and the digital input provides the load.

This means that the input and output works in pairs:

Digital source output – Digital sink input

Digital sink output – Digital source input

Figure 1 shows a sinking digital output that is connected to a sourcing digital input.

In this circuit, the load is pulled to ground because of the sinking digital input provided.

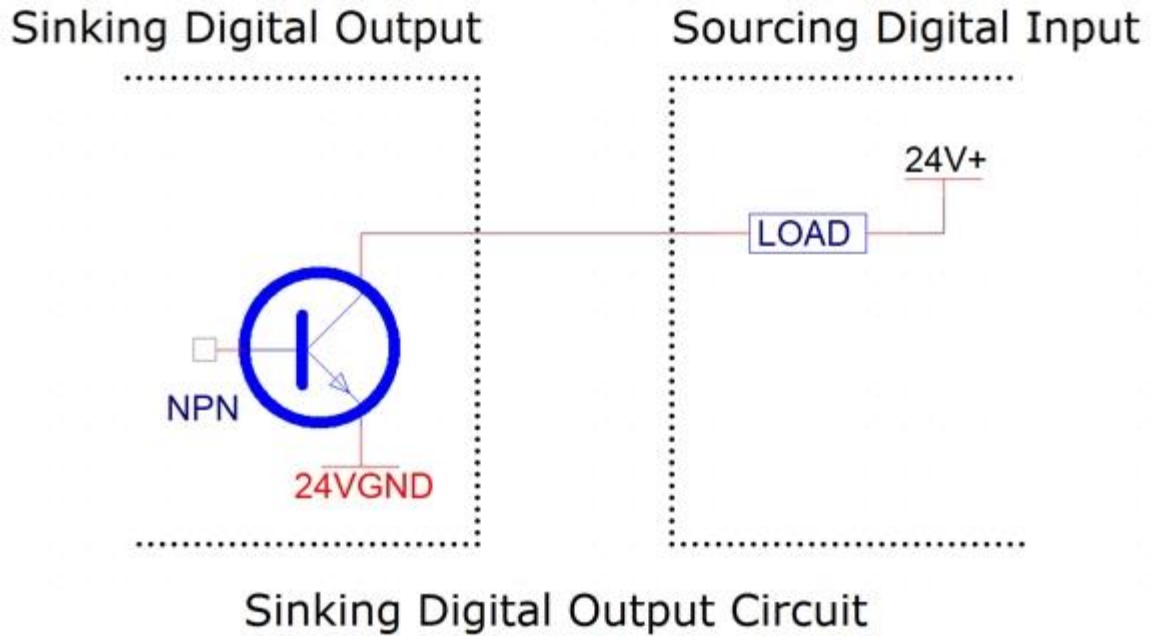
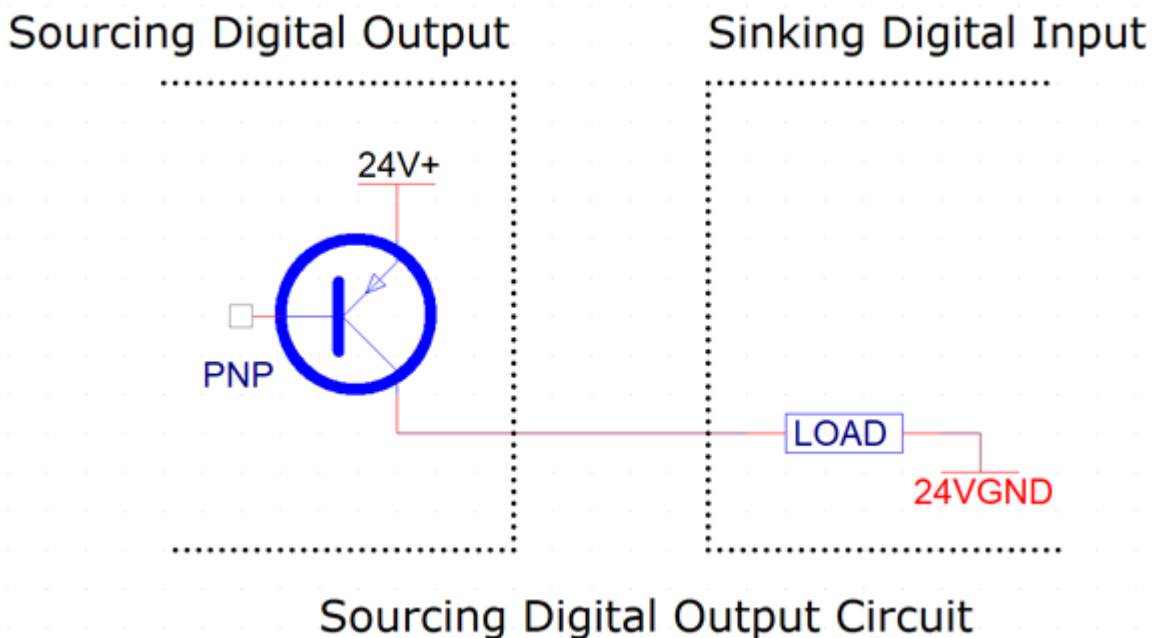


Figure 2 shows a sourcing digital output that is connected to a sinking digital input. In this circuit, the load is pulled up to receive voltage because the sourcing digital input has been provided.



9. Practice Quiz

1. For an n-channel E-MOSFET $V_{th}=5V$, what is the condition to turn

ON the device?

- a) $V_{ds} > 5V$
- b) $V_{gs} < 5V$
- c) $V_{gs} > 5V$**
- d) $V_{ds} = 5V$

2. Which of the following statements is true for E-MOSFET?

- a) It is operated in both depletion mode and enhancement mode
- b) It is capable of operating only in the depletion mode
- c) It is capable of operating only in the enhancement mode**
- d) Neither capable of depletion mode nor in enhancement mode

3. In MOSFET amplifier, the input is applied as:

- a) Voltage across gate and source**
- b) Voltage across drain and source
- c) Current at gate
- d) Current at drain

4. Input impedance of MOSFET amplifier in Common Source configuration is:

- a) Very high at high frequencies
- b) Very high at low frequencies**
- c) Very low at high frequencies
- d) Very low at low frequencies

5. The MOSFET is said to be in diode connected configuration if:

- a) A diode is placed between supply and drain
- b) A diode is placed between source and ground
- c) Source and gate are connected
- d) Drain and gate are connected**

6. The diode connected MOSFET acts as:

- a) Active element for amplification
- b) Voltage source
- c) Current source

d) Load impedance

7. The advantage of using source degeneration resistor in Common source amplifier is to provide:

- a) Huge gain
- b) Non Linearity behavior of amplifier
- c) **Linearity behavior of amplifier**
- d) Less gain

10. Assignments

S.No	Question	BL	CO
1	Explain the single stage amplifier using MOSFET.	2	2
2	Explain the small signal analysis of common source amplifier.	2	2
3	Explain the small signal analysis of common drain amplifier.	2	2

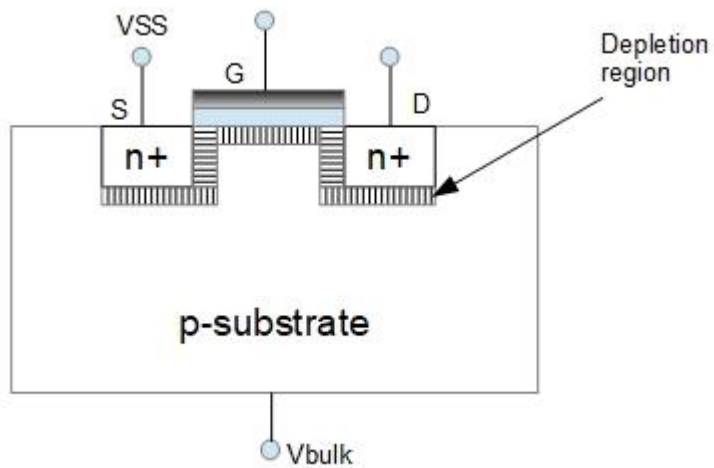
11. Part A- Question & Answers

S.No	Question & Answers	BL	CO
1	<p>List out the various regions of MOSFET</p> <p>Ans. Cut off region, Linear region, saturated region</p>	1	1
2	<p>Draw the model of a transistor.</p> <p>Ans.</p>	1	1

3 Define Body bias effect.

Ans.

Body effect refers to the change in the threshold voltage of the device when there is a difference between substrate (body) and source voltages. Body bias is usually the lowest voltage in the chip (in case of p-substrate).

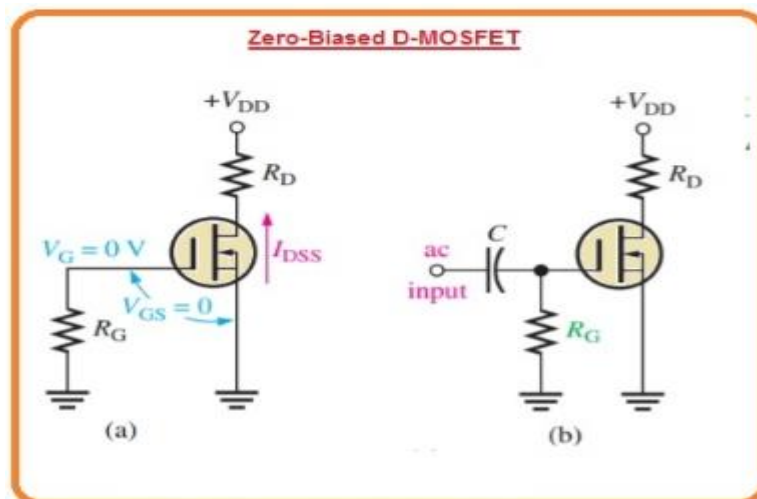


1

1

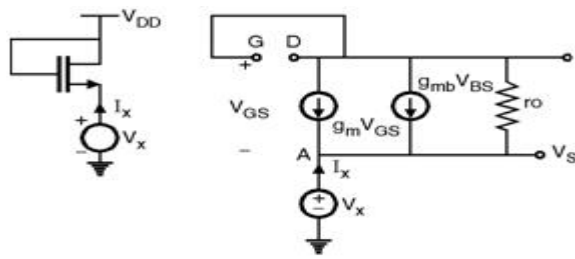
4. Draw the D-MOSFET Bias Configuration.

Ans.



1

1

5.	Draw the CS Amplifier with Active Load.	1	1
	<p>Ans.</p>  <p>Diode connected NMOS load</p>	1	1

SNo	Question	BL	CO
1	Explain the various regions of MOSFET with characteristics.	2	2
2	Explain the concept of common source amplifier.	2	2
3	Explain the concept of common gate amplifier.	2	2
4	Explain the concept of common drain amplifier.	2	3

Lecture Notes

4.1 Complementary CMOS- introduction

Static CMOS Design

The most widely used logic style is static complementary CMOS. The static CMOS style is really an extension of the static CMOS inverter to multiple inputs. The primary advantage of the CMOS structure is robustness (i.e., low sensitivity to noise), good performance, and low power consumption with no static power dissipation. Most of those properties are carried over to large fan-in logic gates implemented using a similar circuit topology.

The complementary CMOS circuit style falls under a broad class of logic circuits called static circuits in which at every point in time (except during the switching transients), each gate output is connected to either VDD or Vss via a low-resistance path. Also, the outputs of the gates assume at all times the value of the Boolean function implemented by the circuit (ignoring, once again, the transient effects during switching periods). This is in contrast to the dynamic circuit class, which relies on temporary storage of signal values on the capacitance of high-impedance circuit nodes. The latter approach has the advantage that the resulting gate is simpler and faster. Its design and operation are however more involved and prone to failure due to an increased sensitivity to noise. The design of various static circuit flavors includes complementary CMOS, ratioed logic (pseudo-NMOS and DCVSL), and pass transistor logic.

Large systems are composed of sub-systems, known as Leaf-Cell. The most basic leaf cell is the common logic gate (inverter, and, etc.). Structured Design-High Regularity-Leaf cells replicated many times and interconnected to form the system. Logical and systematic approach to VLSI design is essential.

Complementary CMOS

A static CMOS gate is a combination of two networks, called the pull-up network (PUN) and the pull-down network (PDN) (Figure 1). The figure shows a generic N input logic gate where all inputs are distributed to both the pull-up and pull-down networks. The function of the PUN is to provide a connection between the output and VDD anytime the output of the logic gate is meant to

be 1 (based on the inputs). Similarly, the function of the PDN is to connect the output to VSS when the output of the logic gate is meant to be 0. The PUN and PDN networks are constructed in a mutually exclusive fashion such that one and only one of the networks is conducting in steady state. In this way, once the transients have settled, a path always exists between VDD and the output F, realizing a high output ("one"), or, alternatively, between VSS and F for a low output ("zero"). This is equivalent to stating that the output node is always a low-impedance node in steady state.

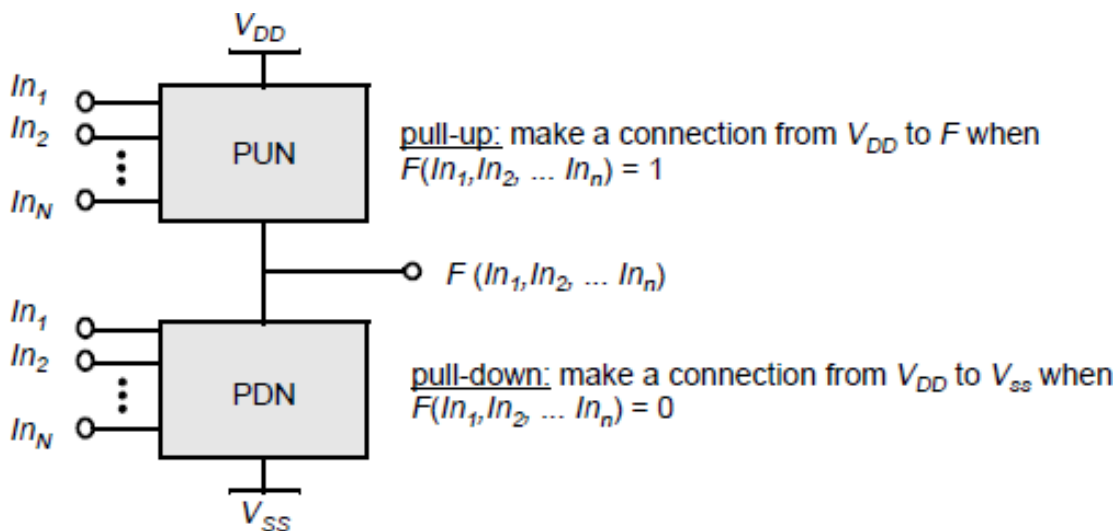


Fig 4.1.1: Complementary logic gate as a combination of a PUN (pull-up network) and a PDN (pull-down network)

In constructing the PDN and PUN networks, the following observations should be kept in mind:

- A transistor can be thought of as a switch controlled by its gate signal. An NMOS switch is on when the controlling signal is high and is off when the controlling signal is low. A PMOS transistor acts as an inverse switch that is on when the controlling signal is low and off when the controlling signal is high.
- The PDN is constructed using NMOS devices, while PMOS transistors are used in the PUN. The primary reason for this choice is that NMOS transistors produce "strong zeros," and PMOS devices generate "strong ones". To illustrate this, consider the examples shown in Figure 2. In Figure 2.a, the output capacitance is initially charged to VDD. Two possible discharge scenarios are shown. An NMOS device pulls the output all the way down to GND, while a PMOS lowers

the output no further than $|V_{Tp}|$ — the PMOS turns off at that point, and stops contributing discharge current. NMOS transistors are hence the preferred devices in the PDN. Similarly, two alternative approaches to charging up a capacitor are shown in Figure 2.b, with the output initially at GND. A PMOS switch succeeds in charging the output all the way to V_{DD} , while the NMOS device fails to raise the output above $V_{DD}-V_{Tn}$. This explains why PMOS transistors are preferentially used in a PUN.

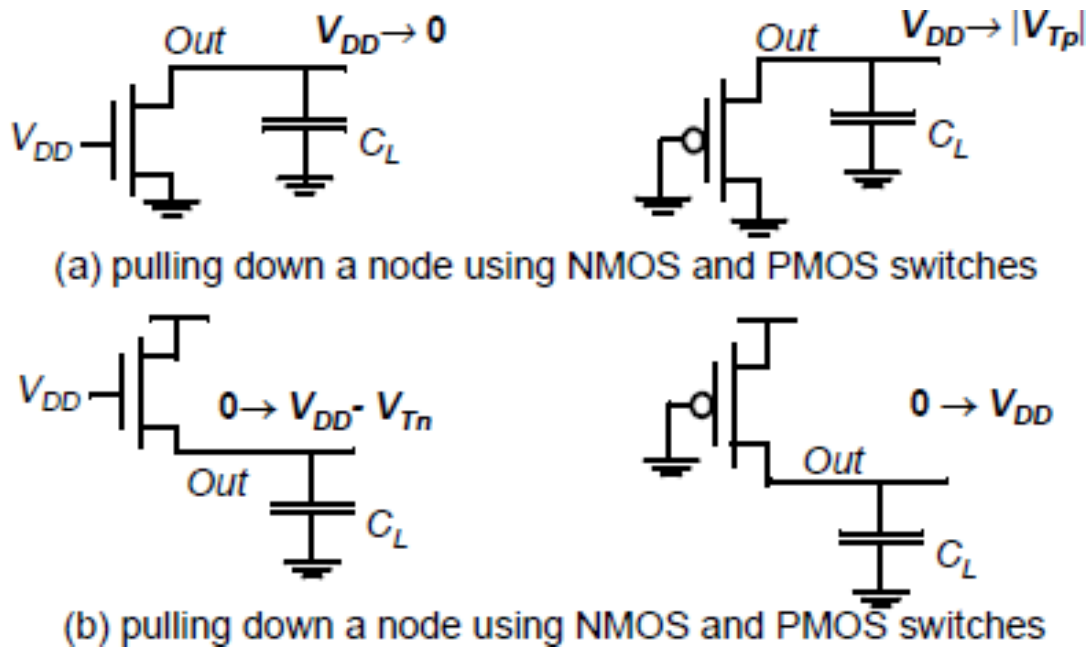


Fig 4.1.2: Simple examples illustrate why an NMOS should be used as a pull-down, and a PMOS should be used as a pull-up device.

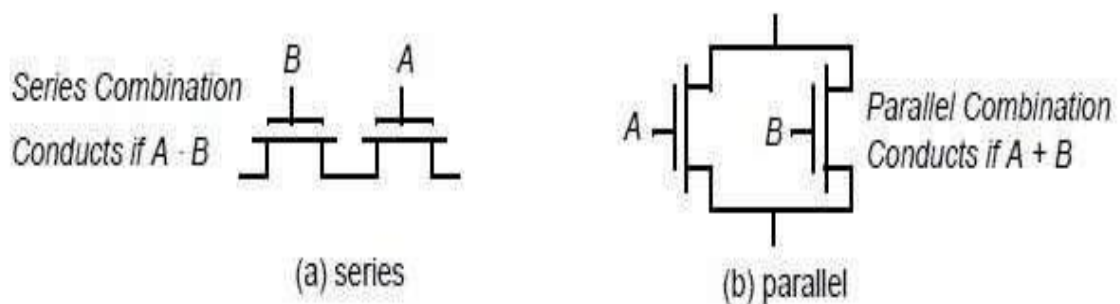


Fig 4.1.3: NMOS logic rules — series devices implement an AND, and parallel devices implement an OR.

A set of construction rules can be derived to construct logic functions (Figure 4). NMOS devices connected in series corresponds to an AND function. With all the inputs high, the series combination conducts and the value at one end of the

chain is transferred to the other end. Similarly, NMOS transistors connected in parallel represent an OR function. A conducting path exists between the output and input terminal if at least one of the inputs is high. Using similar arguments, construction rules for PMOS networks can be formulated. A series connection of PMOS conducts if both inputs are low, representing a NOR function ($A \cdot B = A + B$), while PMOS transistors in parallel implement a NAND ($A + B = A \cdot B$).

- Using De Morgan's theorems ($(A + B) = A \cdot B$ and $A \cdot B = A + B$), it can be shown that the pull-up and pull-down networks of a complementary CMOS structure are dual networks. This means that a parallel connection of transistors in the pull-up network corresponds to a series connection of the corresponding devices in the pull-down network, and vice versa. Therefore, to construct a CMOS gate, one of the networks (e.g., PDN) is implemented using combinations of series and parallel devices. The other network (i.e., PUN) is obtained using duality principle by walking the hierarchy, replacing series sub-nets with parallel sub-nets, and parallel sub-nets with series sub-nets. The complete CMOS gate is constructed by combining the PDN with the PUN.

The complementary gate is naturally inverting, implementing only functions such as NAND, NOR, and XNOR. The realization of a non-inverting Boolean function (such as AND OR, or XOR) in a single stage is not possible, and requires the addition of an extra inverter stage.

The number of transistors required to implement an N-input logic gate is $2N$.

4.2 RATIOED CIRCUITS

Ratioed circuits depend on the proper size or resistance of devices for correct operation. For example, in the 1970s and early 1980s before CMOS technologies matured, circuits were often built with only nMOS transistors, as shown in Figure 9.12. Conceptually, the ratioed gate consists of an nMOS pulldown network and some pullup device called the static load. When the pulldown network is OFF, the static load pulls the output to 1. When the pulldown network turns ON, it fights the static load. The static load must be weak enough that the output pulls down to an acceptable 0. Hence, there is a ratio constraint between the static load and pulldown network. Stronger static loads produce faster rising outputs, but increase V_{OL} , degrade the noise margin, and burn more static power when

the output should be 0. Unlike complementary circuits, the ratio must be chosen so the circuit operates correctly despite any variations from nominal component values that may occur during manufacturing. CMOS logic eventually displaced nMOS logic because the static power became unacceptable as the number of gates increased. However, ratioed circuits are occasionally still useful in special applications.

A resistor is a simple static load, but large resistors consume a large layout area in typical MOS processes. Another technique is to use an nMOS transistor with the gate tied to VGG. If $V_{GG} = V_{DD}$, the nMOS transistor will only pull up to $V_{DD} - V_t$. Worse yet, the threshold is increased by the body effect. Thus, using $V_{GG} > V_{DD}$ was attractive. To eliminate this extra supply voltage, some nMOS processes offered depletion mode transistors. These transistors, indicated with the thick bar, are identical to ordinary enhancement mode transistors except that an extra ion implantation was performed to create a negative threshold voltage. The depletion mode pullups have their gate wired to the source so $V_{gs} = 0$ and the transistor is always weakly ON.

Pseudo-nMOS

Figure 4.2.1 shows a *pseudo-nMOS* inverter. Neither high-value resistors nor depletion mode transistors are readily available as static loads in most CMOS

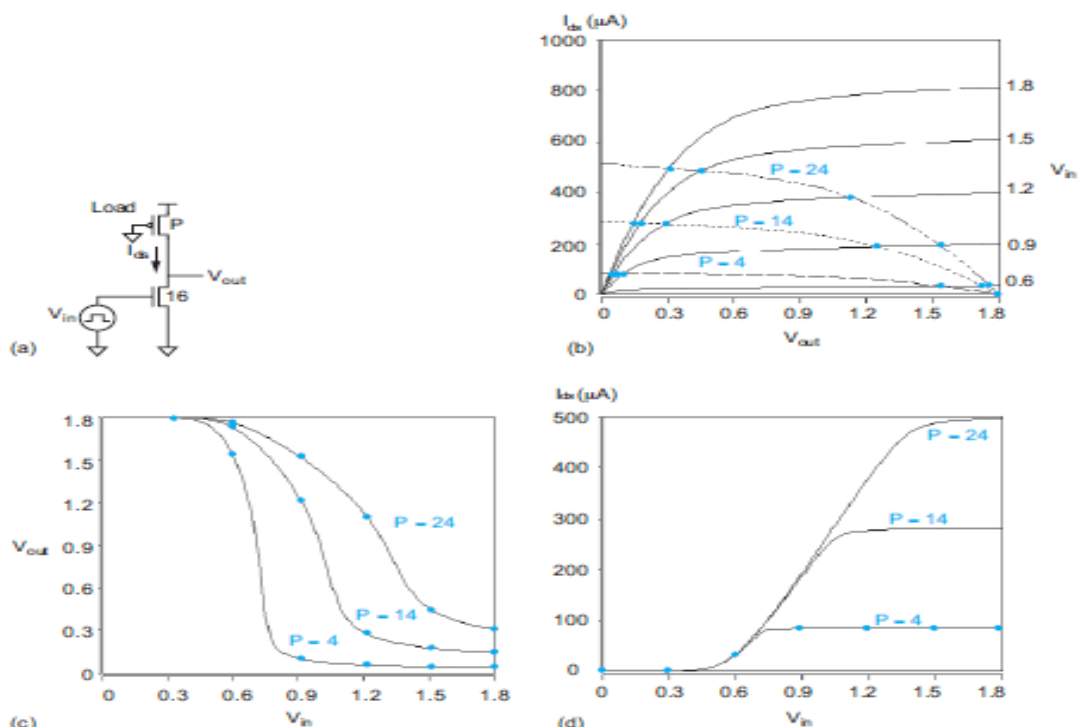


Fig 4.2.1: Pseudo-nMOS inverter and DC transfer characteristics

processes. Instead, the static load is built from a single pMOS transistor that has its gate grounded so it is always ON. The DC transfer characteristics are derived by finding V_{out} for which $I_{dsn} = |I_{dsp}|$ for a given V_{in} , as shown in Figure 9.13(b–c) for a 180 nm process. The beta ratio affects the shape of the transfer characteristics and the VOL of the inverter. Larger relative pMOS transistor sizes offer faster rise times but less sharp transfer characteristics. Figure 4.2.1 shows that when the nMOS transistor is turned on, a static DC current flows in the circuit. Figure 4.2.1 shows several pseudo-nMOS logic gates. The pulldown network is like that of an ordinary static gate, but the pullup network has been replaced with a single pMOS transistor that is grounded so it is always ON. The pMOS transistor widths are selected to be about 1/4 the strength (i.e., 1/2 the effective width) of the nMOS pulldown network as a compromise between noise margin and speed; this best size is process-dependent, but is usually in the range of 1/3 to 1/6.

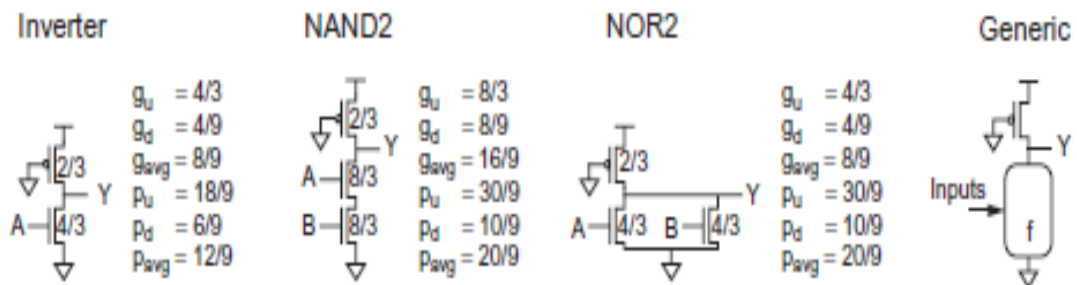


Fig 4.2.2: Pseudo-nMOS logic gates

To calculate the logical effort of pseudo-nMOS gates, suppose a complementary CMOS unit inverter delivers current I in both rising and falling transitions. For the widths shown, the pMOS transistors produce $I/3$ and the nMOS networks produce $4I/3$. The logical effort for each transition is computed as the ratio of the input capacitance to that of a complementary CMOS inverter with equal current for that transition. For the falling transition, the pMOS transistor effectively fights the nMOS pulldown. The output current is estimated as the pulldown current minus the pullup current, $(4I/3 - I/3) = I$. Therefore, we

will compare each gate to a unit inverter to calculate g_d . For example, the logical effort for a falling transition of the pseudo-nMOS inverter is the ratio of its input capacitance ($4/3$) to that of a unit complementary CMOS inverter (3), i.e., $4/9$. g_u is three times as great because the current is $1/3$ as much.

The parasitic delay is also found by counting output capacitance and comparing it to an inverter with equal current. For example, the pseudo-nMOS NOR has 10 units of diffusion capacitance as compared to 3 for a unit-sized complementary CMOS inverter, so its parasitic delay pulling down is $10/9$. The pullup current is $1/3$ as great, so the parasitic delay pulling up is $10/3$. As can be seen, pseudo-nMOS is slower on average than static CMOS for NAND structures. However, pseudo-nMOS works well for NOR structures. The logical effort is independent of the number of inputs in wide NORs, so pseudo-nMOS is useful for fast wide NOR gates or NOR-based structures like ROMs and PLAs when power permits.

Pseudo-nMOS gates will not operate correctly if $V_{OL} > V_{IL}$ of the receiving gate. This is most likely in the SF design corner where nMOS transistors are weak and pMOS transistors are strong. Designing for acceptable noise margin in the SF corner forces a conservative choice of weak pMOS transistors in the normal corner. A biasing circuit can be used to reduce process sensitivity.

The goal of the biasing circuit is to create a V_{bias} that causes P2 to deliver $1/3$ the current of N2, independent of the relative mobilities of the pMOS and nMOS transistors. Transistor N2 has width of $3/2$ and hence produces current $3I/2$ when ON. Transistor N1 is tied ON to act as a current source with $1/3$ the current of N2, i.e., $I/2$. P1 acts as a current mirror using feedback to establish the bias voltage sufficient to provide equal current as N1, $I/2$. The size of P1 is noncritical so long as it is large enough to produce sufficient current and is equal in size to P2. Now, P2 ideally also provides $I/2$. In summary, when A is low, the pseudo-nMOS gate pulls up with a current of $I/2$. When A is high, the pseudo-nMOS gate pulls down with an effective current of $(3I/2 - I/2) = I$. To first order, this biasing technique sets the relative currents strictly by transistor widths, independent of relative pMOS and nMOS mobilities.

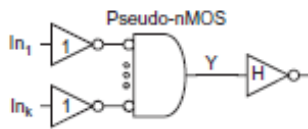


Fig 4.2.3: k-input AND gate driving load of H

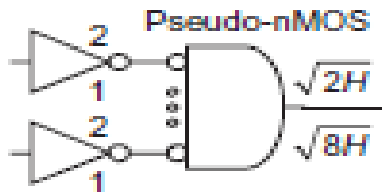


Fig 4.2.4: k-input AND marked with transistor widths

Ganged CMOS:

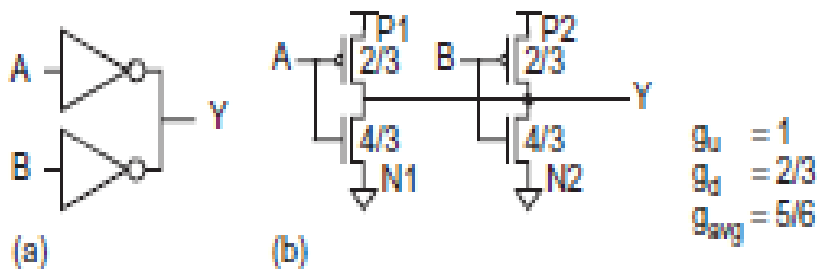


Fig 4.2.5: Symmetric 2-input NOR gate

in Table 4.2, showing that the pair compute the NOR function. Such a circuit is sometimes called a symmetric 2 NOR [Johnson88], or more generally, ganged CMOS [Schultz90]. When one input is 0 and the other 1, the gate can be viewed as a pseudo-nMOS circuit with appropriate ratio constraints. When both inputs are 0, both pMOS transistors turn on in parallel, pulling the output high faster than they would in an ordinary pseudo nMOS gate. Moreover, when both inputs are 1, both pMOS transistors turn OFF, saving static power dissipation. As in pseudo-nMOS, the transistors are sized so the pMOS are about 1/4 the strength of the nMOS and the pulldown current matches that of a unit inverter. Hence, the symmetric NOR achieves both better performance a lower power dissipation than a 2-input pseudo-nMOS NOR.

A	B	N1	P1	N2	P2	Y
0	0	OFF	ON	OFF	ON	1
0	1	OFF	ON	ON	OFF	~ 0
1	0	ON	OFF	OFF	ON	~ 0
1	1	ON	OFF	ON	OFF	0

Table: Operation of Symmetric NOR

Johnson also showed that symmetric structures can be used for NOR gates with more inputs and even for NAND gates. The 3-input symmetric NOR also works well, but the logical efforts of the other structures are unattractive.

4.3 Pass-Transistor Circuits & 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, (VDD and GND) are the source of the strongest 1s and 0s. An nMOS transistor is an almost perfect switch when passing a 0 and thus we say it passes a strong 0. However, the nMOS transistor is imperfect at passing a 1. The high voltage level is somewhat less than VDD, as will be explained in Section 2.5.4. We say it passes a degraded or weak 1. A pMOS transistor again has the opposite behavior, passing strong 1s but degraded 0s. The transistor symbols and behaviors are summarized in Figure 1.20 with g, s, and d indicating gate, source, and drain. When an nMOS or pMOS is used alone as an imperfect switch, we sometimes call it a pass transistor. By combining an nMOS and a pMOS transistor in parallel (Figure 4.3(a)), we obtain a switch that turns on when a 1 is applied to g (Figure 4.3(b)) in which 0s and 1s are both passed in an acceptable fashion (Figure 4.3(c)). We term this a transmission gate or pass gate. In a circuit where only a 0 or a 1 has to be passed, the appropriate transistor (n or p) can be deleted, reverting to a single nMOS or pMOS device.

In the circuit families we have explored so far, inputs are applied only to the gate terminals of transistors. In pass-transistor circuits, inputs are also 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. Many authors have claimed substantial area, speed, and/or

power improvements for pass transistors compared to static CMOS logic. In specialized circumstances this can be true; for example, pass transistors are essential to the design of efficient 6-transistor static RAM cells used in most modern systems (see Section 12.2). Full adders and other circuits rich in XORs also can be efficiently constructed with pass transistors. In certain other cases, we will see that pass-transistor circuits are essentially equivalent ways to draw the fundamental logic structures we have explored before. An independent evaluation finds that for most general-purpose logic, static CMOS is superior in speed, power, and area [Zimmermann97]. For the purpose of comparison, Figure 9.47 shows a 2-input multiplexer constructed in a wide variety of pass-transistor circuit families along with static CMOS, pseudo nMOS, CVSL, and single- and dual-rail domino. Some of the circuit families are dual rail, producing both true and complementary outputs, while others are single-rail and may require an additional inversion if the other polarity of output is needed. XOR can be computed with exactly the same logic using $S = U, S = \bar{U}, A = V, B = \bar{V}$.

This shows that static CMOS is particularly poorly suited to XOR because the complex gate and two additional inverters are required; hence, pass-transistor circuits become attractive. In comparison, static CMOS NAND and NOR gates are relatively efficient and benefit less from pass transistors.

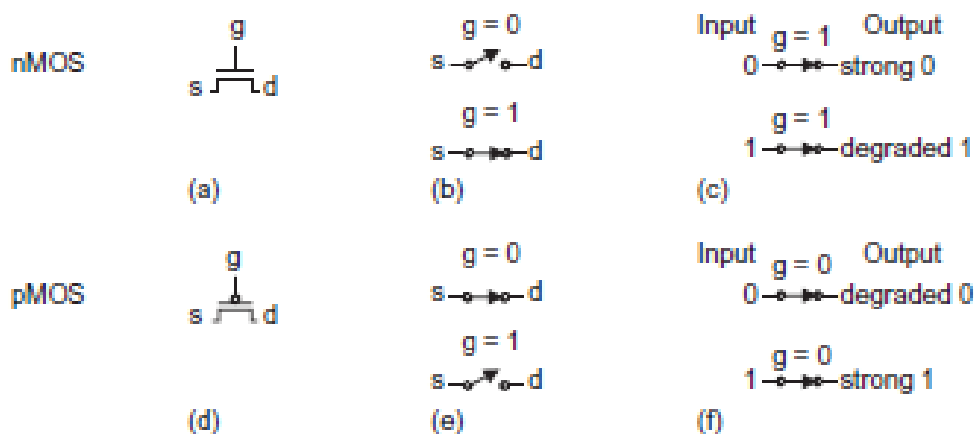


FIG 4.3.1: Pass transistor strong and degraded outputs

Note that both the control input and its complement are required by the transmission gate. This is called double rail logic. Some circuit symbols for the transmission gate are shown in Figure 1.21 (d). None are easier to draw than the simple schematic, so we will use the schematic version to represent a

transmission gate in this book. In all of our examples so far, the inputs drive the gate terminals of nMOS transistors in the pull-down network and pMOS transistors in the complementary pull-up network, as was shown in Figure 1.14. Thus, the nMOS transistors only need to pass 0s and the pMOS only pass 1s, so the output is always strongly driven and the levels are never degraded. This is called a fully restored logic gate and simplifies circuit design considerably.

In contrast to other forms of logic, where the pull-up and pull-down switch networks have to be ratioed in some manner, static CMOS gates operate correctly independently of the physical sizes of the transistors. Moreover, there is never a path through 'ON' transistors from the 1 to the 0 supplies for any combination of inputs (in contrast to single-channel MOS, GaAs technologies, or bipolar). As we will find in subsequent chapters, this is the basis for the low static power dissipation in CMOS.

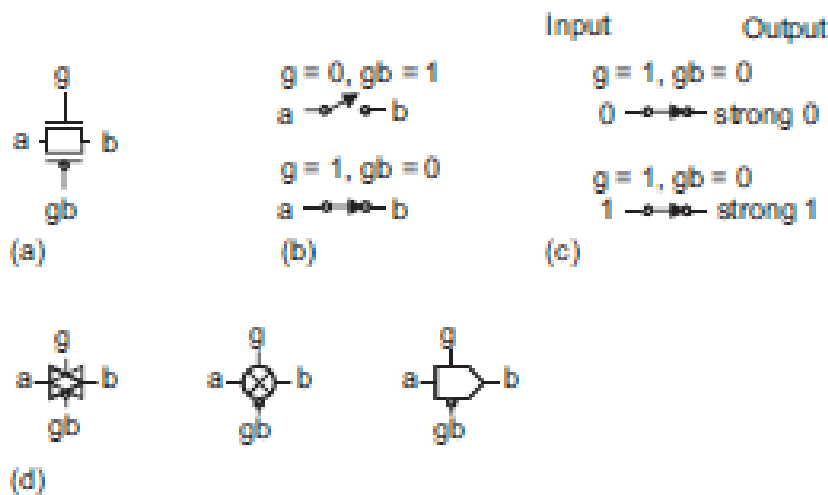


FIG 4.3.2: Transmission gate

A consequence of the design of static CMOS gates is that they must be inverting. The nMOS pull-down network turns ON when inputs are 1, leading to 0 at the output. We might be tempted to turn the transistors upside down to build a noninverting gate. For example, Figure 4.3.3 shows a noninverting buffer. Unfortunately, now both the nMOS and pMOS transistors produce degraded outputs, so the technique should be avoided. Instead, we can build noninverting functions from multiple stages of inverting gates. Figure 4.3.4 shows several ways to build a 4-input AND gate from two levels of inverting static CMOS gates. Each design has different speed, size, and power trade-offs.

Similarly, the compound gate could be built with two AND gates, an OR gate, and an inverter. The AND and OR gates in turn could be constructed from NAND/NOR gates and inverters, using a total of 20 transistors, as compared to eight. Good CMOS logic designers exploit the efficiencies of compound gates rather than using large numbers of AND/OR gates.

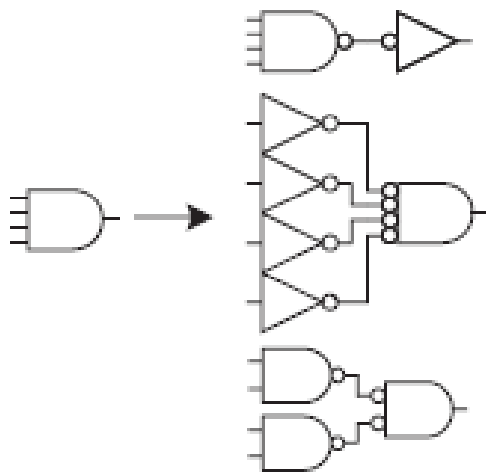


FIG 4.3.3: Various implementations of a CMOS 4-input AND gate

4.4 Dynamic logic – introduction

It was noted earlier that static CMOS logic with a fan-in of N requires $2N$ devices. A variety of approaches were presented to reduce the number of transistors required to implement a given logic function including pseudo-NMOS, pass transistor logic, etc. The pseudo-NMOS logic style requires only $N + 1$ transistors to implement an N input logic gate, but unfortunately it has static power dissipation. In this section, an alternate logic style called dynamic logic is presented that obtains a similar result, while avoiding static power consumption. With the addition of a clock input, it uses a sequence of pre charge and conditional evaluation phases.

4.5 Dynamic Logic: Basic Principles

The basic construction of an (n -type) dynamic logic gate is shown in Figure 4.5.a. The PDN (pull-down network) is constructed exactly as in complementary

CMOS. The operation of this circuit is divided into two major phases: precharge and evaluation, with the mode of operation determined by the clock signal CLK.

Precharge

When $CLK = 0$, the output node Out is precharged to VDD by the PMOS transistor M_p . During that time, the evaluate NMOS transistor M_e is off, so that the pull-down path is disabled. The evaluation FET eliminates any static power that would be consumed during the precharge period (this is, static current would flow between the supplies if both the pull-down and the precharge device were turned on simultaneously).

Evaluation

For $CLK = 1$, the precharge transistor M_p is off, and the evaluation transistor M_e is turned on. The output is conditionally discharged based on the input values and the pull-down topology. If the inputs are such that the PDN conducts, then a low resistance path exists between Out and GND and the output is discharged to GND. If the PDN is turned off, the precharged value remains stored on the output capacitance CL, which is a combination of junction capacitances, the wiring capacitance, and the input capacitance of the fan-out gates. During the evaluation phase, the only possible path between the output node and a supply rail is to GND. Consequently, once Out is discharged, it cannot be charged again till then next precharge operation. The inputs to the gate can therefore make at most one transition during evaluation. Notice that the output can be in the high-impedance state during the evaluation period if the pull-down network is turned off. This behavior is fundamentally different from the static counterpart that always has a low resistance path between the output and one of the power rails.

As an example, consider the circuit shown in Figure 4.5b. During the precharge phase ($CLK=0$), the output is precharged to VDD regardless of the input values since the evaluation device is turned off. During evaluation ($CLK=1$), a conducting path is created between Out and GND if (and only if) $A \cdot B + C$ is TRUE. Otherwise, the output remains at the precharged state of VDD. The following function is thus realized:

$$Out = CLK + \cdot AB + C() \cdot CLK$$

A number of important properties can be derived for the dynamic logic gate:

1. The logic function is implemented by the NMOS pull-down network. The construction of the PDN proceeds just as it does for static CMOS.

2. The number of transistors (for complex gates) is substantially lower than in the static case: $N + 2$ versus $2N$.

3. It is non-ratioed. The sizing of the PMOS precharge device is not important for realizing proper functionality of the gate. The size of the precharge device can be made large to improve the low-to-high transition time (of course, at a cost to the high-to-low transition time). There is however, a trade-off with power dissipation since a larger precharge device directly increases clock-power dissipation.

4. It only consumes dynamic power. Ideally, no static current path ever exists between VDD and GND. The overall power dissipation, however, can be significantly higher compared to a static logic gate.

5. The logic gates have faster switching speeds. There are two main reasons for this. The first (obvious) reason is due to the reduced load capacitance attributed to the lower number of transistors per gate and the single-transistor load per fan-in. Second, the dynamic gate does not have short circuit current, and all the current provided by the pull-down devices goes towards discharging the load capacitance.

The low and high output levels V_{OL} and V_{OH} are easily identified as GND and VDD and are not dependent upon the transistor sizes. The other VTC parameters are dramatically different from static gates. Noise margins and switching thresholds have been defined as static quantities that are not a function of time. To be functional, a dynamic gate requires a periodic sequence of precharges and evaluations. Pure static analysis, therefore, does not apply. During the evaluate period, the pull-down network of a dynamic inverter starts to conduct when the input signal exceeds the threshold voltage (V_{Tn}) of the NMOS pull-down transistor. Therefore, it is reasonable to set the switching threshold (V_M) as well as V_{IH} and V_{IL} of the gate equal to V_{Tn} . This translates to

a low value for the NML.

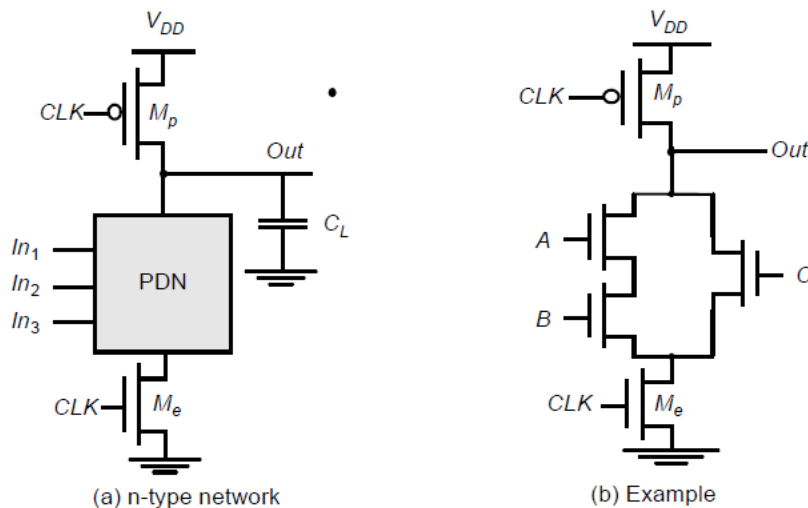


Fig 4.5: Basic concepts of dynamic gate

Design Consideration

It is also possible to implement dynamic logic using a complimentary approach, where the out-put node is connected by a pre-discharge NMOS transistor to GND, and the evaluation PUN network is implemented in PMOS. The operation is similar: during precharge, the output node is discharged to GND. During evaluation, the output is conditionally charged to VDD. This p-type dynamic gate has the disadvantage of being slower than the n-type due to the lower current drive of the PMOS transistors.

4.6 Speed and Power Dissipation of Dynamic Logic

The main advantages of dynamic logic are increased speed and reduced implementation area. Fewer devices to implement a given logic function implies that the overall load capacitance is much smaller. The analysis of the switching behavior of the gate has some interesting peculiarities to it. After the precharge phase, the output is high. For a low input signal, no additional switching occurs. As a result, $t_{pLH} = 0$! The high-to-low transition, on the other hand, requires the discharging of the output capacitance through the pull-down network. Therefore t_{pHL} is proportional to C_L and the current-sinking capabilities of the pull-down network. The presence of the evaluation transistor slows the gate somewhat, as it presents an extra series resistance. Omitting this transistor, while functionally not forbidden, may result in static power dissipation and potentially a performance loss.

The above analysis is somewhat unfair, because it ignores the influence of the pre-charge time on the switching speed of the gate. The precharge time is determined by the time it takes to charge CL through the PMOS precharge transistor. During this time, the logic in the gate cannot be utilized. However, very often, the overall digital system can be designed in such a way that the precharge time coincides with other system functions. For instance, the precharge of the arithmetic unit in a microprocessor can coincide with the instruction decode. The designer has to be aware of this “dead zone” in the use of dynamic logic, and should carefully consider the pros and cons of its usage, taking the overall system requirements into account.

Example: A Four-Input Dynamic NAND Gate

Figure 4.6.1 shows the design of a four-input NAND example designed using the dynamic-circuit style. Due to the dynamic nature of the gate, the derivation of the voltage-transfer characteristic diverges from the traditional approach. As we had discussed above, we will assume that the switching threshold of the gate equals the threshold of the NMOS pull-down transistor. This results in asymmetrical noise margins, as shown in Table 6.9.

The dynamic behavior of the gate is simulated with SPICE. It is assumed that all inputs are set high as the clock transitions high. On the rising edge of the clock, the output node is discharged. The resulting transient response is plotted in Figure 4.6.1, and the propagation delays are summarized in Table 6.9. The duration of the precharge cycle can be adjusted by changing the size of the PMOS precharge transistor. Making the PMOS too large should be avoided, however, as it both slows down the gate, and increases the capacitive load on the clock line. For large designs, the latter factor might become a major design concern as the clock load can become excessive and hard to drive.

Table 6.9 The dc and ac parameters of a four-input dynamic NAND.

Transistors	V_{OH}	V_{OL}	V_M	NM_H	NM_L	t_{pHL}	t_{pLH}	t_{pre}
6	2.5 V	0 V	V_{TN}	$2.5 - V_{TN}$	V_{TN}	110 psec	0 nsec	83psec

As mentioned earlier, the static parameters are time-dependent. To illustrate this, consider the four input NAND gate with all inputs tied together, and making a partial low-to-high transition. Figure 6.54 shows a transient simulation of the output voltage for three different input transitions—to 0.45V, 0.5V and 0.55V, respectively. Above, we have defined the switching threshold of the dynamic gate as the device threshold. However, notice that the amount by which the output voltage drops is a strong function of the input voltage and the available evaluation time. The noise voltage needed to corrupt the signal has to be larger if the evaluation time is short. In other words, the switching threshold is really a function of the evaluation time.

When evaluating the power dissipation of a dynamic gate, it would appear that dynamic logic presents a significant advantage. There are three reasons for this. First, the physical capacitance is lower since dynamic logic uses fewer transistors to implement a given function. Also, the load seen for each fanout is one transistor instead of two. Second, dynamic logic gates by construction can at most have one transition per clock cycle. Glitching (or dynamic hazards) does not occur in dynamic logic. Finally, dynamic gates do not exhibit short circuit power since the pull-up path is not turned on when the gate is evaluating.

While these arguments are generally true, they are offset by other considerations: (i) the clock power of dynamic logic can be significant, particularly since the clock node has a guaranteed transition on every single clock cycle; (ii) the number of transistors is higher than the minimal set required for implementing the logic; (iii) short-circuit power may exist when leakage-combatting devices are added (as will be discussed further); (iv) and, most importantly, dynamic logic generally displays a higher switching activity due to the periodic pre charge and discharge operations. Earlier, the transition probability for a static gate was shown to be $p_0 p_1 = p_0 (1-p_0)$. For dynamic logic, the output transition probability does not depend on the state (history) of the inputs, but rather on the signal probabilities only. For an n-tree dynamic

gate, the output makes a 0→1 transition during the precharge phase only if the output was discharged during the preceding evaluate phase. The 0→1 transition probability for an n-type dynamic gate hence equals

$$\alpha_{0 \rightarrow 1} = p_0$$

where p_0 is the probability that the output is zero. This number is always larger or equal to $p_0 p_1$. For uniformly distributed inputs, the transition probability for an N-input gate is:

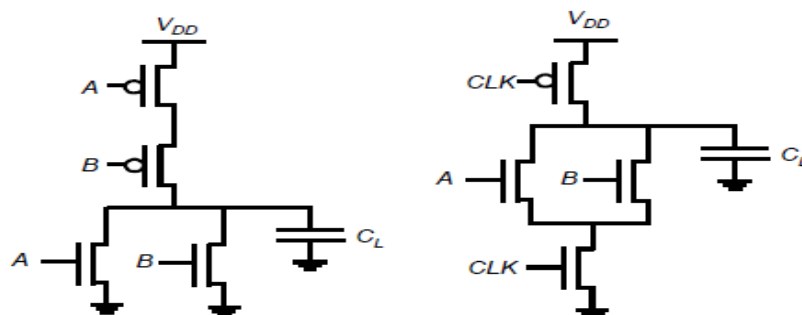
$$\alpha_{0 \rightarrow 1} = \frac{N_0}{2^N}$$

where N_0 is the number of zero entries in the truth table of the logic function.

Activity estimation in dynamic logic

To illustrate the increased activity for a dynamic gate, once again consider a 2 input NOR gate. An n-tree dynamic implementation is shown in Figure 6.55 along with its static counter-part. For equiprobable inputs, there is then a 75% probability that the output node of the dynamic gate will discharge immediately after the pre charge phase, implying that the activity for such a gate equals 0.75 (i.e. $P_{NOR} = 0.75 \text{ CLV}_{dd} 2f_{clk}$). The corresponding activity is a lot smaller, 3/16, for a static implementation. For a dynamic NAND gate, the transition probability is 1/4 (since there is a 25% probability the output will be discharged) while it is 3/16 for a static implementation. Though these examples illustrate that the switching activity of dynamic logic is generally higher, it should be noted that dynamic logic has lower physical capacitance. Both factors must be accounted for when choosing a logic style.

Figure 4.6.1 Static NOR versus n-type dynamic NOR.



4.7 Issues in Dynamic Design

Dynamic logic clearly can result in high performance solutions compared to static circuits. However, there are several important considerations that must be taken into account if one wants dynamic circuits to function properly. This include charge leakage, charge sharing, back gate (and in general capacitive) coupling, and clock feedthrough. Some of these issues are highlighted in this section.

Charge Leakage

The operation of a dynamic gate relies on the dynamic storage of the output value on a capacitor. If the pull-down network is off, the output should ideally remain at the pre-charged state of VDD during the evaluation phase. However, this charge gradually leaks away due to leakage currents, eventually resulting in a malfunctioning of the gate. Figure 4.7.1 shows the sources of leakage for the basic dynamic inverter circuit.

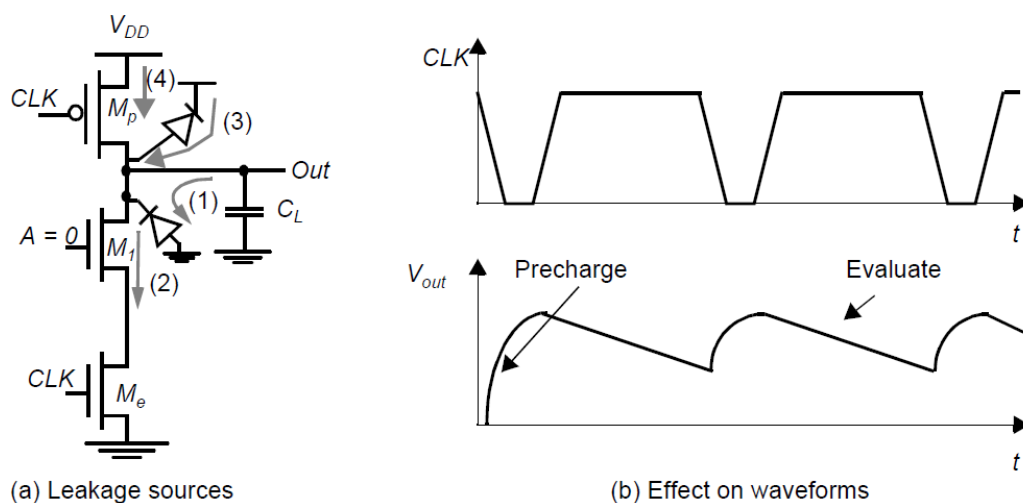


Fig 4.7.1: Leakage issues in dynamic circuits

Source 1 and 2 are the reverse-biased diode and sub-threshold leakage of the NMOS pull-down device M1, respectively. The charge stored on CL will slowly leak away due these leakage sources, assuming that the input is at zero during evaluation. Charge leakage causes a degradation in the high level (Figure 6.56b). Dynamic circuits therefore require a minimal clock rate, which is typically on the order of a few kHz. This makes the usage of dynamic techniques unattractive for low performance products such as watches, or processors that use conditional clocks (where there are no guarantees on minimum clock

rates). Note that the PMOS precharge device also contributes some leakage current due to the reverse bias diode (source 3) and the subthreshold conduction (source 4). To some extent, the leakage current of the PMOS counteracts the leakage of the pull-down path. As a result, the output voltage is going to be set by the resistive divider composed of the pull-down and pull-up paths.

Example: Leakage in dynamic circuits

Consider the simple inverter with all devices set at $0.5\mu\text{m}/0.25\mu\text{m}$. Assume that the input is low during the evaluation period. Ideally, the output should remain at the precharged state of VDD. However, as seen from Figure 4.7.2 the output voltage drops. Once the output drops below the switching threshold of the fan-out logic gate, the output is interpreted as a low voltage. Notice that the output settles to an intermediate voltage. This is due to the leakage current provided by the PMOS pull-up.

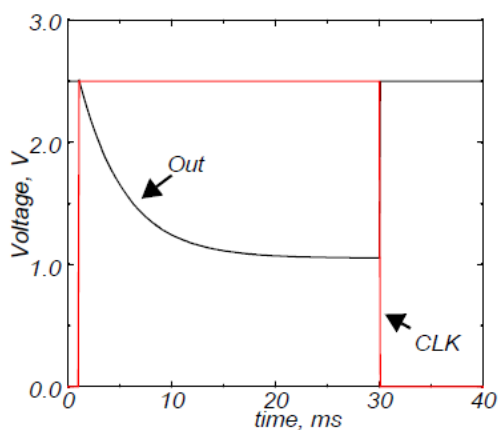


Figure 4.7.2: Impact of charge leakage. The output settles to an intermediate voltage determined by a resistive divider of the pull-down and pull up devices.

Leakage is caused by the high impedance state of the output node during the evaluate mode, when the pull down path is turned off. The leakage problem can be counteracted by reducing the output impedance on the output node during evaluation. This is often done by adding a bleeder transistor as shown in Figure 6.58a. The only function of the bleeder—a pseudo-NMOS-like pull-up device—is to compensate for the charge lost due to the pull-down leakage paths. To avoid the ratio problems associated with this style of circuit and the associated static power consumption, the bleeder resistance is made high, or,

in other words, the device is kept small. This allows the (strong) pull-down devices to lower the Out node substantially below the switching threshold of the inverter. Often, the bleeder is implemented in a feedback configuration to eliminate the static power dissipation (Figure 4.7.3).

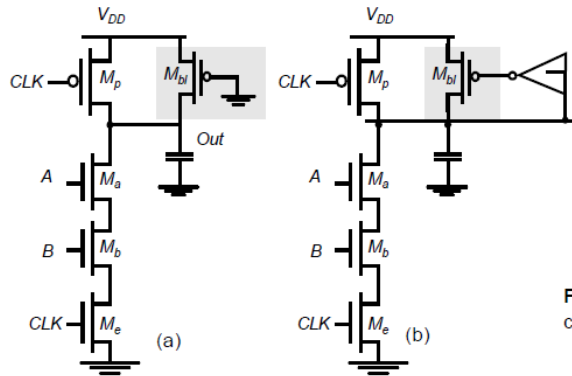


Figure 4.7.3 Static bleeders compensates for the charge-leakage.

lower the Out node substantially below the switching threshold of the inverter. Often, the bleeder is implemented in a feedback configuration to eliminate the static power dissipation (Figure 4.7.3).

Charge Sharing

Another important concern in dynamic logic is the impact of charge sharing. Consider the circuit of Figure 4.7.4. During the precharge phase, the output node is precharged to V_{DD} . Assume that all inputs are set to 0 during precharge, and that the capacitance C_a is dis-charged. Assume further that input B remains at 0 during evaluation, while input A makes a $0 \rightarrow 1$ transition, turning transistor M_a on. The charge stored originally on capacitor C_L is redistributed over C_L and C_a . This causes a drop in the output voltage, which cannot be recovered due to the dynamic nature of the circuit.

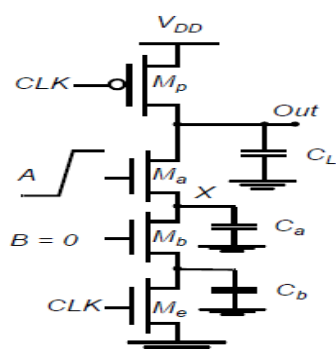


Figure 4.7.4 Charge sharing in dynamic networks.

The influence on the output voltage is readily calculated. Under the above assumptions, the following initial conditions are valid: $V_{out}(t = 0) = V_{DD}$ and $V_X(t = 0) = 0$. Two possible scenarios must be considered:

1. $\Delta V_{out} < V_{Tn}$ — In this case, the final value of V_X equals $V_{DD} - V_{Tn}(V_X)$.
ervation yields

$$C_L V_{DD} = C_L V_{out}(t) + C_a [V_{DD} - V_{Tn}(V_X)]$$

or

$$\Delta V_{out} = V_{out}(t) - V_{DD} = -\frac{C_a}{C_L} [V_{DD} - V_{Tn}(V_X)]$$

2. $\Delta V_{out} > V_{Tn}$ — V_{out} and V_X reach the same value:

$$\Delta V_{out} = -V_{DD} \left(\frac{C_a}{C_a + C_L} \right)$$

Which of the above scenarios is valid is determined by the capacitance ratio. The boundary condition between the two cases can be determined by setting ΔV_{out} equal to V_{Tn} in Eq.(6.38), yielding

$$\frac{C_a}{C_L} = \frac{V_{Tn}}{V_{DD} - V_{Tn}}$$

Overall, it is desirable to keep the value of ΔV_{out} below $|V_{Tp}|$. The output of the dynamic gate might be connected to a static inverter, in which case the low level of V_{out} would cause static power consumption. One major concern is circuit malfunction if the output voltage is brought below the switching threshold of the gate it drives.

Capacitive Coupling

The high impedance of the output node makes the circuit very sensitive to crosstalk effects. A wire routed over a dynamic node may couple capacitively and destroy the state of the floating node. Another equally important form of capacitive coupling is the back-gate (or output-to-input) coupling. Consider the circuit shown in Figure 6.62 in which a dynamic two-input NAND gate drives a static NAND gate. A transition in the input In of the static gate may cause the output of the gate ($Out2$) to go low. This output transition couples capacitively

to the other input of the gate, the dynamic node Out1, through the gate-source and gate-drain capacitances of transistor M4. A simulation of this effect is shown in Figure 6.63, and demonstrates that the output of the dynamic gate can drop significantly. This further causes the output of the static NAND gate not to drop all the way down to 0V, and a small amount of static power is dissipated. If the voltage drop is large enough, the circuit can evaluate incorrectly, and the NAND output may not go low. When designing and laying out dynamic circuits, special care is needed to minimize capacitive coupling.

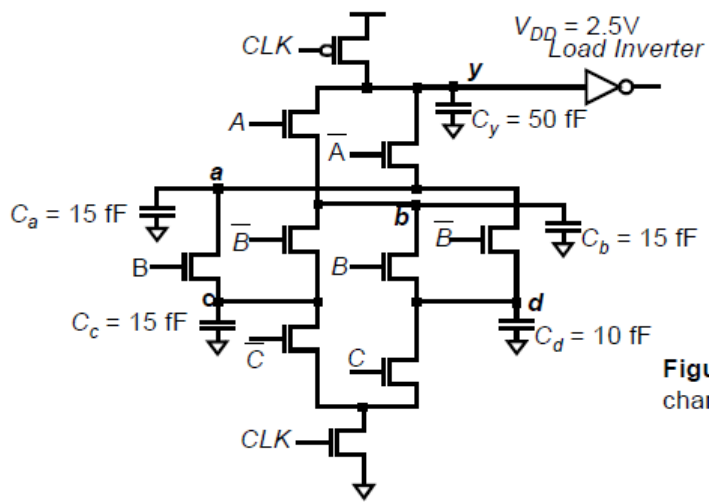


Figure 4.7.5 Example illustrating the charge sharing effect in dynamic logic.

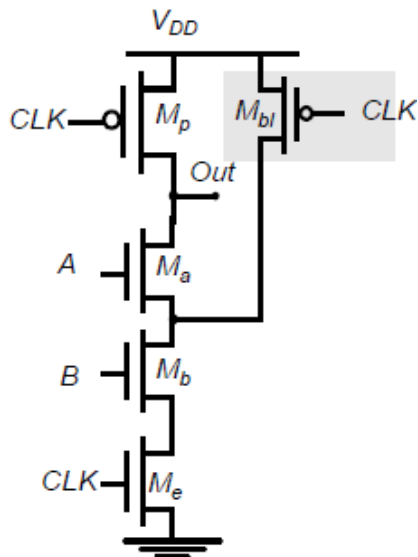


Fig 4.7.6: Dealing with charge-sharing by precharging internal nodes. An NMOS precharge transistor may also be used, but this requires an inverted clock.

operation. CMOS latchup might be another result of this injection. For all purposes, high-speed dynamic circuits should be carefully simulated to ensure that clock-feedthrough effects stay within bounds.

All the above considerations demonstrate that the design of dynamic circuits is rather tricky and requires extreme care. It should therefore only be attempted when high performance is required.

4.8 Cascading Dynamic gates

Besides the signal integrity issues, there is one major catch that complicates the design of dynamic circuits: straightforward cascading of dynamic gates to create more complex structures does not work. The problem is best illustrated with the two cascaded n-type dynamic inverters, shown in Figure 4.8.a. During the precharge phase (i.e., $CLK = 0$), the outputs of both inverters are precharged to V_{DD} . Assume that the primary input In makes a $0 \rightarrow 1$ transition (Figure 4.8.b). On the rising edge of the clock, output Out_1 starts to discharge. The second output should remain in the precharged state of V_{DD} as its expected

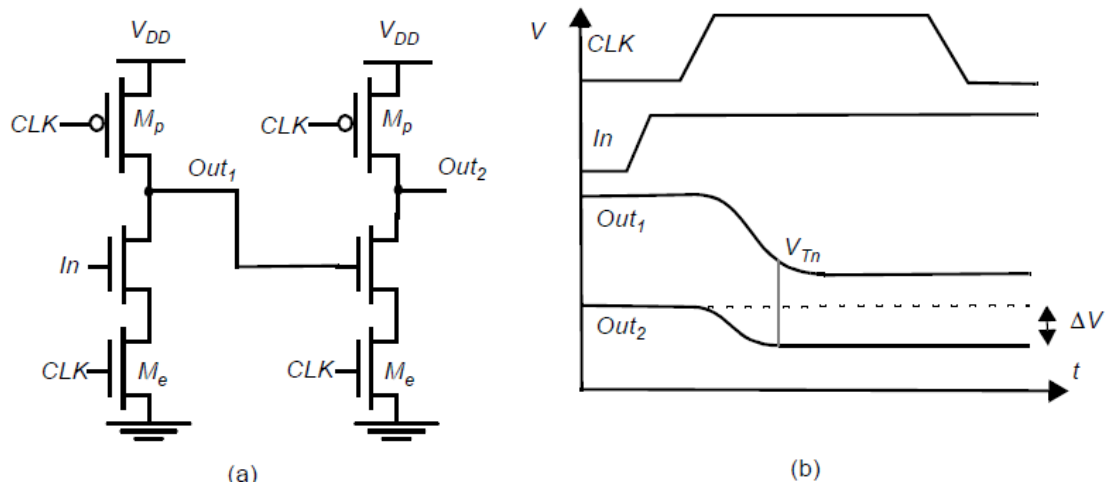


Fig 4.8: Cascade of dynamic n-type blocks.

value is 1 (Out_1 transitions to 0 during evaluation). However, there is a finite propagation delay for the input to discharge Out_1 to GND. Therefore, the second output also starts to discharge. As long as Out_1 exceeds the switching threshold of the second gate, which approximately equals V_{Tn} , a conducting path exists between Out_2 and GND, and precious charge is lost at Out_2 . The conducting path is only disabled once Out_1 reaches V_{Tn} , and turns off the

NMOS pull-down transistor. This leaves Out2 at an intermediate voltage level. The correct level will not be recovered, as dynamic gates rely on capacitive storage in contrast to static gates, which have dc restoration. The charge loss leads to reduced noise margins and potential malfunctioning.

The cascading problem arises because the outputs of each gate—and hence the inputs to the next stages—are precharged to 1. This may cause inadvertent discharge in the beginning of the evaluation cycle. Setting all the inputs to 0 during precharge addresses that concern. When doing so, all transistors in the pull-down network are turned off after precharge, and no inadvertent discharging of the storage capacitors can occur during evaluation. In other words, correct operation is guaranteed as long as the inputs can only make a single $0 \rightarrow 1$ transition during the evaluation period². Transistors are only be turned on when needed, and at most once per cycle. A number of design styles complying with this rule have been conceived. The two most important ones are discussed below.

4.9 Choosing a logic style

In the preceding sections, we have discussed several gate-implementation approaches using the CMOS technology. Each of the circuit styles has its advantages and disadvantages. Which one to select depends upon the primary requirement: ease of design, robustness, area, speed, or power dissipation. No single style optimizes all these measures at the same time. Even more, the approach of choice may vary from logic function to logic function.

The static approach has the advantage of being robust in the presence of noise. This makes the design process rather trouble-free and amenable to a high degree of automation. This ease-of-design does not come for free: for complex gates with a large fan-in, complementary CMOS becomes expensive in terms of area and performance. Alternative static logic styles have therefore been devised. Pseudo-NMOS is simple and fast at the expense of a reduced noise margin and static power dissipation. Pass-transistor logic is attractive for the implementation of a number of specific circuits, such as multiplexers and XOR-dominated logic such as adders.

Dynamic logic, on the other hand, makes it possible to implement fast and small

complex gates. This comes at a price. Parasitic effects such as charge sharing make the design process a precarious job. Charge leakage forces a periodic refresh, which puts a lower bound on the operating frequency of the circuit.

The current trend is towards an increased use of complementary static CMOS. This tendency is inspired by the increased use of design-automation tools at the logic design level. These tools emphasize optimization at the logic rather than the circuit level and put a premium on robustness. Another argument is that static CMOS is more amenable to voltage scaling than some of the other approaches discussed in this chapter.

Designing Logic for Reduced Supply Voltages

We projected that the supply voltage for CMOS processes will continue to drop over the coming decade, and may go as low as 0.6V by 2010. To maintain performance under those conditions, it is essential that the device thresholds scale as well. Figure 4.9a shows a plot of the (V_T , V_{DD}) ratio required to maintain a given performance level (assuming that other device characteristics remain identical). This trade-off is not without penalty. Reducing the threshold voltage, increases the subthreshold leakage current exponentially as we derived in Eq. (3.40) (repeated here for the sake of clarity).

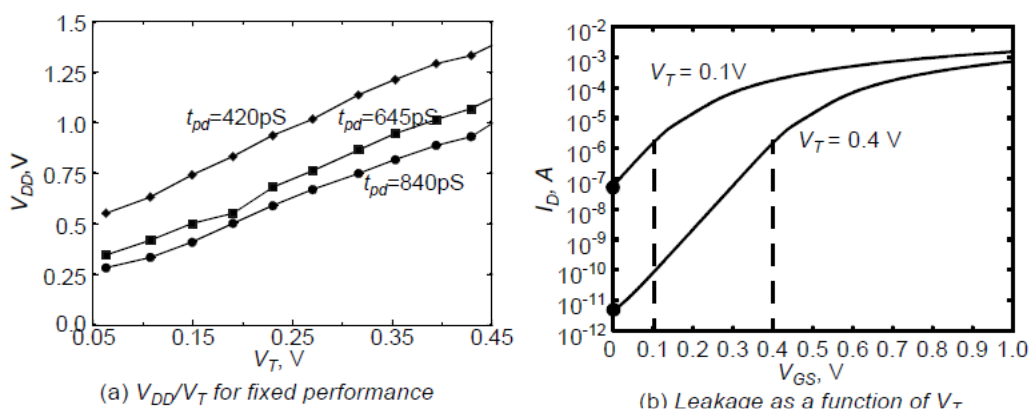


Fig 4.9.1: Voltage Scaling (V_{DD}/V_T on delay and leakage)

$$I_{leakage} = I_S 10^{\frac{V_{GS} - V_{Th}}{S}} \left(1 - 10^{-\frac{nV_{DS}}{S}} \right)$$

with S the slope factor of the device. The subthreshold leakage of an inverter is the current of the NMOS for $V_{in} = 0\text{V}$ and $V_{out} = V_{DD}$ (or the PMOS current for

$V_{in} = V_{DD}$ and $V_{out} = 0$). The exponential increase in inverter leakage for decreasing thresholds illustrated in Figure 4.9b.

These leakage currents are particularly a concern for designs that feature intermittent computational activity separated by long periods of inactivity. For example, the processor in a cellular phone remains in idle mode for a majority of the time. While the processor is shutdown mode, the system should ideally consume zero or near-zero power. This is only possible if leakage is low—this is, the devices have a high threshold voltage. This is in contradictory to the scaling scenario that we just depicted, where high performance under low supply voltage means reduced thresholds. To satisfy the contradicting requirements of high-performance during active periods, and low leakage during standby, several process modifications or leakage-control techniques have been introduced in CMOS processes. Most processes with feature sizes at and below $0.18 \mu\text{m}$ CMOS support devices with different thresholds—typically a device with low threshold for high performance circuits, and a transistor with high threshold for leakage control. Another approach that is gaining ground is the dynamic control of the threshold voltage of a device by exploiting the body effect of the transistor. To use this approach for the control of individual devices requires a dual-well process.

Clever circuit design can also help to reduce the leakage current, which is a function of the circuit topology and the value of the inputs applied to the gate. Since V_T depends on body bias (V_{BS}), the sub-threshold leakage of an MOS transistor depends not only on the gate drive (V_{GS}), but also on the body bias. In an inverter with $I_n = 0$, the sub-threshold leakage of the inverter is set by the NMOS transistor with its $V_{GS} = V_{BS} = 0$ V. In more complex CMOS gates, the leakage current depends upon the input vector. For example, the sub-threshold leakage current of a two-input NAND gate is the least when $A = B = 0$. Under these conditions, the intermediate node X settles to,

$$V_X \approx V_{th} \ln(1 + n)$$

The NAND gate sub-threshold leakage is then set by the top-most NMOS transistor with $V_{GS} = V_{BS} = -V_X$. Clearly, the sub-threshold leakage under this condition is slightly smaller than that of the inverter. This reduction in sub-

threshold leakage due to stacked transistors is called the stack effect. Figure 4.9.2 shows the leakage components for a simple two-input NAND gate.

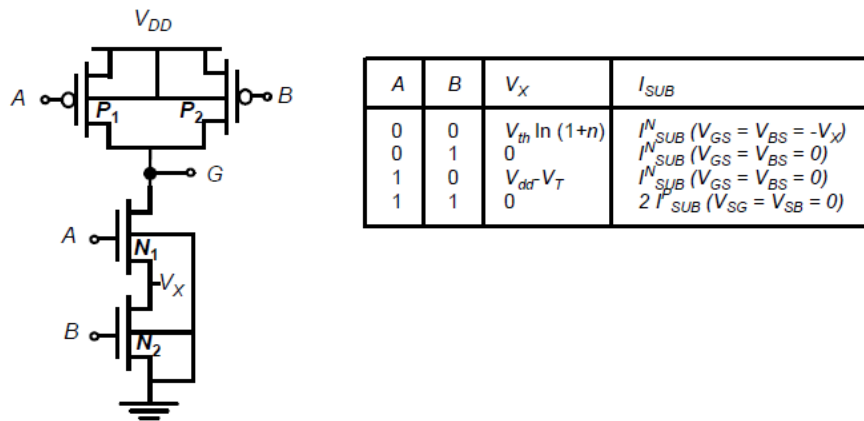


Figure 4.9.2 Sub-threshold leakage reduction due to stack effect in a two-input NAND gate.

In short-channel MOS transistors, the sub-threshold leakage current depends not only on the gate drive (V_{GS}) and the body bias (V_{BS}), but also depends on the drain voltage (V_{DS}). The threshold voltage of a short-channel MOS transistor decreases with increasing V_{DS} due to drain induced barrier lowering (DIBL). Typical value for DIBL can range from 20-150 mV change in V_T per volt change in V_{DS} . Figure 4.9.2 illustrates the impact on the sub-threshold leakage as a result of

- a decrease in gate drive—point A to B
- an increase in body bias—point A to C
- an increase in drain voltage—point A to D.

Because of the latter, the impact of the stack effect is even more significant for short-channel transistors. The intermediate voltage reduces the drain-source voltage of the top-most device, and hence reduces its leakage.

In summary, the sub-threshold leakage in complex stacked circuits can be significantly lower than in individual devices. Observe that the maximum leakage reduction occurs when all the transistors in the stack are off, and the intermediate node voltage reaches its steady state value. Exploiting this effect requires a careful selection of the input signals to every gate during standby or sleep mode.

4.10 GATE DESIGN IN ULTRA DEEP SUBMICRON ERA

A submicron core process flow, that is, a common skeleton to which different technologies could be grafted was agreed and a common set of design rules was developed. To investigate the possible evolution toward a 0.5 micron process, special advanced rules were developed.

The deep sub micron -scale (below 100 nm) represents the current state of the art for the chips at the heart of the world's computers, cellphones and electronics.

Modern and future ultra-deep-submicron (UDSM) technologies introduce several new problems in analog design. Nonlinear output conductance in combination with reduced voltage gain pose limits in linearity of (feedback) circuits. Gate-leakage mismatch exceeds conventional matching tolerances. Increasing area does not improve matching any more, except if higher power consumption is accepted or if active cancellation techniques are used. Another issue is the drop in supply voltages. Operating critical parts at higher supply voltages by exploiting combinations of thin- and thick-oxide transistors can solve this problem. Composite transistors are presented to solve this problem in a practical way. Practical rules of thumb based on measurements are derived for the above phenomena.

Modern and future ultra-deep-submicron (UDSM) technologies introduce several new problems in analog design. Nonlinear output conductance in combination with reduced voltage gain pose limits in linearity of (feedback) circuits. Gate-leakage mismatch exceeds conventional matching tolerances. Increasing area does not improve matching any more, except. Applications typically apply transistors biased at low and moderate gate-overdrive voltages. The corresponding of a technology is then inside a relatively small frequency band. Fig. 4.10.1

shows such bands for four technologies as derived from measurements. This figure clearly illustrates that signal frequencies for which the input impedance appears to be resistive change from roughly 0.1 Hz in 180-nm technologies to about 1 MHz in 65-nm .

Deep-submicron technology allows billions of transistors on a single die, potentially running at gigahertz frequencies. According to Semiconductor Industry Association (SIA) projections, the number of transistors per chip and the local clock frequencies for high-performance microprocessors will continue to grow exponentially in the near future. This ensures that future microprocessors will become ever more complex. However, physical and program behavioral constraints will limit the usefulness of this complexity. Physical constraints include interconnect and device limits, as well as practical limits on power and cost. Program behavioral constraints result from program control and data dependencies, and from unpredictable events during execution. Other challenges include the need for advanced CAD tools to combat the negative effect of greater complexity on design time. Designers will also have to make improvements to preserve computational integrity, reliability, and diagnostic features. Successful implementations will depend on the processor architect's ability to foresee technology trends and understand the changing design trade-offs for specific applications, beginning with the differing requirements for client versus server processors. This article discusses these trade-offs in light of industry projections and the many considerations affecting deep submicron technology.

1. Ultra-deep submicron technology

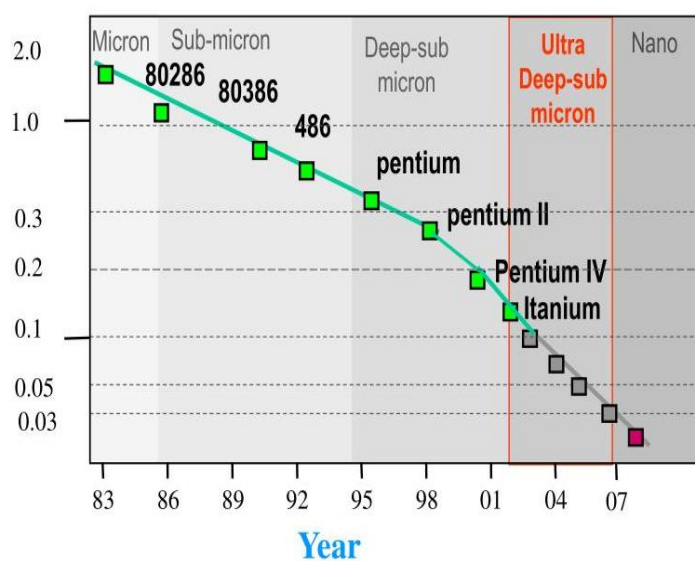


Figure 4.10.1 ultra-deep submicron technology

APPLICATIONS OF ULTRA DEEP SUBMICRON

- Health,
- nutritional,
- cosmetic
- pharmaceutical applications.

4.11 LATCH VS REGISTER

Pipelining is a popular design technique often used to accelerate the operation of the data-paths in digital processors. The idea is easily explained with the example of Figure 7.40a. The goal of the presented circuit is to compute $\log(|a - b|)$, where both a and b represent streams of numbers, that is, the computation must be performed on a large set of input value's. The minimal clock period T_{min} necessary to ensure correct evaluation is given as:

$$T_{min} = t_{c-q} + t_{pd,logic} + t_{su}$$

where t_{c-q} and t_{su} are the propagation delay and the set-up time of the register, respectively. We assume that the registers are edge-triggered D registers. The term $t_{pd,logic}$ stands for the worst-case delay path through the combinational network, which consists of the adder, absolute value, and logarithm functions. In conventional systems (that don't push the edge of technology), the latter delay is generally much larger than the delays associated with the registers and dominates the circuit performance. Assume that each logic module has an equal propagation delay. We note that each logic module is then active for only 1/3 of the clock period (if the delay of the register is ignored).

For example, the adder unit is active during the first third of the period and remains idle—this is, it does no useful computation—during the other 2/3 of the period. Pipelining is a technique to improve the resource utilization, and increase the functional throughput. Assume that we introduce registers between the logic blocks, as shown in Figure 7.40b. This causes the computation for one set of input data to spread over a number of clock periods, as shown in Table 7.1.

The result for the data set (a1, b1) only appears at the output after three clock-periods. At that time, the circuit has already performed parts of the computations for the next data sets, (a2, b2) and (a3,b3). The computation is performed in an assembly-line fashion, hence the name pipeline.

Table: Example of pipelined computations.

Clock Period	Adder	Absolute Value	Logarithm
1	$a_1 + b_1$		
2	$a_2 + b_2$	$ a_1 + b_1 $	
3	$a_3 + b_3$	$ a_2 + b_2 $	$\log(a_1 + b_1)$
4	$a_4 + b_4$	$ a_3 + b_3 $	$\log(a_2 + b_2)$
5	$a_5 + b_5$	$ a_4 + b_4 $	$\log(a_3 + b_3)$

The advantage of pipelined operation becomes apparent when examining the mini-mum clock period of the modified circuit. The combinational circuit block has been partitioned into three sections, each of which has a smaller propagation delay than the original function. This effectively reduces the value of the minimum allowable clock period:

$$T_{min,pipe} = t_{c-q} + \max(t_{pd,add}, t_{pd,abs}, t_{pd,log})$$

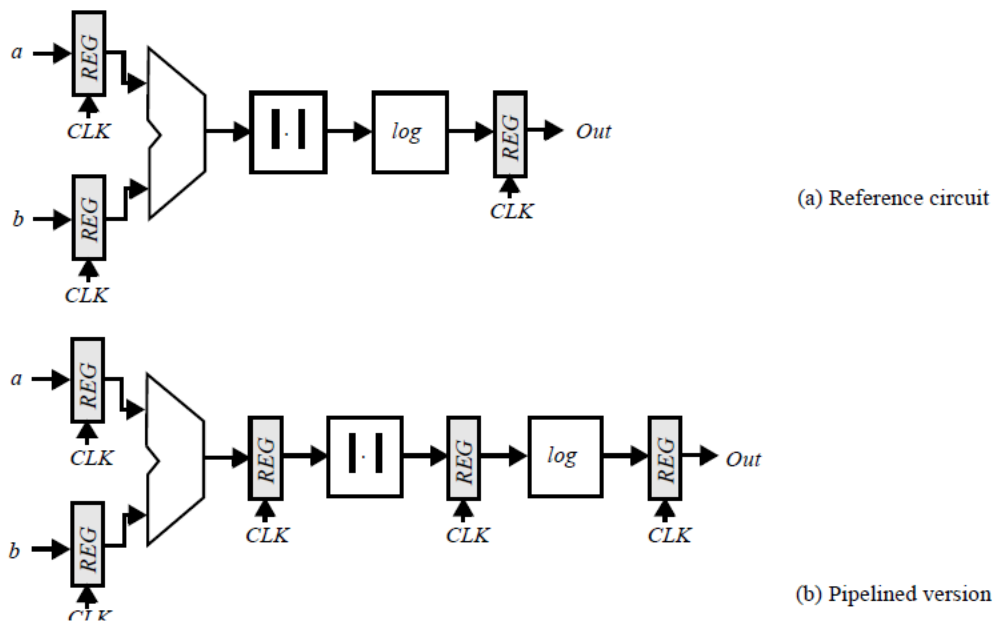


Figure 4.11.1 Datapath for the computation of $\log(|a + b|)$.

Suppose that all logic blocks have approximately the same propagation delay, and that the register overhead is small with respect to the logic delays. The pipelined network outperforms the original circuit by a factor of three under these assumptions, or $T_{min,pipe} = T_{min}/3$. The increased performance comes at

the relatively small cost of two additional registers, and an increased latency.² This explains why pipelining is popular in the implementation of very high-performance data paths.

Latch- vs. Register-Based Pipelines

Pipelined circuits can be constructed using level-sensitive latches instead of edge-triggered registers. Consider the pipelined circuit of Figure 7.41. The pipeline system is implemented based on pass-transistor-based positive and negative latches instead of edge-triggered registers. That is, logic is introduced between the master and slave latches of a master-slave system. In the following discussion, we use without loss of generality the CLK-CLK notation to denote a two-phase clock system.

Latch-based systems give significantly more flexibility in implementing a pipelined system, and often offers higher performance. When the clocks CLK and \overline{CLK} are non-overlapping, correct pipeline operation is obtained. Input data is sampled on C1 at the negative edge of CLK and the computation of logic block F starts; the result of the logic block F is stored on C2 on the falling edge of CLK, and the computation of logic block G starts.

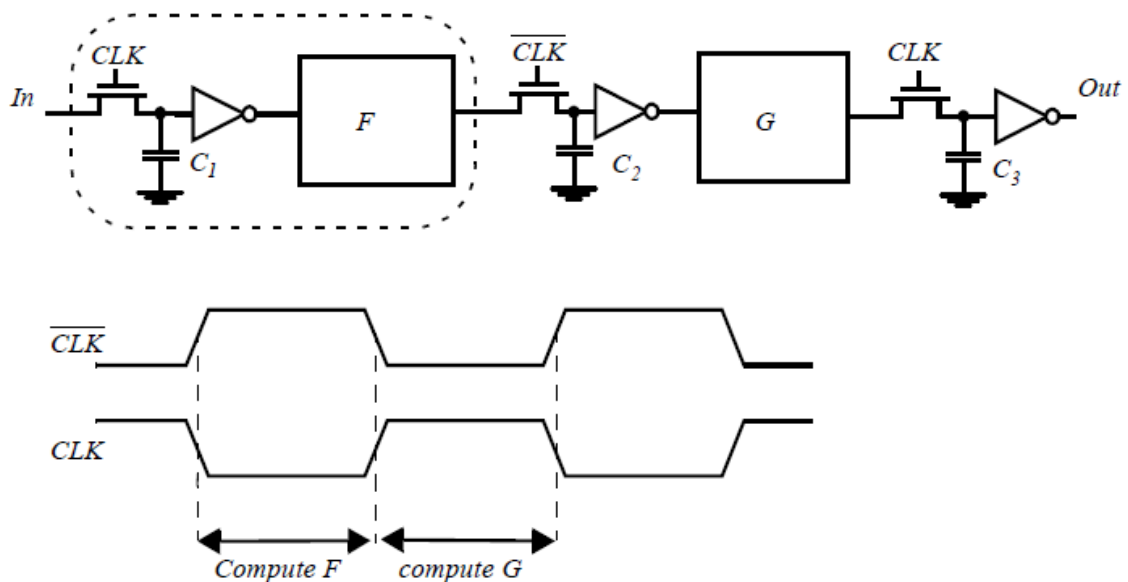


Figure 4.11.2 Operation of two-phase pipelined circuit using dynamic registers.

The non-overlapping of the clocks ensures correct operation. The value stored on C2 at the end of the CLK low phase is the result of passing the previous input (stored on the falling edge of CLK on C1) through the logic function F. When

overlap exists between CLK and CLK, the next input is already being applied to F, and its effect might propagate to C2 before CLK goes low (assuming that the contamination delay of F is small). In other words, a race develops between the previous input and the current one. Which value wins depends upon the logic function F, the overlap time, and the value of the inputs since the propagation delay is often a function of the applied inputs. The latter factor makes the detection and elimination of race conditions non-trivial.

Latency is defined here as the number of clock cycles it takes for the data to propagate from the input to the output. For the example at hand, pipelining increases the latency from 1 to 3. An increased latency is in general acceptable, but can cause a global performance degradation if not treated with care.

4.12 Latch based design (clock)

While the use of registers in a sequential circuit enables a robust design methodology, there are significant performance advantages to using a latch-based design in which combinational logic is separated by transparent latches. In an edge-triggered system, the worst-case logic path between two registers determines the minimum clock period for the entire system. If a logic block finishes before the clock period, it has to idle till the next input is latched in on the next system clock edge. The use of a latch-based methodology (as illustrated in Figure 10.26) enables more flexible timing, allowing one stage to pass slack to or steal time from following stages. This flexibility, allows an overall performance increase. Note that the latch-based methodology is nothing more than adding logic between latches of a master-slave flip-flop.

For the latch-based system in Figure 10.26, assume a two-phase clocking scheme. Assume furthermore that the clock are ideal, and that the two clocks are inverted versions of each other (for sake of simplicity). In this configuration, a stable input is available to the combinational logic block A (CLB_A) on the falling edge of CLK1 (at edge 2) and it has a maximum time equal to the $T_{CLK}/2$ to evaluate (that is, the entire low phase of CLK1). On the falling edge of CLK2 (at edge 3), the output CLB_A is latched and the computation of CLK_B is launched. CLB_B computes on the low phase of CLK2 and the output is

available on the falling edge of CLK1 (at edge 4).

This timing appears equivalent to having an edge-triggered system where CLB_A and CLB_B are cascaded and between two edge-triggered registers (Figure 10.27). In both cases, it appears that the time available to perform the combination of CLB_A and CLB_B is TCLK.

However, there is an important performance related difference. In a latch-based system, since the logic is separated by level sensitive latches, it is possible for a logic block to utilize time that is left over from the previous logic block and this is referred to as slack borrowing [Bernstein00]. This approach requires no explicit design changes, as the passing of slack from one block to the next is automatic. The key advantage of slack borrowing is that it allows logic between cycle boundaries to use more than one clock cycle while satisfying the cycle time constraint. Stated in another way, if the sequential system works at a particular clock rate and the total logic delay for a complete cycle is larger than the clock period, then unused time or slack has been implicitly borrowed from preceding stages. This implies that the clock rate can be higher than the worst-case critical path!

Slack passing happens due to the level sensitive nature of latches. In Figure 10.26, the input to CLB_A should be valid by the falling edge of CLK1 (edge 2). What happens if the combinational logic block of the previous stage finishes early and has a valid input data for CLB_A before edge 2? Since a latch is transparent during the entire high phase of the clock, as soon as the previous stage has finished computing, the input data for CLB_A is valid. This implies that the maximum time available for CLB_A is its phase time (i.e., the low phase of CLK1) and any left-over time from the previous computation. Formally stated, slack passing has taken place if $TCLK < t_{pd, A} + t_{pd, B}$ and the logic functions correctly (for simplicity, the delay associated with latches are ignored).

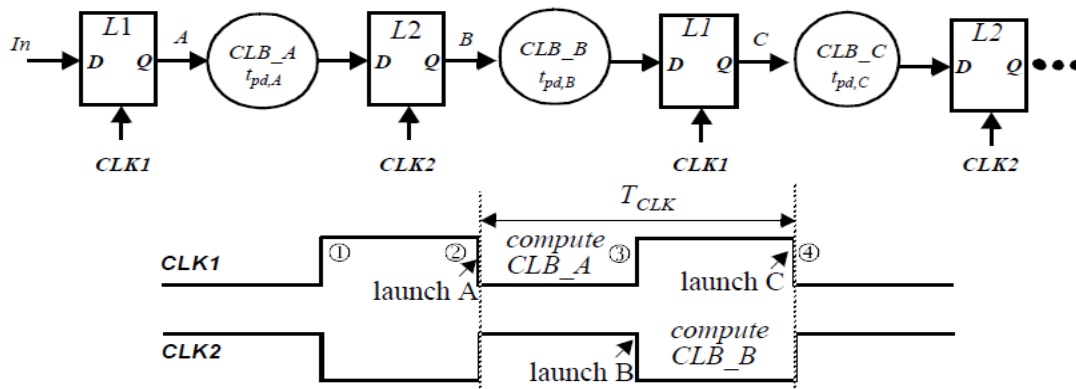


Figure 4.12.1 Latch-based design- transparent latches are separated by combinational logic

However, there is an important performance related difference. In a latch based system, since the logic is separated by level sensitive latches, it is possible for a logic block to utilize time that is left over from the previous logic block and this is referred to as slack borrowing [Bernstein00]. This approach requires no explicit design changes, as the passing of slack from one block to the next is automatic. The key advantage of slack borrowing is that it allows logic between cycle boundaries to use more than one clock cycle while satisfying the cycle time constraint. Stated in another way, if the sequential system works at a particular clock rate and the total logic delay for a complete cycle is larger than the clock period, then unused time or slack has been implicitly borrowed from preceding stages. This implies that the clock rate can be higher than the worst-case critical path!

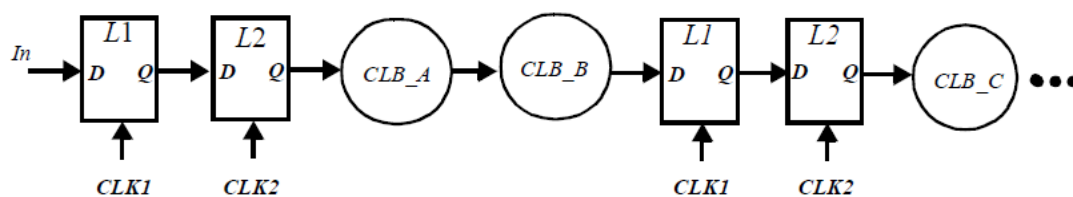


Figure 4.12.2 Edge-triggered pipeline (back-to-back latches for edge-triggered registers) of the logic in Figure 4.12.1

4.13 Timing decimation

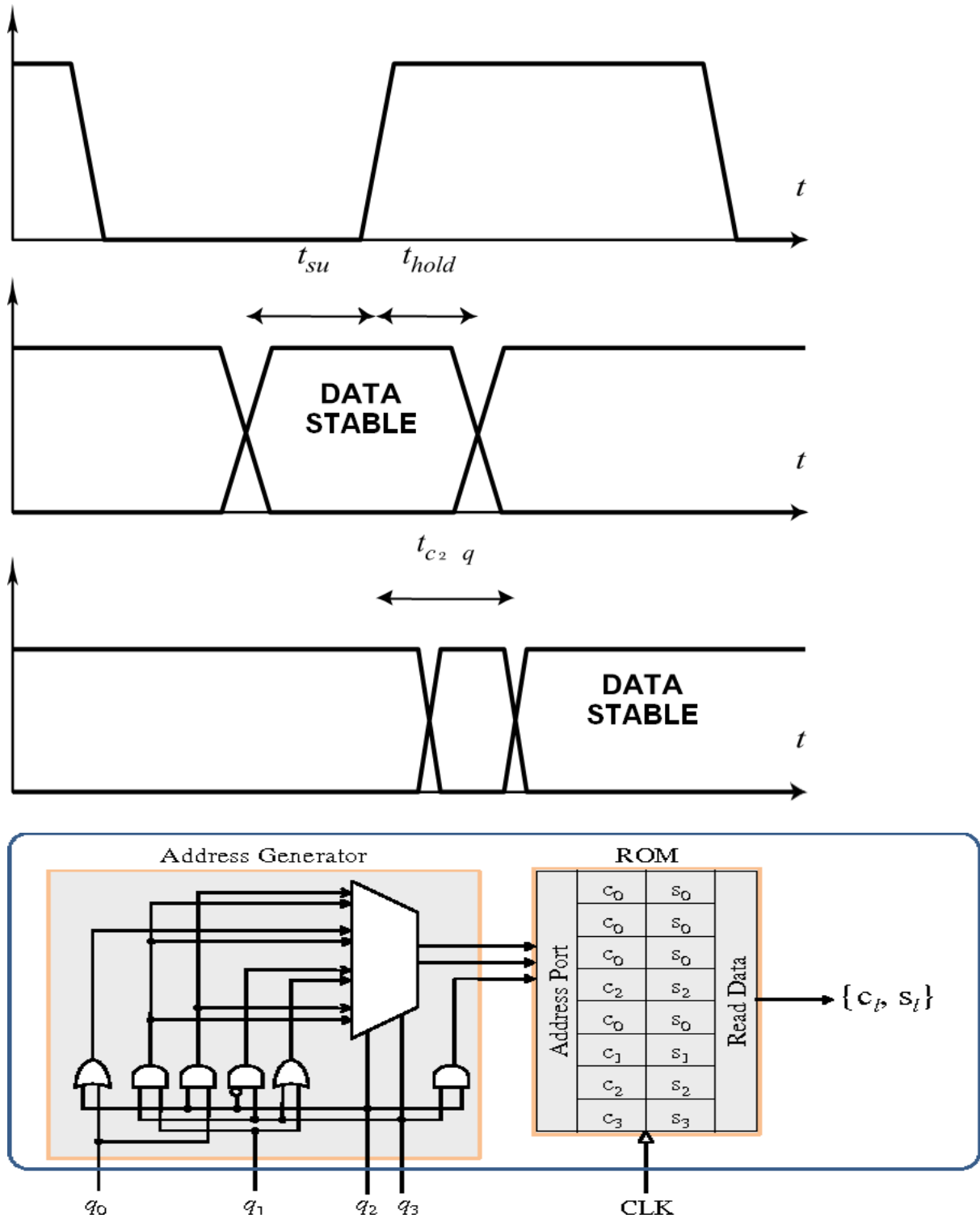


Figure 4.13.1 Timing decimation diagram

The decimation-in-time (DIT) fast Fourier transform (FFT) very often has advantage over the decimation-in-frequency (DIF) FFT for most real-valued applications, like speech/image/video processing, biomedical signal processing, and time-series analysis, etc

Digital audio application such HiFi CD and DAT systems often use sigma delta A/D converters. The quality of sigma delta modulator is recognized by the order and its resolution. In this project the sampling frequency represent as 6.144 MHz with the over sampling ration of 128. Nyquist frequency is selected to be 48kHz to support Bf equal to 24 kHz with the frequency response ripple less than 0.0002 db. Figure 1 shows 3rd order sigma delta modulator with multirate decimation filter. A multirate decimation filter system was chosen to realize the needed performance. The filter system was thus organized as an initial filter stage having a 16:1 decimation ratio followed by a third stage having an 8:1 decimation ratio.

Digital Decimation Process:

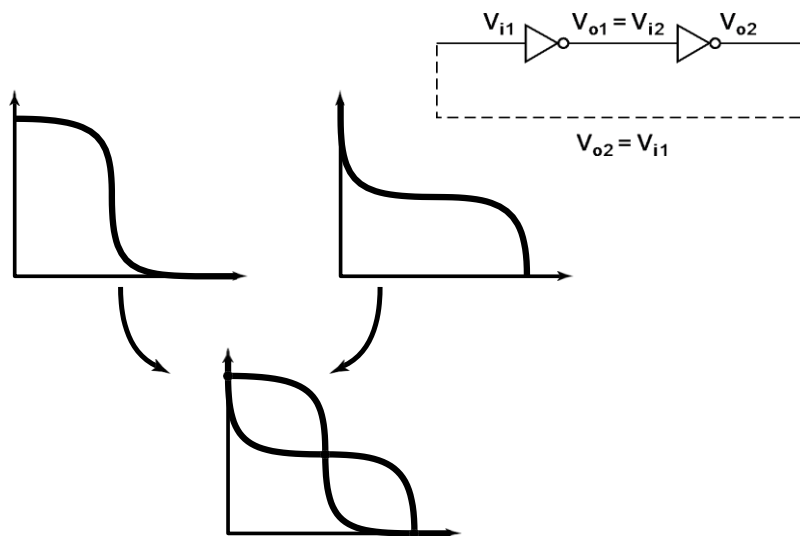
CIC filter is located after sigma delta modulator and decimate the frequency by the ratio of 16. The packing of modulator and CIC filter minimized the noise by decrease the number of parallel pad drivers. Then CIC filter increase the sigma delta resolution to improve Signal to Noise ratio. The two half band filters (Brandt & Wooley 1994) are used to reduce remain sampling rate reduction to the Nyquist output rate. First half band filter and second half band filter make the frequency response flatter and sharper similar to ideal filter specially second half band filter due to higher order of the filter ($R=40$), has most effect to make the frequency response sharp. All the even coefficients of half band filters are zero exception last one which is 0.5. This particular make them efficient for 2:1 decimation ratio and reduce the computational complexity by near 50% as compared to general direct form filter Architecture. Droop correction filter is allocated to compensate pass band attenuation which is created by CIC filter.

Frequency response of decimation system Similar Decimation process has been done high speed Cascaded Integrator Comb filter is designed and implemented to accomplish decimation task, remove quantization noise and avoid aliasing to the signal.

4.14 Positive feedback

Bistability is required for an effective binary switch Now imagine a version of our model signaling system in which ... The impact of the positive feedback will vary a great deal depending on the relative rates of the various binding, ...

Their model described the bistability in extrinsic apoptosis within the context of c as permeated positive feedback.



4.15 Instability

When an electric field is applied across a gate oxide, dangling bonds called *traps* develop at the Si-SiO₂ interface. The threshold voltage increases as *m* traps form, reducing the drive current until the circuit fails. The process is most pronounced for pMOS transistors with a strong negative bias (i.e., a gate voltage of 0 and source voltage of *VDD*) at elevated temperature. It has become the most important oxide wear out mechanism for many nano meter processes. When a field $E_{ox} = VDD/t_{ox}$ is applied for time *t*, the threshold voltage shift can be modelled as

$$\Delta V_t = k \frac{E_{ox}}{E_0} t^{0.25}$$

The high stress during burn-in can lock in most of the threshold voltage shift expected from NBTI; this is good because it allows testing with full NBTI degradation. During design, a chip should be simulated under the worst-case NBTI shift expected over its lifetime.

4.16 Metastability

A latch is a bistable device; i.e., it has two stable states (0 and 1). Under the right conditions, that latch can enter a metastable state in which the output is at an indeterminate level between 0 and 1. For example, Figure 10.42 shows a simple model for a static latch consisting of two switches (probably transmission gates in practice) and two inverters. While the latch is transparent, the sample switch is closed and the hold switch open (Figure 10.42(a)). When the latch goes opaque, the sample switch opens and the hold switch closes (Figure 10.42(b)). Figure 10.42(c) shows the DC transfer characteristics of the two inverters. Because $A = B$ when the latch is opaque, the stable states are $A = B = 0$ and $A = B = V_{DD}$. The metastable state is $A = B = V_m$, where V_m is an invalid logic level.

This point is called metastable because the voltages are self-consistent and can remain there indefinitely. However, any noise or other disturbance will cause A and B to switch to one of the two stable states. Figure 10.42(d) shows an analogy of a ball delicately balanced on a hill. The top of the hill is a metastable state. Any disturbance will cause the ball to roll down to one of the two stable states on the left or right side of the hill.

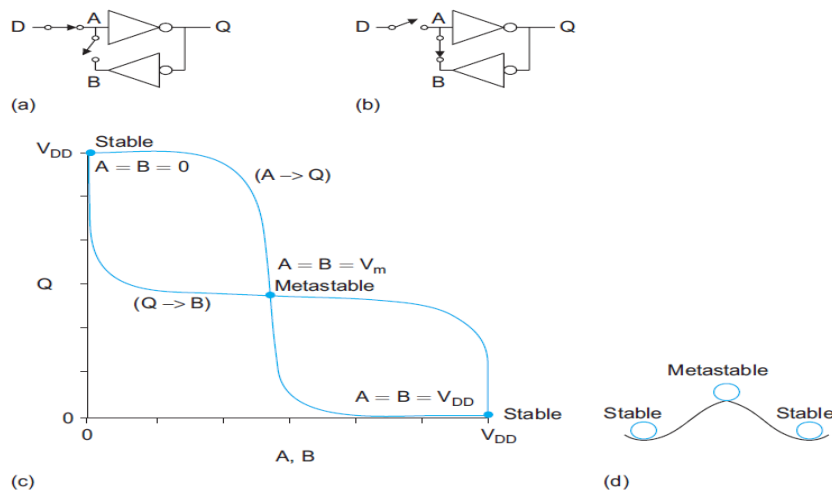


Figure 4.16.1 Metastable state in static latch

Figure 10.43(a) plots the output of the latch from Figure 10.17(g) as the data transitions near the falling clock edge. If the data changes at just the wrong time t_m within the aperture, the output can remain at the metastable point for some time before settling to a valid logic level. Figure 10.43(b) plots t_{DQ} vs. t_{DC}

– t_m on a semilogarithmic scale for a rising input and output. The delay is less than or equal to t_{pdq} for inputs that meet the setup time and increases for inputs that arrive too close to t_m . The points marked on the graph will be used in the example at the end of this section.

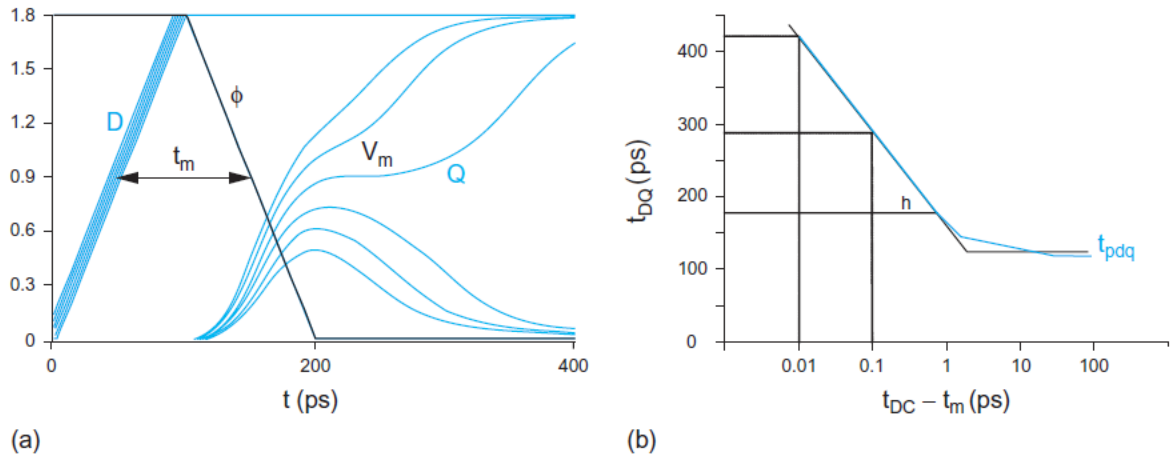


Figure 4.16.2 Metastable transients and propagation delay

The cross-coupled inverters behave like a linear amplifier with gain G when A is near the metastable voltage V_m . The inverter delay can be modeled with an output resistance R and load capacitance C . We can predict the behavior in metastability by assuming that the initial voltage on node A when the latch becomes opaque at time $t = 0$ is

$$A(0) = V_m + a(0)$$

where $a(0)$ is a small signal offset from the metastable point. Figure 10.44 shows a small-signal model for $a(t)$. The behavior after time 0 is given by the first-order differential equation

$$\frac{Ga(t) - a(t)}{R} = C \frac{da(t)}{dt}$$

Solving this equation shows that the positive feedback drives $a(t)$ exponentially away from the metastable point with a time constant determined by the gain and RC delay of the cross-coupled inverter loop.

$$a(t) = a(0)e^{\frac{t}{\tau_s}}; \tau_s = \frac{RC}{G-1}$$

Suppose the node is defined to reach a legal logic level when $|a(t)|$ exceeds some deviation ΔV . The time to reach this level is

$$t_{DQ} = \tau_s [\ln \Delta V - \ln a(0)]$$

This shows that the latch propagation delay increases as $A(0)$ approaches the metastable point and $a(0)$ approaches 0. The delay approaches infinity if $a(0)$ is precisely 0, but this can never physically happen because of noise. However, there is no upper bound on the possible waiting time t required for the signal to become valid. If the input $A(t)$ is a ramp that passes through V_m at time t_m , $a(0)$ is proportional to $t_{DC} - t_m$. Observe that EQ (10.24) is a good fit to the log-linear portion of Figure 10.43(b). The time constant τ_s is essentially the reciprocal of the gain-bandwidth product [Flannagan85]. Therefore, the feedback loop in a latch should have a high gain-bandwidth product to resolve from meta-stability quickly.

Designers need to know the probability that latch propagation delay exceeds some time t' . Longer propagation delays are less likely because they require $a(0)$ to be closer to 0. This probability should decrease with the clock period T_c because a uniformly distributed input change is less likely to occur near the critical time. Projecting through EQ (10.24) shows that it should also decrease exponentially with waiting time t' . Theoretical and experimental studies [Chaney83, Veendrick80, Horstmann89] find that the probability can be expressed as

$$P(t_{DQ} > t') = \frac{T_0}{T_c} e^{-\frac{t'}{\tau_s}} \text{ for } t' > h$$

where T_0 and τ_s can be extracted through simulation [Baghini02] or measurement. Intuitively, T_0/T_c describes the probability that the input would change during the aperture, causing metastability, and the exponential term describes the probability that the output has not resolved after t' if it did enter metastability. The model is only valid for sufficiently long propagation delays (h significantly greater than t_{pdq}).

We have seen that a good synchronizer latch should have a feedback loop

with a high-gain-bandwidth product. Conventional latches have data and clock transistors in series, increasing the delay (i.e., reducing the bandwidth). Figure 10.45 shows a synchronizer flip-flop in which the feedback loops simplify to cross-coupled inverter pairs [Dike99]. Furthermore, the flip-flop is reset to 0, and then is only set to 1 if $D = 1$ to minimize loading on the feedback loop.

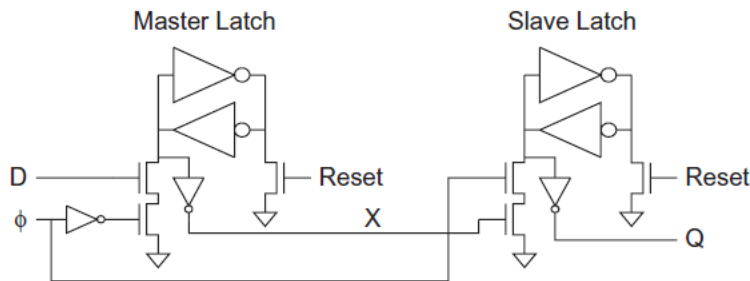


Figure 4.16.3 Fast synchronizer flip-flop

The flip-flop consists of master and slave jamb latches. Each latch is reset to 0 while $D = 0$. When D rises before ϕ , the master output X is driven high. This in turn drives the slave output Q high when ϕ rises. The pulldown transistors are just large enough to over-power the cross-coupled inverters, but should add as little stray capacitance to the feedback loops as possible. X and Q are buffered with small inverters so they do not load the feedback loops.

4.17 Multiplexer based latches

There are many approaches for constructing latches. One very common technique involves the use of transmission gate multiplexers. Multiplexer based latches can provide similar functionality to the SR latch, but has the important added advantage that the sizing of devices only affects performance and is not critical to the functionality.

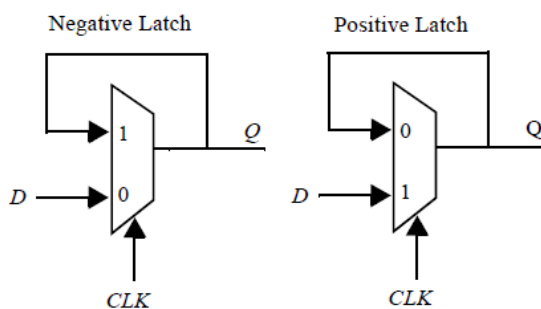


Figure 4.17.1 Negative and positive latches based on multiplexers.

Figure 7.11 shows an implementation of static positive and negative latches based on multiplexers. For a negative latch, when the clock signal is low, the input 0 of the multiplexer is selected, and the D input is passed to the output. When the clock signal is high, the input 1 of the multiplexer, which connects to the output of the latch, is selected. The feedback holds the output stable while the clock signal is high. Similarly in the positive latch, the D input is selected when clock is high, and the output is held (using feedback) when clock is low.

A transistor level implementation of a positive latch based on multiplexers is shown in Figure 7.12. When CLK is high, the bottom transmission gate is on and the latch is transparent - that is, the D input is copied to the Q output. During this phase, the feedback loop is open since the top transmission gate is off. Unlike the SR FF, the feedback does not have to be overridden to write the memory and hence sizing of transistors is not critical for realizing correct functionality. The number of transistors that the clock touches is important since it has an activity factor of 1. This particular latch implementation is not particularly efficient from this metric as it presents a load of 4 transistors to the CLK signal.

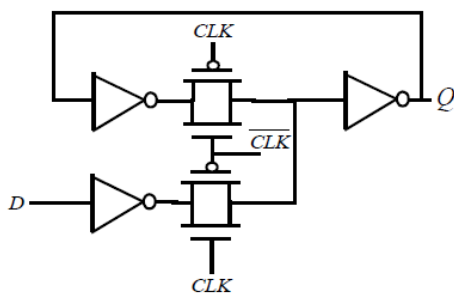


Figure 4.17.2 Positive latch built using transmission gates.

It is possible to reduce the clock load to two transistors by using implement multiplexers using NMOS only pass transistor as shown in Figure 7.13. The advantage of this approach is the reduced clock load of only two NMOS devices. When CLK is high, the latch samples the D input, while a low clock-signal enables the feedback-loop, and puts the latch in the hold mode. While attractive for its simplicity, the use of NMOS only pass transistors results in the passing of a degraded high voltage of $V_{DD} - V_{Tn}$ to the input of the first inverter. This impacts both noise margin and the switching performance, especially in the case of low values of V_{DD} and high values of V_{Tn} . It also causes static power dissipation in first inverter, as already pointed out in Chapter 6. Since the

maximum input-voltage to the inverter equals $V_{DD}-V_{Tn}$, the PMOS device of the inverter is never turned off, resulting in a static current flow.

4.18 Master-Slave Based Edge Triggered Register

The most common approach for constructing an edge-triggered register is to use a master-slave configuration, as shown in Figure 7.14. The register consists of cascading a negative latch (master stage) with a positive latch (slave stage). A multiplexer-based latch is used in this particular implementation, although any latch could be used. On the low phase of the clock, the master stage is transparent, and the D input is passed to the master stage output, Q_M . During this period, the slave stage is in the hold mode, keeping its previous value using feedback. On the rising edge of the clock, the master slave stops sampling the input, and the slave stage starts sampling. During the high phase of the clock, the slave stage samples the output of the master stage (Q_M), while the master stage remains in a hold mode. Since Q_M is constant during the high phase of the clock, the output Q makes only one transition per cycle. The value of Q is the value of D right before the rising edge of the clock, achieving the positive edge-triggered effect. A negative edge-triggered register can be constructed using the same principle by simply switching the order of the positive and negative latch (this is, placing the positive latch first).

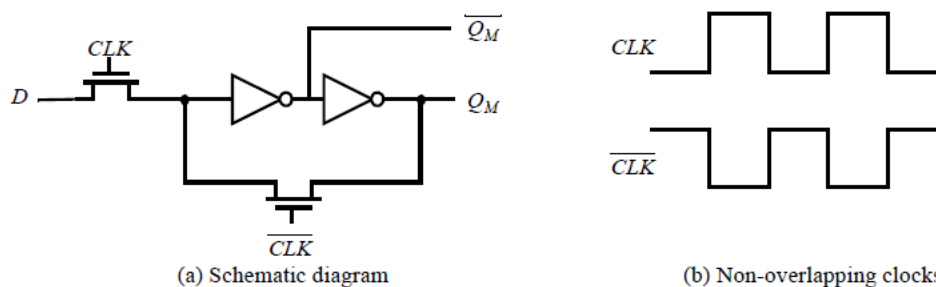


Figure 4.18.1 Multiplexer-based NMOS latch using NMOS-only pass transistors.

A complete transistor-level implementation of the master-slave positive edge-triggered register is shown in Figure 7.15. The multiplexer is implemented using transmission gates as discussed in the previous section. When the clock is low ($CLK = 1$), T_1 is on and T_2 is off, and the D input is sampled onto node Q_M . During this period, T_3 is off and T_4 is on and the cross-coupled inverters (I_5, I_6) holds the state of the slave latch. When the clock goes high, the master stage stops sampling the input and goes into a hold mode. T_1 is off and T_2 is on, and

the cross coupled inverters I3 and I4 holds the state of QM. Also, T3 is on and T4 is off, and QM is copied to the output Q.

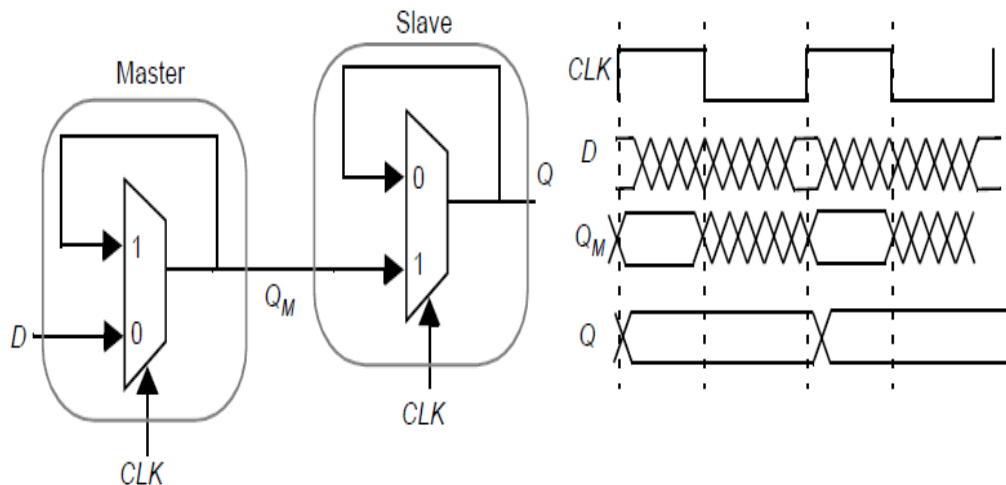


Figure 4.18.2 Multiplexer-based NMOS latch using NMOS-only pass transistors.

Timing Properties of Multiplexer-based Master-Slave Registers

Registers are characterized by three important timing parameters: the set-up time, the hold time and the propagation delay. It is important to understand the factors that affect these timing parameters, and develop the intuition to manually estimate them. Assume that the propagation delay of each inverter is t_{pd_inv} , and the propagation delay of the transmission gate is t_{pd_tx} . Also assume that the contamination delay is 0 and the inverter delay to derive CLK from CLK has a delay equal to 0.

The set-up time is the time before the rising edge of the clock that the input data D must become valid. Another way to ask the question is how long before the rising edge does the D input have to be stable such that QM samples the value reliably. For the transmission gate multiplexer-based register, the input D has to propagate through I1, T1, I3 and I2 before the rising edge of the clock. This is to ensure that the node voltages on both terminals of the transmission gate T2 are at the same value. Otherwise, it is possible for the cross-coupled pair I2 and I3 to settle to an incorrect value. The set-up time is therefore equal to $3 * t_{pd_inv} + t_{pd_tx}$.

The propagation delay is the time for the value of QM to propagate to the output Q. Note that since we included the delay of I2 in the set-up time, the output of I4 is valid before the rising edge of clock. Therefore the delay t_{c-q} is

simply the delay through T3 and I6 ($t_{c-q} = t_{pd_tx} + t_{pd_inv}$).

The hold time represents the time that the input must be held stable after the rising edge of the clock. In this case, the transmission gate T1 turns off when clock goes high and therefore any changes in the D-input after clock going high are not seen by the input. Therefore, the hold time is 0.

4.19 Clock to q delay

The time required, to propagate is 1 transmission gate delay + 1 inverter delay
Clk-Q delay = 1 transmission gate delay + 1 inverter delay
Hold Time is the time for which 'D' input remain valid after clock edge. In this case, 'Tr1' is OFF after rising 'CLK'.

An edge-triggered flip-flop, the clock-to-Q time is the time it takes for the register output to be in a stable state after a clock edge occurs. One nuance: suppose the clock changes at $t=0$. Clock-to-Q delay is the time d when Q may start to change. Q does not necessarily settle at time d .

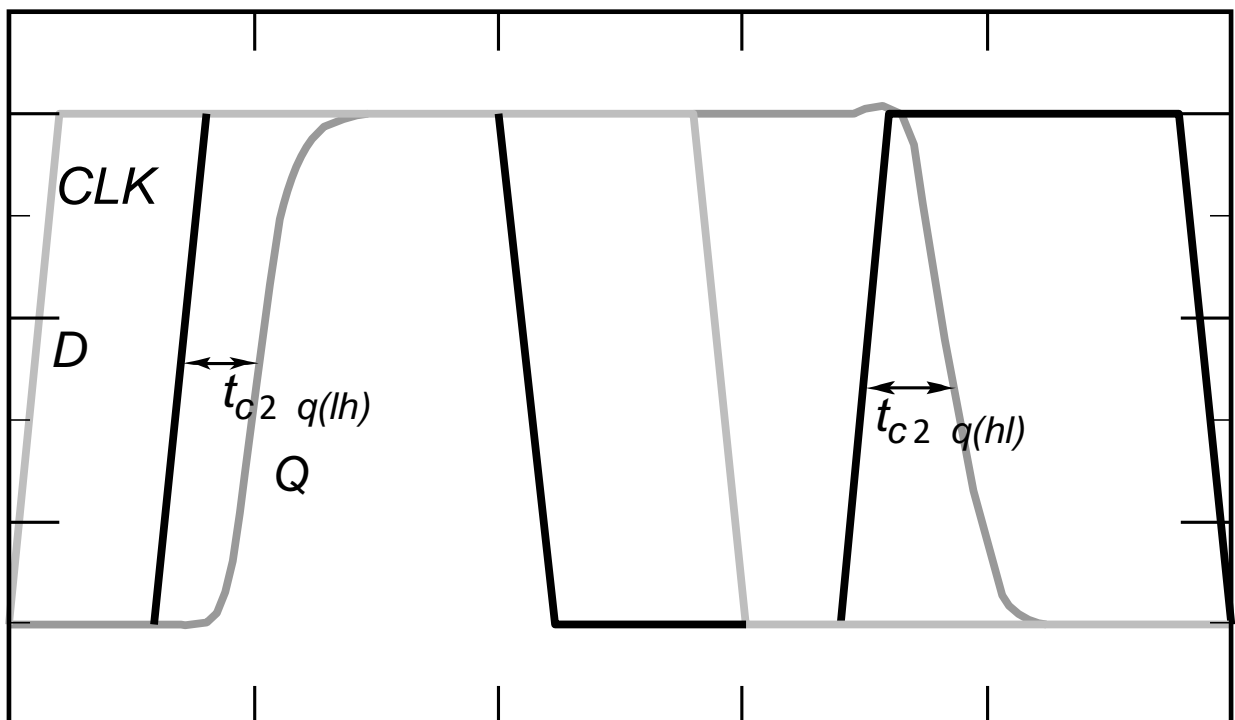


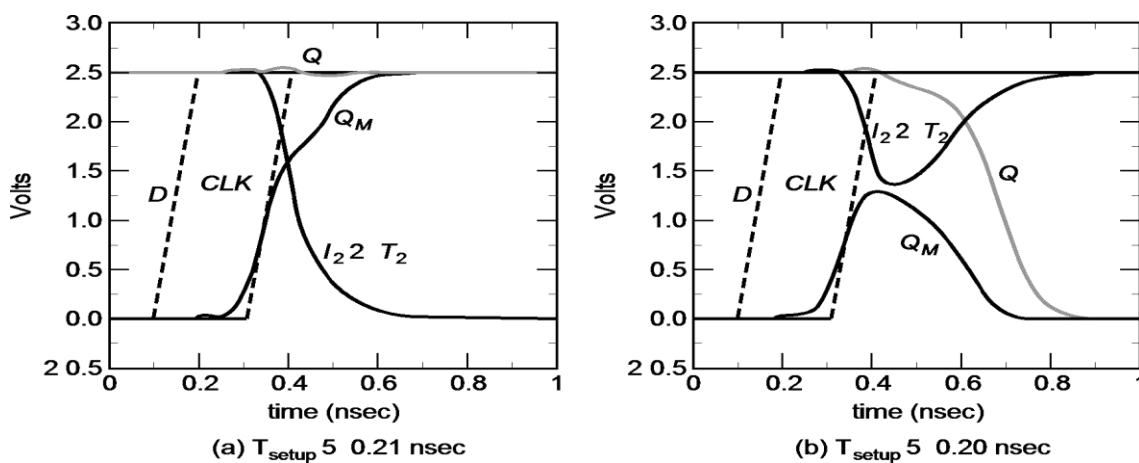
Figure 4.19.1 Clock to q delay

4.20 Setup time and Hold time

Setup time is defined as the minimum amount of time BEFORE the clock's active edge by which the data must be stable for it to be latched correctly. Any violation in this required time causes incorrect data to be captured and is known as a setup violation.

Setup time is the interval needed to adjust the settings on a machine, so that it is ready to process a job. Shortening the amount of setup time is critical for engaging in short production runs, so that a business can more easily engage in just-in-time production. Setup time is the amount of time required for the input to a Flip-Flop to be stable before a clock edge. Hold time is similar to setup time, but it deals with events after a clock edge occurs. Hold time is the minimum amount of time required for the input to a Flip-Flop to be stable after a clock edge.

HOLD TIME: Hold time is defined as the minimum amount of time AFTER the clock's active edge during which the data must be stable. Any violation in this required time causes incorrect data to be latched and is known as a hold violation.

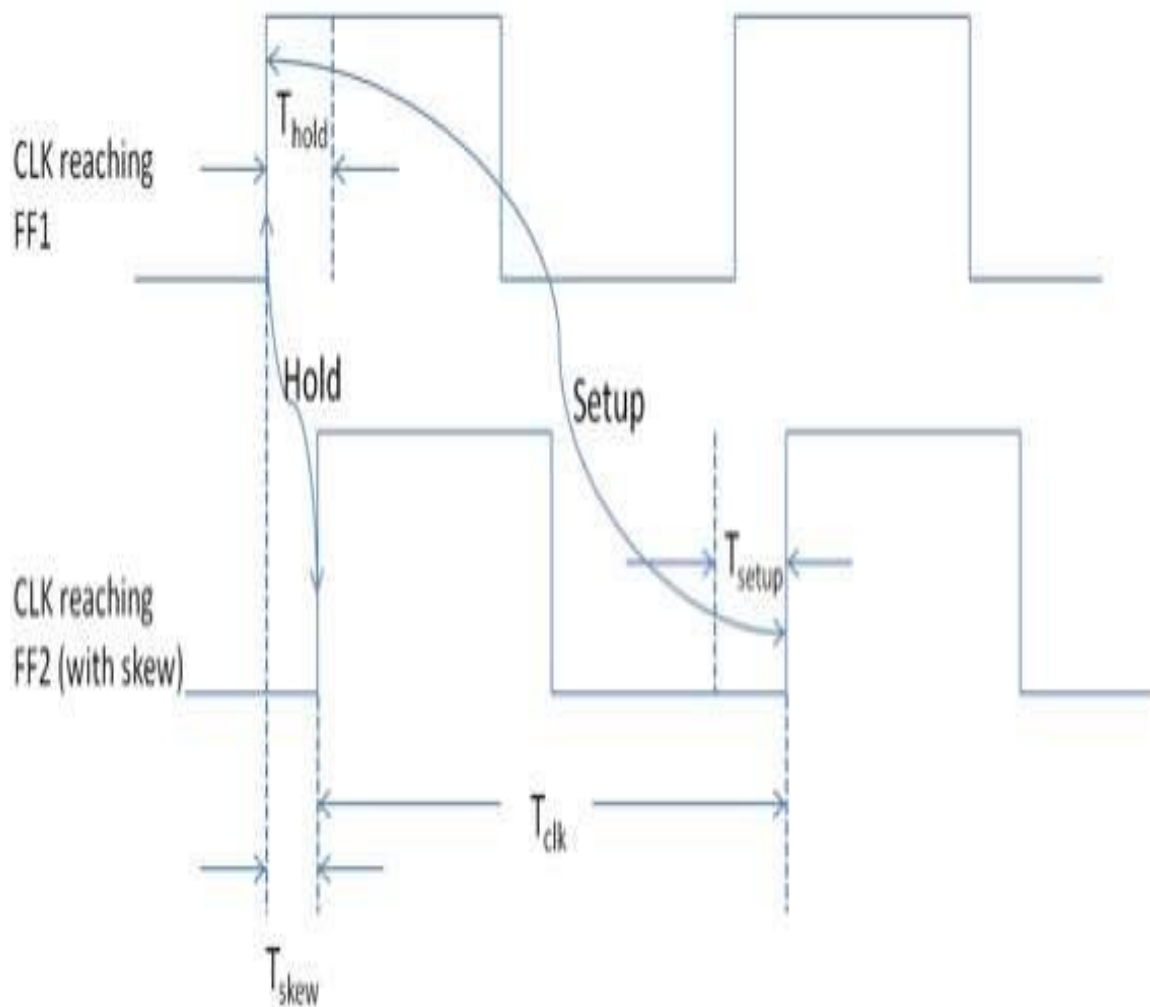


In the diagram above, at time zero FF1 is to process D2 and FF2 is to process D1. Time taken for the data D2 to propagate to FF2, counting from the clock edge at FF1, is invariably $= T_{c2q} + T_{\text{comb}}$ and for FF2 to successfully latch it, this D2 has

to be maintained at D of FF2 for T_{setup} time before the clock tree sends the next positive edge of the clock to FF2. Hence to fulfill the setup time requirement, the equation should be like the following.

$$T_{c2q} + T_{comb} + T_{setup} \leq T_{clk} + T_{skew} \quad (1)$$

In the diagram above, at time zero FF1 is to process D2 and FF2 is to process D1. Time taken for the data D2 to propagate to FF2, counting from the clock edge at FF1, is invariably $= T_{c2q} + T_{comb}$ and for FF2 to successfully latch it, this D2 has to be maintained at D of FF2 for T_{setup} time before the clock tree sends the next positive edge of the clock to FF2. Hence to fulfill the setup time requirement, the equation should be like the following.



$$T_{c2q} + T_{comb} + T_{setup} \leq T_{clk} + T_{skew} \quad (1)$$

In the diagram above, at time zero FF1 is to process D2 and FF2 is to process D1. Time taken for the data D2 to propagate to FF2, counting from the clock edge at FF1, is invariably = $T_{c2q} + T_{comb}$ and for FF2 to successfully latch it, this D2 has to be maintained at D of FF2 for T_{setup} time before the clock tree sends the next positive edge of the clock to FF2. Hence to fulfill the setup time requirement, the equation should be like the following.

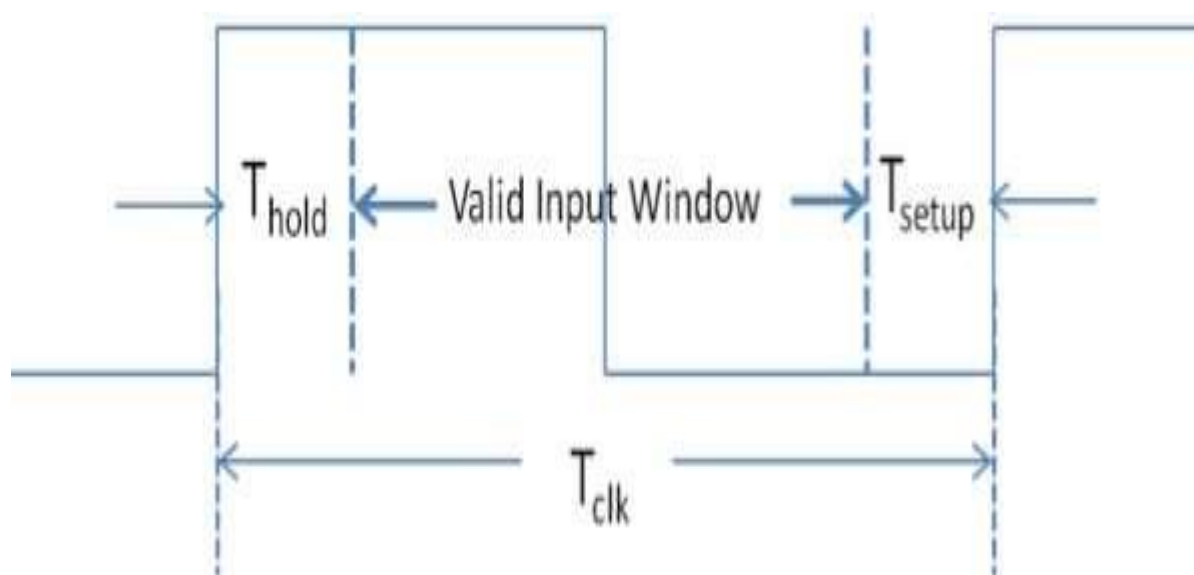
$$T_{c2q} + T_{comb} + T_{setup} \leq T_{clk} + T_{skew} \quad (1)$$

Let's have a look at the timing diagram below to have a better understanding of the setup and hold time.

Now, to avoid the hold violation at the launching flop, the data should remain stable for some time (T_{hold}) after the clock edge. The equation to be satisfied to avoid hold violation looks somewhat like below:

$$T_{c2q} + T_{comb} \geq T_{hold} + T_{skew} \quad (2)$$

As seen from the above two equations, it can be easily judged that positive skew is good for setup but bad for hold. The only region where the input can vary is the 'valid input window' as shown in Figure.



4.21 CLOCKED CMOS REGISTER

Figure 4.21.1 shows an ingenious positive edge-triggered register, based on a master-slave concept insensitive to clock overlap. This circuit is called the C2MOS (Clocked CMOS) register [Suzuki73], and operates in two phases.

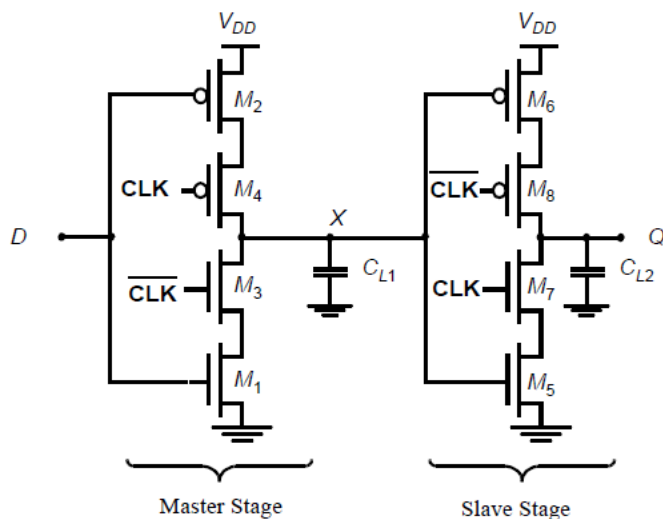


Figure 4.21.1 C2MOS master-slave positive edge-triggered register.

1. CLK = 0 (CLK = 1): The first tri-state driver is turned on, and the master stage acts as an inverter sampling the inverted version of D on the internal node X. The master stage is in the evaluation mode. Meanwhile, the slave section is in a high-impedance mode, or in a hold mode. Both transistors M7 and M8 are off, decoupling the output from the input. The output Q retains its previous value stored on the output capacitor CL2.

2. The roles are reversed when CLK = 1: The master stage section is in hold mode (M3-M4 off), while the second section evaluates (M7-M8 on). The value stored on CL1 propagates to the output node through the slave stage which acts as an inverter.

The overall circuit operates as a positive edge-triggered master-slave register — very similar to the transmission-gate based register presented earlier. However, there is an important difference:

A C2MOS register with CLK-CLK clocking is insensitive to overlap, as long as the rise and fall times of the clock edges are sufficiently small.

To prove the above statement, we examine both the (0-0) and (1-1) overlap cases (Figure 7.25). In the (0-0) overlap case, the circuit simplifies to the network shown in Figure 7.27a in which both PMOS devices are on during this period. The question is does any new data sampled during the overlap window propagate to the output Q. This is not desirable since data should not change on the negative edge for a positive edge-triggered register. Indeed new data is sampled on node X through the series PMOS devices M2-M4, and node X can make a 0-to-1 transition during the overlap period. However, this data cannot propagate to the output since the NMOS device M7 is turned off. At the end of the overlap period, CLK=1 and both M7 and M8 turn off, putting the slave stage in the hold mode. Therefore, any new data sampled on the falling clock edge is not seen at the slave output Q, since the slave state is off till the next rising edge of the clock. As the circuit consists of a cascade of inverters, signal propagation requires one pull-up followed by a pull-down, or vice-versa, which is not feasible in the situation presented.

The (1-1) overlap case (Figure 7.27b), where both NMOS devices M3 and M7 are turned on, is somewhat more contentious. The question is again if new data sampled during the overlap period (right after clock goes high) propagates to the Q output. A positive edge-triggered register may only pass data that is presented at the input before the rising edge. If the D input changes during the overlap period, node X can make a 1-to-0 transition, but cannot propagate to the output. However, as soon as the overlap period is over, the PMOS M8 is turned on and the 0 propagates to output. This effect is not desirable. The

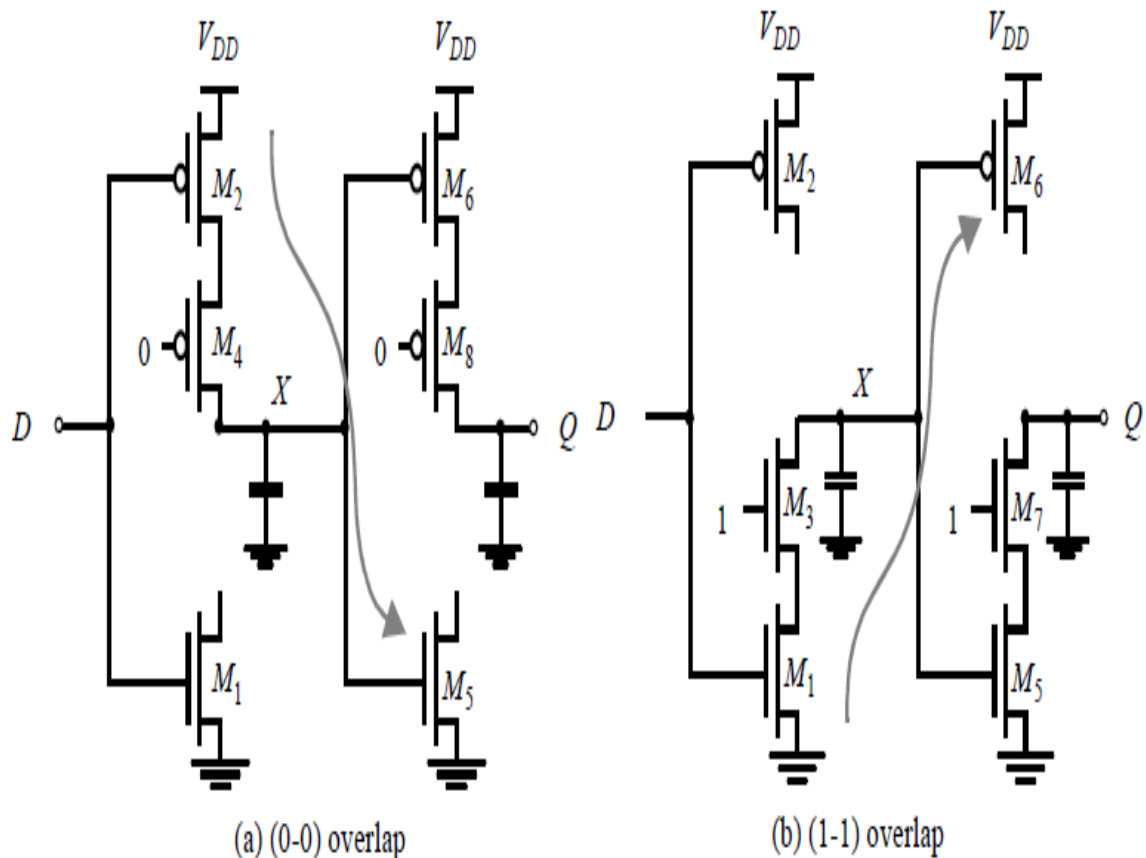


Figure 4.21.2 C2MOS D-FF during overlap periods. No feasible signal path can exist between In and D, as illustrated by the arrows.

problem is fixed by imposing a hold time constraint on the input data, D , or, in other words, the data D should be stable during the overlap period.

In summary, it can be stated that the C2MOS latch is insensitive to clock overlaps because those overlaps activate either the pull-up or the pull-down networks of the latches, but never both of them simultaneously. If the rise and fall times of the clock are sufficiently slow, however, there exists a time slot where both the NMOS and PMOS transistors are conducting. This creates a path between input and output that can destroy the state of the circuit. Simulations have shown that the circuit operates correctly as long as the clock rise time (or fall time) is smaller than approximately five times the propagation delay of the register. This criterion is not too stringent, and is easily met in practical designs.

4.21 C

4.22 CROSS COUPLED NAND AND NOR

The cross-coupled inverter pair shown in the previous section provides an approach to store a binary variable in a stable way. However, extra circuitry must be added to enable control of the memory states. The simplest incarnation accomplishing this is the well-know SR —or set-reset— flip-flop, an implementation of which is shown in Figure 7.6a. This circuit is similar to the cross-coupled inverter pair with NOR gates replacing the inverters. The second input of the NOR gates is connected to the trigger inputs (S and R), that make it possible to force the outputs Q and \bar{Q} to a given state. These outputs are complimentary (except for the SR = 11 state). When both S and R are 0, the flip-flop is in a qui-escent state and both outputs retain their value (a NOR gate with one of its input being 0 looks like an inverter, and the structure looks like a cross coupled inverter). If a positive (or 1) pulse is applied to the S input, the Q output is forced into the 1 state (with \bar{Q} going to 0). Vice versa, a 1 pulse on R resets the flip-flop and the Q output goes to 0.

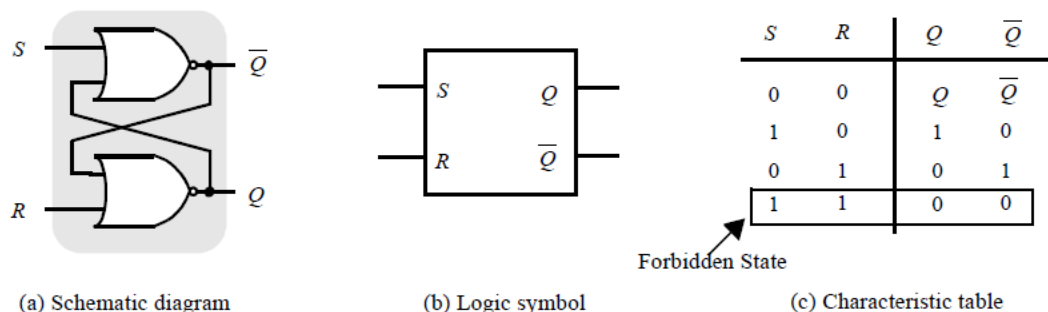


Figure 4.22.1 NOR-based SR flip-flop

These results are summarized in the characteristic table of the flip-flop, shown in Figure 7.6c. The characteristic table is the truth table of the gate and lists the output states as functions of all possible input conditions. When both S and R are high, both Q and \bar{Q} are forced to zero. Since this does not correspond with our constraint that Q and \bar{Q} must be complementary, this input mode is considered to be forbidden. An additional problem with this condition is that when the input triggers return to their zero levels, the resulting state of the latch is unpredictable and depends on whatever input is last to go low. Finally, Figure 7.6 shows the schematics symbol of the SR flip-flop.

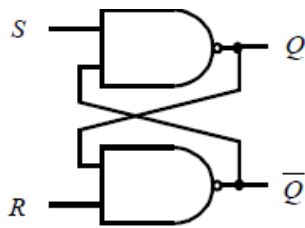


Figure 4.22.2 NAND-based SR flip-flop.

The SR flip-flops discussed so far are asynchronous, and do not require a clock signal. Most systems operate in a synchronous fashion with transition events referenced to a clock. One possible realization of a clocked SR flip-flop—a level-sensitive positive latch—is shown in Figure 7.8. It consists of a cross-coupled inverter pair, plus 4 extra transistors to drive the flip-flop from one state to another and to provide clocked operation. Observe that the number of transistors is identical to the implementation of Figure 7.6, but the circuit has the added feature of being clocked. The drawback of saving some transistors over a fully-complimentary CMOS implementation is that transistor sizing becomes critical in ensuring proper functionality. Consider the case where Q is high and an R pulse is applied. The combination of transistors M4, M7, and M8 forms a ratioed inverter. In order to make the latch switch, we must succeed in bringing Q below the switching threshold of the inverter M1-M2. Once this is achieved, the positive feedback causes the flip-flop to invert states. This requirement forces us to increase the sizes of transistors M5, M6, M7, and M8.

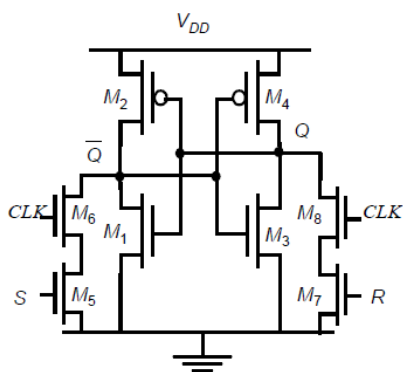


Figure 4.22.3 CMOS clocked SR flip-flop.

The presented flip-flop does not consume any static power. In steady-state, one inverter resides in the high state, while the other one is low. No static paths between VDD and GND can exist except during switching.

4.23 SR MASTER SLAVE REGISTER

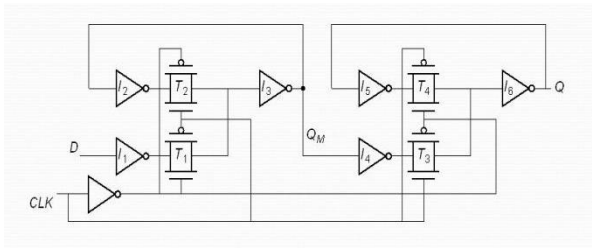


Figure 4.23.1 SR MASTER SLAVE REGISTER

The slave stage is in hold mode, keeping the previous value by using feedback. When CLK=1, the slave stage samples the output of the master stage (Q_M), while master goes into hold mode.

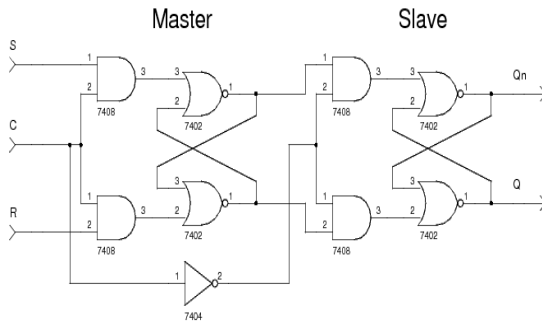


Figure 4.23.2 SR MASTER SLAVE REGISTER (GATE FORM)

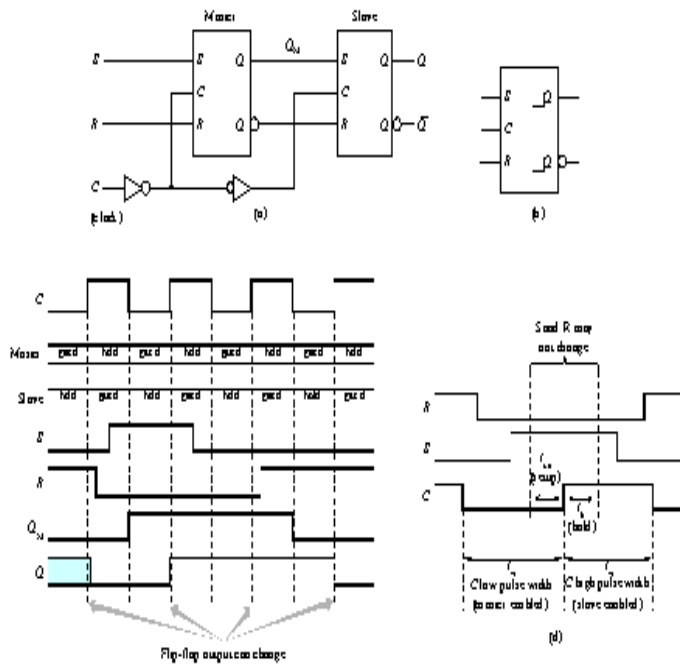


Figure 4.23.3 SR MASTER SLAVE REGISTER WAVEFORM

4.24 STORAGE MECHANISM

Storage Mechanisms are two types

1.static

2.dynamic

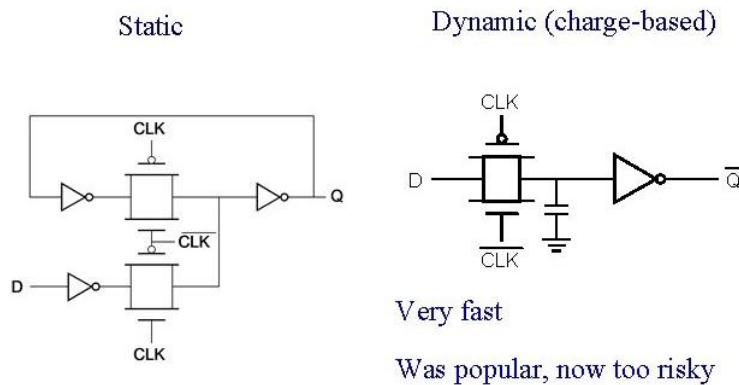


Figure 4.24.1 STORAGE MECHANISM

Setup and hold times determined by transmission gate—must ensure that value stored on transmission gate is solid.

Static v.s. Dynamic

- Static Logic Gates
- Valid logic levels are steady-state operating points
- Outputs are generated in response to input voltage levels after a certain time delay, and it can preserve its output levels as long as there is power.
- All gate output nodes have a conducting path to VDD or GND, except when input changes are occurring.
- Dynamic Logic Gates
- The operation depends on temporary storage of charge in parasitic node capacitances.
- The stored charge does not remain indefinitely, so must be updated or refreshed. This requires establishment of an update or recharge path to the capacitance frequently enough to preserve valid voltage levels.

Static v.s. Dynamic (Continued)

- Advantages of Dynamic Logic Gates
- Allow implementation of simple sequential circuits with memory functions.
- Use of common clock signals throughout the system enables the synchronization of various circuit blocks.
- Implementation of complex circuits requires a smaller silicon area than static circuits.
- Often consumes less dynamic power than static designs, due to smaller parasitic capacitances.

Latch versus Register • Latch stores data when clock is low • Register stores data when clock rises

D Q D Q Clk Clk Clk Clk D D Q Q

static or dynamic Regs • Registers can be static or dynamic. A static register holds state as long as the power supply is turned on • It is ideal for memory that is accessed infrequently (e.g., reconfiguration registers or control information) • Dynamic memory is based on temporary charge store on capacitors • The primary advantage is the reduced complexity and higher performance/lower power.

static and dynamic Regs • However, charge on a dynamic node leak away with time, and hence dynamic circuits have a minimum clock frequency. • several fundamentally different approaches towards building a register. • The most common and widely used approach is the master-slave configuration • which involves cascading a positive latch and negative latch (or vice-versa).

static and dynamic Regs • Storage in a static sequential circuit relies on the concept that a cross-coupled inverter pair produces a bistable element and can thus be used to memorize binary values. • This approach has the useful property that a stored value remains valid as long as the supply voltage is applied to the circuit, hence the name static. • The major disadvantage of the static gate, however, is its complexity

static and dynamic Regs • When registers are used in computational structures, that are constantly clocked such as pipelined datapath, the requirement that the memory should hold state for extended periods of time can be significantly relaxed. • This results in a class of circuits based on temporary storage of charge on parasitic capacitors. The principle is exactly identical to the one used in dynamic logic

static and dynamic Regs • Charge stored on a capacitor can be used to represent a logic signal. The absence of charge denotes a 0, while its presence stands for a stored 1. • No capacitor is ideal, unfortunately, and some charge leakage is always present. • A stored value can hence only be kept for a limited amount of time, typically in the range of milliseconds. If one wants to preserve signal integrity, a periodic refresh of its value is necessary. Hence the

name dynamic storage.

static and dynamic Regs • Reading the value of the stored signal from a capacitor without disrupting the charge requires the availability of a device with a high input impedance.

4.25 PIPELINING CONCEPT

Pipelining is a technique where multiple instructions are overlapped during execution. Pipeline is divided into stages and these stages are connected with one another to form a pipe like structure. Instructions enter from one end and exit from another end. Pipelining increases the overall instruction throughput.

Pipeline system is like the modern-day assembly line setup in factories. For example, in a car manufacturing industry, huge assembly lines are setup and at each point, there are robotic arms to perform a certain task, and then the car moves on ahead to the next arm.

PURPOSE OF PIPELINING:

Pipelining is a technique for breaking down a sequential process into various sub-operations and executing each sub-operation in its own dedicated segment that runs in parallel with all other segments.

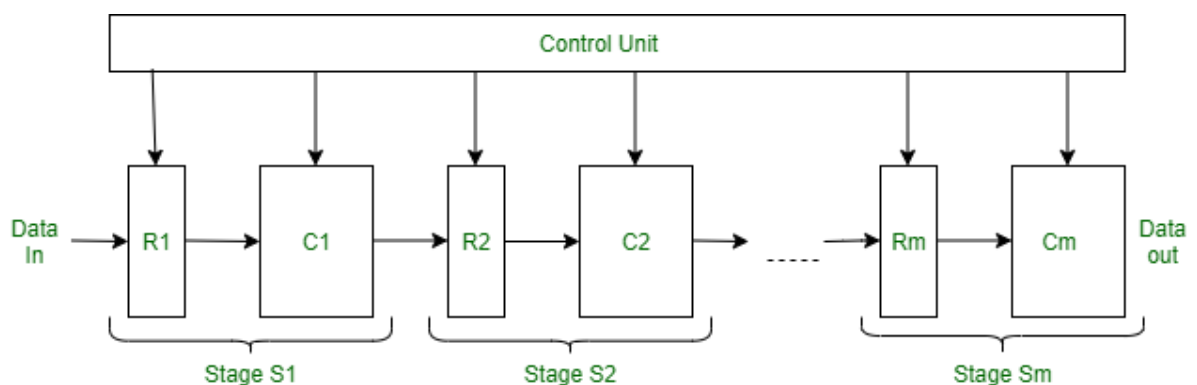


Figure - Structure of a Pipeline Processor

Figure 4.25.1 STRUCTURE OF PIPELINE PROCESSOR

5. Practice Quiz

1. Which clock is preferred in storage devices?

- a) single phase overlapping clock signal
- b) single phase non overlapping clock signal
- c) two phase overlapping clock signal
- d) two phase non overlapping clock signal**

2. Which occupies lesser area?

- a) nMOS**
- b) pMOS
- c) CMOS
- d) BiCMOS

3. In which design, dissipation is less?

- a) nMOS
- b) pMOS
- c) CMOS**
- d) BiCMOS

4. How many transistors might bring up latch up effect in p-well structure?

- a) two**
- b) three
- c) one
- d) four

5. What can be introduced to reduce the latch-up effect?

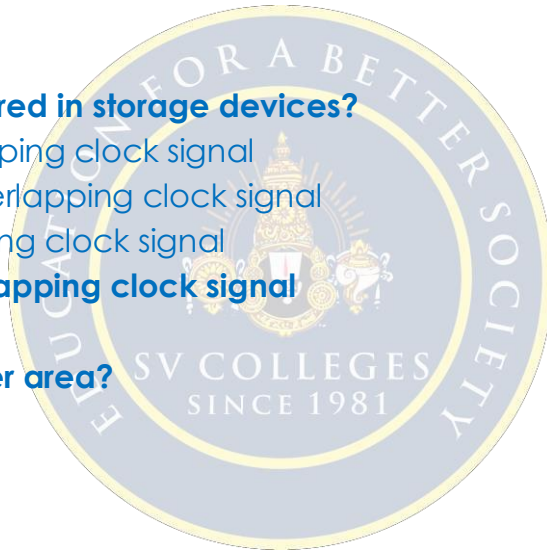
- a) latch-up rings
- b) guard rings**
- c) latch guard rings
- d) substrate rings

6. Which process produces a circuit which is less prone to latch-up effect?

- a) CMOS
- b) nMOS
- c) pMOS
- d) BiCMOS**

7. The reduction in carrier lifetime brings about _____

- a) reduction in alpha
- b) reduction in beta**
- c) reduction in current
- d) reduction in voltage



8. Latch-up is the generation of _____

a) low impedance path

b) high impedance path

c) low resistance path

d) high resistance path

9. BJT gain should be _____ to avoid latch-up effect.

a) increased

b) decreased

c) should be maintained constant

d) changed randomly

10. NOR type flash allows _____ to be read or written independently.

a) one machine cycle

b) one machine word

c) one machine sentence

d) one bit

11. NAND type flash memories are used in

a) Memory cards

b) USB

c) Solid state drivers

d) All of the mentioned

12. Which is a comparatively slower device?

a) ROM

b) RAM

c) flash memory

d) SRAM

13. In NOR type flash memory, each cell has one end connected to

a) source

b) drain

c) gate

d) ground

14. In NOR type flash memory, data is erased

a) bitwise

b) byte wise

c) block wise

d) sentence wise

15. The transistors in NAND type flash are connected in

- a) series
- b) parallel
- c) cascade
- d) randomly

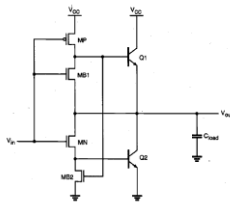
16. Which allows random access to read?

- a) NOR type flash
- b) NAND type flash
- c) all of the mentioned
- d) none of the mentioned

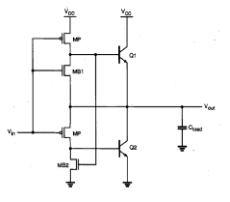
17. The BiCMOS are preferred over CMOS due to _____

- a) Switching speed is more compared to CMOS
- b) Sensitivity is less with respect to the load capacitance
- c) High current drive capability
- d) All of the mentioned

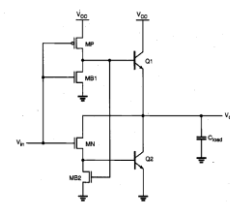
18. Which is the proper BiCMOS inverter circuit?



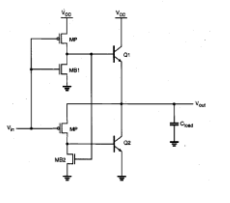
a)



b)



c)



d)

19. What is the work of BJT in BiCMOS?

- a) Current controlled Voltage source
- b) Voltage controlled Current source
- c) Current controlled current source
- d) Voltage controlled current source

20. In BiCMOS, the analysis of the operation of BJT is well explained by _____

- a) RC Model
- b) Emitter resistor model
- c) Ebers Moll Model**
- d) Hybrid model

21. Drain to source current is due to

- a) flow of majority carriers from drain to source
- b) flow of minority carriers from drain to source
- c) flow of majority carriers from source to drain**
- d) flow of majority carriers from drain to source

22. Transit time can be given as the ratio of

- a) channel length to velocity**
- b) electron distance to velocity
- c) source length to velocity
- d) drain length to velocity

23. The magnitude of the depletion region decreases when

- a) V_{gs} decreases
- b) V_{gs} increases**
- c) V_{ds} increases
- d) V_{ds} decreases

24. Velocity saturation occurs in

- a) low electric field
- b) high electric field**
- c) low magnetic field
- d) high magnetic field

25. Increasing fan-out _____ the propagation delay.

- a) increases**
- b) decreases
- c) does not affect
- d) exponentially decreases

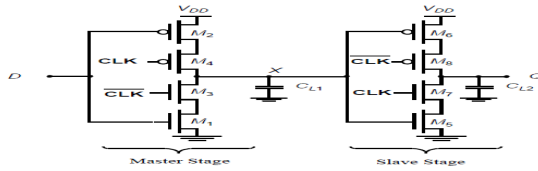
10. Assignments

S.No	Question	BL	CO
1	Compare the cross coupled NAND and NOR SR Master slave registers?	2	3
2	Design a master-slave based edge triggered Register?	5	1
3	Describe the concept of metastability in latches?	5	2
4	Differences between latch versus Register in ultra-deep-submicron era?	4	3
5	Describe the dynamic logic principles in brief manner.	4	3

11.Part A- Question & Answers

S.No	Question & Answers	BL	CO
1	<p>Define Static Power?</p> <p>Static power is consumed even when a chip is not switching. CMOS has replaced nMOS processes because contention current inherent to nMOS logic limited the number of transistors that could be integrated on one chip. Static CMOS gates have no contention current. Prior to the 90 nm node, leakage power was of concern primarily during sleep mode because it was negligible compared to dynamic power. In nanometer processes with low threshold voltages and thin gate oxides, leakage can account for as much as a third of</p>	2	4
2	<p>Define Metastability?</p> <p>A latch is a bistable device; i.e., it has two stable states (0 and 1). Under the right conditions, that latch can enter a metastable state in which the output is at an indeterminate level between 0 and 1.</p>	2	3
3	<p>What are Programmable Interconnects?</p> <p>In a PAL, the device is programmed by changing the characteristics of the switching element. An alternative would be to program the routing.</p>	2	3

4 Explain clocked CMOS register?



2

4

1. CLK = 0 (CLK = 1): The first tri-state driver is turned on, and the master stage acts as an inverter sampling the inverted version of D on the internal node X. The master stage is in the evaluation mode. Meanwhile, the slave section is in a high-impedance mode, or in a hold mode. Both transistors M7 and M8 are off, decoupling the output from the input. The output Q retains its previous value stored on the output capacitor CL2.

2. The roles are reversed when CLK = 1: The master stage section is in hold mode (M3-M4 off), while the second section evaluates (M7-M8 on). The value stored on CL1 propagates to the output node through the slave stage which acts as an inverter.

12.Part B- Questions

S.No	Question	BL	CO
1	Explain the different dynamic logic styles involved in CMOS design?	2	1
2	Explain clearly about the CMOS ratioed logic?	2	1
3	Explain the speed and power dissipation of dynamic logic circuits?	3	1
4	Design a master-slave based edge triggered Register?	5	1
5	Describe the concept of metastability in latches?	5	2
6	Differences between latch versus Register in ultra-deep-submicron era?	4	3

13.Supportive Online Certification Courses

1. Digital circuits By Prof. Santanu Chattopdhayay, conducted by IIT Kharagpur on NPTEL – 12 weeks
2. Digital Electronic Circuits By Prof. Goutam Saha, conducted IIT Kharagpur on NPTEL – 12 weeks
3. System Design Using Verilog in Udemy – 6 weeks.

14.Real Time Applications

S.No	Application	CO
1	Design and implementation of FPGA based traffic control system	3
2	FPGA based Digital hearing aid chip	3
3	Security logging system based on FPGA	3

15.Contents Beyond the Syllabus

Multi-threshold CMOS

Multi-threshold CMOS (MTCMOS) is a variation of CMOS chip technology which has transistors with multiple threshold voltages (V_{th}) in order to optimize delay or power. The V_{th} of a MOSFET is the gate voltage where an inversion layer forms at the interface between the insulating layer (oxide) and the substrate (body) of the transistor. Low V_{th} devices switch faster, and are therefore useful on critical delay paths to minimize clock periods. The penalty is that low V_{th} devices have substantially higher static leakage power. High V_{th} devices are used on non-critical paths to reduce static leakage power without incurring a delay penalty. Typical high V_{th} devices reduce static leakage by 10 times compared with low V_{th} devices.

One method of creating devices with multiple threshold voltages is to apply different bias voltages (V_b) to the base or bulk terminal of the transistors. Other methods involve adjusting the gate oxide thickness, gate oxide dielectric constant (material type), or dopant concentration in the channel region beneath the gate oxide.

A common method of fabricating multi-threshold CMOS involves simply adding additional photolithography and ion implantation steps. For a given fabrication process, the V_{th} is adjusted by altering the concentration of dopant atoms in the channel region beneath the gate oxide. Typically, the concentration is adjusted by ion implantation method. For example, photolithography methods are applied to cover all devices except the p-MOSFETs with photoresist. Ion implantation is then completed, with ions of the chosen dopant type

penetrating the gate oxide in areas where no photoresist is present. The photoresist is then stripped. Photolithography methods are again applied to cover all devices except the n-MOSFETs. Another implantation is then completed using a different dopant type, with ions penetrating the gate oxide. The photoresist is stripped. At some point during the subsequent fabrication process, implanted ions are activated by annealing at an elevated temperature.

In principle, any number of threshold voltage transistors can be produced. For CMOS having two threshold voltages, one additional photomasking and implantation step is required for each of p-MOSFET and n-MOSFET. For fabrication of normal, low, and high V_{th} CMOS, four additional steps are required relative to conventional single- V_{th} CMOS.

The most common implementation of MTCMOS for reducing power makes use of sleep transistors. Logic is supplied by a virtual power rail. Low V_{th} devices are used in the logic where fast switching speed is important. High V_{th} devices connecting the power rails and virtual power rails are turned on in active mode, off in sleep mode. High V_{th} devices are used as sleep transistors to reduce static leakage power.

The design of the power switch which turns on and off the power supply to the logic gates is essential to low-voltage, high-speed circuit techniques such as MTCMOS. The speed, area, and power of a logic circuit are influenced by the characteristics of the power switch.

16. Prescribed Text Books:

1. Kamran Eshraghian, "Essentials of VLSI Circuits and Systems", Douglas and A. Pucknell and Sholeh Eshraghian, Prentice-Hall of India Private Limited, 2005 Edition.
2. Behzad Razavi, "Design of Analog CMOS Integrated Circuits", McGraw Hill, 2003
3. Jan M. Rabaey, "Digital Integrated Circuits", Anantha Chandrakasan and Borivoje Nikolic, Prentice-Hall of India Pvt. Ltd, 2nd edition, 2009.

References:

1. John P. Uyemura, "Introduction to VLSI Circuits and Systems", John Wiley & Sons, reprint 2009.

17. Mini Project Suggestion

1). Timing Synchronization Technique with a Symbol Rate for Wireless OFDM Systems with Low Power

This proposed system mainly used to improve the act of wireless OFDM (Orthogonal Frequency Division Multiplexing) system through decreasing the power of the entire baseband with the help of a clock generator with phase tunable & dynamic sample-timing controller.

2). Accumulator based Low Power & High-Speed Multiplier Implementation with SPST Adder & Verilog

This project is used to design a low power & high-speed MAC (multiplier and accumulator) through accepting the false suppression method of power on an MBE (modified booth encoder). By using this design, the power dissipation of entire switching can be avoided.

3). Robot Processor Design & Implementation by Enabling Anti-collision with RFID Technology

The proposed system is mainly used to implement a robot processor with anti-collision to avoid the physical collision of robots in the environment of multi-robot. This algorithm is mainly implemented using VHDL & RFID technology.

4). Designing of Logic Circuit with Power Efficient using Adiabatic Method

This system demonstrates the logic circuit design by efficiently with adiabatic method when compared through conventional CMOS design with the help of circuits using NAND & NOR gates. By using the adiabatic method, the dissipation of power within the network can be reduced as well as recycles the stored energy within the load capacitor.

UNIT-V

CAD TOOLS FOR DESIGN AND SIMULATION

As the design of very large systems is concerned, it is essential to have computer aids to design so that the design can be completed in a reasonable time.

The designer's 'tool box' should include:

1. **Physical design layout and editing capabilities**, either through textual or graphical entry of information;
2. **Structure generation/system composition capabilities**
3. **Physical verification.**

The tools here should include, design rule checking (DRC), circuit extractors, ratio rule and other static checks, and a capability to plot out and/or display for visual checking;

4. **Behavioral verification.**

Simulation at various levels will be required to check out the design before one embarks on the expense of turning out the design in silicon.

Simulators are available for logic (switch level) simulation and timing simulation.

Circuit simulation via such programs as SPICE is also possible, but may be expensive in terms of computing time and therefore impractical for other than small subsystems.

Recent advances in simulators have made it possible to use the software as 'a probe' to examine the simulated responses on various parts of the circuit to input stimuli also provided via the simulator.

ASPECTS OF DESIGN TOOLS

Graphical Entry Layout

1. Textual entry of layouts was at one time quite widely used and special textual entry editors are in existence and may well be used for small subsystem layout.
2. However, such tools have been virtually swept aside by a much more convenient and highly interactive method of producing layouts for which monochrome or color graphics terminals are used, and on which the layout is built up and displayed during the design process.
3. Such systems are mostly 'menu driven', in that menus of possible actions at various stages of the design are displayed on the screen beside the display of the current layout detail.
4. Some form of cursor allows selection and/or placement of geometric features, etc., and the cursor may also allow selection of menu items or, alternatively, these may also be selected from a keyboard.
5. Positioning of the cursor may be effected from the keyboard in simple systems and/or cursor position may be controlled from a bit pad digitizer or from a 'mouse ', etc

6. Two of the earliest available graphical entry layout packages were **KIC** developed at the University of California, Berkeley, and **PLAN**, originally developed at the University of Adelaide.
7. PLAN makes use of low-cost monochrome as well as color graphics terminals and is marketed by Integrated Silicon Design Pty Ltd, Adelaide.
8. The use of an early version of PLAN to generate layouts illustrated in Figures and it is hoped that the inclusion of these figures, which show various stages of design, is sufficient to convey an idea of the nature of the layout process using this class of software tools.

Design Verification Prior to Fabrication

1. It is not enough to have good design tools for producing mask and system layout detail.
2. It is essential that such tools be complemented by equally effective verification software capable of handling large systems and with reasonable computing power requirements.
3. The nature of the tools required will depend on the way in which an integrated circuit design is represented in the computer.
4. Two basic approaches are:
 1. **Mask level layout languages**, such as CIF, which are well suited to physical layout description but not for capturing the design intent.
 2. **Circuit description languages** where the primitives are circuit elements such as transistors, wires, and 'nodes.

In general, such languages capture the design intent but do not directly describe the physical layout associated with the design.

Design Rule Checkers (DRC)

- A. The cost in time and facilities in mask-making and in fabricating a chip from those masks is such that all possible errors must be eliminated before mask-making proceeds.
- B. Once a design has been turned in to silicon there is little that can be done if it doesn't work.

The wise designer will check for errors at all stages of the design, namely:

1. At the pencil and paper stage of the design of leaf-cells;
2. At the leaf-cell level once the layout is complete (e.g. when the CIF code for that leaf-cell has been generated);
3. At the subsystem level to check that butting together and wiring up of leaf-cells is correctly done;
4. Once the entire system layout has been completed.

The nature of physical layout verification 'design rule checking (DRC)' software may depend on whether the design rules are absolute or lambda-based, or on whether or not the layout is on a fixed or virtual grid.

Circuit Extractors

Circuit extractor can describe mask layout in a form suitable to a simulator.

The circuit description contains information about circuit components and their interconnections.

(An example of a circuit extractor program is NET from Integrated Silicon Design Pty Ltd.)

Simulators

The circuit description is subsequently transformed into a set of equations by the simulator from which the predictions of behavior are made.

The topology of the circuit determines two sets of equations:

- Kirchhoff's Current Law-determining the branch currents; and
- Kirchhoffs Voltage Law-determining node voltages.

The electrical behavior is defined by mathematical modeling, the accuracy of which determines two key factors:

- the accuracy of the simulation; and
- the computing power and time needed for the simulation.

Various **types of simulators** are available but generally they fall into the following groups:

- Circuit simulators;
- Timing simulators;
- Logic level (switch level) (functional) simulators;
- System level (functional) simulators.

Circuit simulators are concerned with the electrical behavior of the various parts of the circuit to be implemented in silicon.

Simulation programs such as SPICE can do this quite well, but take a lot of computing time to simulate even relatively small sections of a system.

Timing simulators (PROBE) have attempted to improve matters in these respects by concentrating on active nodes and ignoring quiescent nodes in simulation.

Timing simulators are becoming increasingly important during the design phase because of their speed and consequent interactive qualities.

The structure of these tools ensures that run times are strictly linearly related to the number of devices and nodes being simulated.

Logic level simulators the performance is assessed in terms of logic levels with no or little timing information.

When considering complete systems, logic simulators may be replaced by **system level simulators** which operate at the register transfer level.

Design for Testability*

Design for testability (DFT) makes it possible to:

1. Assure the detection of all faults in a circuit
2. Reduce the cost and time associated with test development
3. Reduce the execution time of performing test on fabricated chips

There are two key concepts underlying all considerations for testability.

They are:

1. Controllability;
2. Observability.

The controllability of a circuit is a measure of the ease (or difficulty) with which the controller (test engineer) can establish a specific signal value at each node by setting values at the circuit input terminals.

(OR)

Controllability of a digital circuit is defined as the difficulty of setting a particular logic signal to 0 or 1.

The observability is a measure of the ease (or difficulty) with which one can determine the signal value at any logic node in the circuit by controlling its primary input and observing the primary output.

(OR)

Observability for a digital circuit is defined as the difficulty of observing the state of a logic signal

The degree of controllability and observability and, thus, the degree of testability of a circuit, can be measured with respect to whether test vectors are generated deterministically or randomly.

These concepts ensure that the designer considers the provision of means of setting or resetting key nodes in the system and of observing the response at key points.

The effects of testability or lack of it are such that it has been predicted that testability will soon become the main design criterion for VLSI circuits.

Design for testability (observability and controllability) is then reduced to a set of design rules or guidelines which, if obeyed, will facilitate test.

A failure during testing at the chip level may be due to a design defect or a poorly controlled fabrication process.

The inputs of the device under test (DUT) are subjected to a test pattern (or test vector) which supplies a set of binary values, in combination and/or in sequence, to detect faults.

The specification of the test vector sequences must involve the designer, while the generation and application of test patterns to a DUT are the problems faced by the test engineer.

Test pattern generation is assisted by using automatic test pattern generators (ATPG), but they are complicated to use properly and ATPG costs tend to rise rapidly with circuit size.

Once the application of a test pattern has revealed a fault, the process of diagnosis must be invoked to localize the fault.

Test coverage

Detecting all the possible faults in a DUT corresponds to 100% 'test coverage'.

In general it is relatively easy to detect the first 80% of faults using various classical test strategies, but when more than an 80% coverage is required, appropriate test strategies must be developed.

In any case, it is not generally possible to anticipate 100% of all faults, so that we tend to talk about a set of fault hypotheses which may then be covered 100%.

Faults may be classified using different **models** and three such are:

- **Mathematical model;**
- **Logical model (stuck-at);**
- **Physical model.**

The latter two are most commonly used.

The 'stuck-at' model has been widely used and was originally developed in the testing of p.c. boards but is not in itself sufficient to test actual VLSI CMOS circuits.

A further set of physical fault models is also used:

- Class 0: A single physical defect such as a faulty contact or via, a transistor stuck on or stuck off, an interconnection through any layer open circuit.
- Class 1: Class 0 with a short circuit between metal lines or diffusion lines.
- Class 2: Class 1 with short circuit(s) between two lines on any layer.

Testing Combinational Logic

The solution to the problem of testing combinational logic is to generate a set of test patterns which will detect all possible fault conditions.

The first approach to testing an N input circuit is to generate all the possible 2^N input signal combinations by means of, say, an N-bit counter (controllability) and observe the output(s) for checking (observability).

This is called exhaustive testing and is very effective, but is only practicable where N is relatively small.

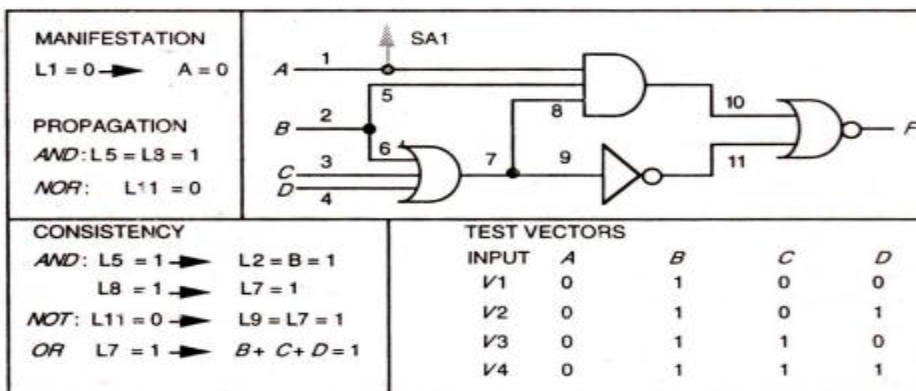
Sensitized path-based testing

The basic idea is to select a path from the site of the possible fault, through a sequence of gates leading to an output of the logic circuitry under test(CUT).

The process comprises three steps:

1. **Manifestation:**
Gate inputs at the site of an assumed fault, say a 'stuck at' (SA) fault, are specified to generate the opposite value to the assumed SA value (0 for SA1, 1 for SA0).
2. **Propagation:**
Inputs of other gates are determined so as to propagate the fault signal along the selected path to the primary output of the circuit. This is done by setting And/Nand inputs to '1' and Or/Nor inputs to '0'.
3. **Consistency (or justification):**
This final step finds the primary input patterns to realize all the necessary values. This is done by tracing backward from the gate inputs to the primary input of the logic.
Examples will help explain the process.

Example 1: Take an SA1 fault on line 1(L 1) in Figure , then



The D-algorithm

The algorithm aims to find an assignment of input values that will allow detection of a particular internal fault by examining the output conditions.

In order to do this the algorithm is based on the hypothesis of the existence of two machines a good machine and a faulty machine.

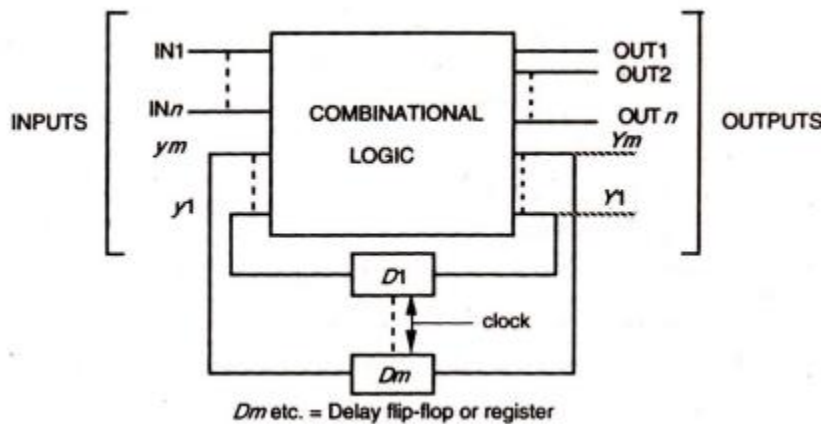
The existence of a fault in the faulty machine will cause a discrepancy between its behavior and that of the good machine for some particular values of inputs.

The D-algorithm provides a systematic means of assigning input values for that particular design so that the discrepancy is driven to an output where it may be observed and thus detected.

The algorithm is extremely time intensive and computing intensive for large circuits and has been the subject of several adoptions, modifications and improvements.

LASAR (Logic Automated Stimulus and Response), PODEM (Path Oriented Decision Making) and FAN (FAN-out oriented test generation) are all improvements on the D-algorithm.

Testing Sequential Logic



Sequential circuits, which may be generally represented as finite state machines, may be modeled as combinational logic with a set of delays and feedback from output to input as shown in Figure.

The 'm' feedback variables constitute the state vector and determine the maximum number of finite states which may be assumed by the circuit.

In the most general case, the next state and the output are both functions of the present state and the independent inputs.

Scan Design Techniques

The scan design techniques are structured approaches to designing sequential circuits so that testability is 'designed in' from the outset.

The major difficulty in sequential circuit testing is in determining the internal state of the circuit.

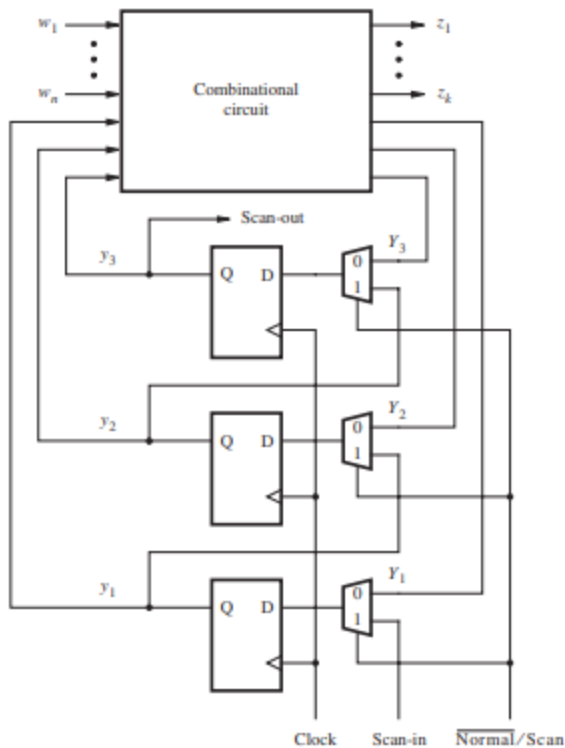
Scan design techniques are directed at improving the controllability and observability of the internal states.

The approach aims to reduce the problem of testing a sequential circuit to that of testing combinational logic.

Scan-Path Technique

A popular technique, called the scan path, uses multiplexers on flip-flop inputs to allow the flip-flops to be used either independently during normal operation of the sequential circuit, or as a part of a shift register for testing purposes.

Figure presents the general scan-path structure for a circuit with three flip-flops.



A 2-to-1 multiplexer connects the D input of each flip-flop either to the corresponding next-state variable or to the serial path that connects all flip-flops into a shift register.

The control signal Normal/Scan selects the active input of the multiplexer.

During the normal operation the flip-flop inputs are driven by the next-state variables, Y_1 , Y_2 , and Y_3 . For testing purposes the shift-register connection is used to scan in the portion of each test vector that involves the present-state variables, y_1 , y_2 , and y_3 .

This connection has Q_i connected to D_{i+1} .

The input to the first flip-flop is the externally accessible pin Scan-in.

The output comes from the last flip-flop, which is provided on the Scan-out pin.

The scan-path technique involves the following steps:

1. The operation of the flip-flops is tested by scanning into them a pattern of 0s and 1s, for example, 01011001, in consecutive clock cycles, and observing whether the same pattern is scanned out.

2. The combinational circuit is tested by applying test vectors on $w_1w_2 \dots w_ny_1y_2y_3$ and observing the values generated on $z_1z_2 \dots z_mY_1Y_2Y_3$. This is done as follows:

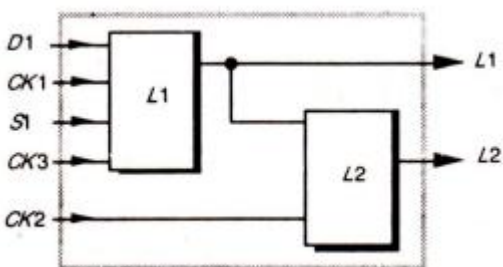
- The $y_1y_2y_3$ portion of the test vector is scanned into the flip-flops during three clock cycles, using Normal/Scan = 1.
- The $w_1w_2 \dots w_n$ portion of the test vector is applied as usual and the normal operation of the sequential circuit is performed for one clock cycle, by setting Normal/Scan = 0. The outputs $z_1z_2 \dots z_m$ are observed. The generated values of $Y_1Y_2Y_3$ are loaded into the flip-flops at this time.
- The select input is changed to Normal/Scan = 1, and the contents of the flip-flops are scanned out during the next three clock cycles, which makes the $Y_1Y_2Y_3$ portion of the test result observable externally.

Level-sensitive scan design (LSSD)

This is a technique, initially developed by IBM,

which incorporates two aspects- level sensitivity and a scan path approach.

The general arrangement is indicated in Figure.



The **level-sensitive aspect** means that the sequential network is designed so that when an input change occurs, the response is independent of the component and wiring delays within the network.

The **scan path aspect** is due to the use of shift register latches (SRL) employed as storage elements. In the test mode they are connected as a long serial shift register.

Each SRL has a specific design similar to a master-slave flip-flop.

It is driven by two non-overlapping clocks which can be controlled readily from the primary inputs to the circuit.

Input DI is the normal data input to the SRL, clocks CK1 and CK2 control the normal operation of the SRL while clocks CK3 and CK2 control scan path movements through the SRL.

The SRL output is derived at L2 in both modes of operation, the mode depending on which clocks are activated. The following advantages are claimed for the LSSD approach:

- The circuit operation is independent of the dynamic characteristics of the logic elements-rise- and fall-times and propagation delays.
- A TP generation is simplified since tests need only be generated for a combinational circuit.
- LSSD methods, when adopted in design, eliminate hazards and races; greatly simplifies test generation and fault simulation.

Built-In-Self-Test (BIST)

As the complexity of individual VLSI circuits and as overall system complexity increase, test generation and application becomes an expensive, and not always very effective, means of testing.

Further, there are also very difficult problems associated with the high speeds at which many VLSI systems are designed to operate.

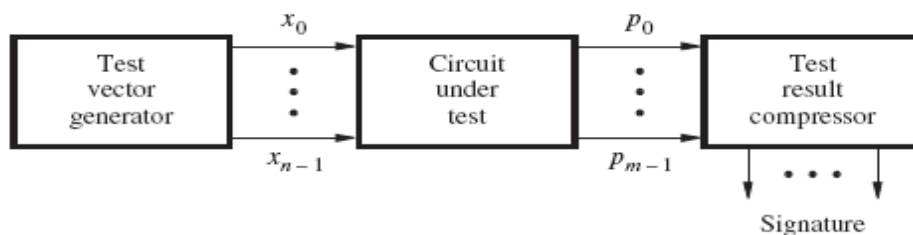
Such problems require the use of very sophisticated, but not always affordable, test equipments. Consequently, BIST .

BIST techniques aim to effectively integrate an automatic test system into the chip design.

BIST objectives are:

1. to reduce test pattern generation costs;
2. to reduce the volume of test data;
3. to reduce test time.

Figure shows a possible BIST arrangement in which a test vector generator produces the test vectors that must be applied to the circuit under test.



Randomly chosen test vectors give good results, with the fault coverage depending on the number of tests performed.

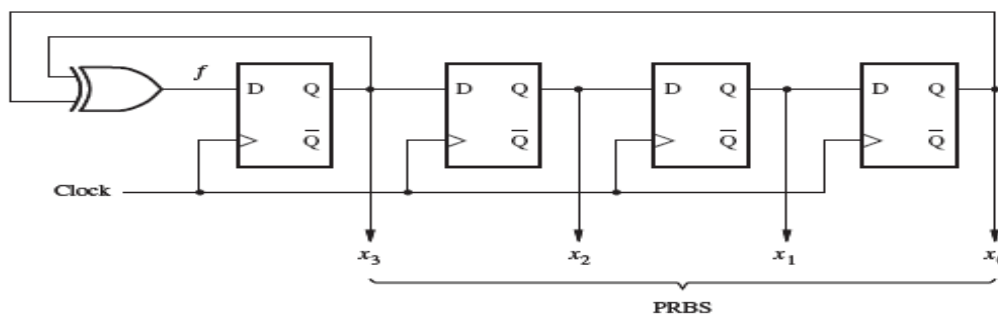
For each test vector applied to the circuit, it is necessary to determine the required response of the circuit.

The response of a good circuit may be determined using the simulator tool of a CAD system.

The expected responses to the applied tests must be stored on the chip so that a comparison can be made when the circuit is being tested.

The generator for pseudorandom tests is easily constructed using a feedback shift-register circuit.

A small example of a possible generator is given in Figure 11.14.

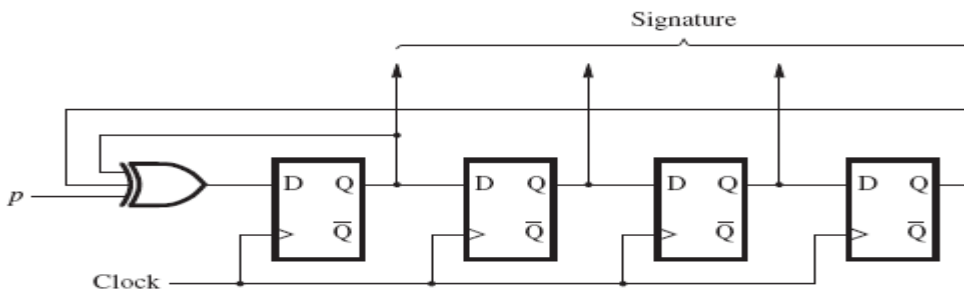


x_3	1	1	1	1	0	1	0	1	1	0	0	1	0	0	0	1	...
x_2	0	1	1	1	1	0	1	0	1	1	0	0	1	0	0	0	...
x_1	0	0	1	1	1	1	0	1	0	1	1	0	0	1	0	0	...
x_0	0	0	0	1	1	1	1	0	1	0	1	1	0	0	1	0	...
f	1	1	1	0	1	0	1	1	0	0	1	0	0	0	1	1	...

A four-bit shift register, with the signals from the first and fourth stages fed back through an XOR gate, generates 15 different patterns during successive clock cycles.

A compressor circuit includes the output signals produced by the circuit under test.

Figure shows a single-input compressor circuit (SIC), which uses the same feedback connections as the PRBSG of Figure



The input p is the output of a circuit under test.

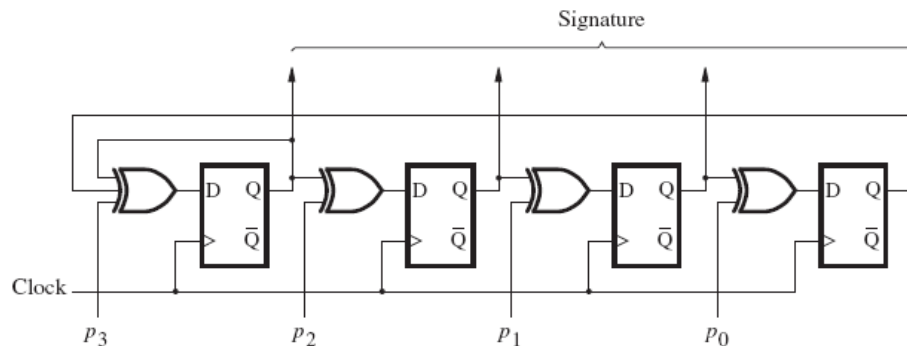
After applying a number of test vectors, the resulting values of p drive the SIC and, coupled with the LFSR functionality, produce a four-bit pattern called a signature.

The signature represents a single pattern that may be interpreted as a result of all the applied tests.

It can be compared against a predetermined pattern to see if the tested circuit is working properly.

If the circuit under test has more than one output, then an LFSR with multiple inputs can be used.

Figure 11.16 illustrates how four inputs, p_0 through p_3 ,



.Again the four-bit signature provides a good mechanism for distinguishing among different sequences of four-bit patterns that may appear on the inputs of this multiple-input compressor circuit (MIC).

BIST for Sequential Circuit:

A complete BIST scheme for a sequential circuit may be implemented as indicated in Figure.

The scan-path approach is used to provide a testable circuit.

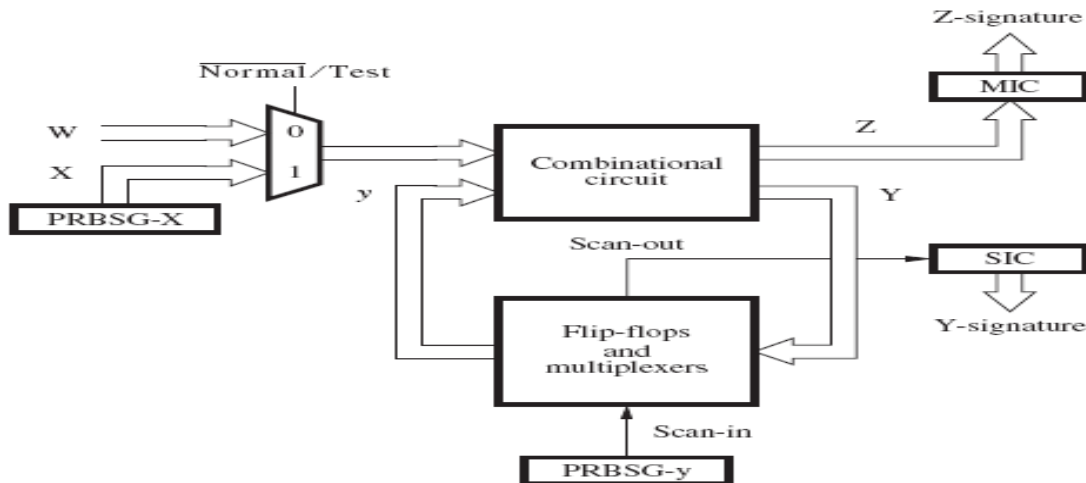
The test patterns that would normally be applied on the primary inputs $W = w_1w_2 \dots w_n$ are generated internally as the patterns on $X = x_1x_2 \dots x_n$.

Multiplexers are needed to allow switching from W to X , as inputs to the combinational circuit.

A pseudorandom binary sequence generator, PRBSG- X , generates the test patterns for X .

The portion of the tests applied via the next-state signals, y , is generated by the second PRBS generator, PRBSG- y .

These patterns are scanned into the flip-flops as explained.



The test outputs are compressed using the two compressor circuits.

The patterns on the primary outputs, $Z = z_1z_2 \dots z_m$, are compressed using the MIC circuit, and those on the next-state wires $Y = Y_1Y_2 \dots Y_k$, by the SIC circuit.

These circuits produce the Z-signature and Y -signature, respectively.

At the end of the testing process the two signatures are compared with the stored patterns.

The effectiveness of the BIST approach depends on the length of the LFSR generator and compressor circuits.

Longer shift registers give better results .

One reason for failing to detect that the circuit under test may be faulty is that the pseudorandomly generated tests do not have perfect coverage of all possible faults.

Another reason is that a signature generated by compressing the outputs of a faulty circuit may coincidentally end up being the same as the signature of the good circuit.

This can occur because the compression process results in a loss of some information, such that two distinct output patterns may be compressed into the same signature. This is known as the aliasing problem

Built-in Logic Block Observer (BILBO)

The essence of BIST is to have internal capability for generation of tests and for compression of the results. Instead of using separate circuits for these two functions, it is possible to design a single circuit that serves both purposes.

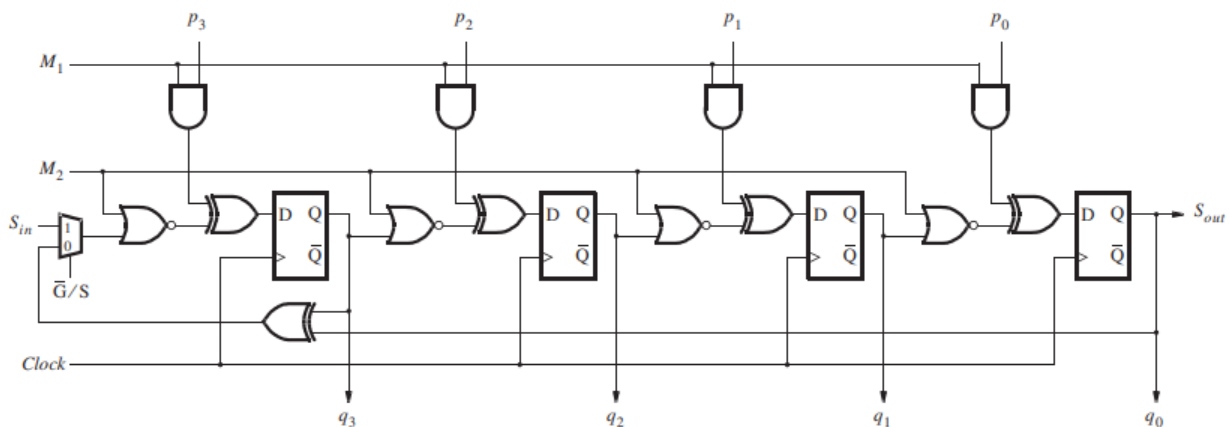
Figure shows the structure of a possible circuit, known as the built-in logic block observer (BILBO).

This four-bit circuit has the same feedback connections.

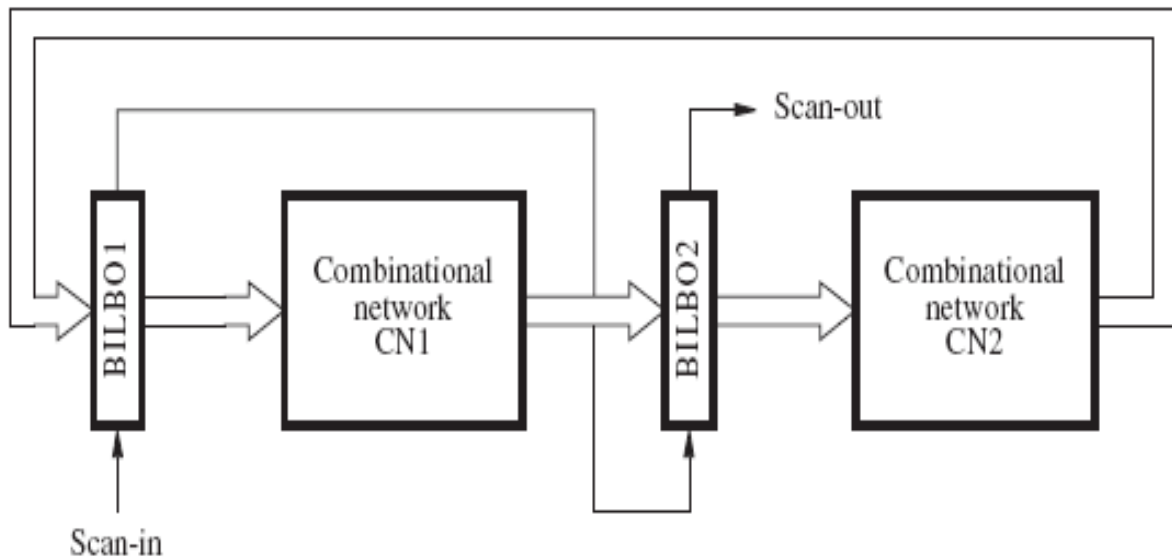
The BILBO circuit has four modes of operation, which are controlled by the mode bits, M1 and M2.

The modes are as follows:

- M1M2 = 11—Normal system mode in which all flip-flops are independently controlled by the signals on inputs p0 through p3. In this mode each flip-flop may be used to implement a state variable of a finite state machine by using p0 to p3 as y0 to y3.
- M1M2 = 00 — Shift-register mode in which the flip-flops are connected into a shift register. This mode allows test vectors to be scanned in, and the results of applied tests to be scanned out, if the control input G/S is equal to 1. If G/S = 0, then the circuit acts as the PRBS generator.
- M1M2 = 10 — Signature mode in which a series of patterns applied on inputs p0 through p3 are compressed into a signature available as a pattern on q0 through q3.
- M1M2 = 01 — Reset mode in which all flip-flops are reset to 0



An efficient way of using BILBO circuits is presented in Figure 11.19.



A combinational circuit can be tested by partitioning it into two (or more) parts.

A BILBO circuit is used to provide inputs to one part and to accept outputs from the other part.

The testing process involves a two-phase approach.

First, BILBO1 is used as a PRBS generator that provides test patterns for combinational network 1 (CN1). During this time BILBO2 acts as a compressor and produces a signature for the test.

The signature is shifted out by placing BILBO2 into the shift-register mode.

Next, the roles of BILBO1 and BILBO2 are reversed, and the process is repeated to test CN2.

The detailed steps in the testing process are

1. Scan the initial test pattern into BILBO1 and reset all flip-flops in BILBO2.
2. Use BILBO1 as the PRBS generator for a given number of clock cycles and use BILBO2 to produce a signature.
3. Scan out the contents of BILBO2 and externally compare the signature; then scan into it the initial test pattern for testing CN2. Reset the flip-flops in BILBO1.
4. Use BILBO2 as the PRBS generator for a given number of clock cycles and use BILBO1 to produce a signature.
5. Scan out the signature in BILBO1 and externally compare it with the required pattern.

The BILBO circuits are used in this way for testing purposes.

At other times the normal system mode is used

Boundary scan test (BST)

This is a technique involving scan path and self-testing to resolve the problems associated with the testing of boards carrying VLSI circuits and/or surface-mounted devices (SMD).

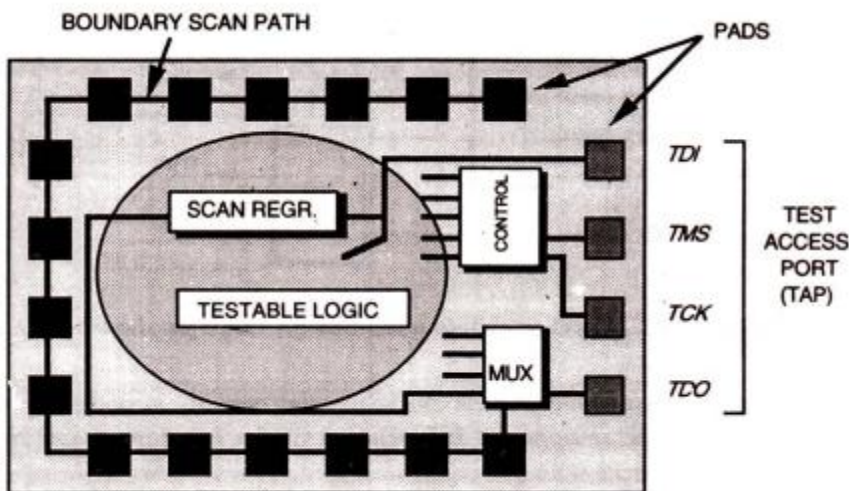
Printed circuit boards (PCBs) are becoming very dense and complex, especially with SMD circuits, so that most test equipment cannot guarantee a good fault coverage.

BST consists of placing a scan path (shift register) cell adjacent to each component pin and to interconnect the cells so as to form a chain around the border of the circuit.

The BST circuits contained on one board are then connected together to form a single path. The general idea is illustrated in Figure .

The boundary scan path is provided with serial input and output pads and appropriate clock pads which make it possible to:

- test the interconnections between the various chips on the board;
- deliver test data to the chips on the board for self-testing;
- test the chips themselves with internal self-test facilities.



BS techniques are grouped by the IEEE standards organization into a 'standard test access port and boundary scan architecture' (namely, IEEE, p. 1149.1-1990).

The advantages of BST are seen as follows:

- no need for complex testers in PCB testing;
- the test engineer's work is simplified and efficient;
- the time spent on test pattern generation and application is reduced;
- fault coverage is increased.

Practical Design for test (OFT) Guidelines

Practical guidelines for testability to facilitate test processes in three main ways:

- facilitate test generation;
- facilitate test application; and
- avoid timing problems.

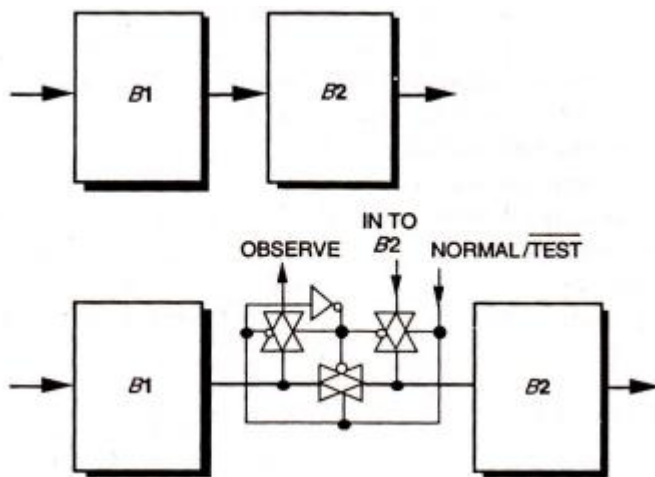
Improve controllability and observability

Design for test methods must ensure that a design is well enough covered to provide for complete and efficient testing.

When a node is difficult to access from primary input or output pads, then a very effective method is to add additional, internal pads to access the desired point.

These additional pads may be accessed using a prober.

If the node is a link between blocks of a circuit, as in Figure, Some additional circuitry will be required and a possible configuration is set out in the figure.



If the Normal/ Test line is set to 1 (Normal) then transmission gates T2 and T3 are open and T1 is closed. Normal transmission between the blocks can take place through T2 but a control input to block 2 can also be applied through T3.

When the Normal/ Test line is set to 0 (Test) then transmission gates T2 and T3 are closed, there will be no transmission between the blocks, and the output (observe) of block 1 can be monitored through f 1 which is now open.

This solution requires three pads and eight transistors in a CMOS environment.

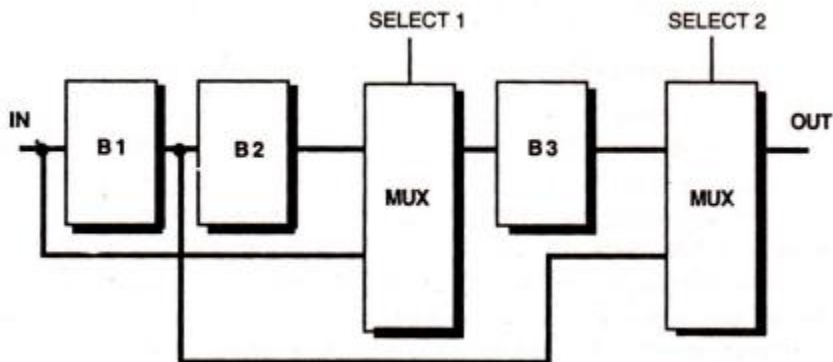
The use of Inter-block multiplexers

Some general attributes are illustrated in Figure.

This arrangement allows the bypassing of blocks.

The addition of demultiplexers also improves observability.

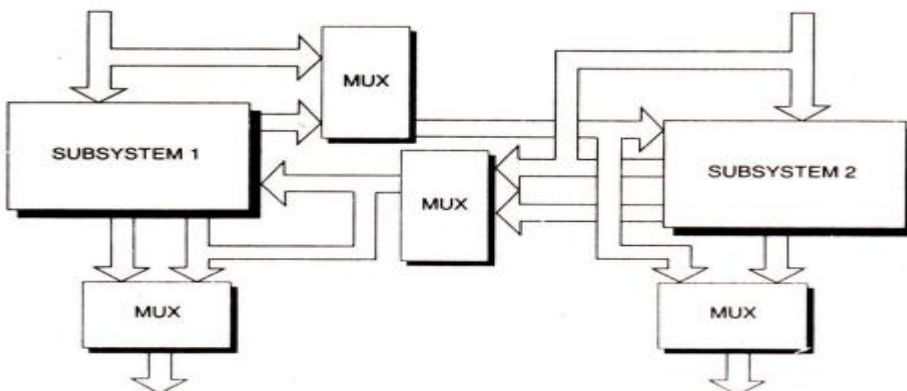
The major penalties incurred here are the numerous extra devices and the added propagation delays through the multiplexers.



The partitioning of large circuits

Partitioning large circuits into smaller subcircuits is an effective way of reducing test generation complexity and test time.

Isolation and control are readily achieved through the use of multiplexers as suggested in Figure .

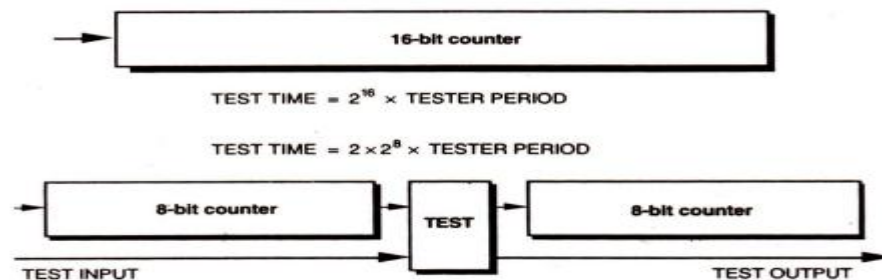


Dividing long counter chains:

Counters are sequential and need a large number of input vectors to be fully tested.

Partitioning into sub-counters can be very effective in reducing test complexity.

For example, the full testing of a 16-bit counter requires the application of $2^{16} = 65,536$ clock pulses. Division of the counter into two 8-bit counters, as Figure , reduces this number to $2 \times 2^8 = 512$ clock pulses.



Initialization of sequential logic

An important problem in sequential logic testing arises at power-up time where the first state will be quite random if there is no initialization.

In this case it is impossible to start a test sequence correctly.

The remedy is to design the circuit using elements which have a preset and/or clear facility (e.g. JK flip-flop elements with Pr. and Clr. inputs).

Asynchronous sequential logic

Asynchronous logic is driven by self-timing state transitions in response to changes of the primary inputs.

Although asynchronous logic is inherently faster than clocked logic it has several serious disadvantages from the test viewpoint as follows:

- testing is difficult;
- sensitivity to tester skew;
- non-deterministic behavior;
- prone to races and other hazards.

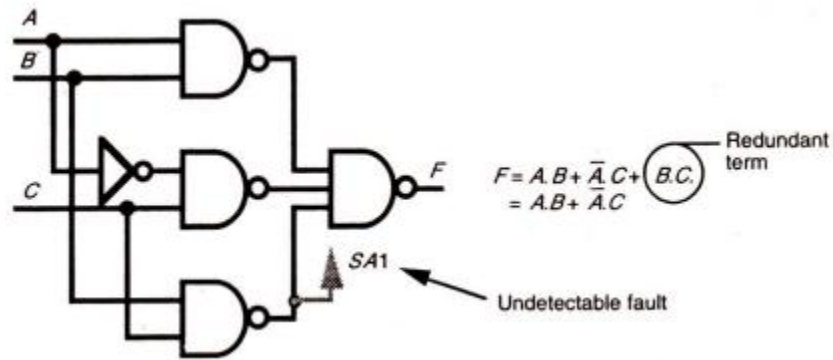
The design processes are more difficult than synchronous logic and must be approached with care, taking due account of critical race and other hazard-generating conditions.

Avoiding logical redundancy

Logical redundancy may be present by design; for example, in order to mask a static hazard condition, or unintentionally as a design bug.

In both cases it is not possible to make a primary output value dependent on the value of the redundant node.

Thus, there are certain fault conditions associated with the node which cannot be detected, Take, for instance, the two sets of conditions outlined in Figure .



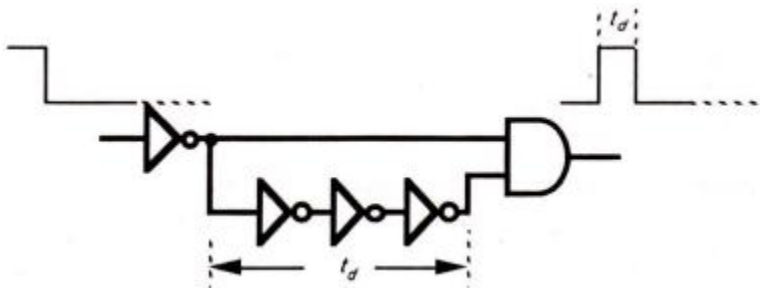
Avoiding delay dependant logic

An example of a delay-dependent circuit is given in Figure.

It will be seen that the presence of a pulse at the AND gate output depends not on the logical performance of the three inverters but rather on their temporal performance.

Automatic test pattern generators (ATPGs) work in the logic domain and view delay-dependent logic as redundant combinational logic.

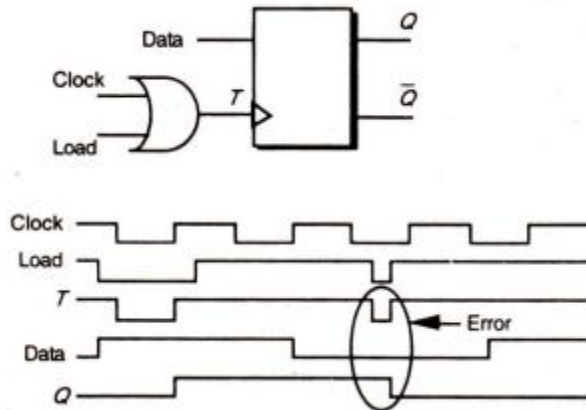
In the case illustrated in Figure , the ATPG will see the Anding of a signal with its complement and will therefore always compute a '0' as the output of the And. gate- rather than a pulse.



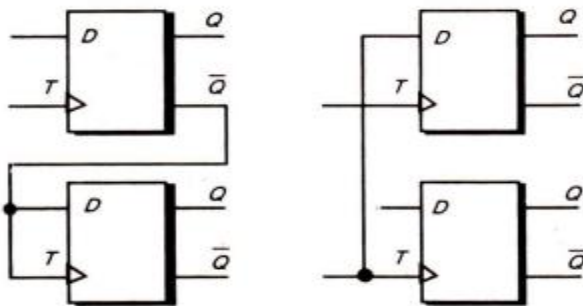
Avoiding gating or asynchronous delays

In the clock line When a clock signal is gated with another signal, such as a load signal coming from a tester, then any skew (or other hazard) on that signal can cause an erroneous output from the associated logic.

This is illustrated in Figure .



Further, another timing situation to avoid is that illustrated in Figure .

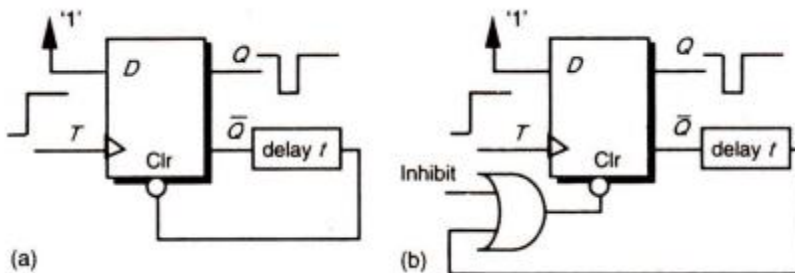


where the tester could not be synchronized if one or more clock is dependent on asynchronous delays (from the D-input to Q-input of the flip-flop in the figure), or when a signal is used both as data and as a clock.

Avoiding self-resetting logic

The problems here are akin to those in asynchronous logic, since the reset input is independent of the system clock. This can result in an erroneous value being read by the tester.

The situation is indicated in Figure(a).

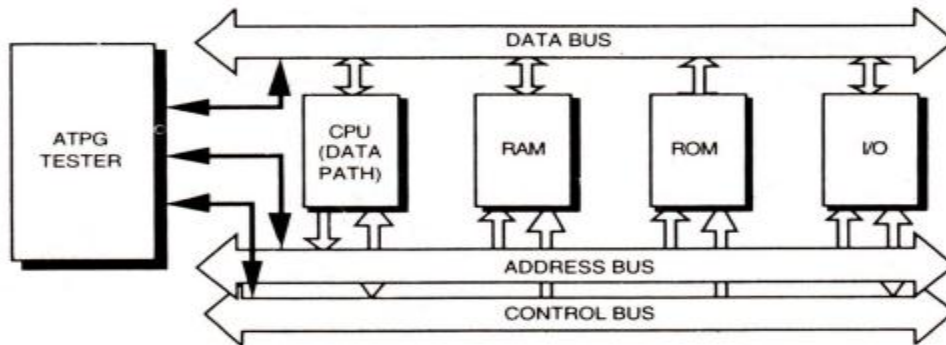


One solution to this problem is to allow the tester to override by adding an Or gate as indicated in Figure (b).

This allows the tester to receive the right response at the right time.

The use of bused structures

This approach is related to the partitioning technique and is very widely used for microprocessor like circuits as illustrated in Figure.



Using this arrangement allows the tester access to all the main subsystems and other modules which the buses interconnect.

The tester can then effectively disconnect any unit or module from the bus by putting its output into the high impedance state. Test patterns can then be applied to each separately.

Separation of analog and digital circuits

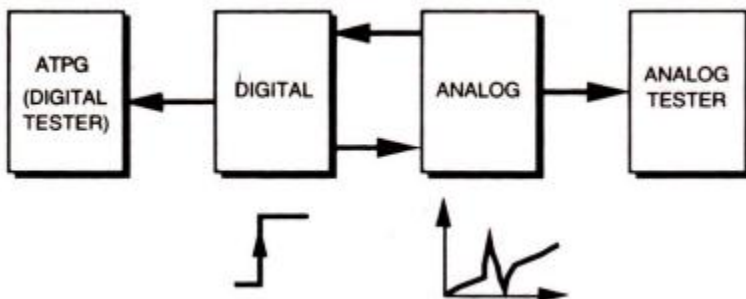
The testing of analog circuits requires a completely different strategy from digital circuits. and is therefore incompatible.

Furthermore, the fast rise and fall times of digital signals can give rise to cross-talk problems in analog signal lines if they are in close proximity.

Where it is essential to route digital signals near analog lines, then consideration must be given to balancing and shielding the digital signals.

In the case of analog-digital converters, it is better to bring out the analog signals for observation before conversion.

For digital-analog conversion the digital signals may also be brought out for observation prior to the converter as outlined in Figure .



ADC TESTING: BRING OUT ANALOG INPUTS FOR TEST OBSERVE DIGITAL OUTPUT

DAC TESTING: BRING OUT DIGITAL INPUTS FOR TEST OBSERVE ANALOG OUTPUT

Bypassing techniques

Bypassing a subsystem consists of providing the facilities for propagating its inputs directly through to its outputs.

The aim is to bypass the sub-system in order to directly access another subsystem to be tested, and, as with partitioning, wide use is made of multiplexers to achieve the bypassing.

To speed up the testing, some subsystems may be tested simultaneously if the propagation paths are associated with other disjoint or separate subsystems.