

Basic VLSI Design  
Unit - IV

- 1) Switch logic pass transistor
- 2) Gate logic inverter,
- 3) NAND gates
- 4) NOR gates
- 5) Pseudo nmos
- 6) Dynamic CMOS Eg of Structured design
- 7). Parity generator,
- 8). Bus arbitration
- 9). multiplexers
- 10). logic function block
- 11). Code converter

Advantages

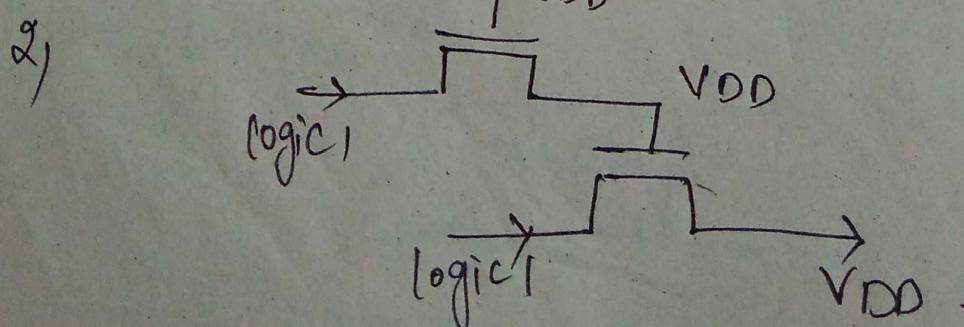
- 1) Requires minimum geometry
- 2) Do not dissipate standby power, Supply to ground.

## Switch logic

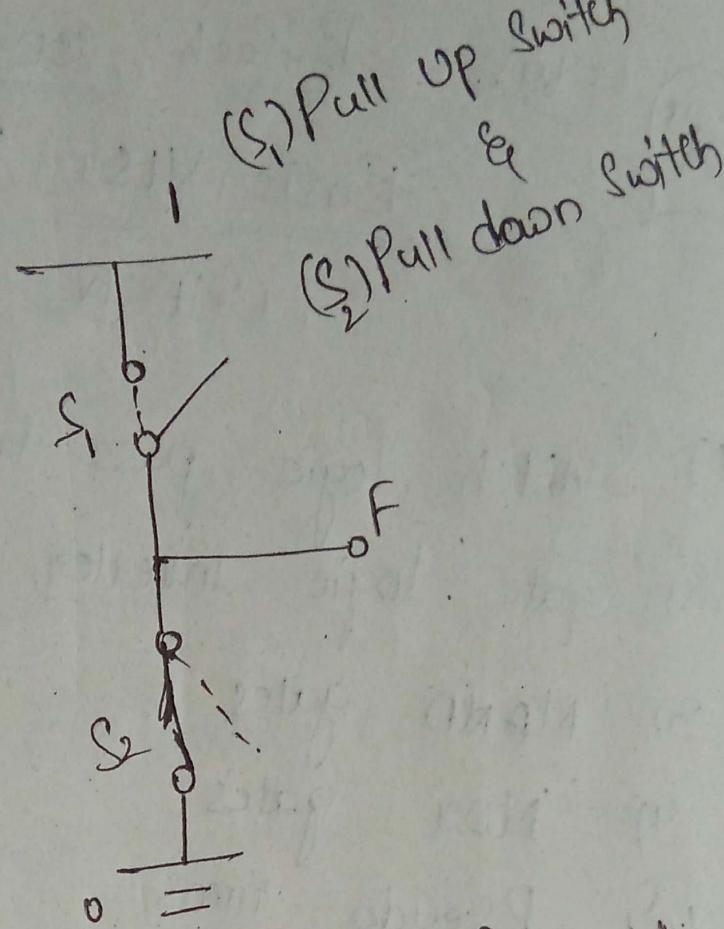
- 1) It is mainly based on Pass-transistor or transmission gates
- 2) It is fast for small arrays and takes no static current from the supply  $V_{DD}$ .
- 3, Hence power dissipation of such array is small since current only flows on switching.
- 4, Switch logic is analogous to logic arrays based on relay contacts, difference in path through each switch

## Pass transistor:-

- 1) This logic uses transistors as switch to carry logic signals from node to node instead of  $V_{DD}$  & ground.



## Switch



\* The Switch  $S_1$  is closed &  $S_2$  is open = 1

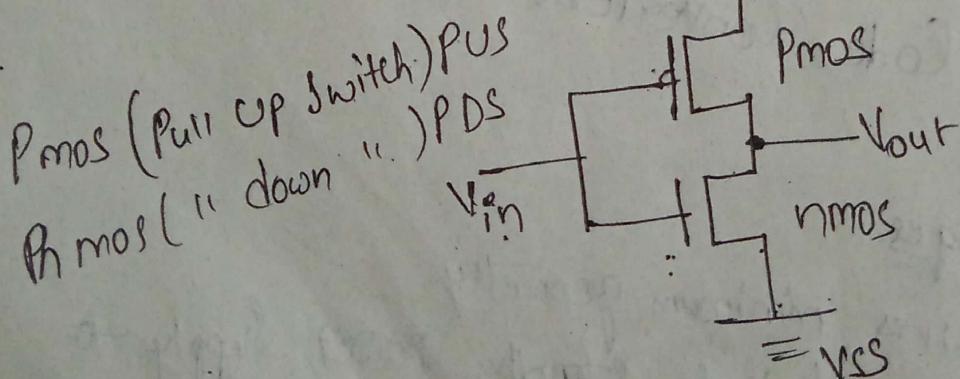
\* " "  $S_1$  is open &  $S_2$  is closed = 0

\* " "  $S_1$  &  $S_2$  is open = 0

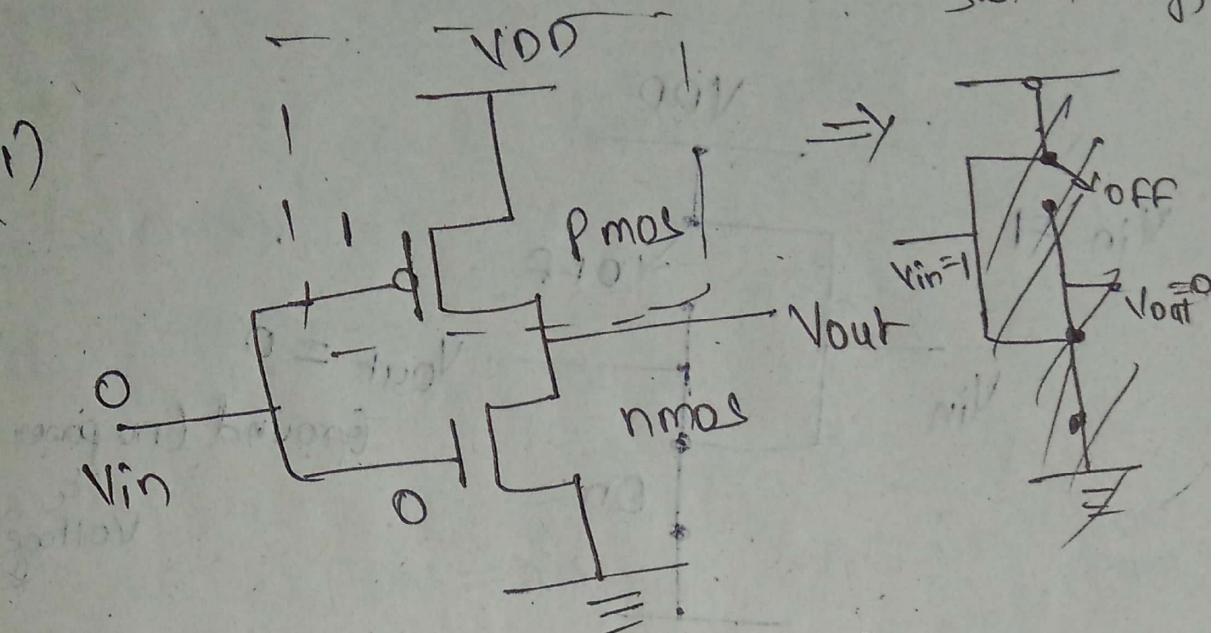
$$CMOS = nMOS + pMOS$$

## CMOS

## Inverter



(1)

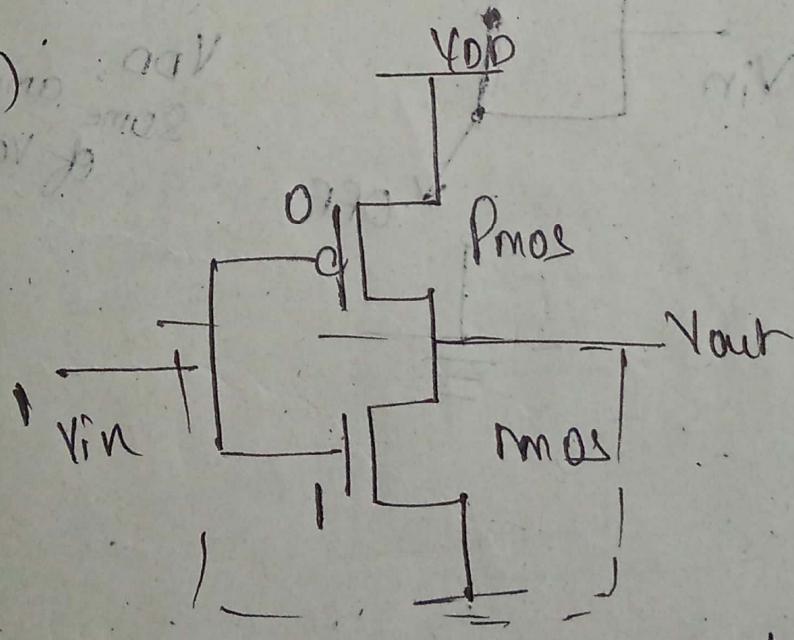


$V_{in} = 0 \Rightarrow$  PMOS is high (1)

NMOS is low (0)

Output of Y is connected to  $V_{DD}$ . So it is pull up network

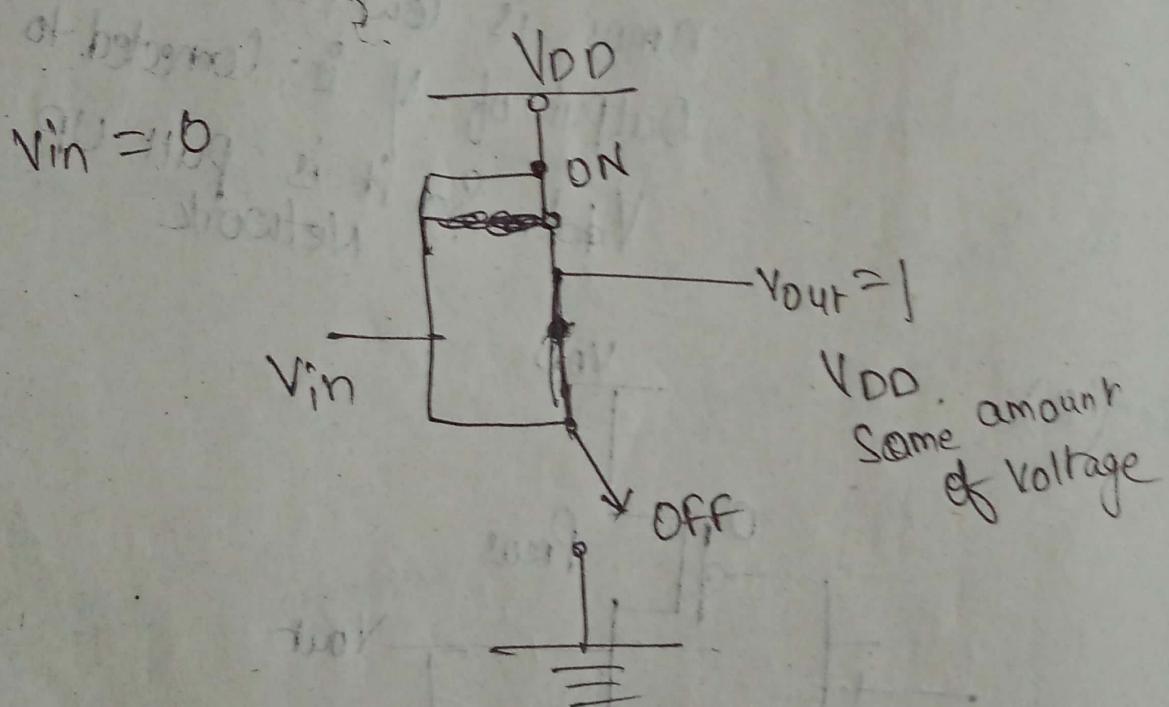
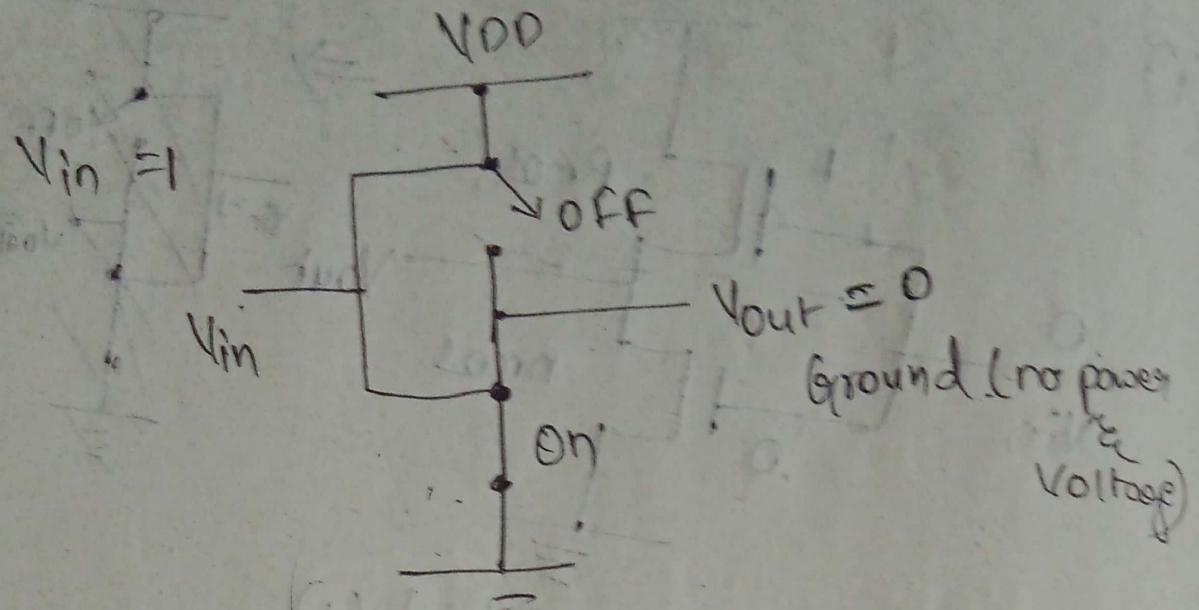
(2)



$V_{in} = 1 \Rightarrow$  PMOS is low (0)

NMOS is high (1)

Output of Y is connected to ground pull down network



# NAND Gate = CMOS

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

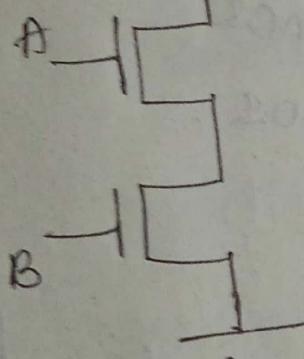
$\overline{A} \cdot \overline{B} \rightarrow$  Parallel.

0	0	1	0
1	0	1	1
0	1	1	0
1	1	0	1

$A \cdot B \rightarrow$  Series

$$A \cdot B (1) P_{DN} = n\text{mos}$$

$V_{DD}$

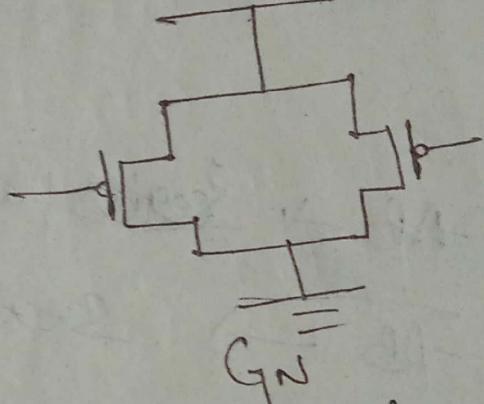


Series

NAND

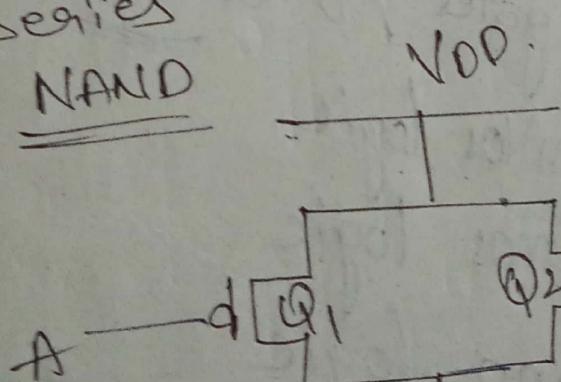
$$A \cdot B (0) P_{UN} = p\text{mos}$$

$V_{DD}$



Parallel.

$V_{DD}$



$P_{MOS}$

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

$A \rightarrow Q_3$  N Mos

$B \rightarrow Q_4$

$\equiv GND$

A	B	Pmos Q <sub>1</sub> ON	Q <sub>2</sub> ON	Nmos Q <sub>3</sub> OFF	Q <sub>4</sub> OFF	Y
0	0					1
0	1		ON, OFF		OFF ON	1
1	0		OFF ON		ON OFF	1
1	1		OFF OFF	ON ON	ON ON	0

~~AB~~ NOR

$AB \rightarrow$  Series parallel — nmos

$\overline{AB} \rightarrow$  Series — pmos

- Alternate logic gates / other CMOS logic forms
- 1) Other logic
  - 2) Pseudo nMOS logic
  - 3) Dynamic CMOS logic
  - 4) CMOS Domino logic
  - 5) Clocked CMOS logic
  - 6) DEPMOS logic



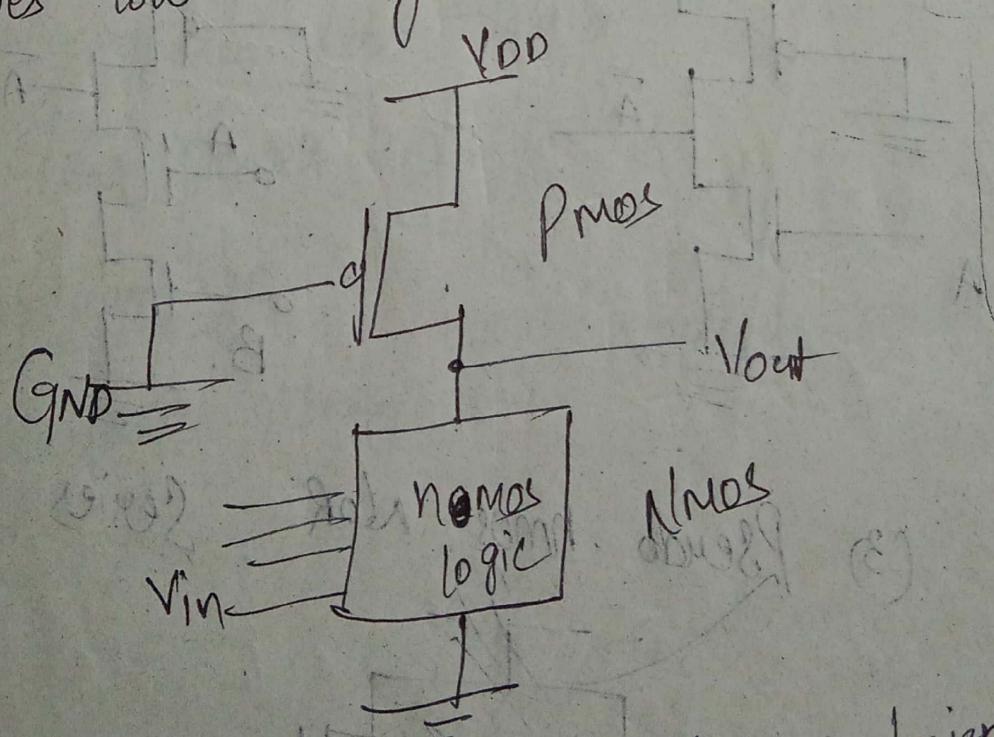
## PSEUDO nMOS logic

Pseudo nmos logic is one type of alternate gate circuit that is used as a supplement for the complementary Mos logic circuit, and higher speed than static mos logic.

1) The gate terminal of the pmos transistor is connected to the ground.

2) It remains permanently in the ON state.

3) Depending on the I/P Combinations output goes low through the PDN



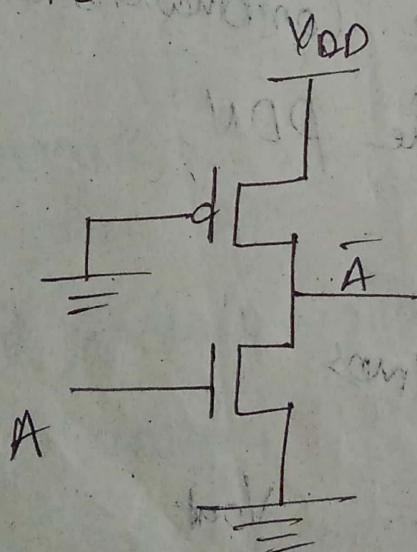
Pseudo nMOS logic design

4) Only the nMOS logic ( $Q_n$ ) is driven by the input Voltage.

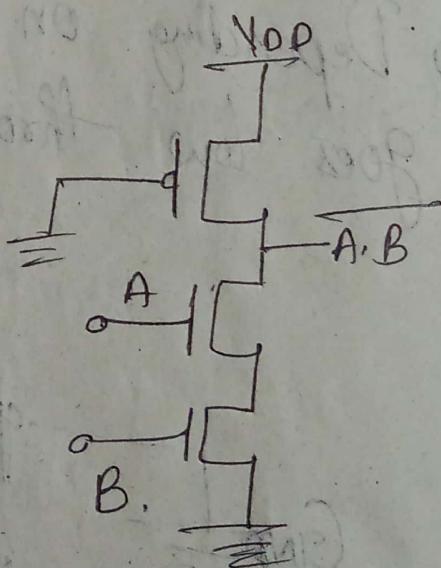
5) While the gate of ptransistor is connected to ground and pmos acts as an active load for  $Q_{out}$ .

6) Except the Pseudo-nMOS gate circuit is identical to the ~~P&N (PDN)~~ of the Complementary CMOS gate.

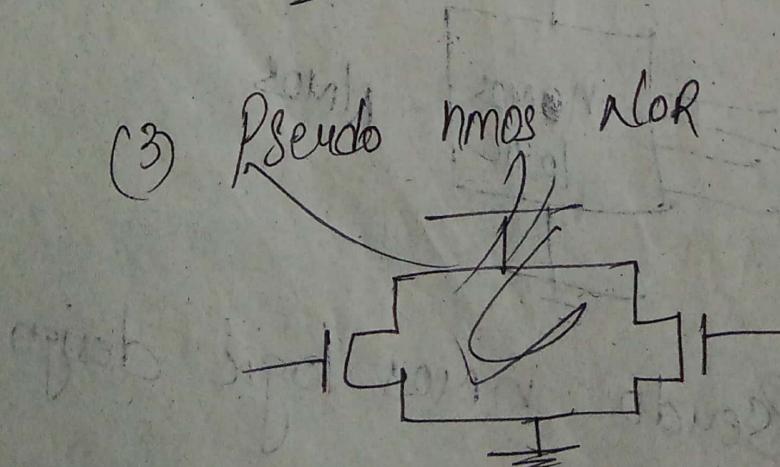
(1) Pseudo nMOS inverter



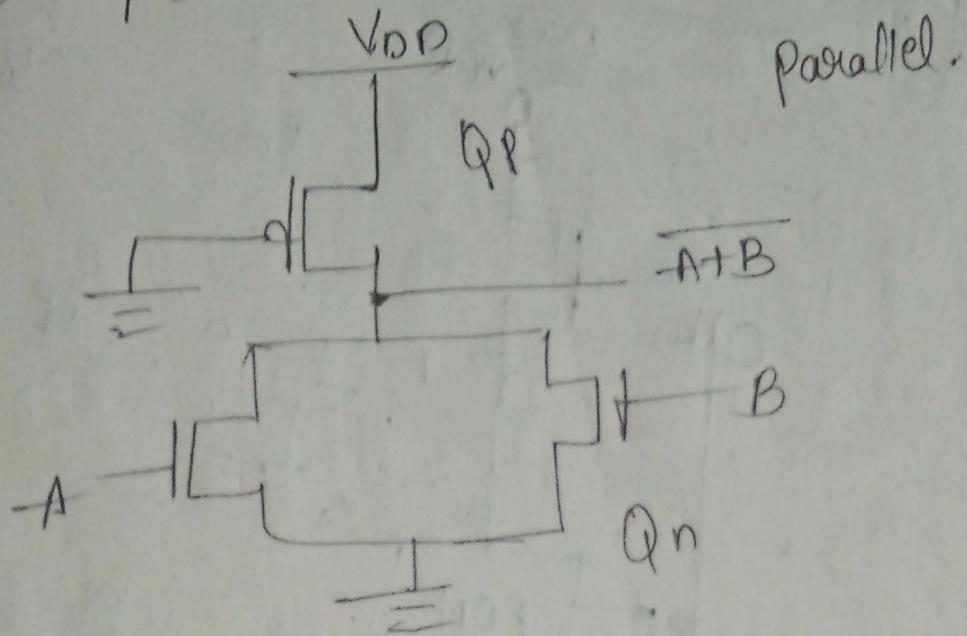
(2) Pseudo nMOS NAND



(3) Pseudo nMOS NOR Series



### (3). Pseudo NMOS NOR

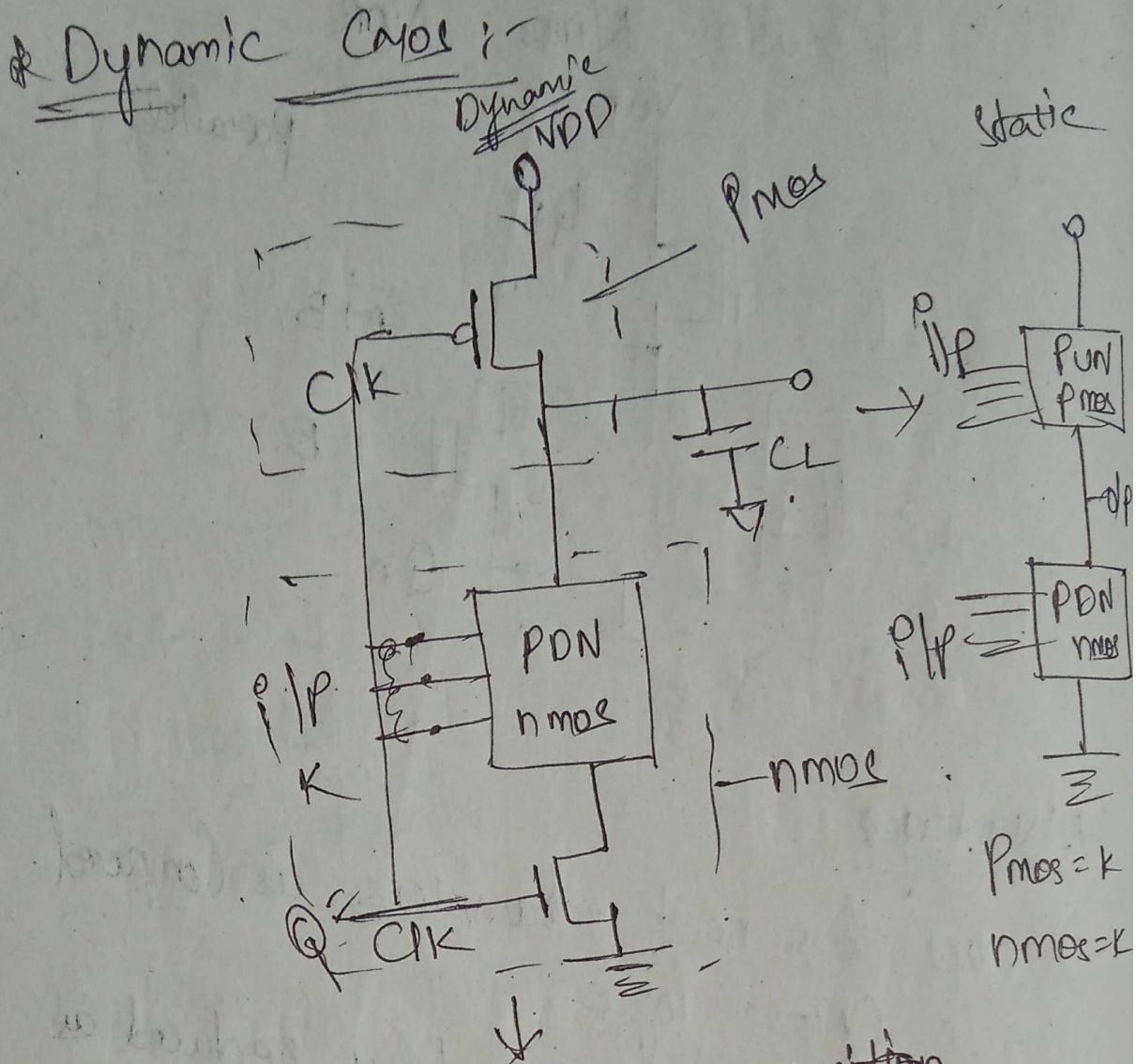


#### Advantages

- 1) uses less no. of transistors as compared to CMOS logic.
- 2) Geometrical area & delay reduced as it requires less transistor.
- 3) low power dissipation.

#### Disadvantage

layout design is critical.



Total no. of  $P_{MOS} = 1.00$  position.

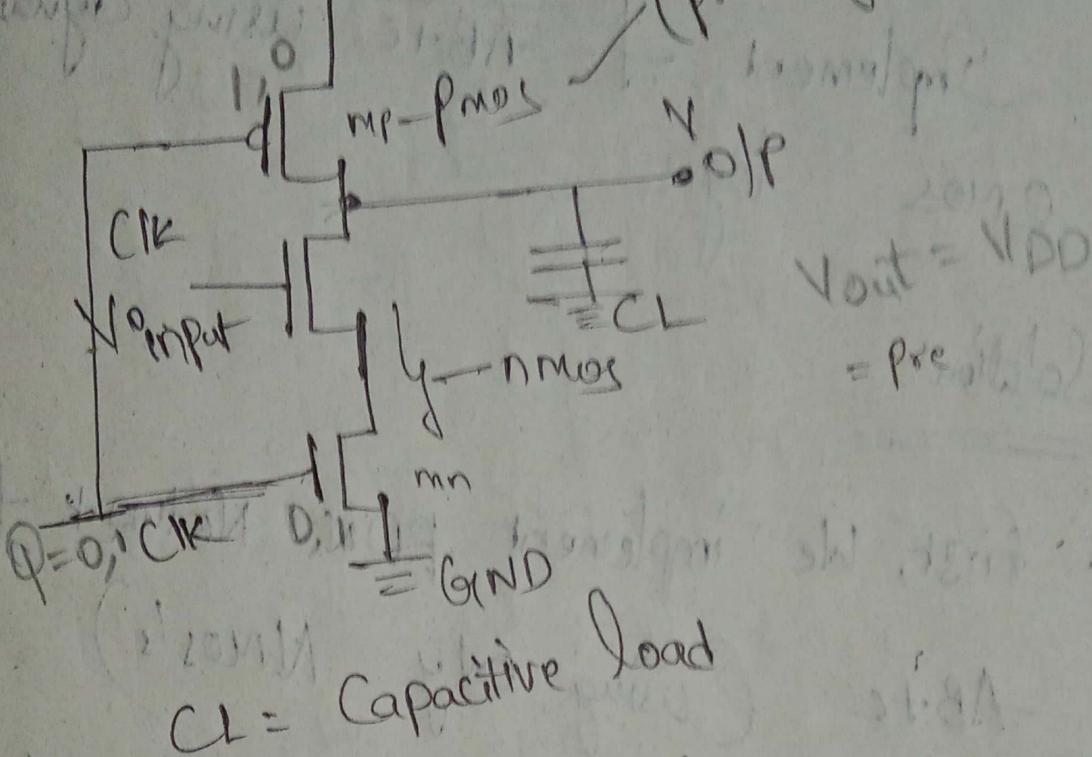
$$n_{MOS} = K + 1$$

\* Compare to Static CMOS we use Dynamic CMOS because less no. of  $P_{MOS}$

\* Total size & loading Capacitive is less  
Dynamic CMOS switching characteristic is faster than

Static CMOS  
\* an alternate logic style called dynamic logic is presented that obtains a similar result while avoiding S.P. Consumption

Dynamic CMOS (It relies on temporary storage of signals and values on the capacitance of high impedance (at precharge) node).



\* Circuit works in 2 modes

- 1) Pre-charging (p-MOSFET) (mp)
- 2) Evaluation. (n-MOSFET) (mn)

1) Pre-charging — When  $Q = 0$ ,  $M_P = 1$ ,  $M_N = 0$

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

It is a precharge FET

2) Evaluation = When  $Q = 1$

~~Eg:~~

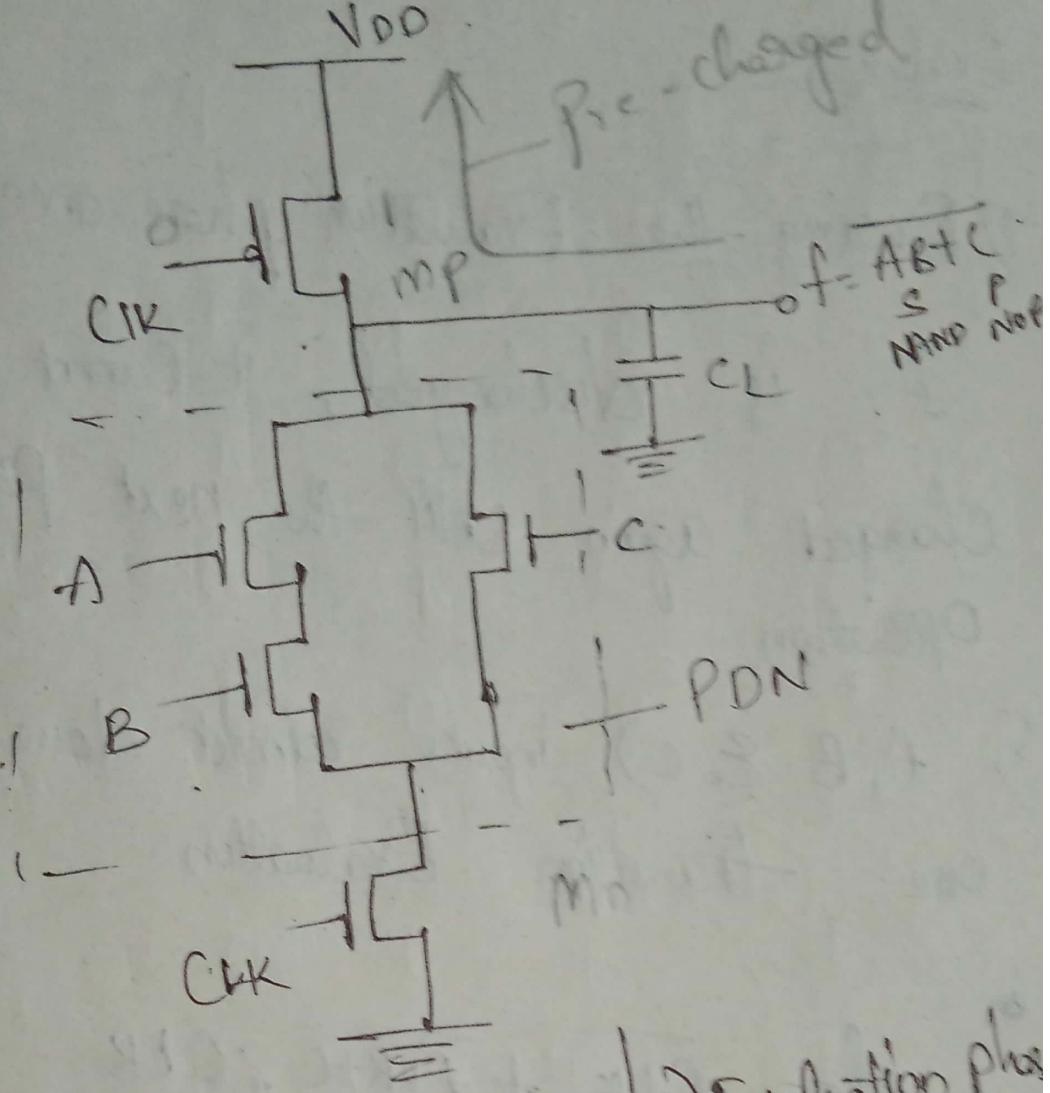
Implement  $f = \overline{AB} + C$  using dynamic CMOS logic

Solution

1. First, we implement PON Network for  $AB + C$  (using only NMOS's)

2. Now, add  $M_p$  &  $M_n$  and apply  $Clk$  to gate of  $M_n$  and  $Clk$  to the gate of  $M_p$ .

4. We also add Capacitances  $C_L$  at the output node.



- 1) Pre-charge phase  $CK^{20}$
- 1)  $CLK=0$  the output node 'f' is pre-charged to  $VDD = ON$
- 2)  $M_P = ON, M_N = OFF$
- 3)  $C_L$  charges to  $VDD$ .
- $\therefore f = VDD$       pre-charged phase

4)

- 2) Evaluation phase  
 $CLK=1$
- 1)  $CLK=1$  the  
 $M_P = OFF, M_N = ON$
- 2)  $C_L$  charges to pull down Network of  $\overline{AB+C}$
- 3) low resistance path exists b/w output f and GND

## Evaluation

4) During the Evaluation phase, once o/p 'f' is discharged, it cannot be charged again until the next pre-charge operation

5) (A, B & C) inputs almost done by one transition Evaluation

6)

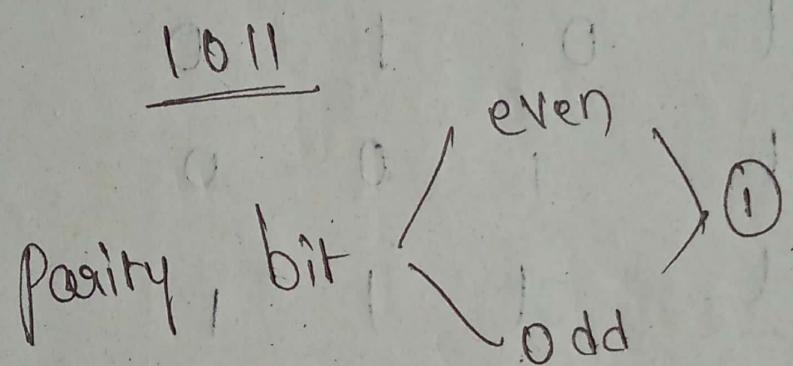
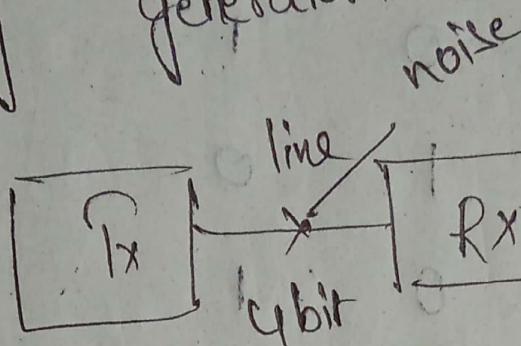
$$f = \overline{C}LK + \overline{A} \cdot \overline{B} + \overline{C} \cdot CLK.$$

$$CLK=1 \quad f = AB + C \quad \text{Evaluation}$$

$$CLK=0 \quad f = NO \quad \text{Precharge}$$

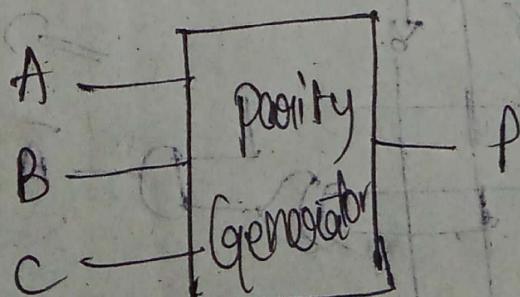
# Parity generator

- \* It is the concept to detect the error
- \* A single bit error is detected by a parity generator.



		Parity				
1	0	Even			odd	
0	1	odd				

~~Odd Even~~ → Even parity generator



Even parity generator

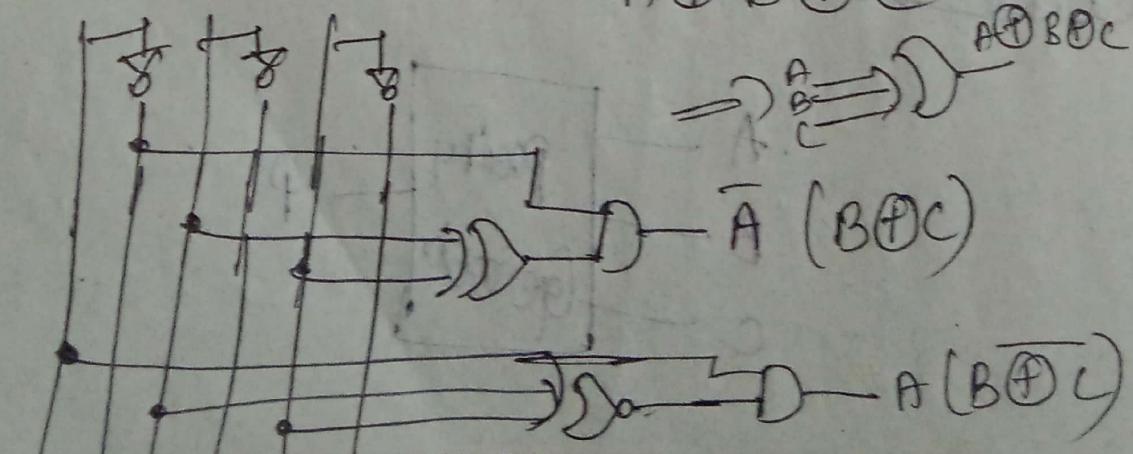
A	B	C	P
0	0	0	0
0	0	1	1
0	1	0	1
0	1	1	0
1	0	0	1
1	0	1	0
1	1	0	0
1	1	1	1

$$P = \overline{A}\overline{B}C + \overline{A}B\overline{C} + A\overline{B}\overline{C} + ABC$$

$$= \overline{A}(\overline{B}C + B\overline{C}) + A(\overline{B}\overline{C} + BC)$$
$$\Rightarrow \overline{AB} + B\overline{A} = A \oplus B \quad \overline{AB} + AB = A \oplus B$$

$$= \overline{A}(B \oplus C) + A(\overline{B} \oplus C)$$

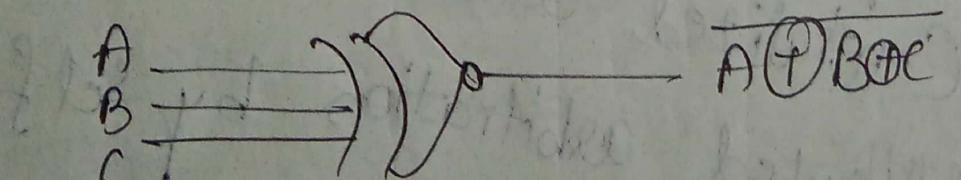
$$A \cdot B \cdot C = A \oplus B \oplus C$$



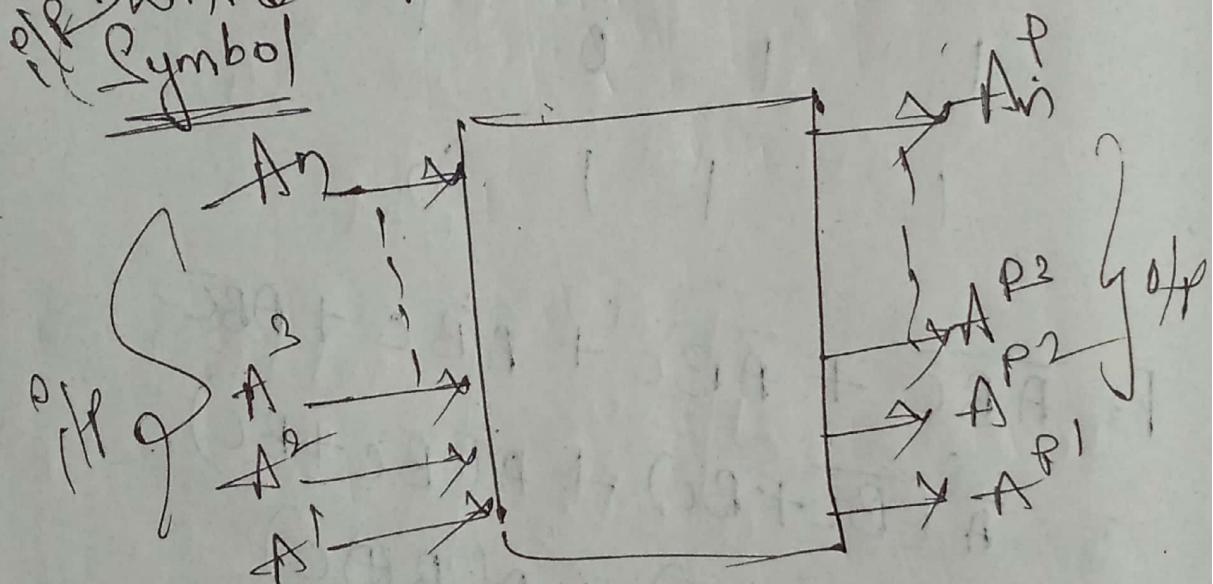
odd parity generator

A	B	C	P
0	0	0	1
0	0	1	0
0	1	0	0
0	1	1	1
1	0	0	0
1	0	1	1
1	1	0	1
1	1	1	0

$$\begin{aligned}
 P &= \overline{ABC} + \overline{A}Bc + A\overline{B}c + ABC \\
 &= \overline{A}(\overline{B}c + Bc) + A(\overline{B}c + Bc) \\
 &= \overline{A}(B \oplus c) + A(cB \oplus c) \\
 &= \overline{A \oplus B \oplus c}
 \end{aligned}$$



Bus arbitration = It is the process by which the current bus master, which accesses the bus and passes it to another bus requesting access, mainly used write operation. It is controlled by processor unit. It is Read(R) output.



### Types of Bus arbitration:-

- \* Daisy chain arbitration.
- \* Centralized arbitration.
- \* Distributed arbitration by self.
- \* Distributed arbitration by collision

## Bus arbitration

Centralized      Distributed

Centralized

bus arbitration

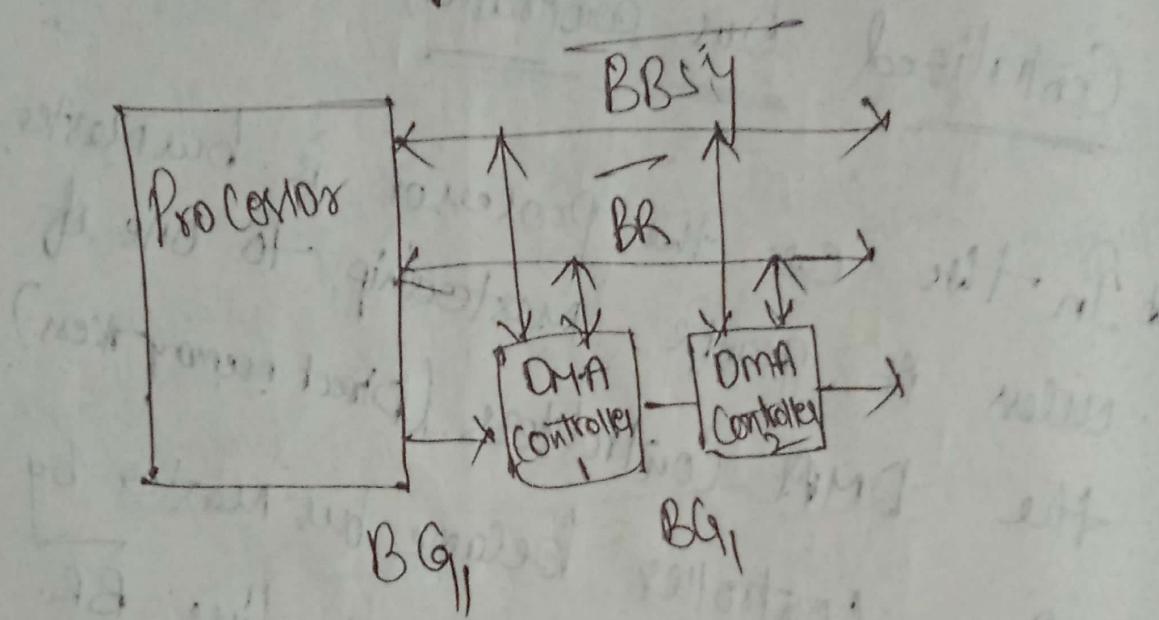
\* In this case the processor is bus master.  
unless it grants mastership to one of  
the DMA controllers (Direct memory access).

\* DMA controller became bus master by  
activating the bus request line BR.

\* The signal on the Bus Request line is the  
of the bus request from all the devices  
connected to it.

A fallen bus-request is activated the  
Processor activates the Bus-grant  
signal, BG1.

\* The Current Bus Master indicates to all devices that is using bus by activating another line called Bus-Busy - BB<sup>SY</sup>

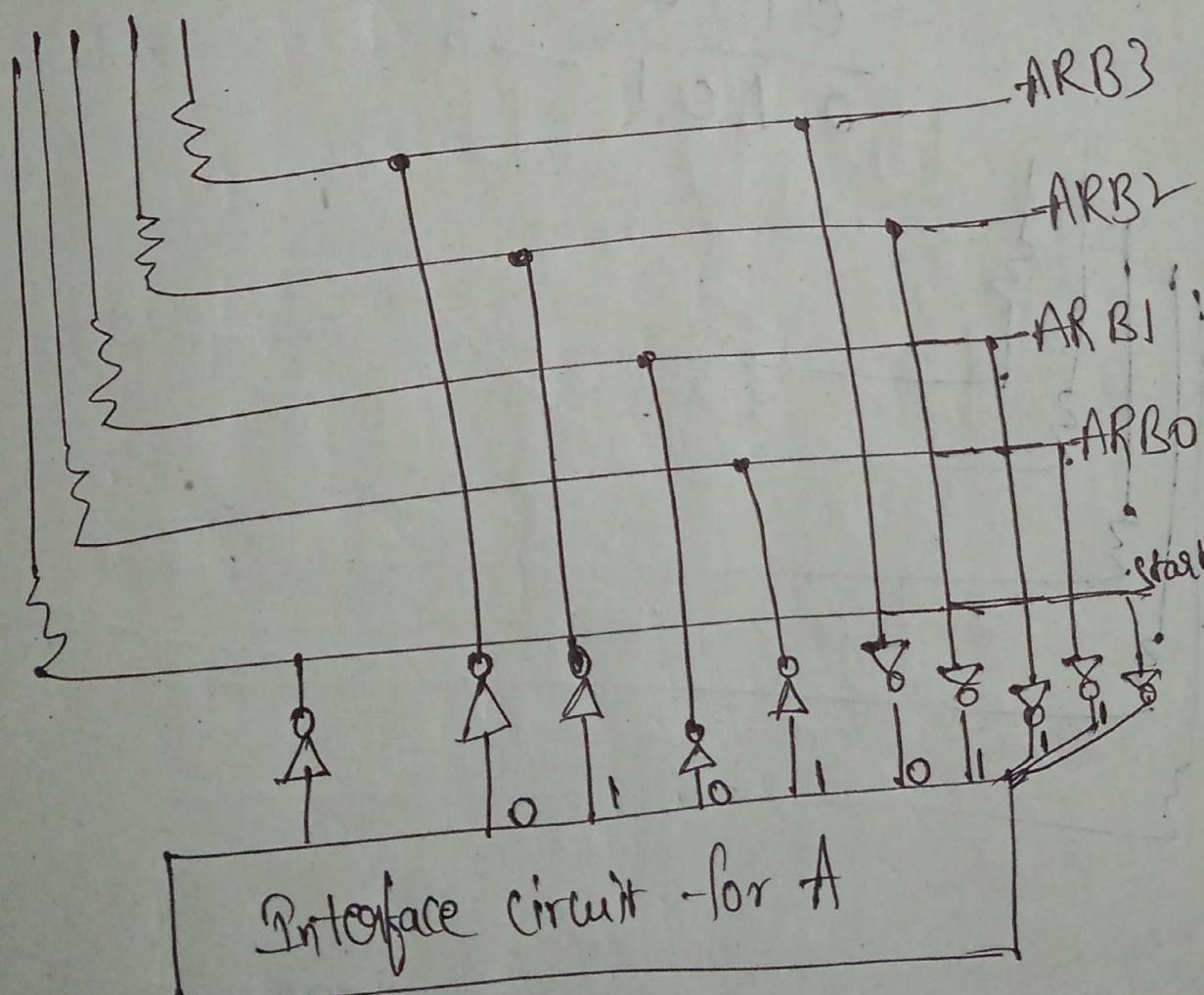


### Distributed Arbitration:

Distributed Arbitration means that all devices have the responsibility in bus. They have equal responsibility to use the bus.

Carrying out the arbitration process without using a central arbiter.

- \* Each device on Bus is assigned a 4-bit identification number
- \* When one (8) more devices request the bus, they assert the Start-Arbitration signal and place their 4-bit ID number on  $\overline{ARB_0}$  to  $\overline{ARB_3}$
- \* A winner is selected as a result of interaction among the signals transmitted over these lines

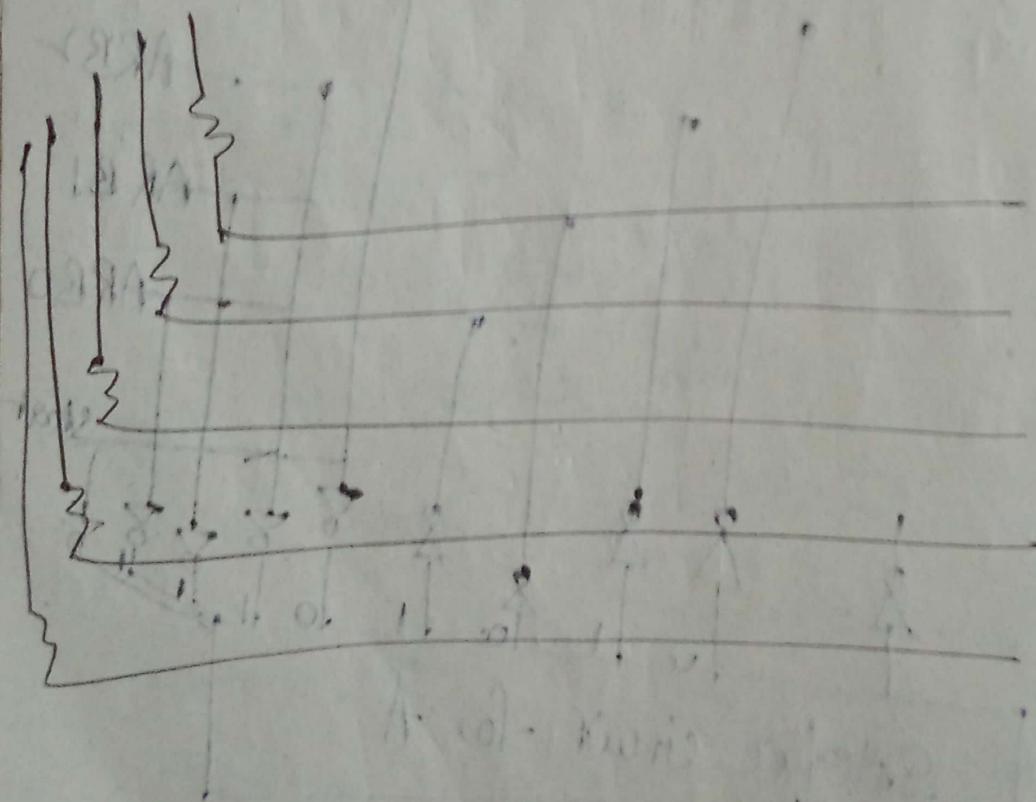


$$\begin{array}{lcl} A = 5 & = & 0 \ 1 \ 0 \ 1 \\ B = 6 & = & 0 \ 1 \ 1 \ 0 \end{array}$$

$$\begin{array}{r} A = 5 \\ - \\ B = 6 \\ \hline 1011 \end{array}$$

$$\begin{array}{r} A = \\ 0111 \\ - \\ 0101 \\ \hline 0100 - 4 \end{array}$$

$$\begin{array}{r} B = \\ 0111 \\ - \\ 0110 \\ \hline 0101 - 5 \end{array}$$



Interface Circuit for B

## Unit - 5

### Subsystem Design and Layout

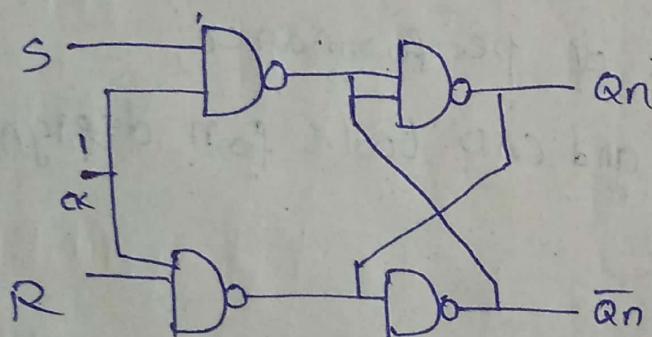
- 1) clocked sequential circuits,
- 2) dynamic shift registers,
- 3) bus lines,
- 4) General considerations,
- 5) 4-bit arithmetic processes,
- 6) 4-bit shifter,
- 7) Regularity Definition & computation.
- 8) practical aspects and testability:
- 9) Some thoughts of performance,
- 10) optimization and CAD tools for design and simulation.

①

## Clocked Sequential Circuits

- 1) SR flip flop.
- 2) D flip flop.
- 3) T flip flop.
- 4) JK flip flop.

SR flip flop :- (Set Reset flip flop).



Truth table :-

S	R	P.S	N.S
0	0	x	No change
0	1	x	Reset
1	0	x	Set
1	1	x	Indeterminate

∴ clock = 1

## Simplification

$$(0,0) \Rightarrow Q_n = 0 \times \bar{Q_n} = \bar{0} = 1 \quad \text{Reciprocal of } \bar{Q_n} = Q_n$$

$$\bar{Q_n} = 0 \times Q_n = \bar{Q_n} =$$

Next stage  $\begin{cases} Q_{n+1} = Q_n \\ \bar{Q_{n+1}} = \bar{Q_n} \end{cases}$ , it is called as No change condition.

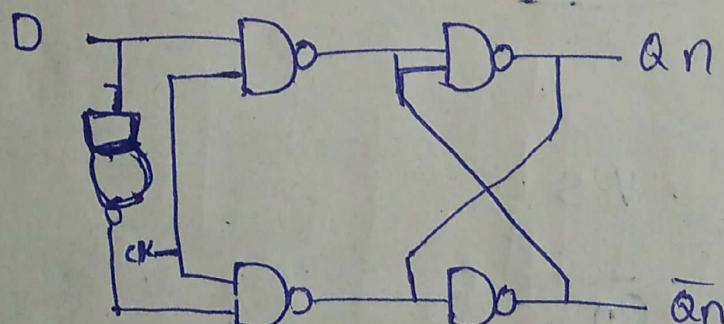
Here, Set = 1, Reset = 0

$$(1,0) \Rightarrow 0 \times \bar{Q_n} = \bar{0} = 1$$

$$1 \times Q_n = Q_n \quad \text{Reciprocal of } \bar{Q_n} = Q_n$$

Based on NAND gate operation, the present & Next stage output is decreased.

D-flip flop :- (delay flip flop.)



Truth table

D	P.S	N.S
---	-----	-----

0	x	Set
---	---	-----

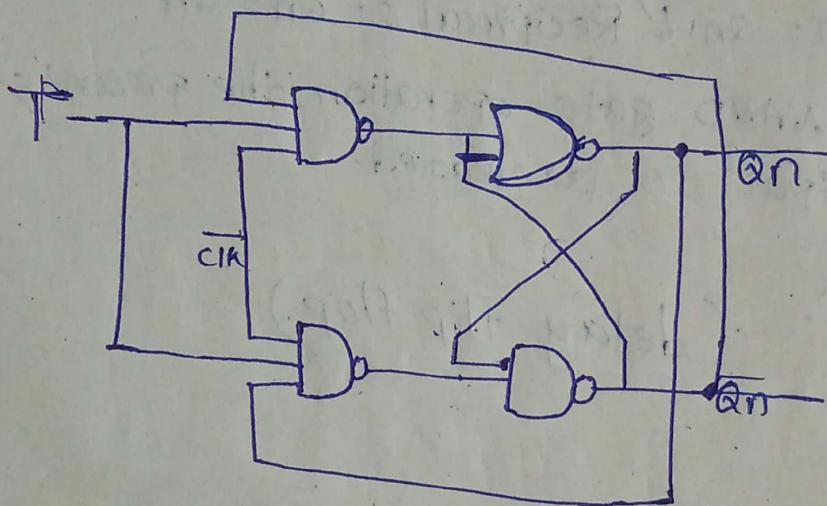
1	x	Reset
---	---	-------

D flip flop is used to donot delay the

Previous Next stage output of the Reset (RS)  
flip flop : Eg:- set & Reset

T flip flop :- (Toggle flip flop)

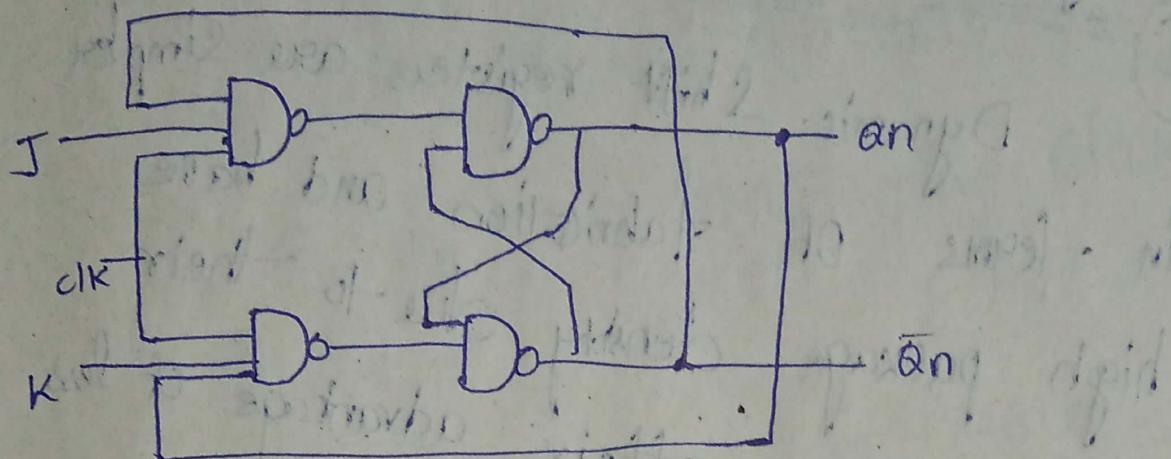
T - flip flop is used to find the  
In determinet stage of SR flip flop.



Truth table :-

T	P-S	N-S
0	x	$\bar{Q}_n$
1	x	$Q_n$

## JK flip flop



Truth table

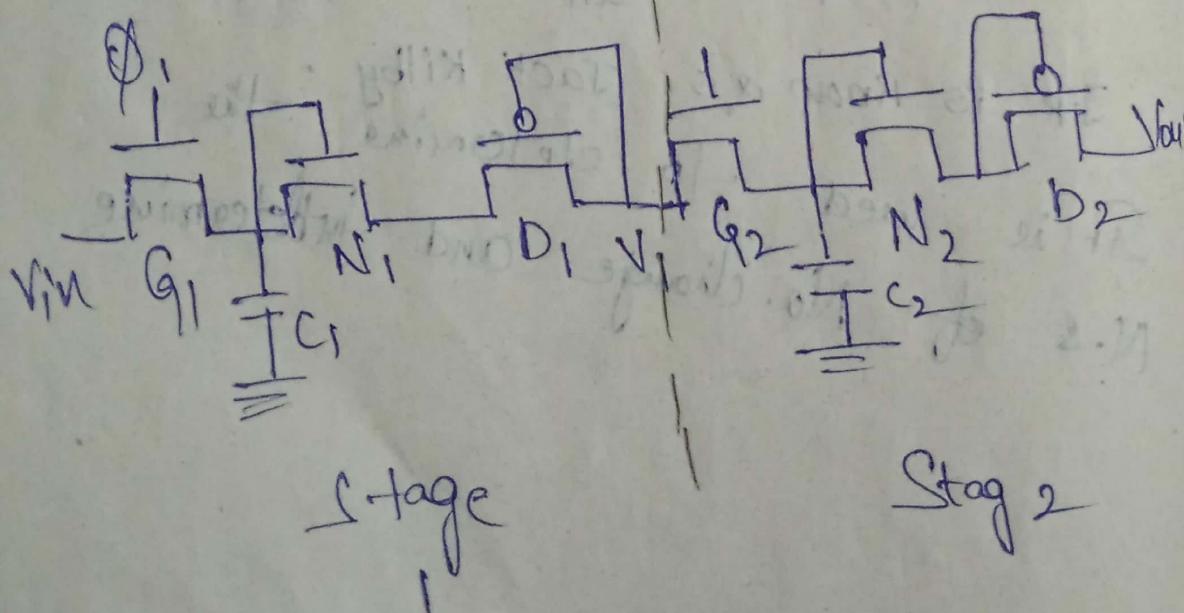
J	K	$Q_{n+1}$
0	0	$Q_n$
0	1	$\bar{Q}_n$
1	0	$\bar{Q}_n$
1	1	$Q_n$

JK is known as Jack Kilby.  
It is used to determine the  
N.S of No. change and indeterminate.

## Dynamic Shift register

②

Dynamic shift registers are simpler in terms of fabrication and have high package density due to their smaller size. Their advantage of less power consumption is caused by the fact that the power consumed increases with the increase in frequency. There are many design variations available in case of dynamic shift registers.



Dynamic Shift Register.

Working:-

\*  $G_1$  and  $G_2$  NMOS depletion mode  
 $(D_2, N_2)$  &  $(D_1, N_1)$  Inverters

\*  $\phi_1$  &  $\phi_2$  non-overlapping clock  
signals

&  $C_1$  and  $C_2$  are gate-to-source  
Stage 1 and Stage 2.

\* Capacitance is a initial state

1.  $V_{in} = 0V$  and  $\phi_1 = V_{DD}$ , the logic state  
0 and 1 respectively.

2. Gate  $G_1$  (-on) the Capacitor  $C_1$   
will remain in its Uncharged State.

3. The Output Voltage level of the inverter  
Circuit in Stage 1 (form by  $D_1$  &  $N_1$ ) to

4.  $g_o$  high  
 $V_1 = V_{DD}$  Zero threshold voltage for all  
the devices in the circuit.

5. When  $\phi_1 = 1$  amount of time, as the charging of the capacitor is a gradual process.

6. the gate  $G_2$  closes due to the capacitor  $C_2$  starts to charge through it gradually to the voltage level of  $V_{DD} = V_1$ .

7. The capacitor  $C_2$  increases, the O/P voltage at the stage 2 decrease due to the circuit formed by formed  $D_1$  and  $N_2$ .

8. The Output Voltage,  $V_{out}$  to go low if

$$V_{in} = V_{out}$$

Similarly

1. If  $V_{in} = V_{DD}$  while  $\phi_1 = V_{DD}$ , then  $C_1$  charges to  $V_{DD}$  through  $G_1$ .

2. The Output Voltage of Stage 1,  $V_1 = 0$

if  $\Phi_2 = V_{DD}$

3.  $G_{12}$  (off). the Capacitor  $C_2$  discharges

the Output Voltage.

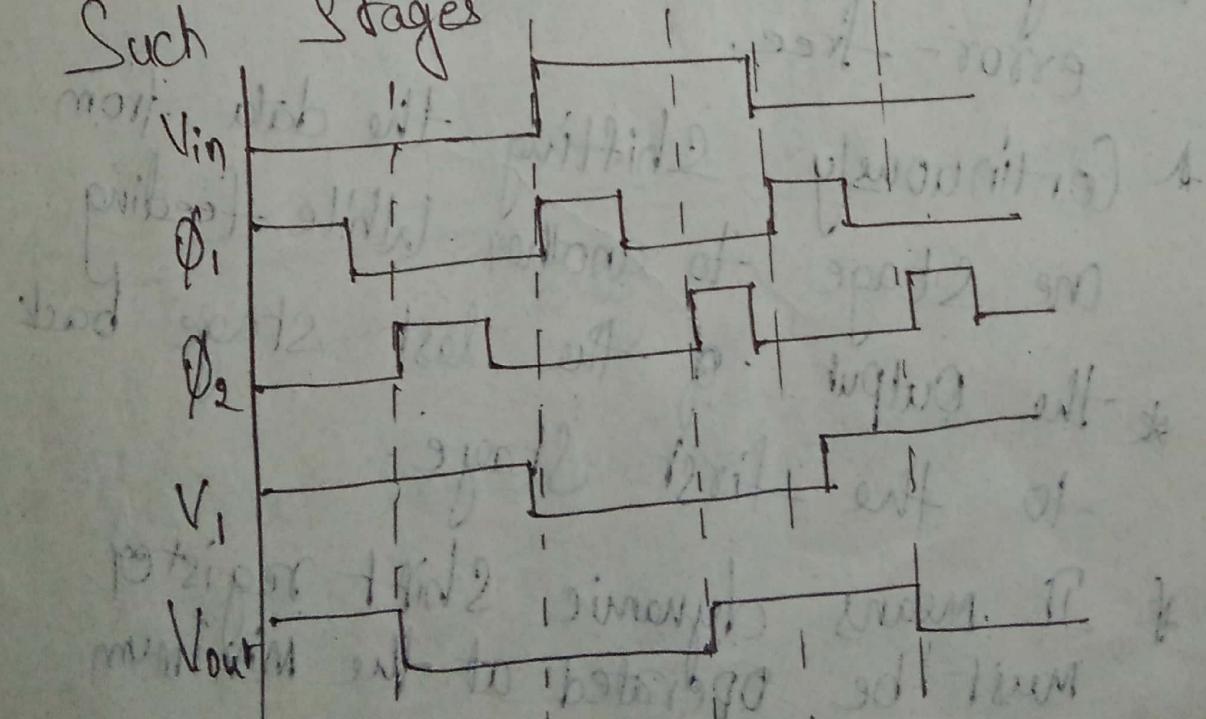
4.  $V_{out}$  increases gradually  $V_{out} = V_{DD}$

reflecting the logic high 1 state of  
input voltage.

5.  $V_{in}$  reflected at  $V_{out}$  which implies  
that the  $V_{in}$  is shifted to  $V_{out}$  under  
the Control of the Clock.

6. The Stage 1 acts as dynamic shift  
register

7. It is designed by Cascading n number of  
Such Stages



# Timing diagram of dynamic shift register

Working:-

- \* The single stage dynamic shift register by timing diagram.
- \* It is store the information in the form of charge on the gate-to-substrate.
- \* This charge is due to leakage and thus one requires refreshing the data periodically in order to ensure that the data stored are error-free.
- \* Continuously shifting the data from one stage to another while feeding the output of the last stage back to the first stage.
- \* It means dynamic shift register must be operated at the minimum

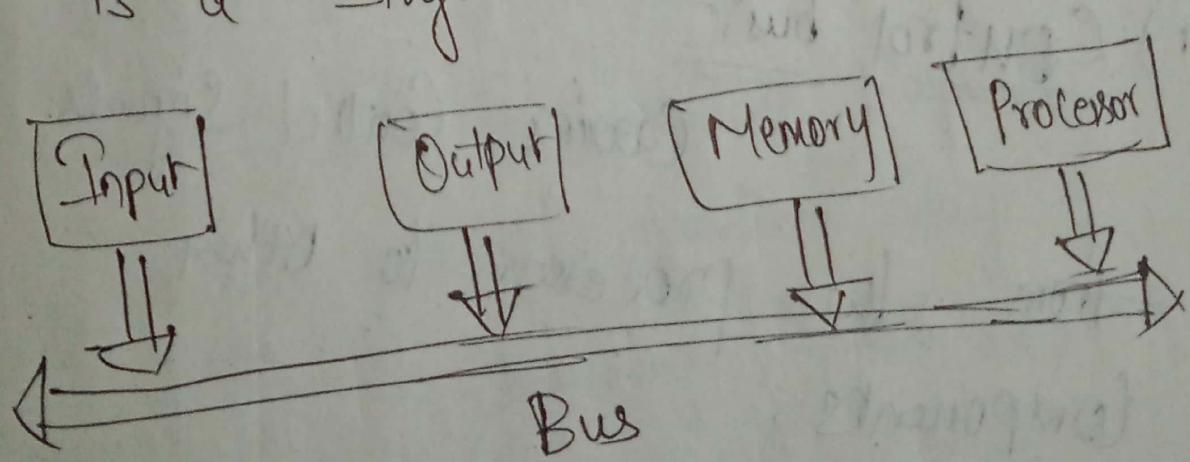
Clock frequency

## Bus lines:- ③

Bus used to Connect different parts of System. A group of lines that serves as a connecting path for several devices is called bus.

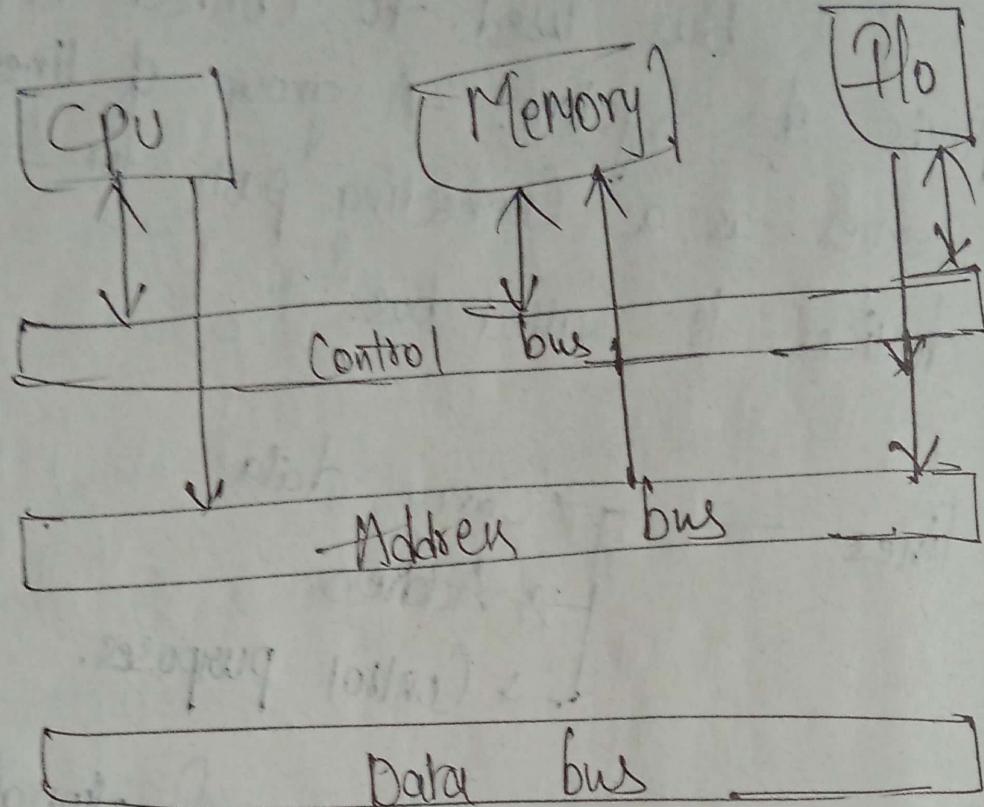
lines →  
→ Carry data.  
→ Address  
→ Control purposes.

\* Simplest way to connect functional units is a single bus



\* This are having some interconnections.

## Architecture:-



### 1) Control bus

Carrier Control Signals

from the processor to other components.

### 2. Address bus

Carrier Memory address

from the processor to other components

such as primary storage and input output devices

3. Data bus:- Carries the data between the Processor and other components.

General Considerations: - ①

The microprocessor includes ALU, Control Unit, I/O Unit and memory. here only ALU (A) data path is considered. the data path by itself is further divided into Subsystem (Shifter) is considered.

\* The Consideration provides ways of holding problems, produces way of designing & realizing Systems which are too to complex.

\* Lower Unit Cost with different approaches Suitable for some requirement

~~Lower Unit Cost is applicable.~~

\* Higher reliability:-

High levels of System  
Integration greatly reduces ~~cost~~ into this  
function provides good reliability.

\* Lower power dissipation:-

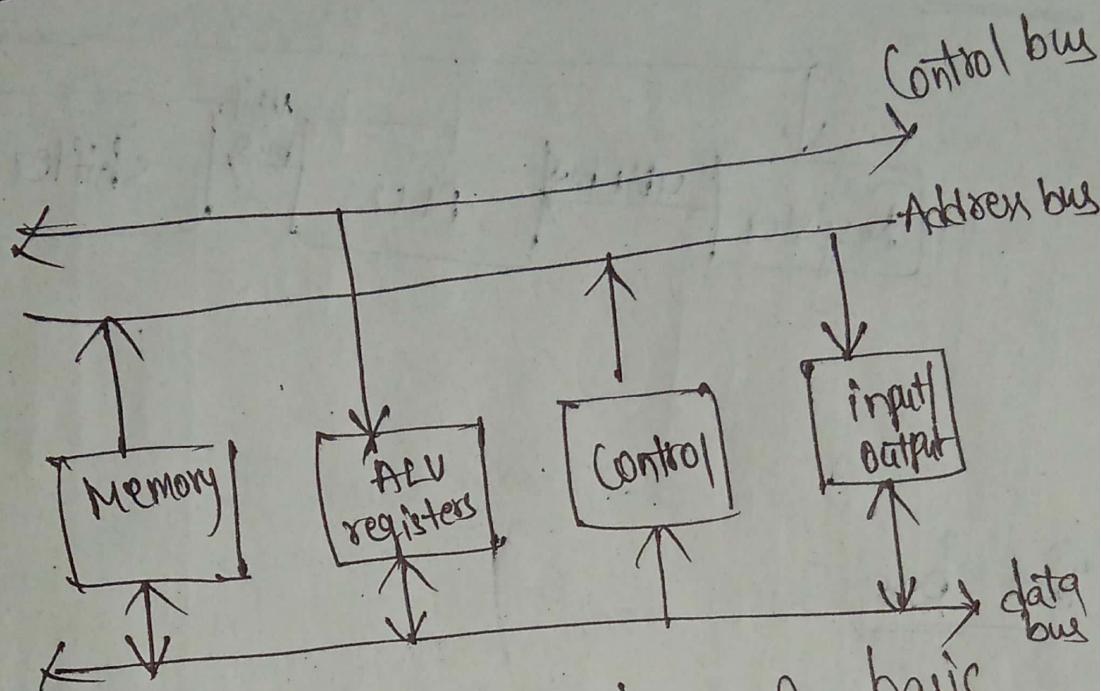
Lower weight & lower  
volume in comparison other approaches.

\* Better performance:- particularly in terms  
of speed power product is good.

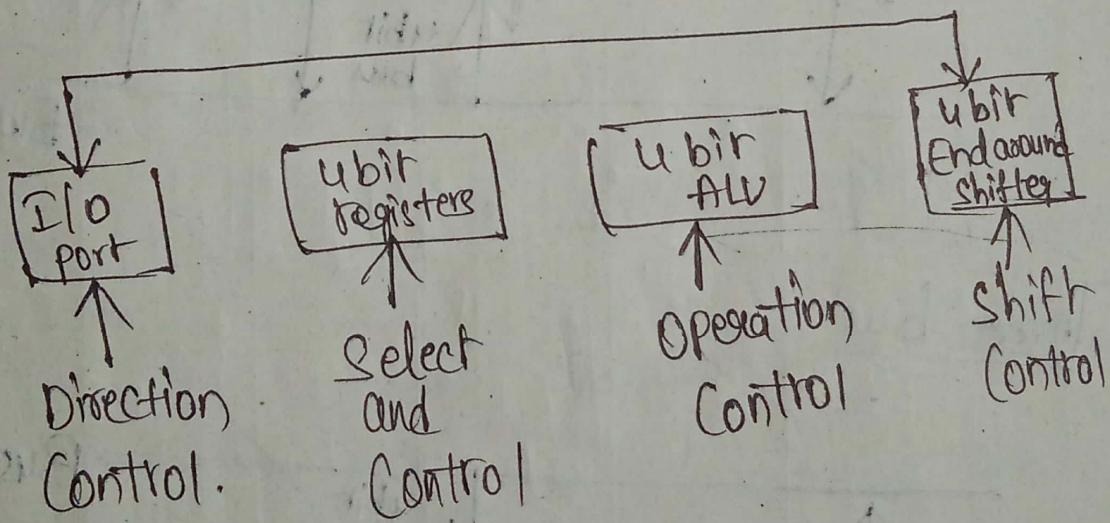
\* Enhanced repeatability:- There are fewer  
processes to be controlled the whole  
system (i) very large part on single  
chip.

\* Possibility of reduced design development period

# 4-bit Arithmetic Processor - ⑤

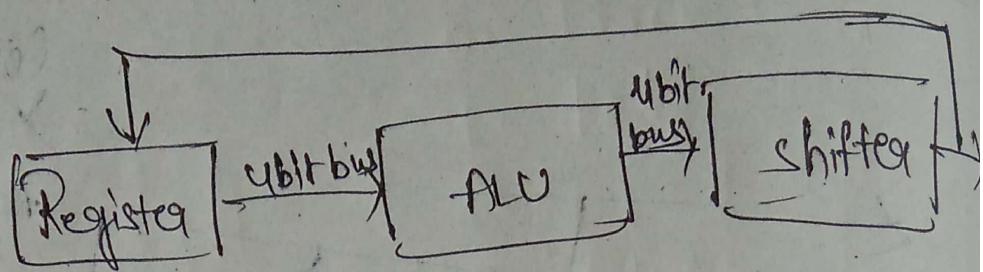


ALU registers having internal basic  
Subsystem it is known as data path.

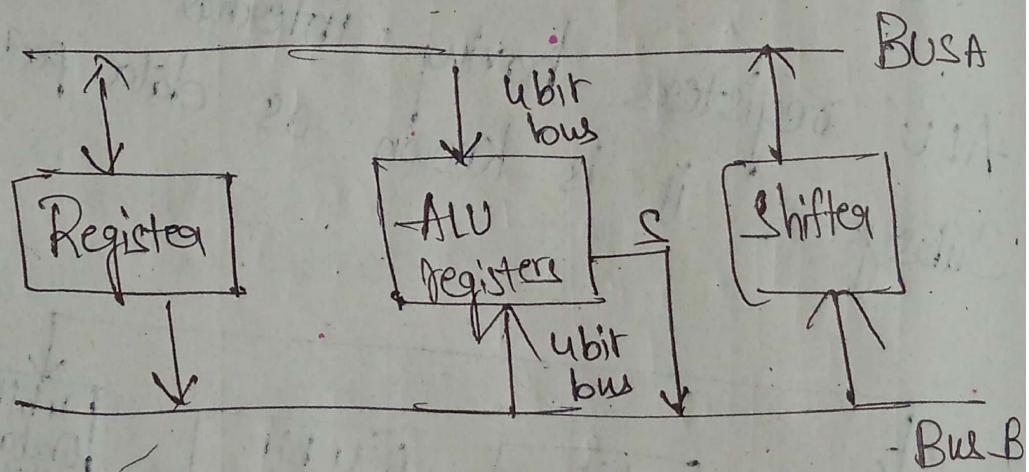


Basic interconnection data path.

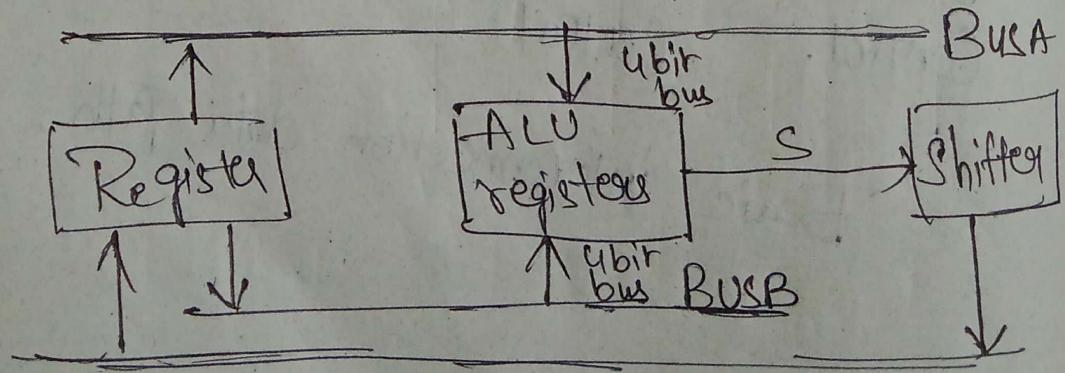
One bus



Two bus



Three bus



## 4-bit Shift register:-

Any n-bit shifter should be able to shift incoming data by up to  $(n-1)$  places in a right-shift direction. (8)

The shifter must have:-

1. Input from a four-line parallel data bus
2. Four output lines for the shifted data
3. Means of transferring input data to output lines with any shift from Zero to three bits inclusive.

## Design of 4-bit Shift register:-

Any general purpose n-bit shifter.

Should be able to shift incoming data by up to  $(n-1)$  places in a right-shift direction. (8)

# Design of 4-bit Shifter

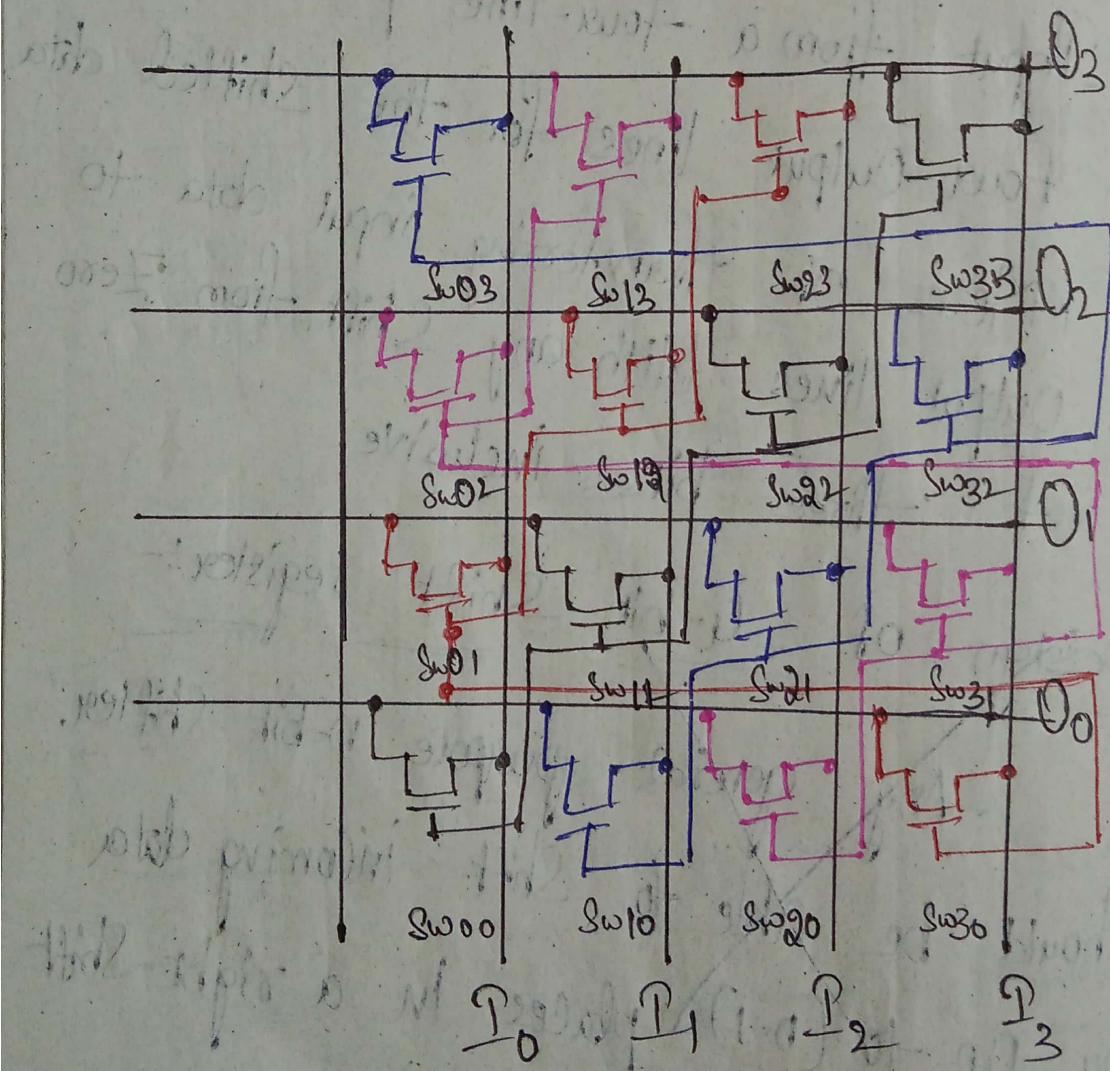
shift	$O_0$	$O_1$	$O_2$	$O_3$
0	$I_0$	$I_1$	$I_2$	$I_3$
1	$I_3$	$I_0$	$I_1$	$I_2$
2	$I_2$	$I_3$	$I_0$	$I_1$
3	$I_1$	$I_2$	$I_3$	$I_0$

0 - black

1 - Red

2 - Pink

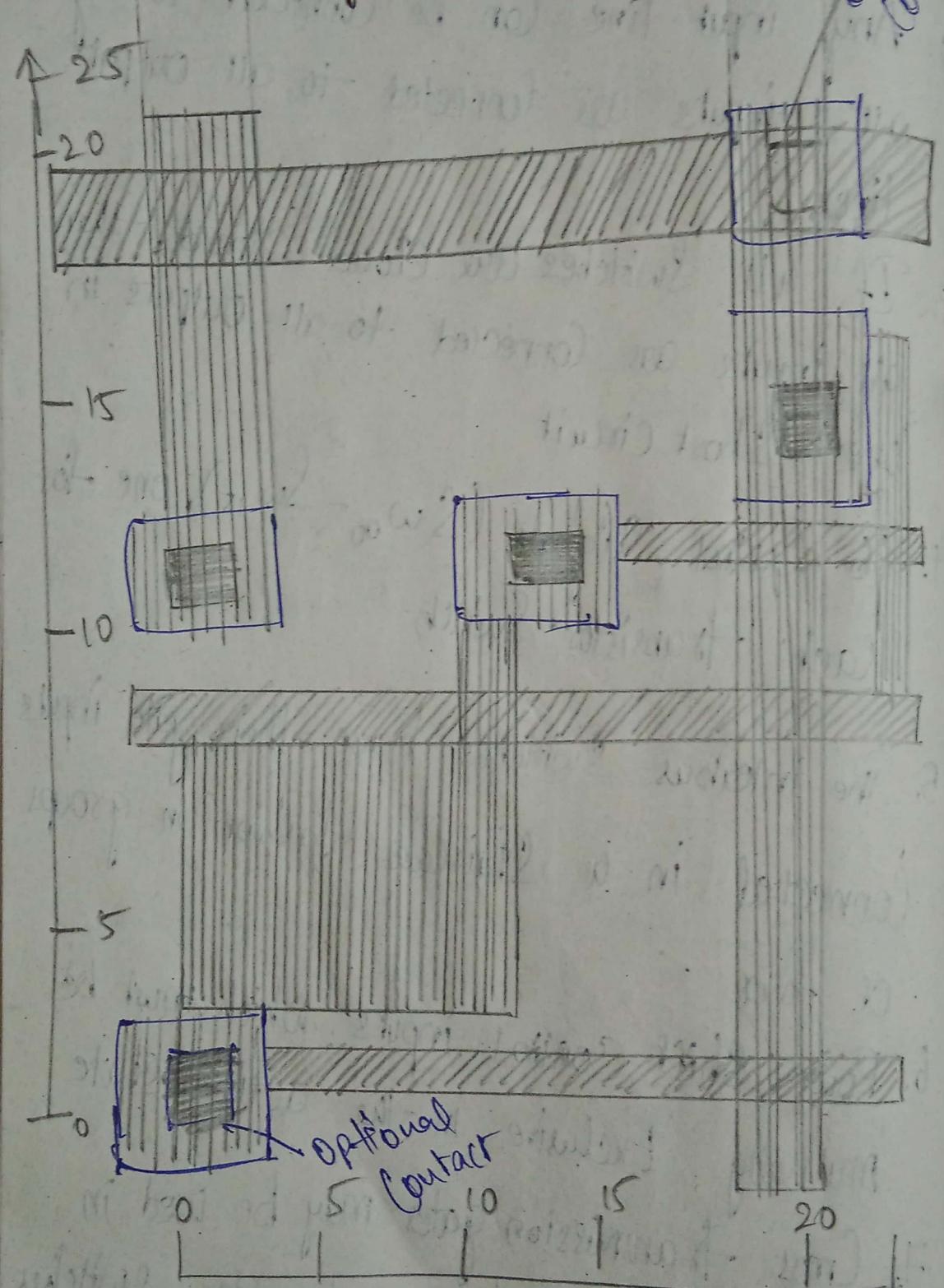
3 - Blue



1. The Arrangement is quite general and may be readily expanded to accommodate  $n$ -bit inputs / outputs.
2. Any input line can be connected to any or all outputs. All inputs are connected to all outputs.
3. If all switches are closed, then all inputs are connected to all outputs in one short circuit.
4. 16 Control Signals ( $S_{W00} - S_{W15}$ ), one for each transistor switch.
5. The interbus switches have their gate inputs connected in a staircase fashion in groups of four.
6. Four Shift Control inputs which must be mutually exclusive in the active state.
7. CMOS transmission gates may be used in place of the simple pair transistor switches.

Barrel Shifter Standard Cell 2-

Mask layout.



## Regularity definition:-

Regularity is used to reduce the design, Simplify the verification and for Correct by Construction.

Eg:- RAM Composed of identical cells

## levels of Regularity:-

- 1) At transistor level
- 2) At identical Gate level
- 3) At micro block
- 4) At macro block

