

Important topics - Unit-3}

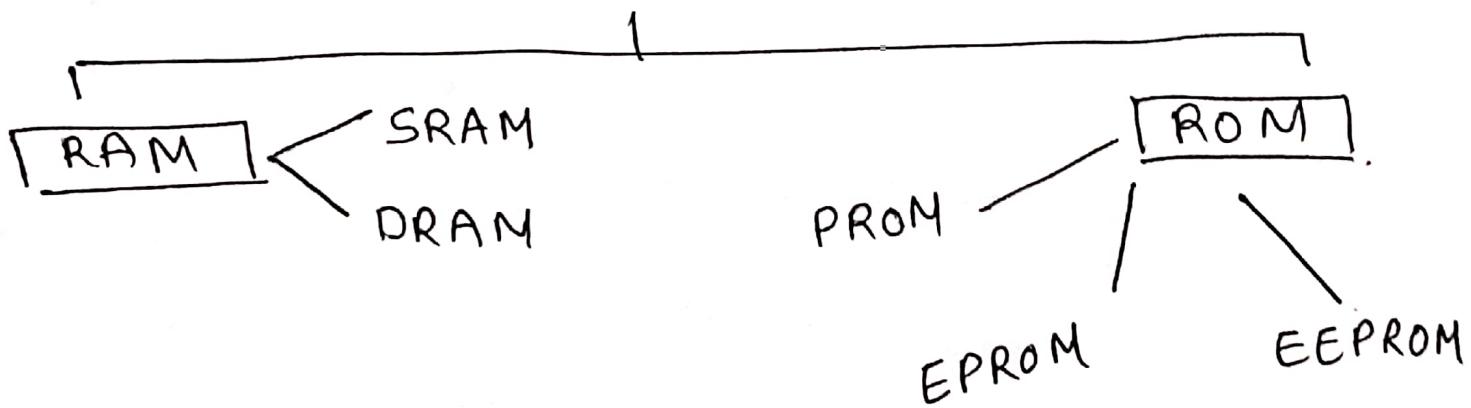
RAM - Random Access memory (RAM) is a type of computer memory that allows data to be read from or written to in almost the same amount of time,

Relationship

ROM - Read only memory type of computer memory used permanently to store data that doesn't need to be modified.
→ Non volatile memory
(Retained even after Power is off)

RAM & ROM both are primary memory

Types of memory



1) MROM

2) PROM (Programmable ROM)

- Each bit is locked by a fuse or antifuse.
- permanently stored & cannot be changed or erased

→ used in firmware & microcode.

3) EPROM (Erasable programmable)

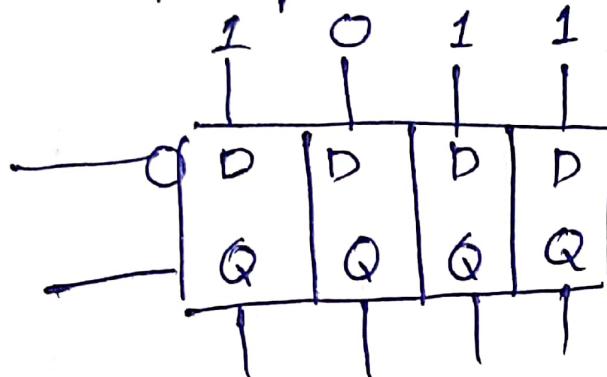
- it can be reprogrammed
- The data can be stored can be erased & reprogrammed again by UV light.
- used in microcontrollers.

4) EEPROM -

- programmed & erased electrically
- no. of times it can be erased.
- duration of erasing programming is near 4ms to 10ms.
- used in remote keyless systems

Registers

- Flip-flop 1 bit memory cell
- used to store single bit data (0 or 1)
- A group of flip-flop known as Registers
(jiski memory 1 bit jyada hoti hai)
- n flip flop $\rightarrow n$ bit data store Karenge.



- Bound to follow the clock.

after every 1 us
what is stored in
memory get changed

iske lie independent control ki zaroorat
hoti hai

It is of two types

- Synchronous - clock ↑ & load ↑
- Asynchronous - only load ↑

If we want to store something for particular time then put the load ↓ for particular that time

Clear input - reset flip flop

classifications of Registers

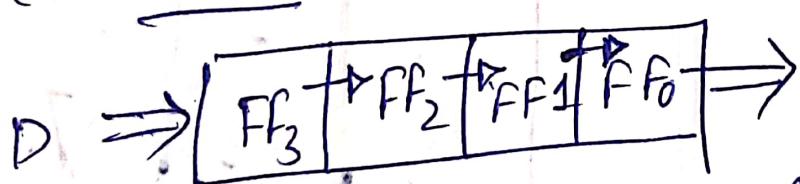
(Depending on input & output)

- SISO (Serial Input, serial output)
- SIPO (Serial Input, parallel output)
- PISO (Parallel Input, serial output)
- PIPO (Parallel Input, Parallel output)

(Depending on application)

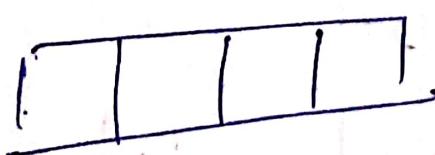
- (a) shift Register
- (b) Storage Register

(a) Shift



When user wants to shift the register from one to another is shift Register.

(b) storage (PIPO)

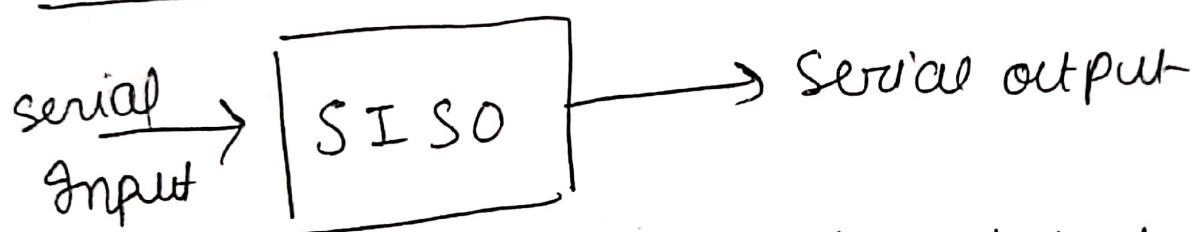


when user only want to store data,

Shift Register

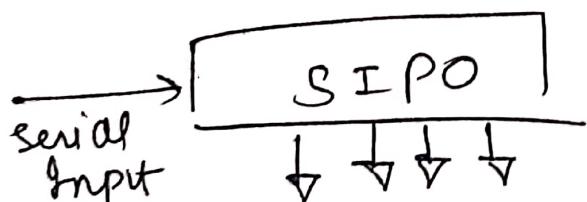
- SISO - serial input, serial output
- SIPO - serial input, parallel output
- PISO - Parallel input serial output
- PIPO - Parallel input, parallel output

• SISO



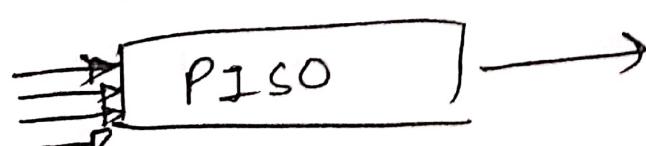
Data ek baar ek saath input hota hai or
ek ek bit ke taur par output hota hai

• SIPO



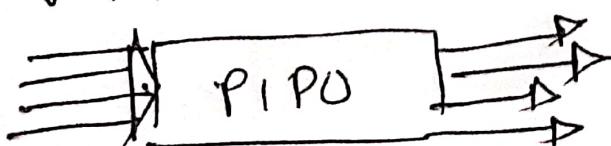
• serially input hota,
lekin output parallel
form mein milta hai

• PISO



• parallel input
hota hai or
series mein output
hota

• PIPO



Parallel Input
Parallel Output

Steps to Design Synchronous counter1. Define counter type -

Requirements ke acc deko konsa Reg chahiye { up counter, Down, Up/Down etc }

2. Determine no. of Bits -

Pta lagao no. of bits kitni chahiye required counter type ke lie.

3. Flip-flops -

choose karo type of (flip-flop)
& typically D flip flop, JK flip flop
jo ki based note hai speed or power

consumption pr state diagram, truth table bhi samani hai

4. State Equations -

state equations Banao jo ki Based hoangi state diagram or truth table pr Express karo next step Har flip flop ka as a function of current state and any control input.

5. Karnaugh maps or logic simplifications -

use karo Karnaugh maps or other methods simplify karne ke lie logic Equations

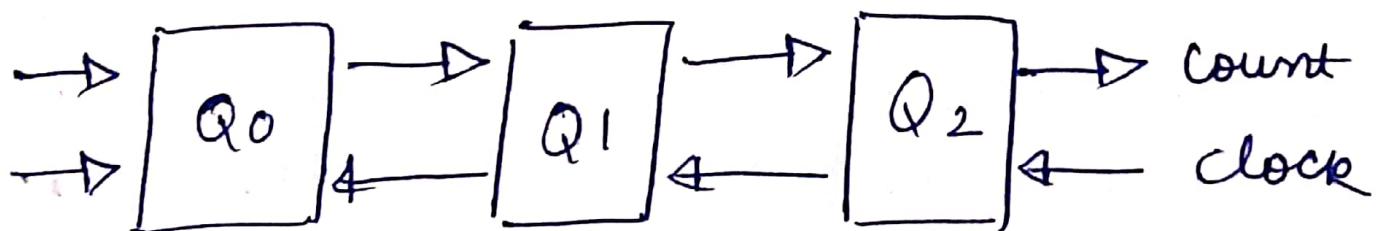
- Implement logic -
Implement Karo jo simplify Kari thi
logic equation logic gates ko use karke
Inputs generate Karne ke lie.
- Feedback & verification
Breadboard & software ka use karke
circuit Banao.
verify Karo

Example

Let we have to create 3 bit synchronously

- no. of bits = 3
- flip-flop : D flip-flop.

State diagram -



State Equation -

$$Q_0 \text{ next} = D_0$$

$$Q_1 \text{ next} = D_1$$

$$Q_2 \text{ next} = D_2$$

Ping counter -

- type of shift Register (output of last flip-flop is connected to input of last flip-flop)
- create circular or Ring like structure
- It operates by shifting th "1" state sequentially through flip-flops, create a rotating pattern of 1's

Characteristics -

- circular structure - output of last flip-flop is connected to input of first flop, create a closed loop.
- one active Bit - Sirf Ek hi flip flop active hota hai ek time pe. or Baaki inactive hote hai.
- shift operators - Active state ~~object and hil ho~~ move / shift karti flip-flops mein clock pulse ke dwara.

Twisted Ring counter

(Better than) Ring counter

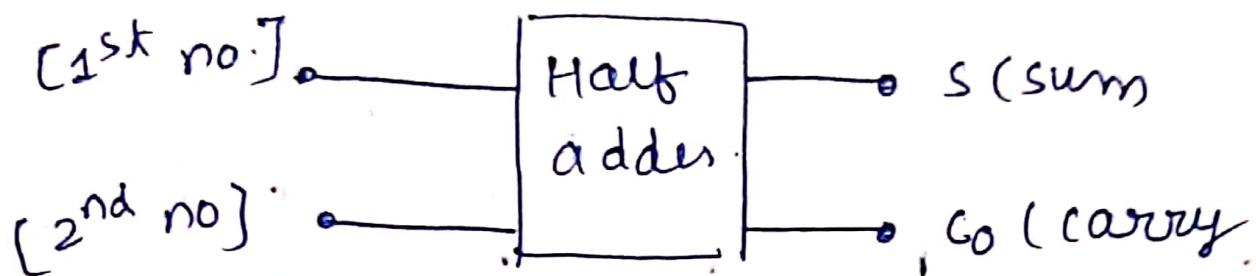
→ Because of no. of states
twisted Ring counter has $= 2 \times \text{no. of ff}$
 $= 2 \times 4$
 $= 8$

→ isko zero se bhi start kar skte hai
(I mean 0 as a input deke)
lekin Ring counter mein input 1 hi hota
hai

→ isme preset nhi dete as input
Baaki clr dete hai
Or isme seedha output se
circular flip flop 1 pe nhi jata
pehle complement leta hai
then uske baad flip flop 1 pe
jata hai jisse input khud
1 & 0 mein convert hota rehta
hai

HALF ADDER

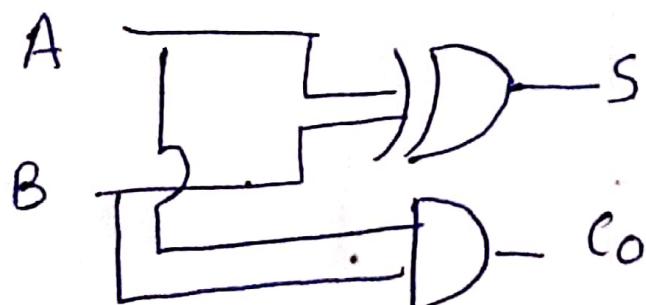
↳ combinational circuits

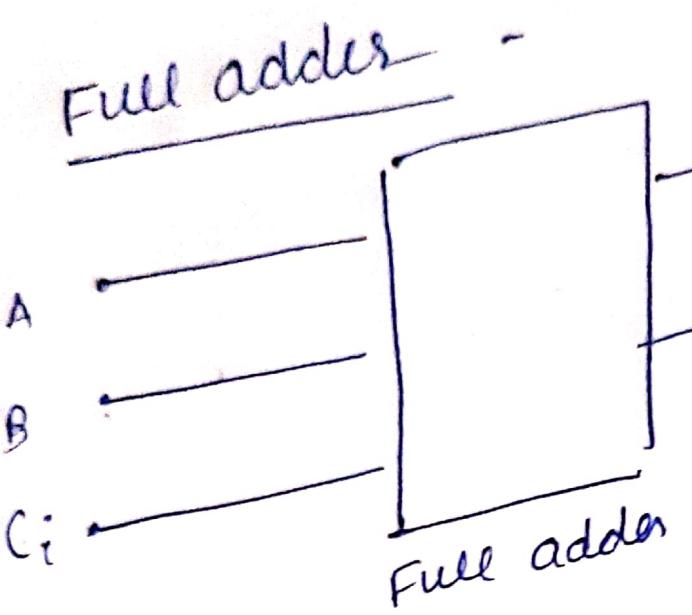


- single bit adder.
- it doesn't take carry from previous sum.

A	B	S	C ₀
0	0	0	0
0	1	1	0
1	0	1	0
1	1	0	1

$$\boxed{S = A \oplus B}$$
$$C_0 = AB$$





A	B	C	S	Co
0	0	0	0	0
0	0	1	1	0
0	1	0	1	0
0	1	1	0	1
1	0	0	1	0
1	0	1	0	1
1	1	0	0	1
1	1	1	1	1

for S

Bc_i	00	01	11	10
A	0	0	1	0
	1	1	0	1
	0	1	0	1

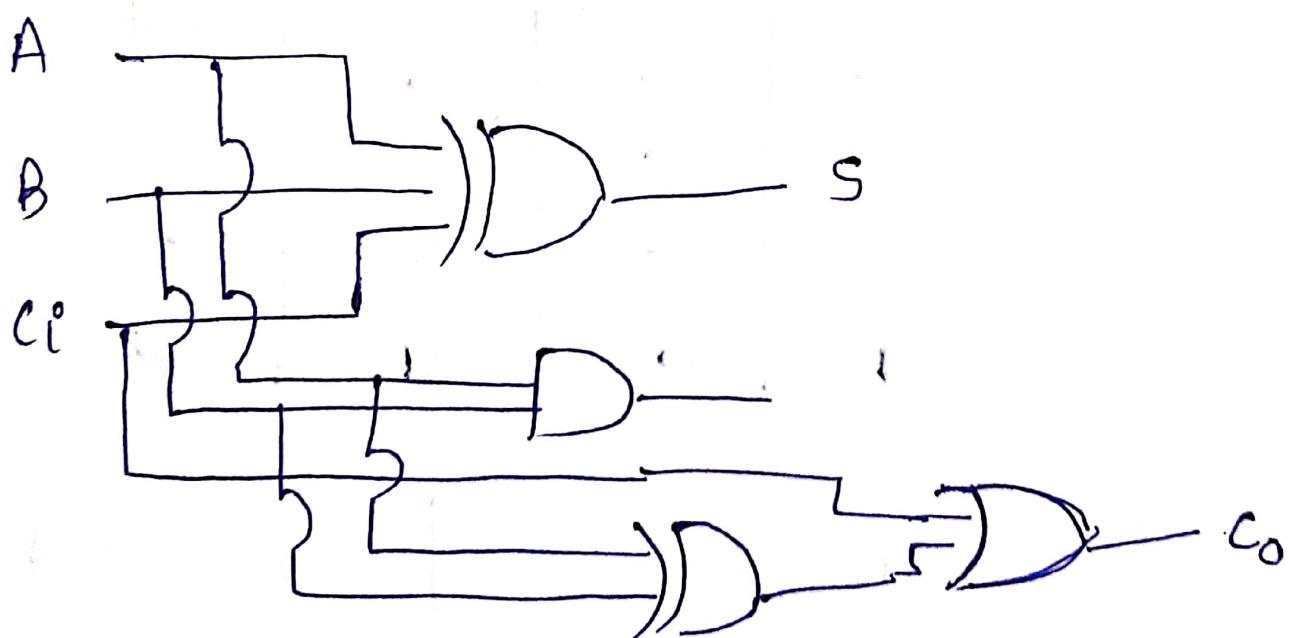
$S = A \oplus B \oplus C_i$

for C_0

Bc_i	00	01	11	10
A	0	0	1	0
	1	0	1	1
	0	1	1	1

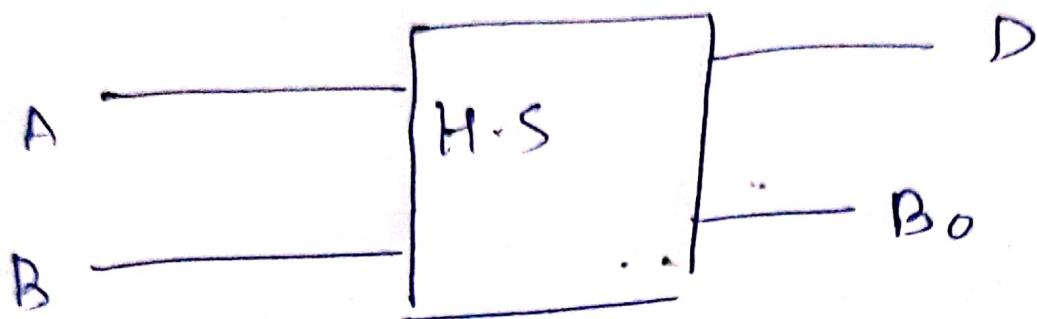
$$C_0 = BC_i + AB + AC_i$$

$$C_0 = AB + C_i(A \oplus B)$$



Half subtractor

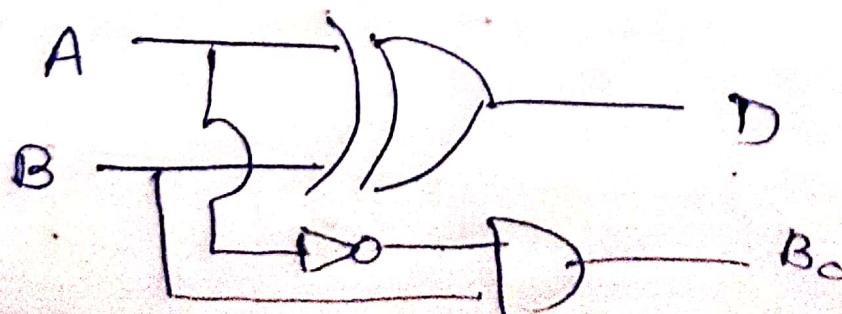
→ for single bit

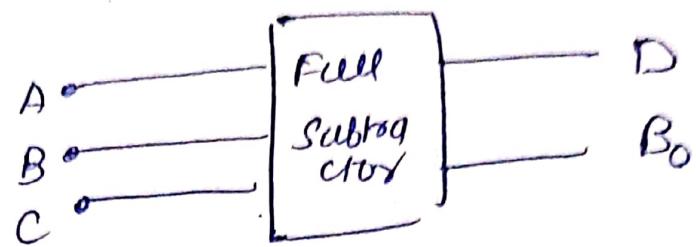


A	B	D	B _o
0	0	0	0
0	1	1	1
1	0	1	0
1	1	0	0

$$D = A \oplus B$$

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



Full subtractor

A	B	C	D	B_0
0	0	0	0	0
0	0	1	1	1
0	1	0	1	1
0	1	1	0	1
1	0	0	1	0
1	0	1	0	0
1	1	0	0	0
.

for D -

	BC	00	01	11	10
A	0	0	1	0	1
	1	1	0	1	0

$$D = A \oplus B \oplus C$$

for B_0

	BC	00	01	11	10
A	0	0	1	1	1
	1	0	0	0	0

$$B_0 = BC + \bar{A}C + \bar{A}B$$

<u>Aspects</u>	combinational	sequential
Inputs Dependency	outputs depends only current inputs	on both inputs & memory.
Feedback	No feedback loops	feedback loops exists
Memory	no memory elements	Contains memory elements
Timing Consideration	no concept of clock or clock cycles	operate based on clock signals
Output	immediate response to input	changes based on inputs & current initial state (so slow)
Examples	Multiplexers adders	flip flop, counters Registers

J-K flip flop

- ↳ type of sequential logic circuit.
- ↳ capable of storing 1-bit of data.

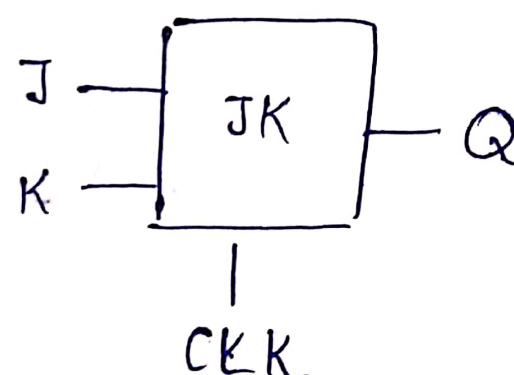
Characteristics -

Inputs - three inputs J(data or set), K(complement of data or reset), & clock point (CLK)

States - two stable states (0, 1)

operation - either hold its state or either changes its state accordingly to inputs

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



$J = K = 0$ {flip holds previous state}

$J = 0 \quad K = 1$ {flip flop resets}

$J = 1, K = 0$ flip flop sets

$J = 1, K = 1$ it toggles or complements its states

* * * Race around condition

It is a condition when both the inputs J & K are high (1) & the clock signal simultaneously change the transition from one state to another { High to low } & { Low to high } which leads to instability & unpredictable behavior.

Truth table

J	K	Q_n	Q_{n+1}
1	1	Q	\bar{Q}
both are high			

How to avoid / prevent Race around condition -

- (1) Synchronous operation - J & K inputs change only during specific clock transitions.
- (2) Avoid configuring / set both values J & K 1 unless you intended to use toggle functionality.

Master slave JK flip flop

- Sequential logic circuit
- Utilizes two J-K flip-flops to reduce the risk of a race around condition
- Consist of master & slave flip flop connected in series
- Allow output to change only the edge of clock signal, Reduce the possibility of unwanted transitions

Master → act as input for slave flip flop
J & K are connected to data inputs, clock input triggers the master flip-flop.

Slave → Takes output from master

The biggest advantage of master flip-flop is

- Reduces race around condition while holding data.

Multiplexer

- combinational circuit that select binary information from any input lines & directs it to o/p lines
- Data Selector.
- No. of output will always be 1
 - when 2 input \rightarrow 2:1 MUX
 - 4 input \rightarrow 4:1 MUX
 - 8 input \rightarrow 8:1 MUX
 - 16 input \rightarrow 16:1 MUX.

Relationship b/w selected lines & no. of I/P

$$n = 2^m \rightarrow \text{no. of selected lines}$$

↓

no. of inputs

$$m = \log_2 n$$

For Exp for 4:1 MUX

$$m = \log_2 2^2$$

$m = 4$

Inputs = 4

$$2 \log_2 2 = 2 \text{ select lines.}$$

	<u>Latch</u>	<u>flip-flop</u>
complexity	Simple design less complex	More complex due to clocked behavior
Timing	Timing issue due to continuous update	No timing issue & more controlled due to clock Edge
Output change	change due to Input change	change due to clock Edge change
Type of storage	Transparent	Edge-triggered
Inputs	Enable (EN) & Data (D) inputs	typically (CLK) clock & Data (D) input
Ex	S R latch Gated latch	J K flip flop D-flip-flop