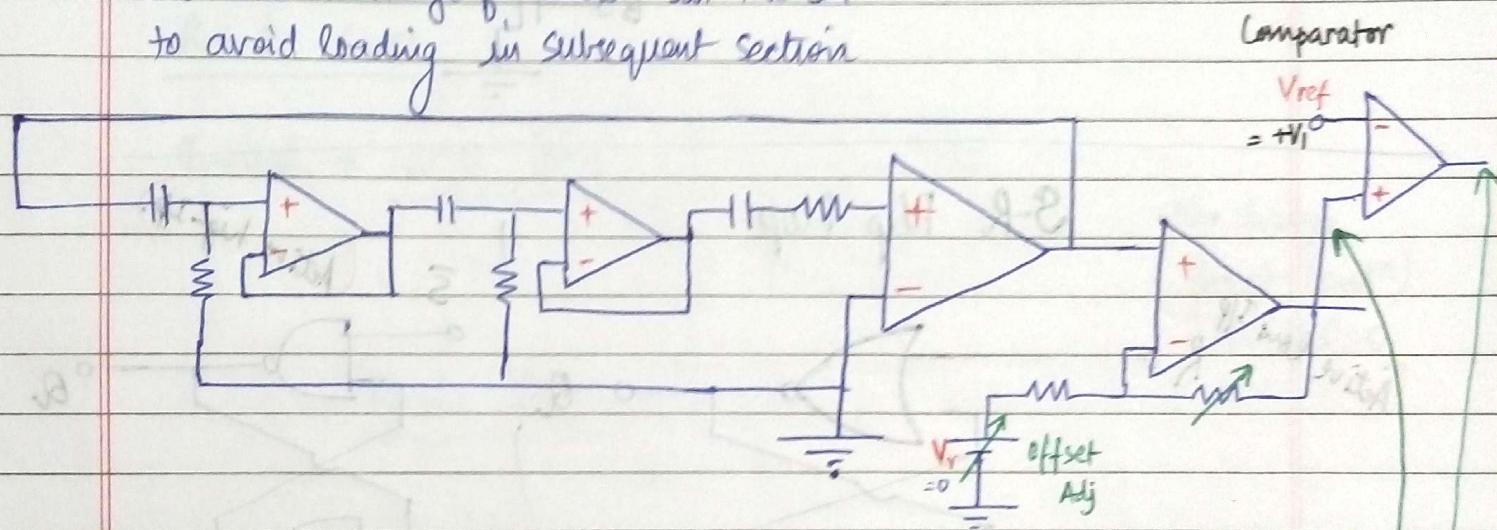


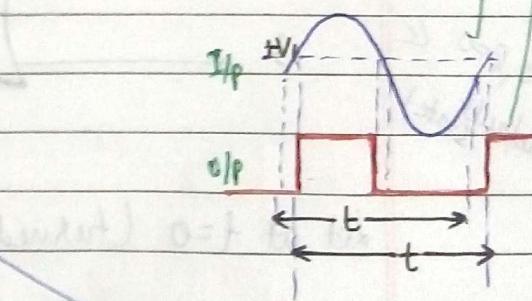
10/11/16

We use voltage followers in the ckt.
to avoid loading in subsequent section



FLIP FLOP OR **LATCHES**

1 0
↓
SR Type
(Set-Reset)
↓
D-type
(Data or Delay)
↓
JK

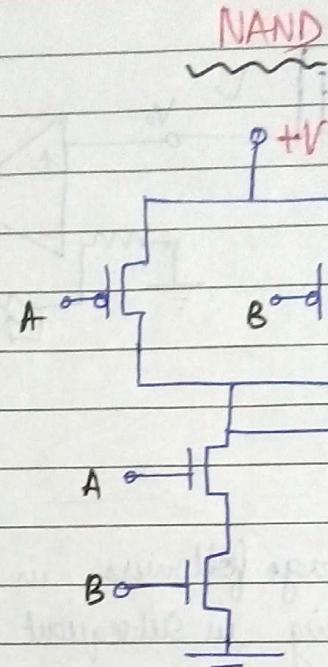
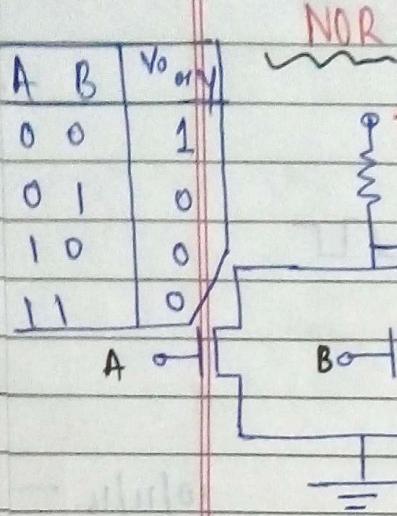


NOR and NAND are
two universal gates.

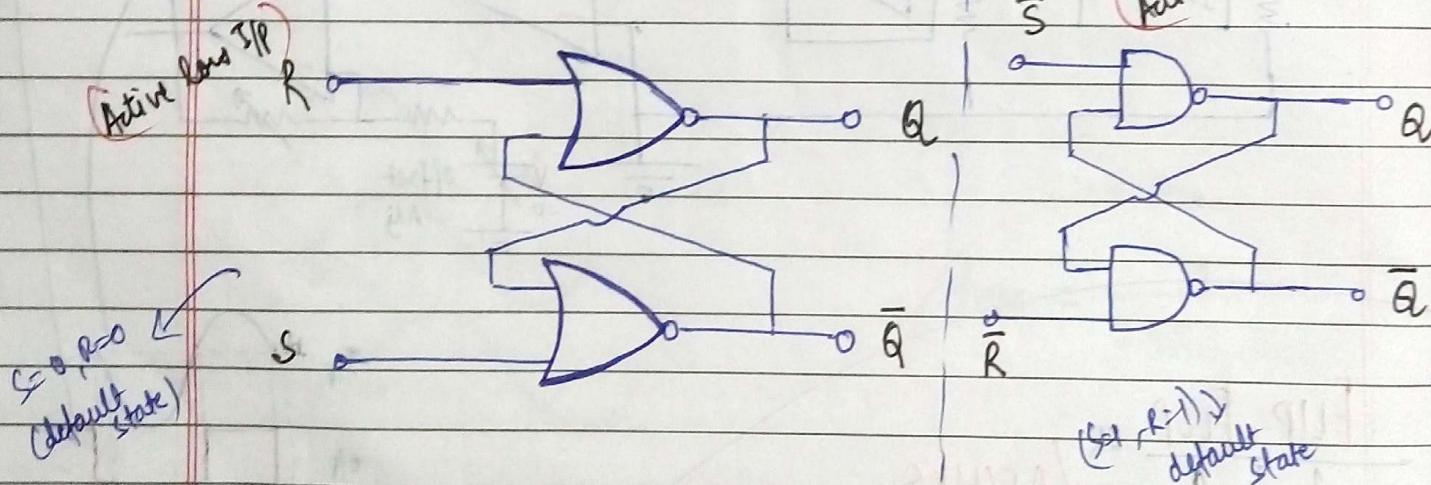
Complementary

CMOS

n & p mos



S-R Flip Flop



Let at $t=0$ (turned on), Let $Q=0$ and $\bar{Q}=1$ (basically, these are undeterminate)

$$\underline{R=S=0}$$

$$\underline{\overset{0}{D}} = 0$$

- No change

$$\underline{\overset{0}{D}} = 1$$

if $Q=1$, $\bar{Q}=0$

$$\underline{R=S=0}$$

$$\underline{\overset{0}{D}} = 1$$

$$\underline{\overset{1}{D}} = 0$$

- No change.

$R=0 \ S=1$ Initial $\rightarrow Q=1, \bar{Q}=0$ $0 \rightarrow D \circ ①$

- No change
(Set condⁿ)
 $S=1 \Rightarrow Q=1.$
 $R=0$

 $Q=0, \bar{Q}=1$
 $0 \rightarrow D \circ ②$
 $1 \rightarrow D \circ ③$

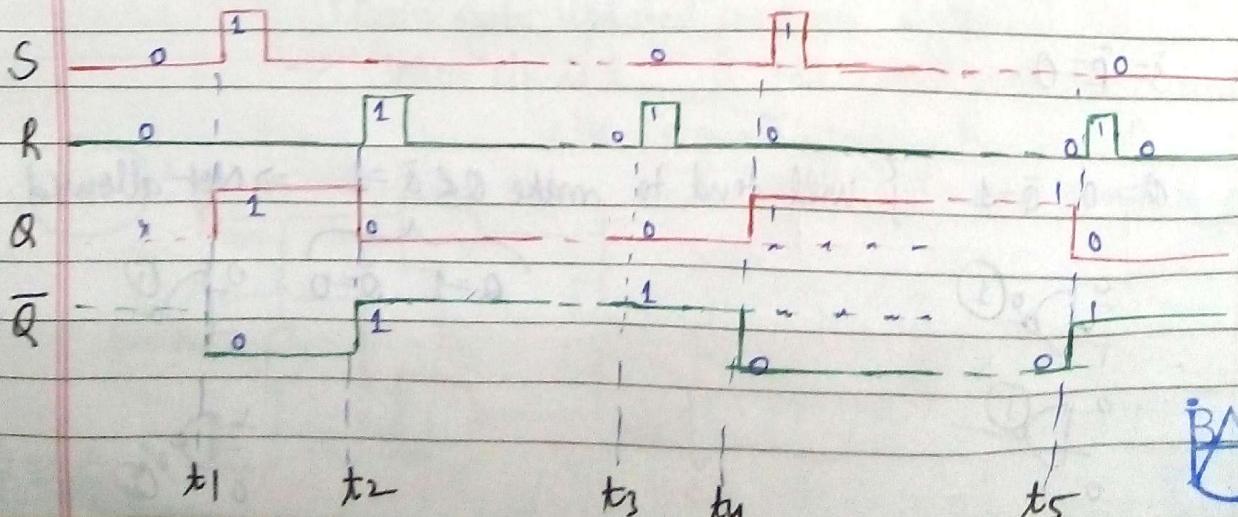
change

 $R=1 \ S=0$ Initial $\rightarrow Q=0, \bar{Q}=1$ $① \rightarrow D \circ ④$

No change
(Reset condⁿ)
 $R=1 \Rightarrow Q=0.$
 $S=0$

 $Q=1, \bar{Q}=0$ $R=1, S=1$

Q and \bar{Q} try to become '0'
⇒ Not allowed.



* $\bar{S}, \bar{R} \rightarrow$ both '1' \rightarrow No change.

$$\alpha=0 \quad \bar{\alpha}=1$$

$${}^1\bar{D}{}^0$$

$${}^0\bar{D}{}^1$$

$$\alpha=1 \quad \bar{\alpha}=0$$

$${}^1\bar{D}{}^1$$

$${}^0\bar{D}{}^0$$

* $\bar{S}=0 \quad \bar{R}=1$

$$\alpha=1 \quad \bar{\alpha}=0$$

$$\alpha=0 \quad \bar{\alpha}=1$$

$${}^0\bar{D}{}^1$$

No change

$${}^1\bar{D}{}^0$$

$${}^0\bar{D}{}^1$$

Change

$${}^1\bar{D}{}^0$$

* $\bar{S}=1 \quad \bar{R}=0$

$$\alpha=1 \quad \bar{\alpha}=0$$

$$\alpha=0 \quad \bar{\alpha}=1$$

$${}^1\bar{D}{}^0$$

Change

$${}^0\bar{D}{}^1$$

$${}^1\bar{D}{}^0$$

No change

$${}^0\bar{D}{}^1$$

* $\bar{S}=\bar{R}=0$.

$\alpha=0 \quad \bar{\alpha}=1$ } Will tend to make $\alpha & \bar{\alpha}=1 \Rightarrow$ Not allowed.

$${}^0\bar{D}{}^1$$

$${}^1\bar{D}{}^0$$

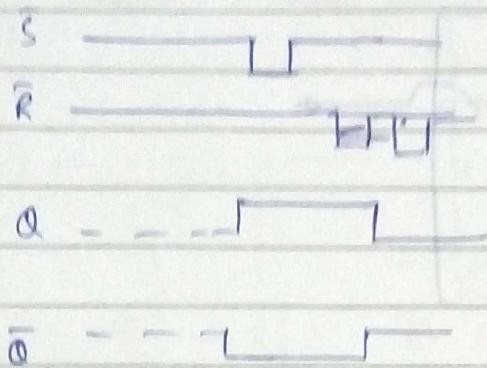
$${}^0\bar{D}{}^1$$

$$\overbrace{\alpha=1 \quad \bar{\alpha}=0}^{Q=1 \quad \bar{Q}=0}$$

$${}^0\bar{D}{}^1$$

$${}^1\bar{D}{}^0$$

$${}^0\bar{D}{}^1$$

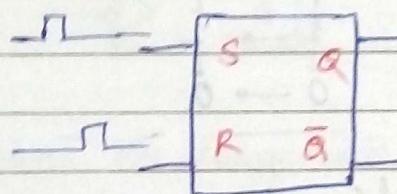


F.F. types:-

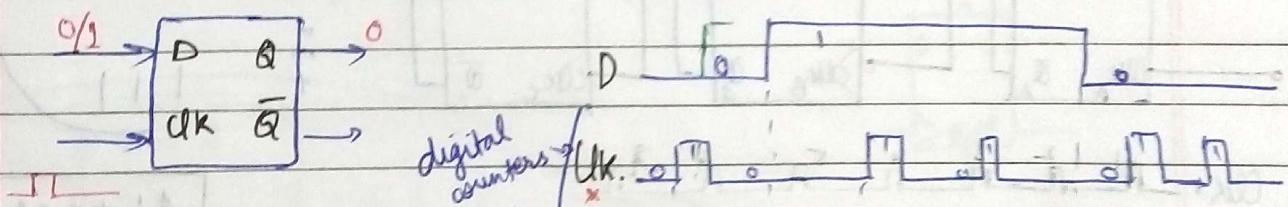
1. S-R
2. D
3. J-K
4. Toggle (T)

Flip Flop or latches
(Memory)

11/11/16



D-type F.F.

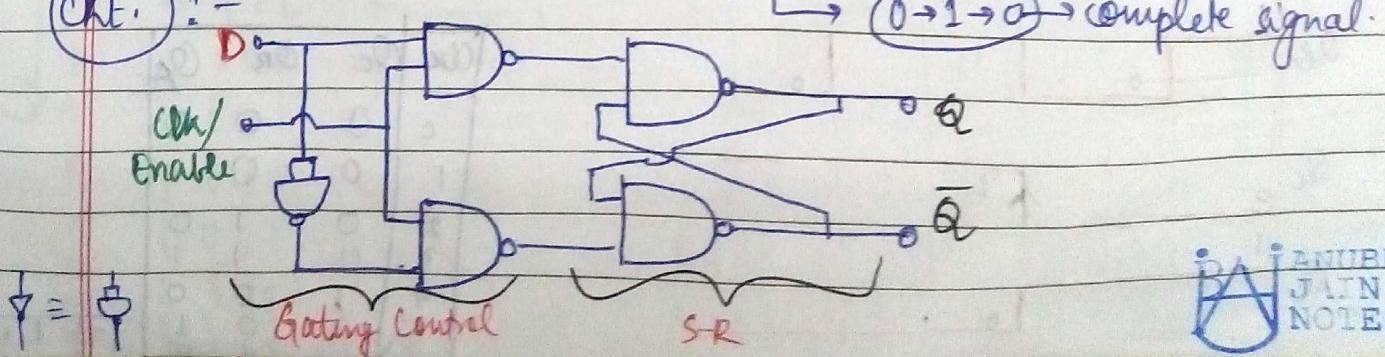


Clock or Enable

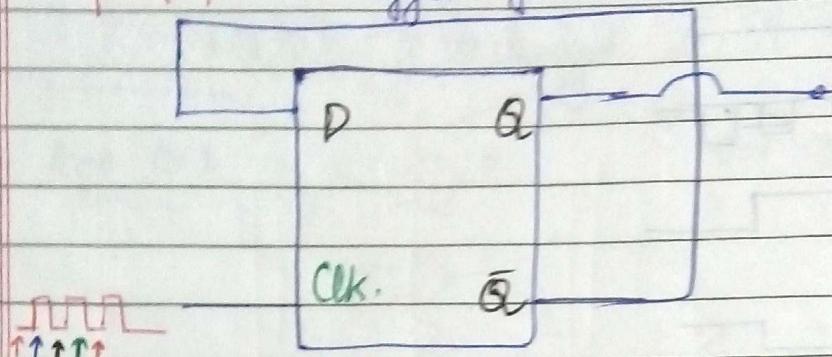
Enable

(Data gets updated ^{only} on clock signal)→ When clk is 1, $Q = D$ and continues until next clk (signal) is received.

Ckt. :-



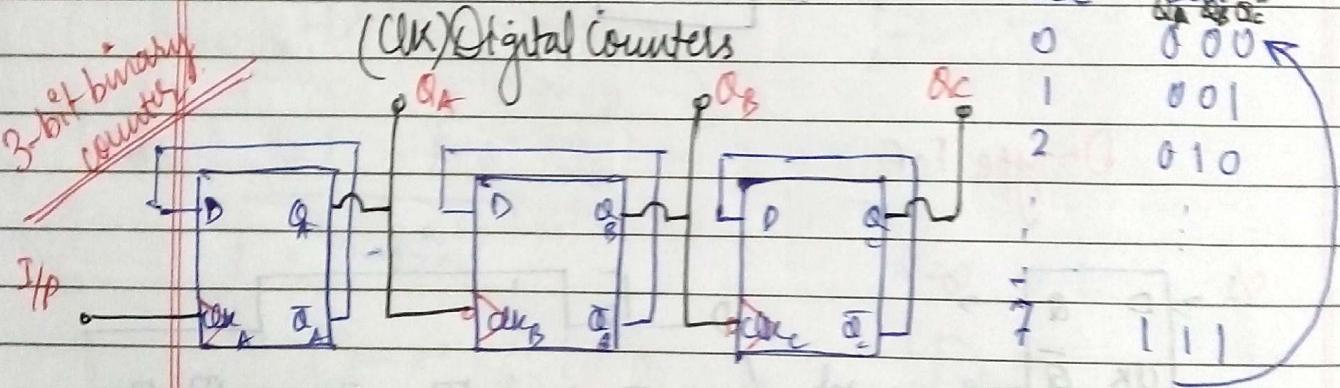
T-F.F. → Toggle signal on each click.



Ck	Q	\bar{Q}	D
→ 0	0	1	1
→ 1	1 ←	0 →	0
→ 0	0 ←	1 →	1
→ 1	1 ←	0 →	0

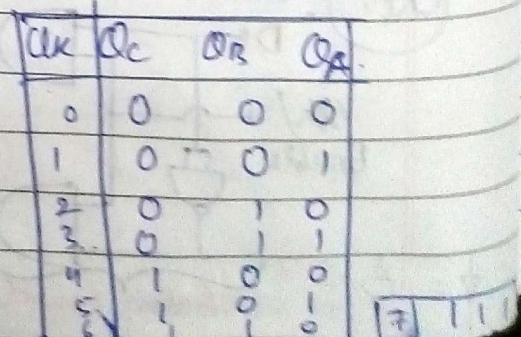
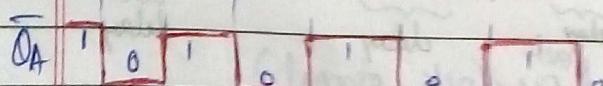
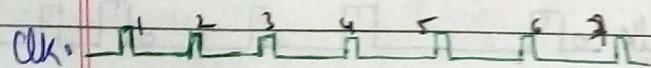
How to make such
e. ~~Bin~~ (countly)

3-bit binary counter

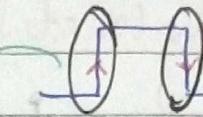
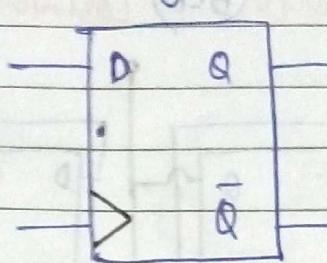


A: the edge triggered

B.C : we edge triggered F.F.



Enable & clock has slight diff.

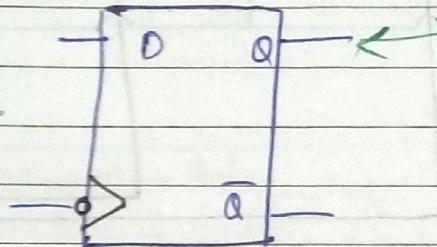


Edge triggered clock.

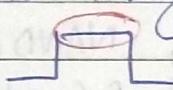
(more commonly used in practice)

The edge triggered FF.

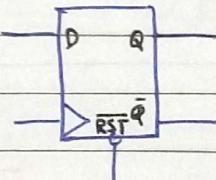
→ edge triggered
→ FF → sensitive to
long going edge.



Whereas, enable sense the changes in D throughout the duration when enable/clk = 1.



Practical flip flops have Reset pins to set up 'D' initially.



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

0 0 1 1
0 1 0 0

0 1 0 1
0 1 1 0

0 1 1 1
1 0 0 0

1 0 0 1
1 0 1 0

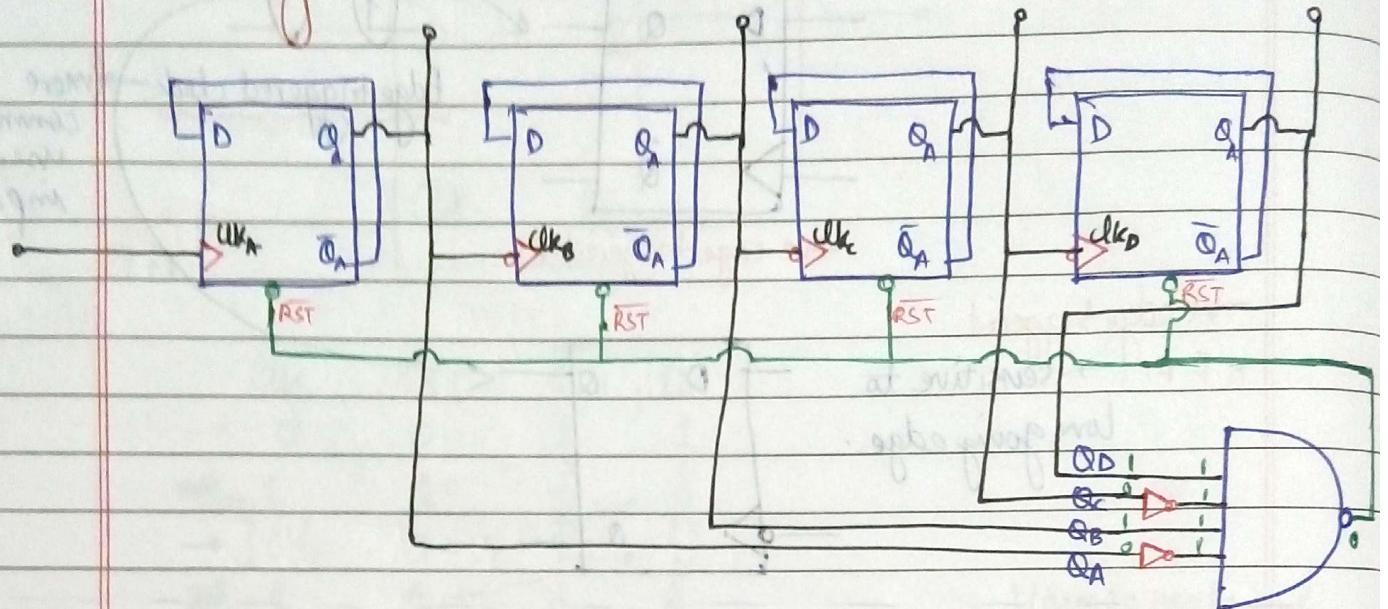
1 0 1 1
1 1 0 0

1 1 0 1
F 1 1 1

How?

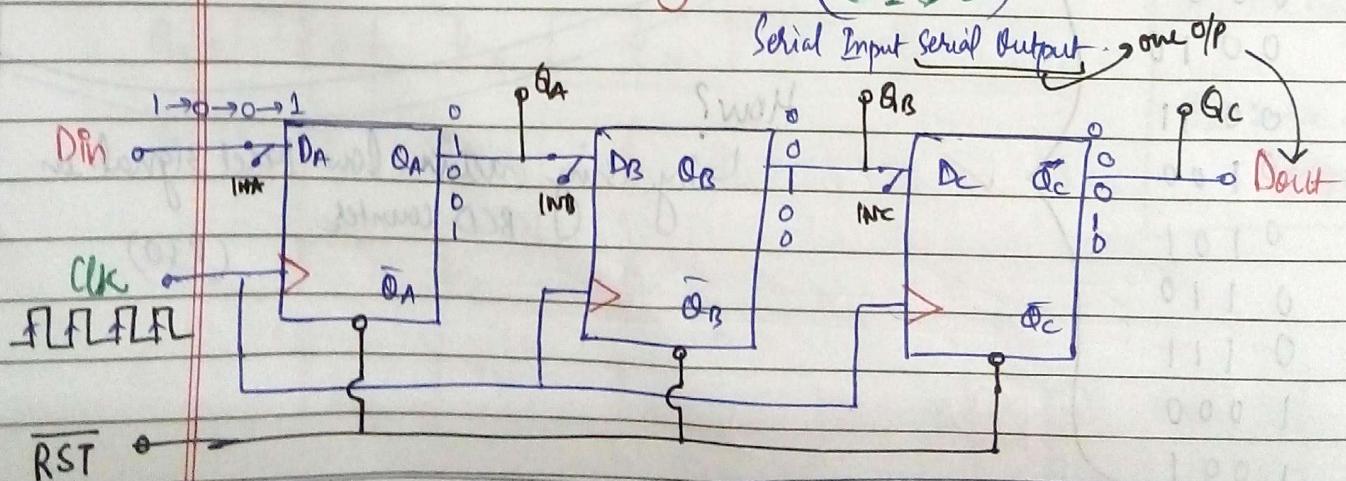
By using active low reset signal in
BCD counter
(PTO)

Binary Coded Decimal (BCD) Counter



When ~~1010~~ 1010 sequence is reached, $\text{NAND O/P} = 0$ and all F.F gets reset.
 \Rightarrow again starts from 0 0 0 0
 \Rightarrow We can limit the counter.

Shift Register - (SISO)

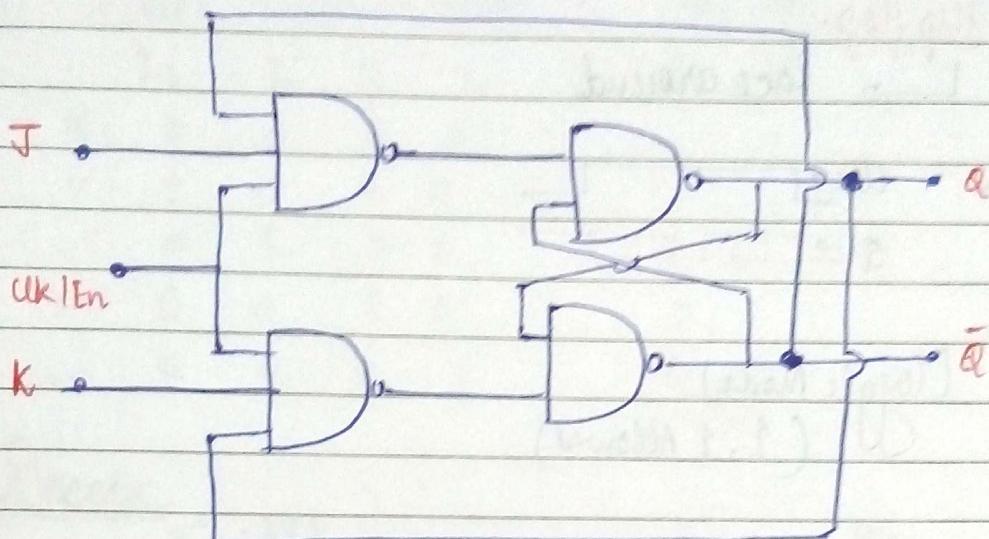


PISO
 (Parallel I/p Serial O/p)

SIPO
 (Serial I/p Parallel O/p)
 Multiple O/p.

PIPO - (Parallel I/p P. O/p)

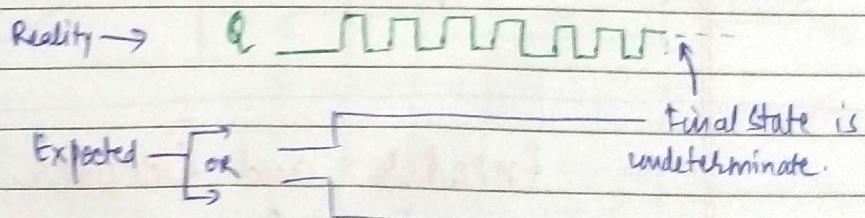
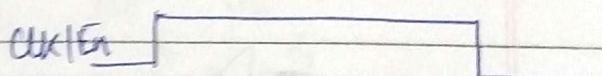
J-K F.F.

Propagation delay \approx ns.

Clk/En	J	K	Q
0	0	0	No change
0	1	0	0
1	0	0	1
1	1	1	Toggle mode

Not allowed
State in SR
but allowed in JK

Problem:-

(Race around situation)
in toggle mode

* If clock width < propagation delay,
then expected mode happens.

* Class Test Friday. (Post Midsem Syllabus) * Extra Class on Thu.

a) End Sem (Whole Syllabus) → Post Midsem (70%)

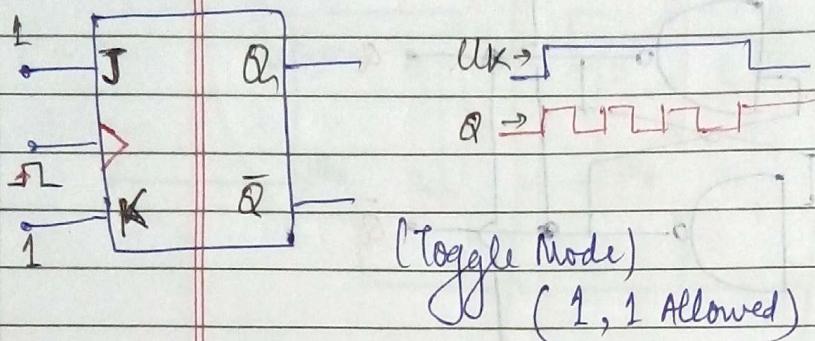
16 ques (5 Marks each)

Orion

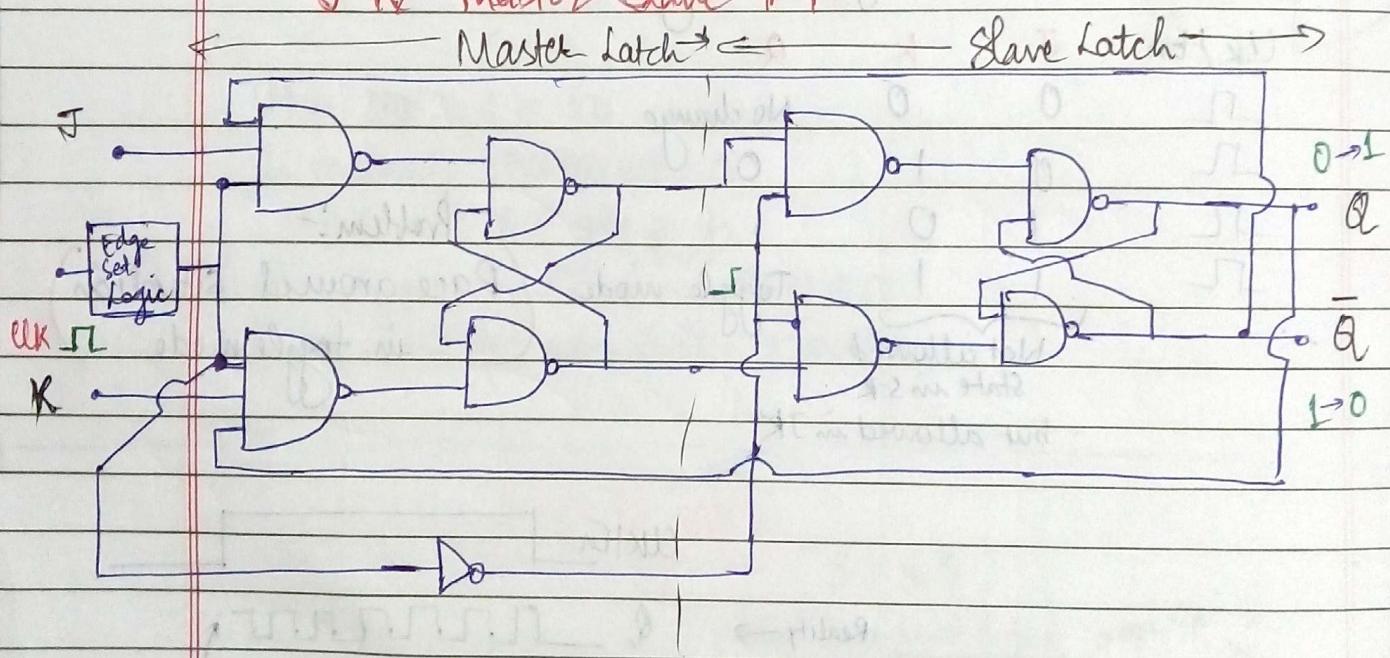
PAGE: 16 / 11 / 16

JK flip flop.

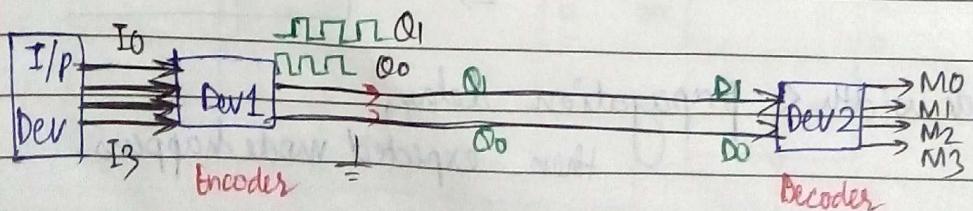
Race around



JK Master Slave F.F.



Encoder and Decoder



Requirement → Operate 4 machines → exactly one at a time.

Encoder

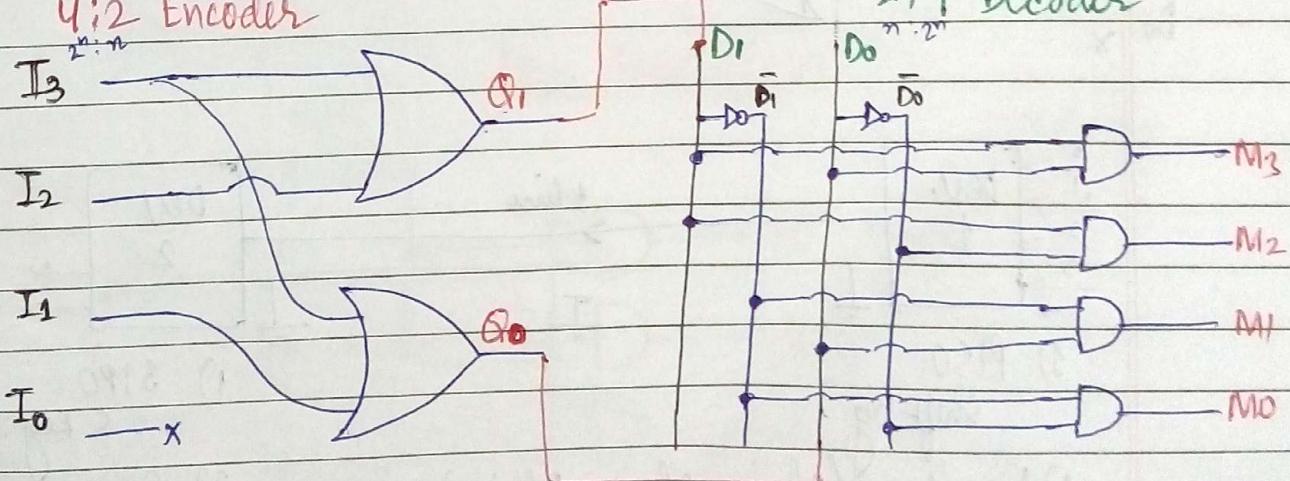
Bin O/p's				Bin O/p's	
(MSB)	(PS)	(LSB)		Q_1	Q_0
I_3	I_2	I_1	I_0	—	—
X	0	0	0	1	1
✓	1	0	0	1	0
✓	0	1	0	0	1
✓	0	0	1	0	0
✓	0	0	0	0	0

Decoder

Bin O/p's		(MSB)		O/p's		(LSB)	
D1	D0	M3	M2	M1	M0	M3	M2
0	0	0	0	0	1	0	1
0	1	0	0	1	0	1	0
1	0	0	1	0	0	0	0
1	1	1	0	0	0	0	0

→ Remotely, we can regenerate original pattern.

16:4
8:3
2:n
4:2 Encoder



$$Q_1 = I_2 + I_3$$

$$Q_0 = I_1 + I_3$$

$$M_3 = D_1 \cdot Q_1 \cdot Q_0 = (I_2 + I_3) \cdot (I_1 + I_2) = I_2 \cdot I_1 + I_2 \cdot I_3 + I_3 \cdot I_1 + I_3 \cdot I_3 = I_2$$

$$M_2 = D_1 \cdot Q_1 \cdot \bar{Q}_0 = I_2$$

$$M_1 = \bar{D}_1 \cdot Q_0 \cdot \bar{Q}_0 = I_1$$

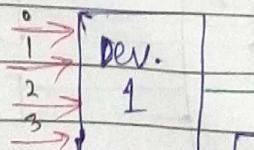
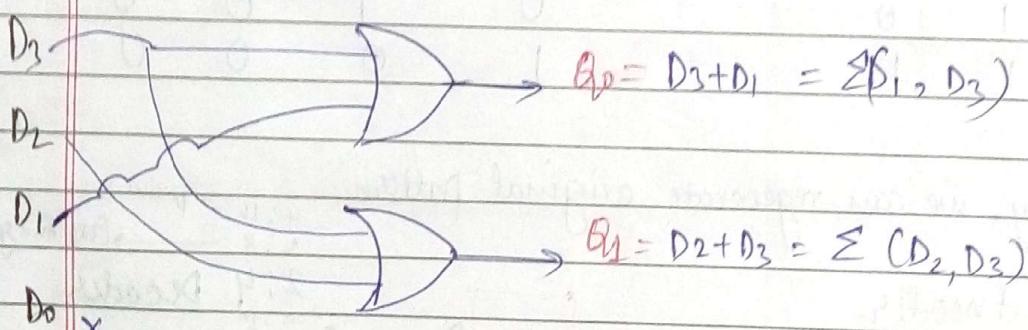
$$M_0 = \bar{D}_1 \cdot \bar{Q}_0 = I_0$$

4:16 Decoder

D_3	D_2	D_1	D_0	M_{15}	M_{14}	M_{13}	M_{12}	M_{11}	M_{10}	M_9	M_8	M_7	M_6	M_5	M_4	M_3	M_2	M_1	M_0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	
0	0	0	1	1	0	0	0	0	0	0	0	0	0	0	0	1	0	0	
0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	
1	1	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	
0	1	1	1	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	
1	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	

17/11/16

4:2 Encoder



1) PISO
shift reg.

- 2) Encoder (2 lines, 3 lines, 4 lines)
3) Multiplexer (mux)

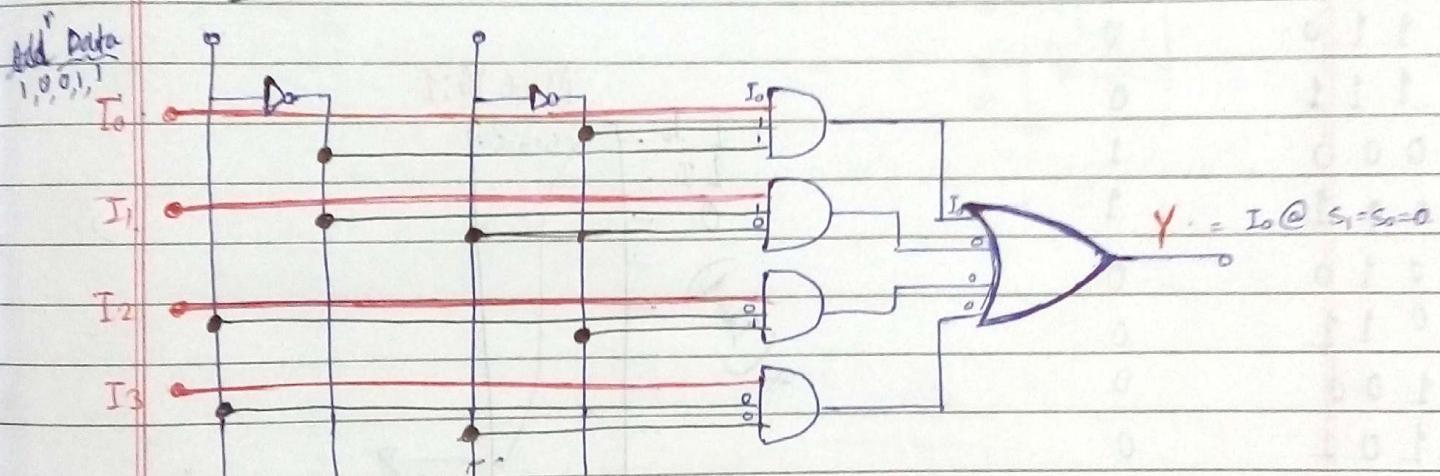
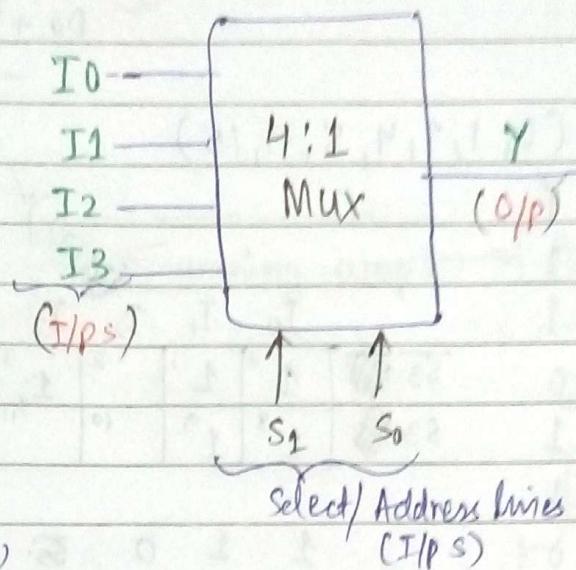
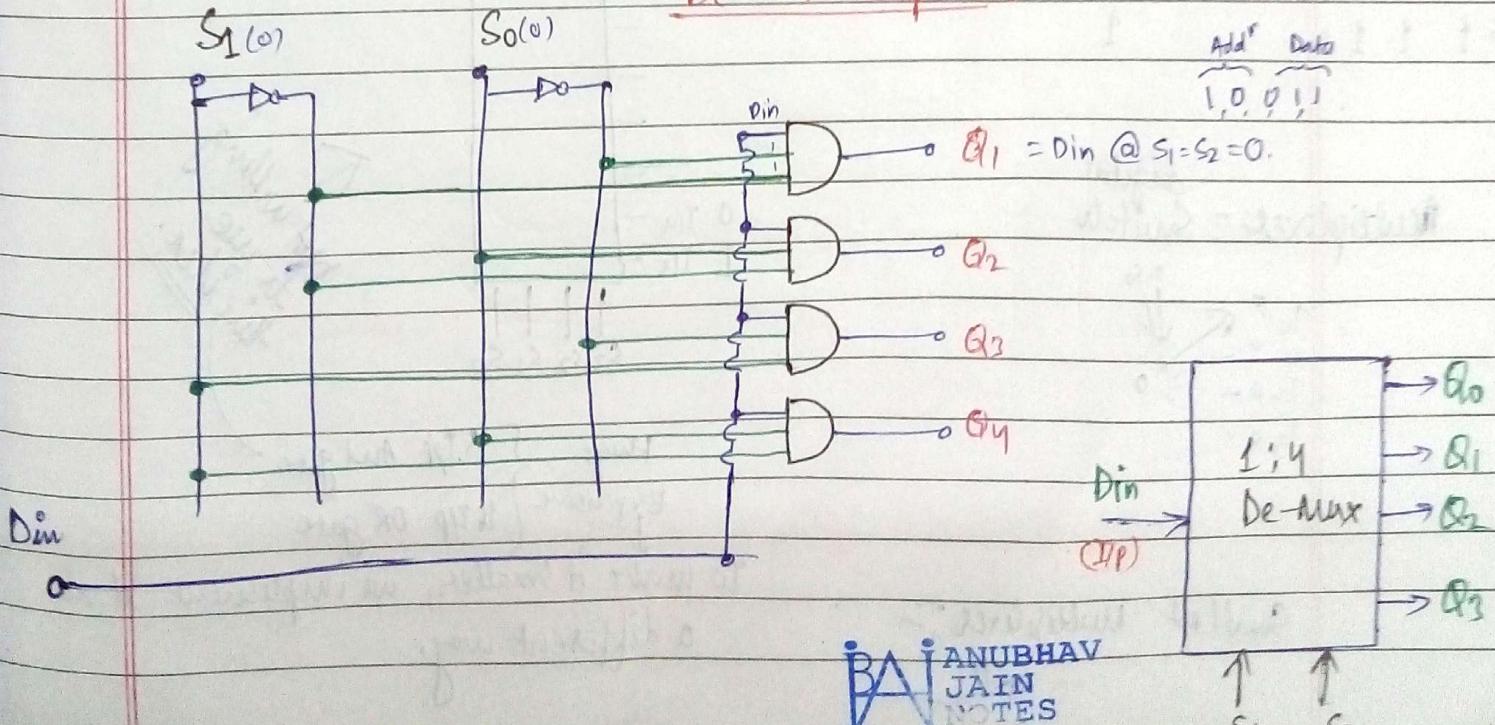


1) SIPO
S.Reg.

- 2) Decoder
3) De-mux

Multiplexer (Multiple I/P, 1 O/P)

(n Select lines)
 2^n I/P lines

De-Multiplexer

$$f = \sum (0, 2, 4, 5, 7, 9)$$

$$D_0 + D_2 + D_4 + D_5 + D_7 + D_9$$

Dec

$$F = \sum (0, 1, 3, 4, 8, 9, 15)$$

 $\downarrow S_3 \ S_2 \ S_1 \ S_0$

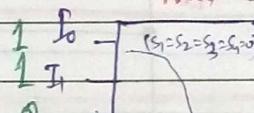
Y

(Index)

8:1 Mux

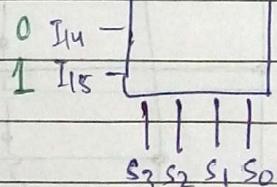
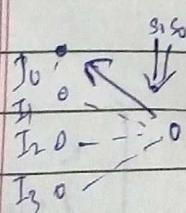
00000	1	I ₀	I ₁	I ₂	I ₃	I ₄	I ₅	I ₆	I ₇
10001	1								
20010	0	S ₃	1 ⁰	1 ¹	2 ²	3 ³	4 ⁴	5 ⁵	6 ⁶
30011	1	S ₃	1 ⁸	1 ⁹	10 ¹⁰	11 ¹¹	12 ¹²	13 ¹³	14 ¹⁴
40100	1								
50101	0		1	1	0	S ₃	S ₃	0	0
60110	0								
70111	0								
81000	1								
91001	1								
101010	0								
111011	0								
121100	0								
131101	0								
141110	0								
151111	1								

Max 16:1



X (Total)

Multiplexer = Switch
Digital

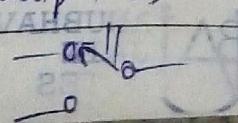


Not willing
to use
this idea.

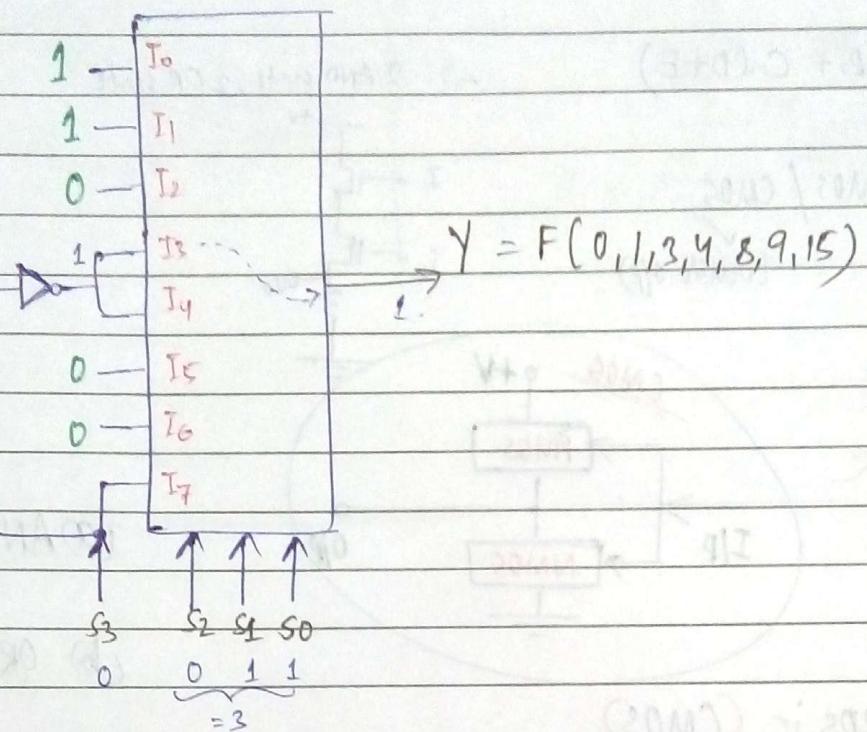
Huge, { 5 I/P And gate
Expensive } 16 I/P OR gate.

To make it smaller, we implement it in
a different way.

Smallest Multiplexer:-



8:1 MUX



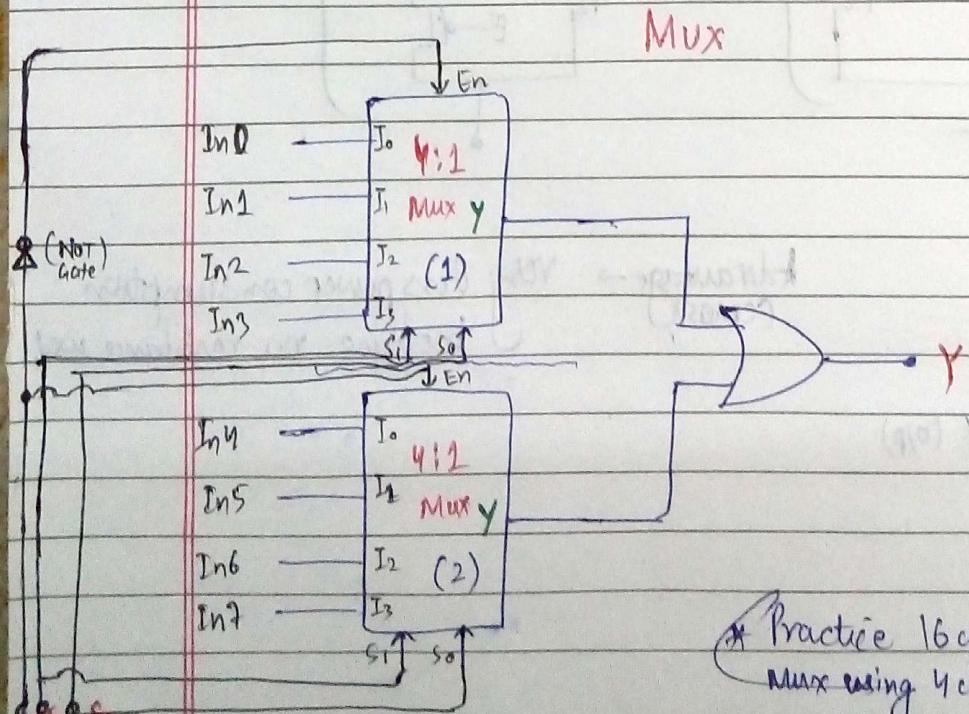
Digital Ckts.

Combinational

eg. Gates
MUX / De-Mux

Sequential

eg. Flip-flop
Clocked ckts
(Shift reg. & counter)



at a time, only one of
two MUX will be enabled
and its o/p will be the
final o/p.

Using this configuration,
we have used 2 4:1 MUX
instead of 1 8:1 MUX.

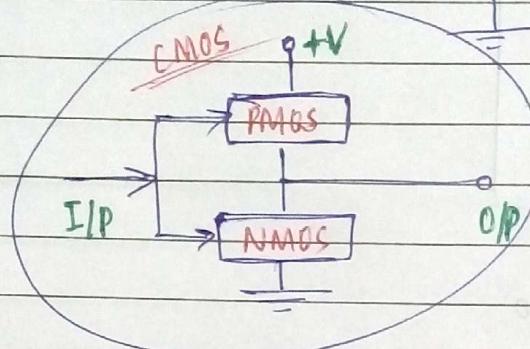
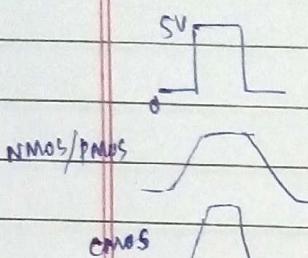
Advantage: Overall cost
reduces (because smaller
gates will be used).

* Practice 16 channel
MUX using 4 channel MUX.

$$Y = AB + C(D+E)$$

→ 2 AND gate, 2 OR gate

NMOS / PMOS / CMOS
(better O/P)

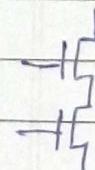


Steps :- (CMOS)

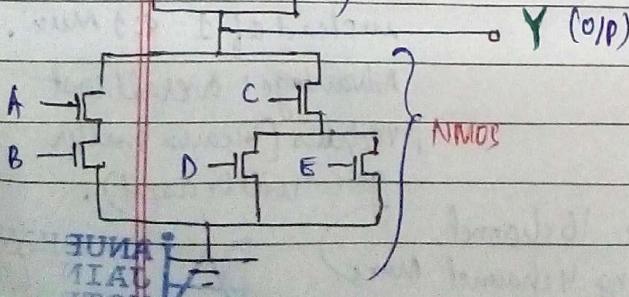
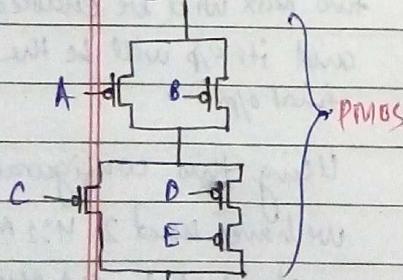
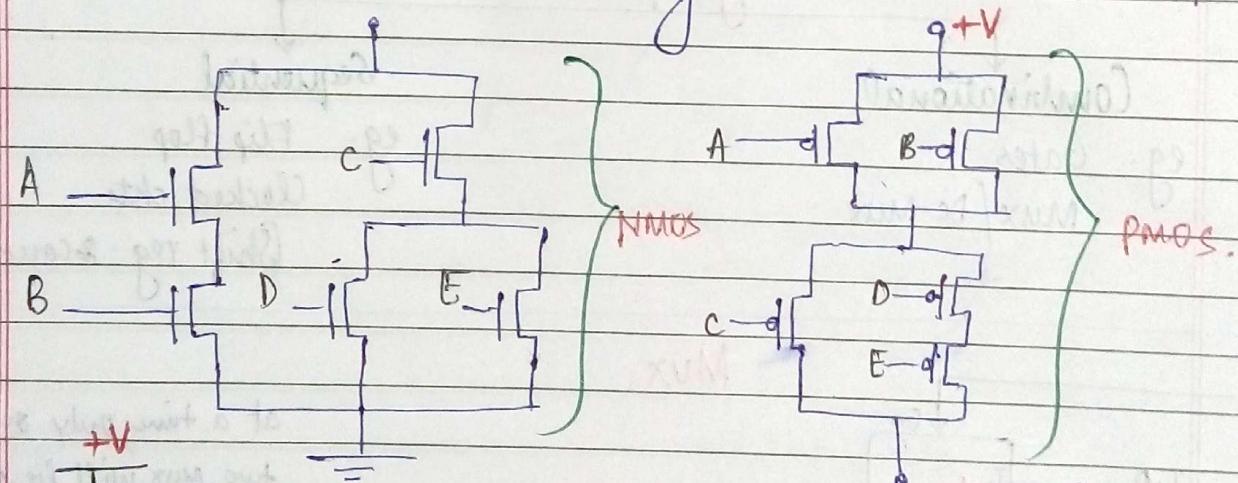
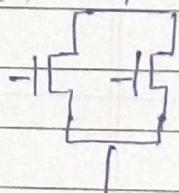
1) Design the NMOS section

2) Draw the dual version using PMOS.

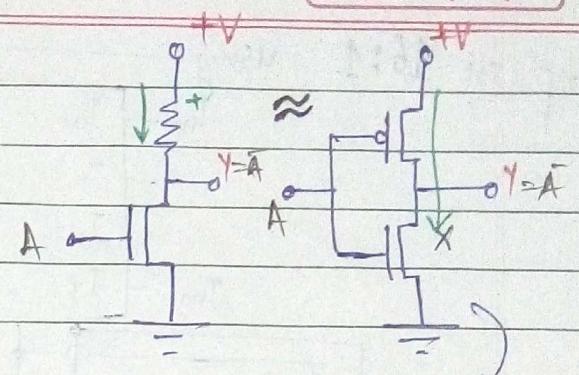
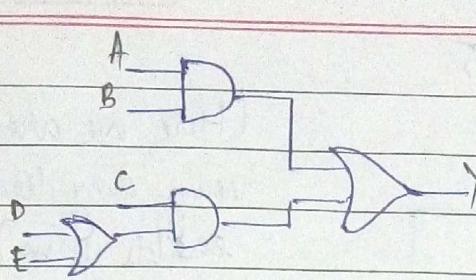
(a) AND(•)



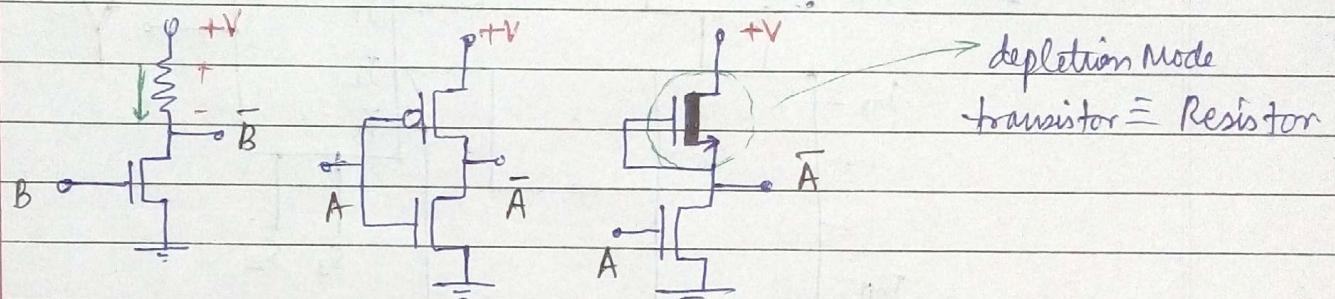
(b) OR(+)



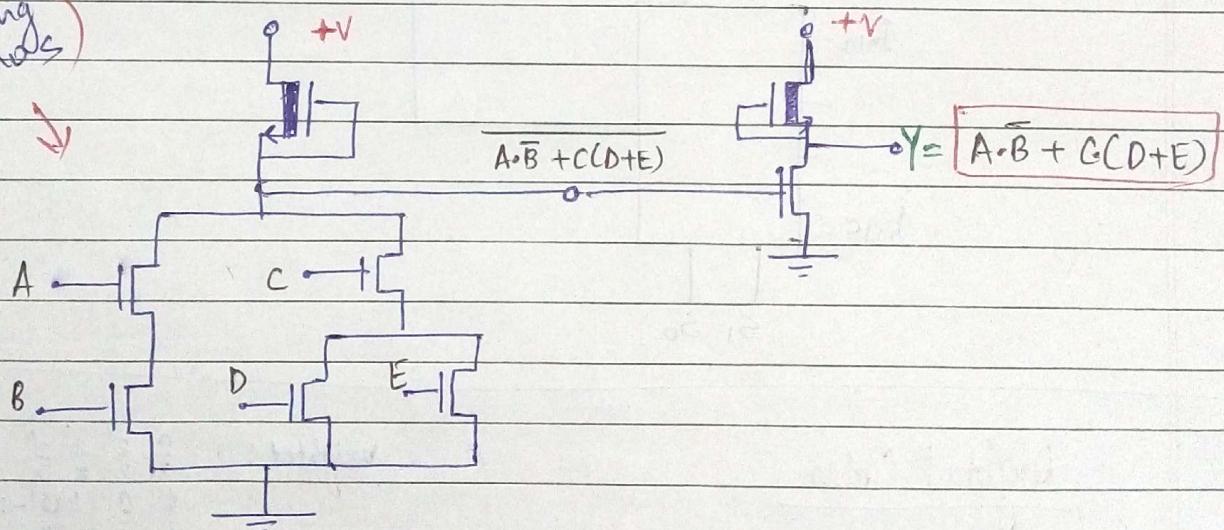
Advantage → very less power consumption
(CMOS)
because no resistance used.



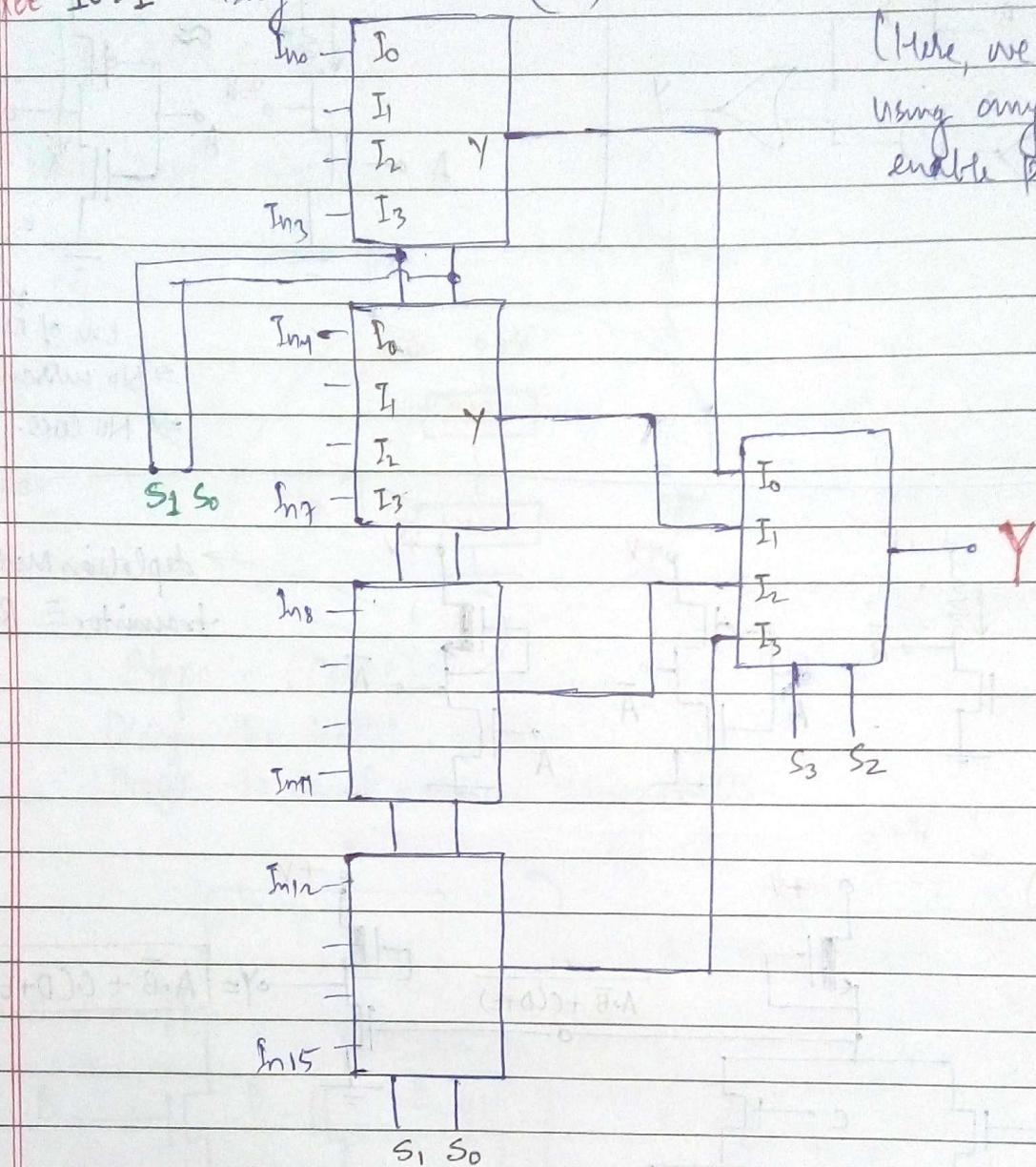
One of MOS if OFF
 \Rightarrow No current path
 \Rightarrow No loss.



(only using NMOS)



Multiplexer 16:1 using $\Rightarrow (4:1) \times 5$



(Here, we are not using any diodes and enable pins)

Digital Codes

$$\begin{array}{r} \text{Weighted : } \\ \begin{array}{ccccccccc} & 2^3 & 2^2 & 2^1 & 2^0 \\ & \times & \times & \times & \times \\ 1 & 0 & 1 & 0 & \rightarrow (10)_d \\ 1 & 1 & 1 & 1 & \\ 2^3 & 0 & 2^2 & 0 \\ \hline & 1 & 1 & 1 & 0 \end{array} \end{array}$$

Weighted 1.

Binary (0000 to 1111)

Codes 2.

Binary Coded Decimal (BCD) (0000 to 1001)

Non-Weighted 4.

Excess-3 (0011 to 1100)

Gray code.

Old form of data encryption.

10_d

BCD : 0001 0000

Excess-3: 0100 0011

Excess-3	Gray	Hexadex	Decimal	Binary	BCD
0011			0	0000	0000
0100	0000	0	1	0001	0001
0101	0001	1	2	0010	0010
0110	0011	2	3	0011	0011
0111	0010	3	4	0100	0100
1000	0110	4	5	0101	0101
1001	0111	5	6	0110	0110
1010	0101	6	7	0111	0111
1011	0100	7	8	1000	1000
1100	1100	8	9	1001	1001
1101	1101	9	10	1010	0001 0000
1110	1111	A	11	1011	0001 0001
1111	1110	B	12	1100	0001 0010
0001 0000	1010	C	13	1101	0001 0011
0001 0001	1011	D	14	1110	0001 0100
0001 0010	1001	E	15	1111	0001 0101
0001 0011	1000	F			

In Gray code, only one digit changes when we move from one no. to next no., whereas in Binary, any no. of change in digits is possible.
 eg. $7 \rightarrow 8$
 Binary $0111 \rightarrow 1000$ (4-digits change)
 Gray $0100 \rightarrow 1100$

\Rightarrow Gray \Rightarrow Less error (Only one flipping)

Dec.	-	10	5	15
Gray	-	1111	0111	1000
Hex	-	F	7	8
Binary	-	1010	0101	1111

Binary to Gray Converter

Orion

PAGE:
DATE:

I/P

O/P

Dec:

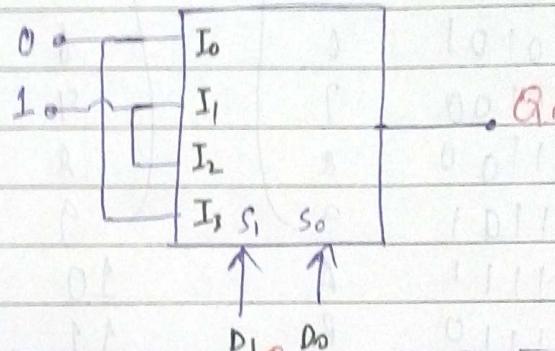
Binary

Gray

Q_0

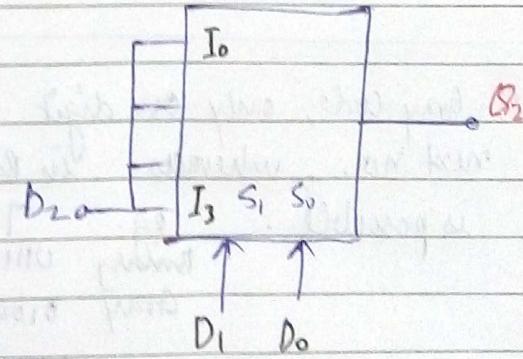
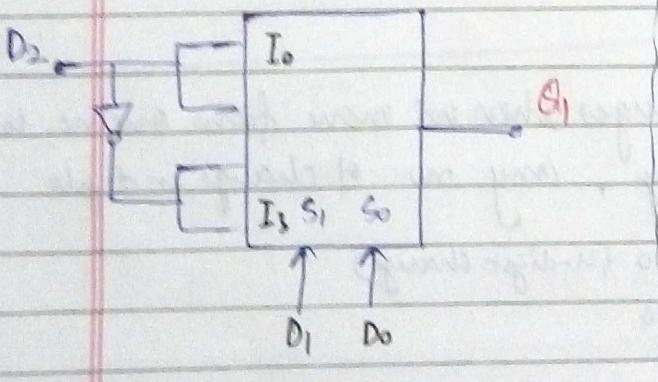
0	000	000
1	001	001
2	010	011
3	011	010
4	100	110
5	101	111
6	110	101
7	111	100

D_2	I_0	I_1	I_2	I_3
0	0	0	1	1
1	0	1	1	0
	0	1	1	0



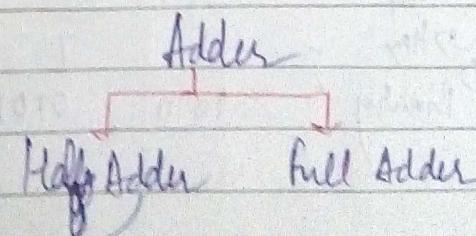
D_2	I_0	I_1	I_2	I_3
0	0	0	1	1
1	1	1	0	0

D_2	I_0	I_1	I_2	I_3
0	0	0	0	0
1	1	1	1	1

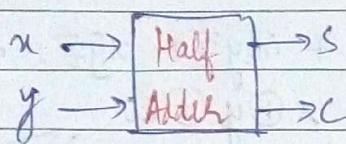


$x \oplus y$ $x + y$ Binary Addition

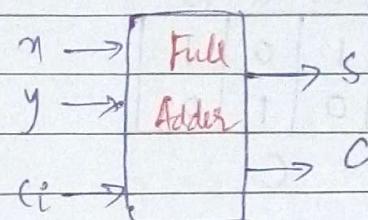
VAH	Y	X	Sum	Carry
0 0	0	0	0	0
0 1	1	0	1	0
1 0	1	1	0	1
1 1	0	1	1	1



Half Adder.



Full Adder.



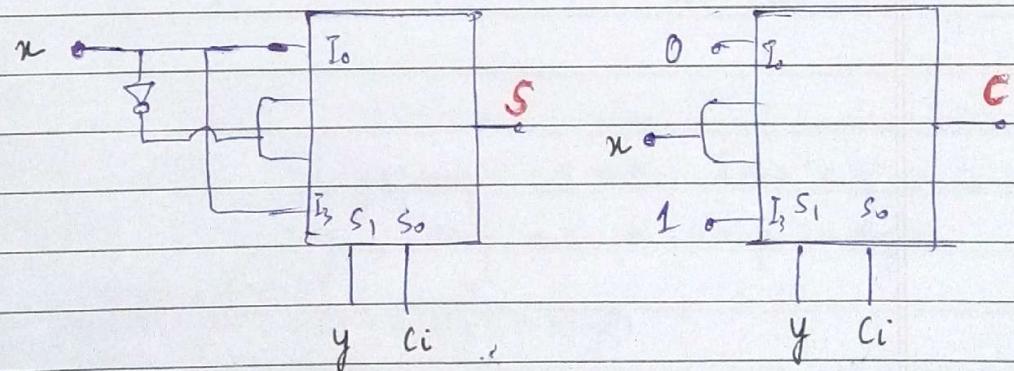
Tips

G/Ps

x	y	c_i	s	c_o
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

x	I_0	I_1	I_2	I_3	s
0	0	1	1	0	0
1	1	0	0	1	1

x	I_0	I_1	I_2	I_3	c
0	0	0	0	1	0
1	0	1	1	1	1



K-map

Karnaugh Map.

Simplification

Technique

AB	CD	00	01	11	10
00					
01					
11					
10					

 $x \backslash y \backslash i$ 00 01 11 10

0	0	1	3	2
1	4	5	7	6

S

$x \cdot y \cdot \bar{z}$	00	01	11	10
0	0	1	0	1
1	1	0	1	0

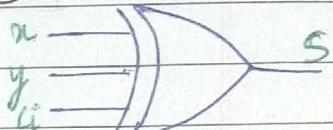
C

$x \cdot y \cdot \bar{z}$	00	01	11	10
0	0	0	1	0
1	0	1	1	1

$$z = \bar{c}i$$

$$S = \bar{x}\bar{y}z + \bar{x}y\bar{z} + x\bar{y}z + xy\bar{z}$$

$$= x \oplus y \oplus ci$$



$$C = xci + xi + yci$$

19/11/16

$$F = \Sigma(0, 1, 3, 4, 7, 12, 14)$$

$$d \leftarrow \Sigma(5, 13)$$

Don't care (0 or 1)

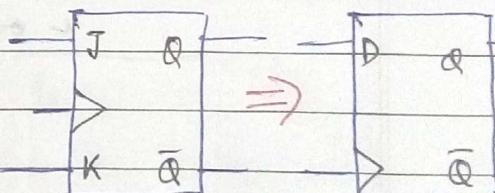
MSB USR

	dc	ba	00	01	11	10
d	1	1	1	0		
c	1	X	1	0		
b	D	X	0	I		
a	0	0	0	0		

$$Y = \bar{b}d + \bar{a}cd + ad.$$

Flip Flop Conversion

$$J-K \rightarrow D \quad (D, Q_n^{(jk)})$$



clk.

 Q_n

D

 Q_{n+1}

J K

1

0

0

0 x

1

0

1

1 x

1

1

0

x 1

1

1

1

x 0

clk.

 Q_n

J

K

 Q_{n+1}

J K

1

0

0

0 1

1

0

x

1 0

1

x

1

0 1

1

x

0

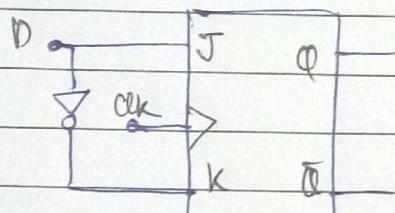
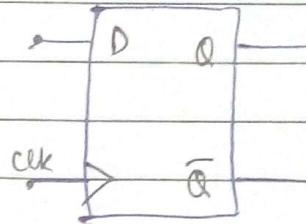
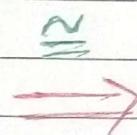
1 1

		J	K
D	Q		
0	0	1	
0	0		X
1	X	X	

$$J = D$$

		K	
D	Q		
0	X	X	
0	X		X
1	1	0	

$$K = \bar{D}$$


$$JK$$

$$D$$