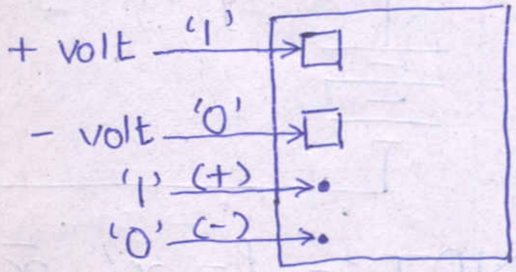
17/7/08

IT - 302

Ch - 1. Digital Electronics

Author - Morris Mano

Digital circuit is based on voltage and current nature.



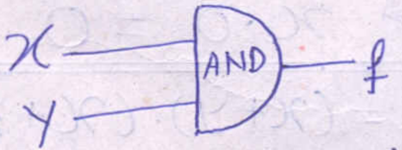
i) **Binary Logic** :- It consists of some binary variables and logical operations carried out upon them.

ii) **Variables** :- A, B, C . . . . Z or a, b, c . . . z [0, 1]
They exhibit either the value of 0 or 1.

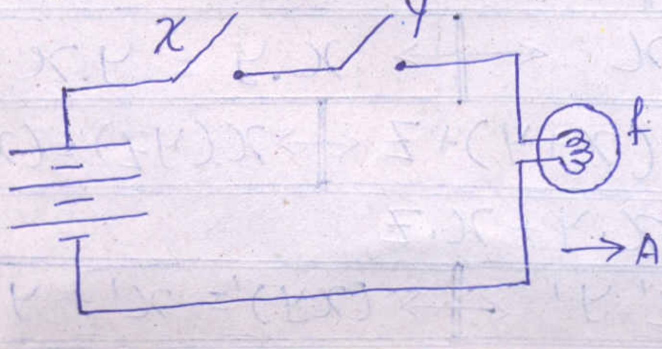
iii) **Logical Operations** :- AND, OR, NOT

iv) **Truth Table** :- A table which takes all possible values and its combinations of input variable on one side and other side shows corresponding output of the function.

a) **AND** :- $f = x$ AND y
↳ logical multiplication (.)



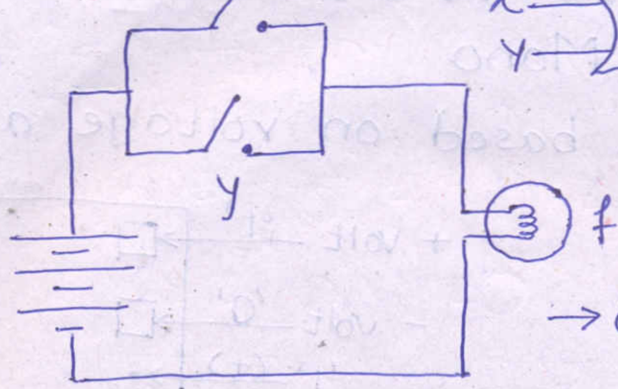
x	y	f
0	0	0
0	1	0
1	0	0
1	1	1



→ AND circuit

b) OR:- $f = x$ OR y

↳ logical addition (+)

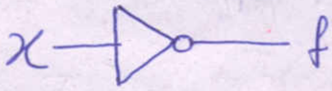


x	y	f
0	0	0
0	1	1
1	0	1
1	1	1

→ OR circuit

c) NOT :- $f = x', \bar{x}$

↳ logical inverter



x	f
0	1
1	0

Boolean Algebra

↳ binary

If x is a boolean function of binary variables - a, b, c, then its output will also be binary.

$$x = f(a, b, c)$$

Postulates and Theorem :-

i) $x + 0 = x$ ↔	$x \cdot 1 = x$
ii) $x + x' = 1$ ↔	$x \cdot x' = 0$
iii) $x + x = x$ ↔	$x \cdot x = x$
iv) $x + 1 = \cancel{x}1$ ↔	$x \cdot 0 = 0$
v) $(x')' = x$	$x + y.z = (x+y).(x+z)$
vi) $x + y = y + x$ ↔	$x.y = y.x$
vii) $x + (y+z) = (x+y) + z$ ↔	$x.(yz) = (xy).z$
viii) $x.(y+z) = x.y + x.z$	
ix) $(x+y)' = x'.y'$ ↔	$(xy)' = x' + y'$
x) $x + x.y = x$	

...ization of above $Th^m s$

i) $x + 0 = x$

$f = x + 0$

Hence, $\boxed{x + 0 = x}$

x	f
0	0
1	1

$\} \; x$

ii) $f = x + x'$, $f' = x + x$

Hence, $\boxed{x + x' = 1}$

and $\boxed{x + x = x}$

x	x'	f	f'
0	1	1	0
1	0	1	1

iv) $f \equiv x + 1$

Hence, $\boxed{x + 1 = 1}$

x	f
0	1
1	1

v) $f \equiv (x')'$

Hence, $\boxed{(x')' = x}$

x	x'	f
0	1	0
1	0	1

$\} \; x$

viii)

x	y	z	$(y+z)$	$x.(y+z)$	$x \cdot y$	$x \cdot z$	$x \cdot y + x \cdot z$
0	0	0	0	0	0	0	0
0	0	1	1	0	0	0	0
0	1	0	1	0	0	0	0
0	1	1	1	0	0	0	0
1	0	0	0	0	0	0	0
1	0	1	1	1	0	1	1
1	1	0	1	1	1	0	1
1	1	1	1	1	1	1	1

Hence, $\boxed{x \cdot (y + z) = (x \cdot y + x \cdot z)}$

ix) $(x+y)' = x'.y'$ — DeMorgan's Thm.

x	y	x'	y'	$(x+y)$	$(x+y)'$	$x'.y'$
0	0	1	1	0	1	1
0	1	1	0	1	0	0
1	0	0	1	1	0	0
1	1	0	0	1	0	0

In DeMorgan's Thm $+ \rightleftharpoons .$

var $\rightleftharpoons$ (var)'

1> $(x + y.z')'$

$\Rightarrow x'.(y.z')'$ (By DeMorgan's Thm)

$\Rightarrow \underline{\underline{x'.(y'+z)}}$

2> $(x'y + xz)'$

$\Rightarrow (x'y)'.(xz)'$ (By DeMorgan's Thm)

$\Rightarrow \underline{\underline{(x+y').(x'+z')}}$

3> $[(x+yz)(x'+y)]'$

$\Rightarrow (x+yz)' + (x'+y)'$

$\Rightarrow x'.(yz)' + x.y'$

$\Rightarrow \underline{\underline{x'.(y'+z') + x.y'}}$

Complement of a Function

1) $f = ABC' + AB + BC$

So, the complement of f is

$\bar{f} = (ABC' + AB + BC)'$

or, $\bar{f} = (ABC')'.(AB)'.(BC)'$ (By DeMorgan's Thm)

or, $\bar{f} = (A'+B'+C).(A'+B').(B'+C')$

2) $f = (x+y')(xyz+y')$

So, the complement of f is

$\bar{f} = [(x+y')(xyz+y')]'$

or, $\bar{f} = (x+y')' + (xyz+y')'$ (By DeMorgan's Thm)

or, $\bar{f} = x'.y + (x'+y'+z').y$

MinTerm & MaxTerm

Dec.	x	y	z	Min term(m_i)	(Max term) M_i	f_1	f_1'
0	0	0	0	$x'.y'.z' - m_0$	$x+y+z - M_0$	0	1
1	0	0	1	$x'.y'.z - m_1$	$x+y+z' - M_1$	1	0
2	0	1	0	$x'.y.z' - m_2$	$x+y'+z - M_2$	0	1
3	0	1	1	$x'.y.z - m_3$	$x+y'+z' - M_3$	0	1
4	1	0	0	$x.y'.z' - m_4$	$x'+y+z - M_4$	1	0
5	1	0	1	$x.y'.z - m_5$	$x'+y+z' - M_5$	0	1
6	1	1	0	$x.y.z' - m_6$	$x'+y'+z - M_6$	0	1
7	1	1	1	$x.y.z - m_7$	$x'+y'+z' - M_7$	1	0

$f = x'y'z + xy'z' + xyz$

'0' is represented by complement

'1' is represented by complement

Minterm and maxterms are complements of each other.

22/7/08

$f_1 = x'y'z + xy'z' + xyz$ → This is sum of product

$= \sum(m_1, m_4, m_7)$

$= \sum(1, 4, 7)$ $\boxed{\sum \Rightarrow \text{Sum of min term}}$

$f_1' = (x'y'z + xy'z' + xyz)'$

or, $f_1' = (x'y'z)' \cdot (xy'z')' \cdot (xyz)'$

or, $f_1' = (x + y + z') \cdot (x' + y + z) \cdot (x' + y' + z')$

or, $f_1' = M_1 \cdot M_4 \cdot M_7$

or, $f_1' = \pi(M_1, M_4, M_7)$

or, $f_1' = \pi(1, 4, 7)$

$\boxed{\pi \Rightarrow \text{Product of max term}}$

Complementing above term

or, $(f_1')' = f_1 = \{\pi(1, 4, 7)\}'$

or, $\boxed{f_1 = \pi(0, 2, 3, 5, 6) = \sum(1, 4, 7)}$

eg:- 1) $f = xy + xz$ → sum of product

To convert into product of sum

$\therefore f = (xy + x) \cdot (xy + z)$ $\{\because x + yz = (x + y) \cdot (x + z)$

or, $f = (x + x)(x + y)(x + z)(\cancel{z} + y)$

or, $\boxed{f = x\,(x + y)(x + z)(y + z)}$

2> $f = (X+Y)(XY+Z)(ZX'+Y')$

To convert into sum of product,

or, $f = [X \cdot XY + XZ + Y \cdot XY + YZ](ZX'+Y')$

or, $f = (XY + XZ + XY + YZ)(ZX'+Y')$

or, $f = XY'Z + X'YZ$

Canonical or Standard Form:-

A boolean function is in canonical form if all of its terms are represented by maximum no. of variables.

eg - $f = AB' + A'B$

$= \Sigma(1,2)$

A	B	f	
0	0	$A'B'$	0
0	1	$A'B$	1
1	0	AB'	2
1	1	AB	3

Here both terms are made up of the two variables, hence they are in canonical form.

Q.1> $f = X + X'Y$

or, $f = X(Y+Y') + X'Y$

or, $f = XY + XY' + X'Y$

or $f = \Sigma(1,2,3)$

X	Y	f	
0	0	$X'Y'$	0
0	1	$X'Y$	1
1	0	XY'	2
1	1	XY	3

Q.2> $f = XZ + Y'Z + XYZ + X'$

or, $f = XZ(Y+Y') + Y'Z(X+X') + XYZ + X'(Y+Y')$

or, $f = \underline{XYZ} + \underline{XY'Z} + \underline{XY'Z} + X'Y'Z + \underline{XYZ} + X'Y + X'Y'$

or, $f = XYZ + XY'Z + X'Y'Z + X'Y(Z+Z') + X'Y'(Z+Z')$

or, $f = XYZ + \underline{XY'Z} + \underline{X'Y'Z} + X'YZ + X'YZ' + \underline{X'Y'Z}$

Or, $f = XYZ + XY'Z + X'Y'Z + X'YZ + X'YZ'$
$\quad + X'Y'Z + X'Y'Z'$

or, $f = \Sigma(7,5,1,3,2,0)$ or $f = \pi(4,6)$

Q.3> $f = (X+YZ').(X'+Y)(X+Z)$

Or, $f = (X+Y).(X+Z')(X'+Y)(X+Z)$

$\{\because X+YZ = (X+Y).(X+Z$

Or, $f = (X+Y)(X'+Y)(X+Z')(X+Z)$

~~Or,~~ $\cancel{f = (XY + X'Y + Y)(X + XZ + XZ')}$

~~Or,~~ $\cancel{f = (XY + XYZ + XYZ' + XY + XYZ + XYZ')}$

~~Or,~~ $\cancel{f = XY + XYZ + XYZ'}$

~~Or,~~ $\cancel{f = XYZ + XYZ' + XYZ + XYZ'}$

~~Or,~~ $\cancel{f = XYZ + XYZ'}$

Or, $f = (X+Y+Z.Z')(X'+Y+ZZ')(X+Z'+YY')$
$\quad (X+Z+YY')$

Or, $f = \underline{(X+Y+Z)}\,\underline{(X+Y+Z')}\,(X'+Y+Z)$
$\quad (X'+Y+Z')\,\underline{(X+Z'+Y)}\,(X+Z'+Y')$
$\quad \underline{(X+Z+Y)}\,(X+Z+Y')$

or, $f = M_0 . M_1 . M_3 . M_4 . M_5 . M_2$

or, $f = \pi(0,1,2,3,4,5)$

* **NOTE**: # For max term, use $x.x'$ and ad

For min term, use $x+x'$ and multipl

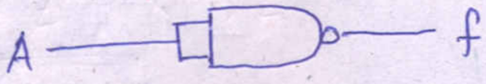
Universal Gate :-

1) NAND gate :-

$f = (A \cdot B)'$

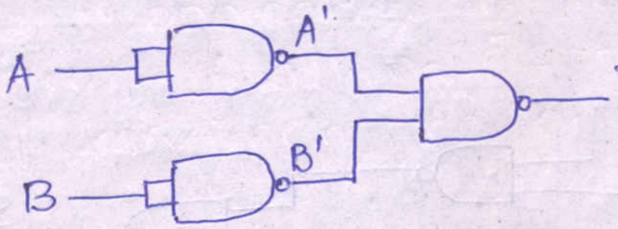
It is universal gate as all gates can be realized by it.

NOT →



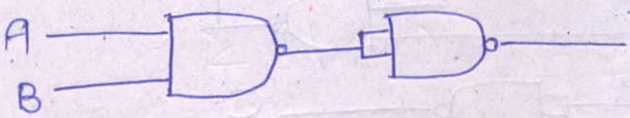
, $f = (A \cdot A)' = A'$

OR →



$f = (A' \cdot B')' = A + B$

AND →



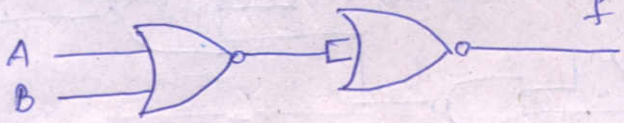
$f = A \cdot B$

2) NOR gate :-

NOT →

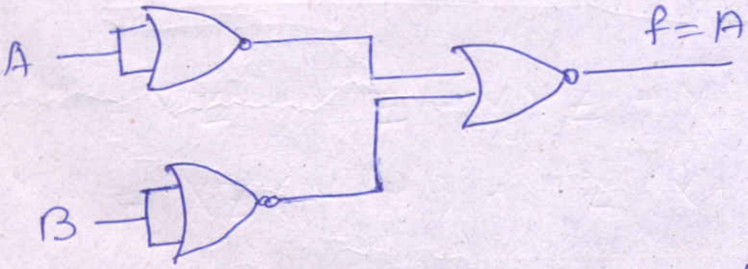
$f = A'$

OR →



$f = A + B$

AND →



$f = A \cdot B$

Q. Take $f = XY' + X'YZ + XYZ'$
Realise this function using NAND gates only, and separately by NOR gates only.

Q. Realise half adder and half subtractor circuits.

Half Adder

x	y	S	C
0	0	0	0
0	1	1	0
1	0	1	0
1	1	0	1

$S = \Sigma(1,2)$

or, $S = X'Y + XY'$

$C = XY$

Half Subtractor

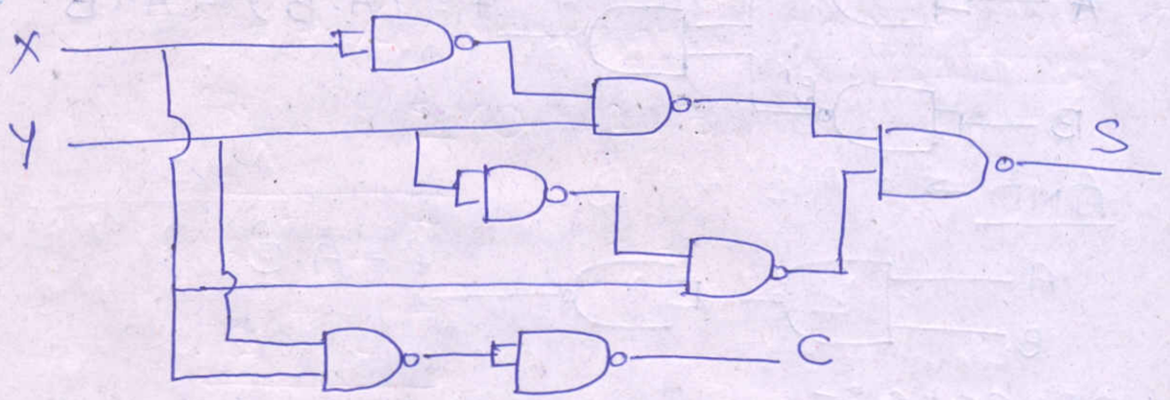
	x	y	d	Borro
0	0	0	0	0
1	0	1	1	1
2	1	0	1	0
3	1	1	0	0

$d = \Sigma(1,2)$

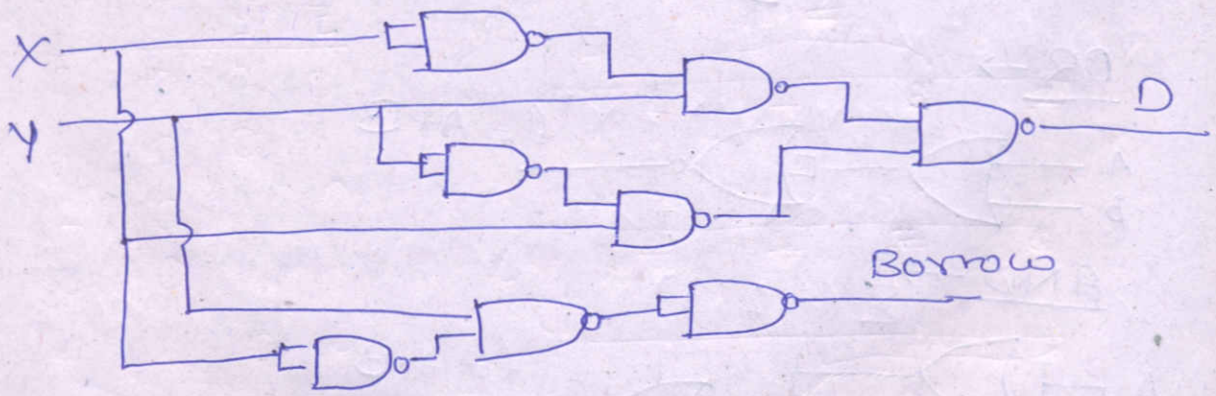
or, $d = X'Y + XY'$

$borrow = X'Y$

Half adder –



Half subtractor –



29/07/08

Simplification of Boolean Function

If $f = f(x, y, z)$

$= \Sigma(1, 2, 3, 4, 5, 6)$

$\Downarrow$

$= \Sigma(\_\_\_)$ (having lesser terms)

Methods for simplification :-

1) Algebraic method

2) Karnough's Map (K-Map)

K-Map

2 variable

	x	y
0	0	0
1	0	1
2	1	0
3	1	1

		y
	00 [0]	01 [1]
x	10 [2]	11 [3]

$\Rightarrow$ x is represented by 2 and 3

$\Rightarrow$ y is represented by 1 and 3

Ex:- $f = \Sigma(1, 2, 3)$

Simplifying by algebraic method

$$f = x'y + xy' + xy$$

or, $f = x'y + x(y + y')$

or, $f = x'y + x = (x + x')(x + y) = x + y$

Simplifying by K-Map

		y
	0 [0]	1 [1]
x	1 [2]	1 [3]

If term is present, put '1' else '0'.

$\Rightarrow$ If there are side by side common blocks, pair them up.

Each pair is represented by one term

so, $x + y$

<u>Eg:-</u> $f = \Sigma(2,3)$

$= xy' + xy = x$

	x	y
0	0	0
1	0	1
2	1	0
3	1	1

		y
	0 [0]	0 [1]
x	1 [2]	1 [3]

$\Rightarrow f = x$ (representing pair by one term)

|||ly for $f = \Sigma(0,1,2,3)$

1 [0]	1 [1]
1 [2]	1 [3]

, $\underline{f = 1}$

For 3-variable system,

	x	y	z
0	0	0	0
1	0	0	1
2	0	1	0
3	0	1	1
4	1	0	0
5	1	0	1
6	1	1	0
7	1	1	1

		z	z	
	[0]	[1]	[3]	[2]
x	[4]	[5]	[7]	[6]
			y	y

eg $f = x'yz + xy'z' + xyz + xyz'$

$= \Sigma(3, 4, 7, 6)$

	[0]	[1]	[3]	[2]
			1	
x	1		1	1

(z over columns 1, 3; y under columns 3, 2)

Note – End blocks can be paired as here in 4 and 6

So, $f = xy + yz + xz'$

Q.1) If $A'C + A'B + AB'C = f$, simply?

Solution

$f = A'C(B+B') + A'B(C+C') + AB'C$

$f = \underline{A'BC} + \underline{A'B'C} + \underline{A'BC} + A'BC' + \underline{AB'C}$

$f = AB'C + A'BC + A'BC' + A'B'C$

$f = \Sigma(5, 1, 3, \cancel{2})$

	[0]	[1]	[3]	[2]
		1	1	1
A		1		$\cancel{1}$

(C over columns 1, 3; B under columns 3, 2)

$f = A'B\cancel{C} + B'C$

Q.2) $F = \Sigma(0, 2, 4, 5, 6)$

	[0]	[1]	[3]	[2]
	1			1
x	1	1		1

(z over columns 1, 3; y under columns 3, 2)

$f = xy' + z'$

NOTE:– i) Combining Two blocks decreases 1 variable, for four blocks decreases by 2 variables for eight blocks decreases by 3 variables

31/07...

Four variable K map :-

Four variables K map

	w	x	y	z
0	0	0	0	0
1	0	0	0	1
2	0	0	1	0
3	0	0	1	1
4	0	1	0	0
5	0	1	0	1
6	0	1	1	0
7	0	1	1	1
8	1	0	0	0
9	1	0	0	1
10	1	0	1	0
11	1	0	1	1
12	1	1	0	0
13	1	1	0	1
14	1	1	1	0
15	1	1	1	1

z (columns 2–3), y (columns 3–4), x (rows 2–3), w (rows 3–4)

0	1	3	2
4	5	7	6
12	13	15	14
8	9	11	10

Q.1 > Simplify $F(w, x, y, z) = \Sigma(0, 1, 2, 4, 5, 6, 8, 9, 12, 13, 14)$

Solution

z (columns 2–3), y (columns 3–4), x (rows 2–3), w (rows 3–4)

0: 1	1: 1	3: 0	2: 1
4: 1	5: 1	7: 0	6: 1
12: 1	13: 1	15: 0	14: 1
8: 1	9: 1	11: 0	10: 0

$F(w, x, y, z)$

$= y' + xz' + w'z'$

Q.2) $F(w, x, y, z) = \Sigma(2, 3, 12, 13, 14, 15)$

Solution

z

0: 0	1: 0	2: 1	3: 1
4: 0	5: 0	7: 0	6: 0
12: 1	13: 1	15: 1	16: 1
8: 0	9: 0	11: 0	10: 0

x W

y

$F(w, x, y, z)$
$= WX + W'X'Y$

Five Variable K-map :- (A, B, C, D, E)

C

0	1	3	2	6	7	5	4
8	9	11	10	14	15	13	12
24	25	27	26	30	31	29	28
16	17	19	18	22	23	21	20

B A

E D E

Simplification of the function

Product of sum :-

~~F = Σ(0,1,2,5,8,9,10)~~

Eg:- $F = \Sigma(1,3,4,6) \Rightarrow$ SOP

	$x'z'$	$x'z$	yz	yz'
x'	0 [0]	1 [1]	1 [3]	0 [2]
x	1 [4]	0 [5]	0 [7]	1 [6]

$F = X'Z + XZ'$

$F = \Sigma(1,3,4,6)$

or, $F = \pi(0,2,5,7) \Rightarrow$ POS

		z		
x'	0 [0]	1 [1]	1 [3]	0 [2]
x	1 [4]	0 [5]	0 [7]	1 [6]

$F' = XZ + x'z'$

or, $F = (X' + Z').(X + Z)$

Q.1) $F(\cancel{w,x,y,z})$ [a,b,c,d] $= \Sigma(0,1,2,5,8,9,10)$

Simplify the function in product of sum form.

Solution

$F = \pi(3,4,6,11,12,13,14,15)$

		D		
	1 [0]	1 [1]	0 [3]	1 [2]
B	0 [4]	1 [5]	0 [7]	0 [6]
B, A	0 [12]	0 [13]	0 [15]	0 [14]
A	1 [8]	1 [9]	0 [11]	1 [10]
			C	C

$F' = AB + CD + BD'$

$F = (A'+B').(C'+D').(B'+D)$

Don't Care Condition :-

These are combinations which are mathematically and logically true, but cannot be implemented.

W	x	y	z	L_0	L_1	L_2 ... L_4 ... L_9
0	0	0	0	1	0	0 - - . . 0
⋮	⋮	⋮	⋮			
0	1	0	0	0	0	0 . . . 1 0
⋮	⋮	⋮	⋮			
1	0	0	1	0	0	0 . . . 0 1
1	0	1	0	X - - - - -		

X ⇒ don't care.

Q.1> Simplify

$F(w, x, y, z) = \Sigma(1, 3, 7, 11, 15)$

$d(w, x, y, z) = \Sigma(0, 2, 5)$

Solution

X [0]	1 [1]	1 [3]	X [2]
0 [4]	X [5]	1 [7]	0 [6]
0 [12]	0 [13]	1 [15]	0 [14]
0 [8]	0 [9]	1 [11]	0 [10]

z (columns 2–3), y (columns 3–4), x (rows 2–3), w (rows 3–4)

X ⇒ not necessarily needed to be combined.

$F = YZ + X'W'$

Simplification of Boolean Function

Quine McClusky Method :- (Tabulation method)

Simplify $F = \Sigma(10, 11, 14, 15)$

Soln

Since, max is 15, 4 variables required,

Dec	w	x	y	z
0	0	0	0	0
1	0	0	0	1
2	0	0	1	0
8	1	0	0	0
10	1	0	1	0
11	1	0	1	1
14	1	1	1	0
15	1	1	1	1

Step 1 :- Categorise minterms according to no. of 1's they contain

Step 2 :- Combine two successive elements differing in only one bit.

Dec	w	x	y	z
0,1	0	0	0	–
0,2	0	0	–	0
0,8	–	0	0	0
2,10	–	0	1	0
8,10	1	0	–	0
10,11	1	0	1	–
10,14	1	–	1	0
11,15	1	–	1	1
14,15	1	1	1	–

$0 = w'x'y'z' + w'x'y'z$
$= w'x'y'$

Quine McClusky Method:- (Fabulation method)

Simplify $F = \Sigma(10, 11, 14, 15)$

Soln

Since, max is 15, 4 variables required,

Dec	w	x	y	z
0	0	0	0	0
1	0	0	0	1
2	0	0	1	0
8	1	0	0	0
10	1	0	1	0
11	1	0	1	1
14	1	1	1	0
15	1	1	1	1

Step 1:- Categorise minterms according to no. of 1's they contain

Step 2:- Combine two successive elements differing in only one bit

Dec	w	x	y	z
0,1	0	0	0	-
0,2	0	0	-	0
0,8	-	0	0	0
2,10	-	0	1	0
8,10	1	0	-	0
10,11	1	0	1	-
10,14	1	-	1	0
11,15	1	-	1	1
14,15	1	1	1	-

$0 = w'x'y'z' + w'x'y'z$

$= w'x'y'$

Step 3:- Repeat step 2.

Dec	w	x	y	z	
0,1	0	0	0	-	$\rightarrow w'x'y'$
(0,2)(8,10)	-	0	-	0	$\rightarrow x'z'$
(0,8)(2,10)	-	0	-	0	$\rightarrow x'z'$
(10,11),(14,15)	1	-	1	-	$\rightarrow wy$
(10,14),(11,15)	1	-	1	-	$\rightarrow wy$

This cannot be simplified further, as noterm with single bit differ

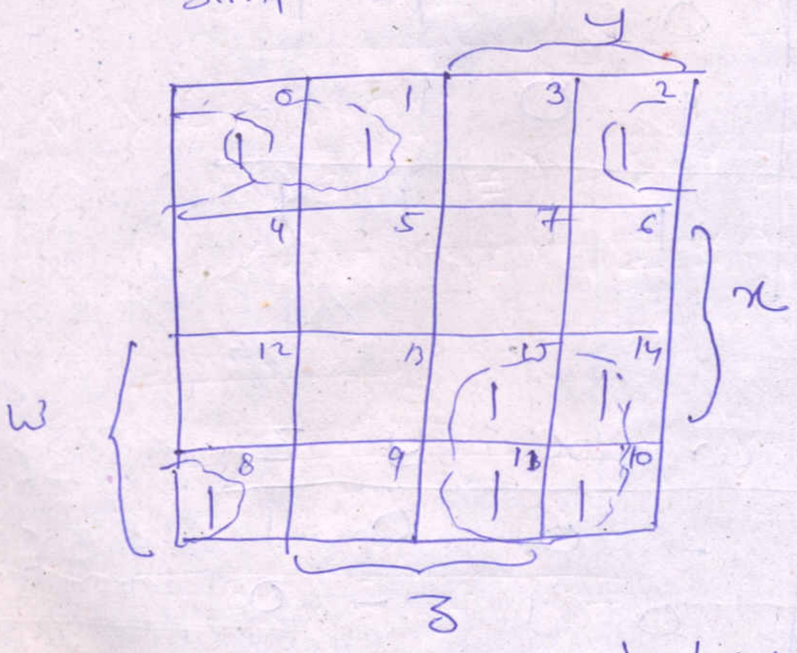
$\therefore$ we write corresponding values on RHS.

$\therefore F = w'x'y' + x'z' + wy$

Now, two cases arise:

i) Prime implicate and simplified term

ii) Essential prime implicant.



Hence, $F = wy + x'z' + w'x'y'$

Q. Simplify the boolean function. Find its prime implicant and essential prime implicant of function

$F(w, x, y, z) = \Sigma(1, 4, 6, 7, 8, 9, 10, 11, 15)$

Soln:

Step I :-

Dec	w	x	y	z
1	0	0	0	1
4	0	1	0	0
8	1	0	0	0
6	0	1	1	0
9	1	0	0	1
10	1	0	1	0
7	0	1	1	1
11	1	0	1	1
15	1	1	1	1

Step II :-

Dec	w	x	y	z
1,9	-	0	0	1
4,6	0	1	-	0
8,9	1	0	0	-
8,10	1	0	-	0
6,7	0	1	1	-
9,11	1	0	-	1
10,11	1	0	1	-
7,15	-	1	1	1
11,15	1	-	1	1

Step III :-

Dec	w	x	y	z	
1, 9	–	0	0	1	$x'y'z$
4, 6	0	1	–	0	$w'xz'$
(8, 9), (10, 11)	1	0	–	–	wx'
6, 7	0	1	1	–	$w'xy$
9, 11	1	0	–	1	–
10, 11	1	0	1	–	–
(7, 15)	–	1	1	1	xyz
(11, 15)	1	–	1	1	wyz

$F = x'y'z + w'xz + w'x' + w'xy + xyz + wyz$

	1	4	6	7	8	9	10	11	15
(1, 9) $x'yz$	X					X			
(8, 9, 10, 11) wx'					X	X	X	X	
(4, 6) $w'xz'$		X	X						
(6, 7) $w'xy$			X	X					
(7, 15) xy				X					X
(11, 15) wyz								X	X

Parity Checker :-

Design circuit that will take 4-bit number and check whether there are odd number of '1' or even number of '1

If all bits are zero, no led will glo

	A	B	C	L_1	L_2
0	0	0	0	0	0
1	0	0	1	0	1
2	0	1	0	0	1
3	0	1	1	1	0
4	1	0	0	0	1
5	1	0	1	1	0
6	1	1	0	1	0
7	1	1	1	0	1

Simplifying by K-map.

L_1//

		1	
	1		1

$L_1 = A'BC + AB'C + AB$

L_2//

	1	1	1
1		1	

$L_2 = A'B'C + A'BC' + AB'C' + ABC$

Binary coded Decimal / Excess-3 code

Decimal	A	B	C	D	W	X	Y	Z
0	0	0	0	0	0	0	1	[illegible]
1	0	0	0	1	0	1	0	[illegible]
2	0	0	1	0	0	1	0	[illegible]
3	0	0	1	1	0	1	1	[illegible]
4	0	1	0	0	0	1	1	[illegible]
5	0	1	0	1	1	0	0	[illegible]
6	0	1	1	0	1	0	0	[illegible]
7	0	1	1	1	1	0	1	[illegible]
[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	1	0	1	[illegible]

Excess-3 code

Q. Design a digital circuit that will convert 4-bit BCD number to excess-3 code.

Soln

decimal	A	B	C	D	W	X	Y	Z
0	0	0	0	0	0	0	1	1
1	0	0	0	1	0	1	0	0
2	0	0	1	0	0	1	0	1
3	0	0	1	1	0	1	1	0
4	0	1	0	0	0	1	1	1
5	0	1	0	1	1	0	0	0
6	0	1	1	0	1	0	0	1
7	0	1	1	1	1	0	1	0
8	1	0	0	0	1	0	1	1
9	1	0	0	1	1	1	0	0
10	1	0	1	0	X	X	X	X
⋮	⋮	⋮	⋮	⋮	X	X	X	X

$W = \Sigma(5,6,7,8,9) + d\Sigma(10,11,12,13,14,15)$

$X = \Sigma(1,2,3,4,9) + d\Sigma(10,11,12,13,14,15)$

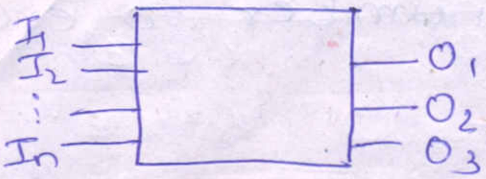
$Y = \Sigma(0,3,4,7,8) + d\Sigma(10,11,12,13,14,15)$

$Z = \Sigma(0,2,4,6,8) + d\Sigma(10,11,12,13,14,15)$

[$\because$ d = Don't care]

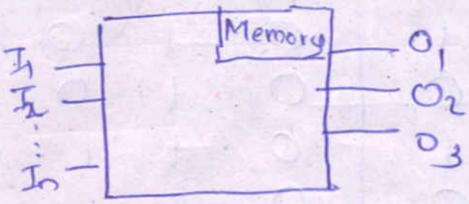
Combinational Logic

5/4/



Output depends only on input.

Sequential logic



Output depends not only on input but also on values processed before (which are kept in memory)

Examples of combinatorial logic :-

i) Adder
ii) Subtractor
iii) code converter
iv) Multiplexer
v) De-multiplexer

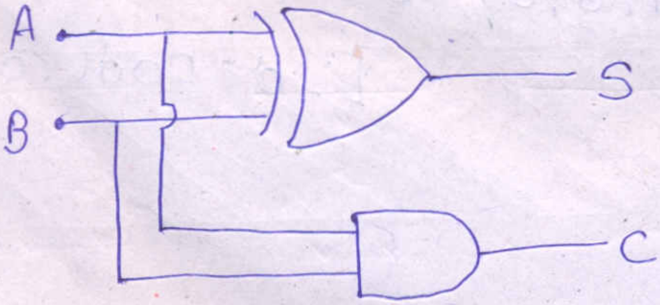
a) Half Adder -

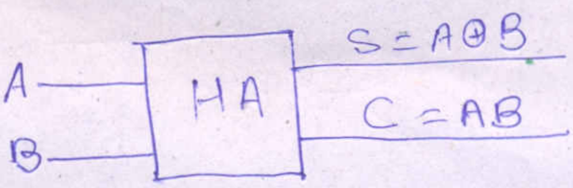
A	B	S	C
0	0	0	0
0	1	1	0
1	0	1	0
1	1	0	1

$S = AB' + A'B$

$S = A \oplus B$

$C = A.B$





b) Full Adder –

A	B	C	S	C
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

$S = \Sigma(1, 2, 4, 7)$

or, $S = A'B'C + A'BC' + AB'C' + ABC$

or, $S = A'(B'C + BC') + A(B'C' + BC)$

or, $S = A'(B \oplus C) + A(B'C' + BC)$

or, $S = A'(B \oplus C) + A(BC' + B'C)'$

or, $S = A'(B \oplus C) + A(B \oplus C)'$

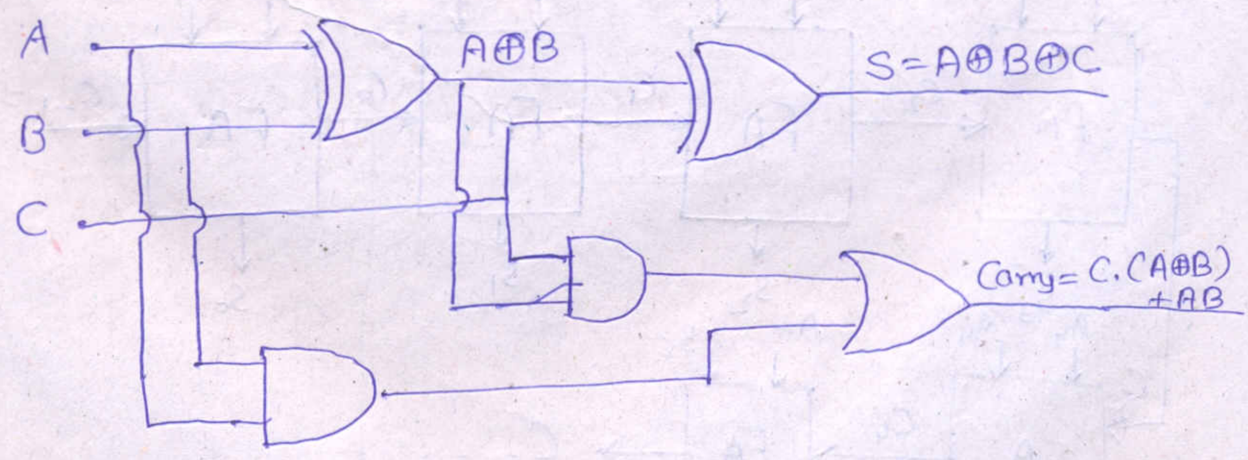
or, $\boxed{S = A \oplus B \oplus C}$

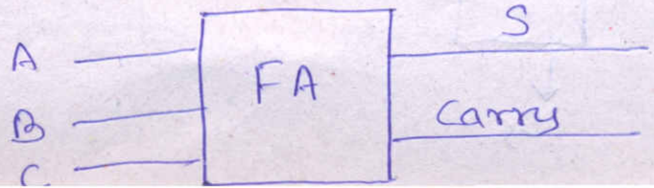
$Carry = \Sigma(3, 5, 6, 7)$

or, $Carry = A'BC + AB'C + ABC' + ABC$

or, $Carry = C(A'B + AB') + AB(C + C')$

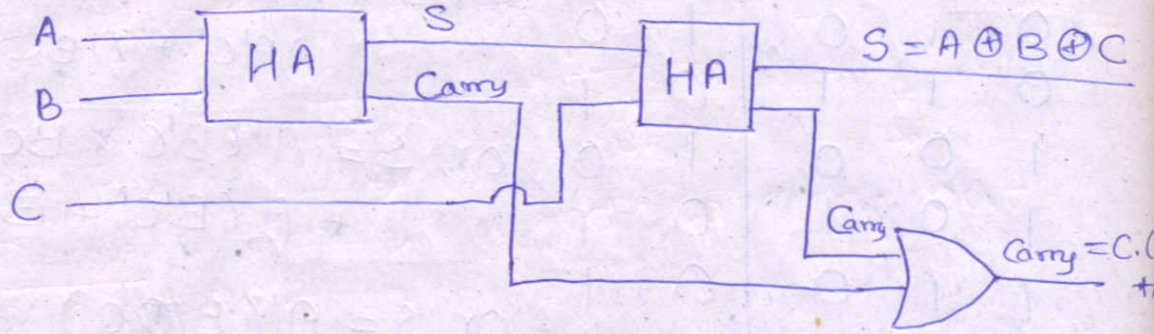
or, $\boxed{Carry = C.(A \oplus B) + AB}$





Q. Design a full adder circuit using two half adder and an or gate.

Soln



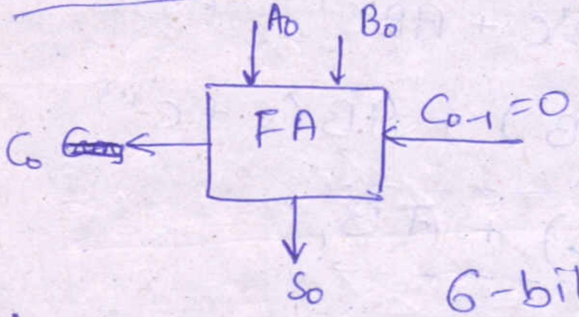
eg:- A = 10111
B = 10011

A+B =
$$\begin{array}{r} 10111 \\ 10011 \\ \hline 101010 \end{array}$$

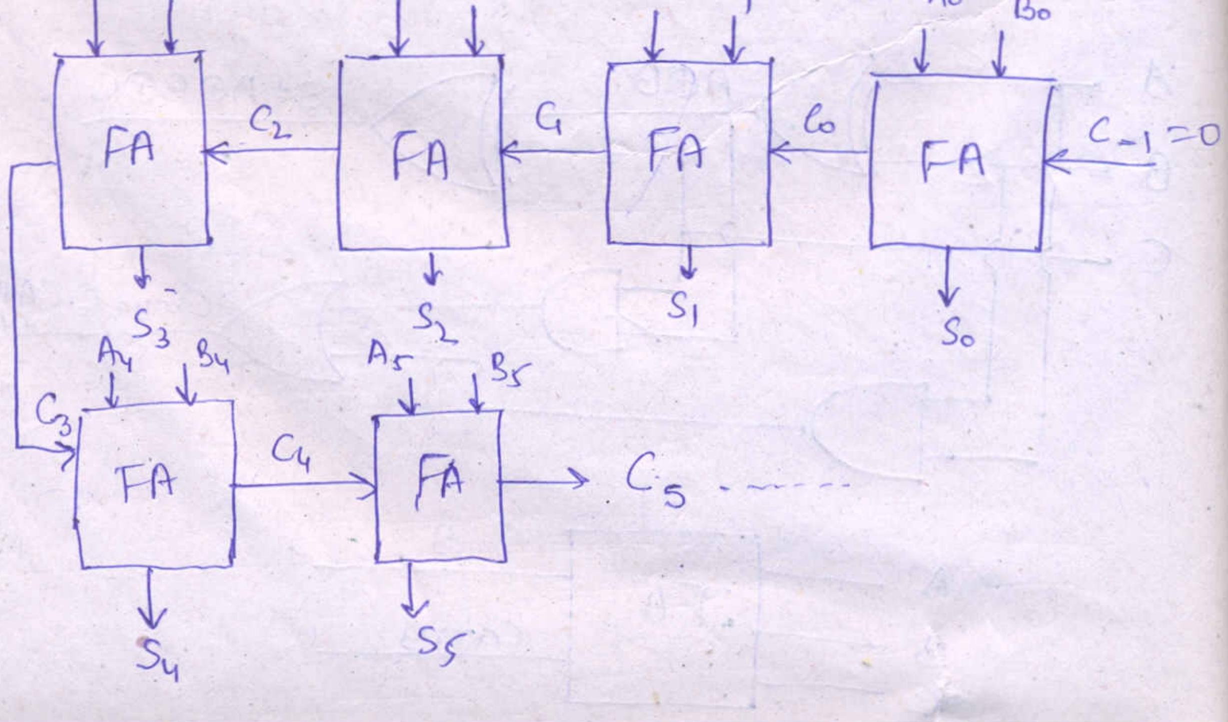
x = 1011
y = 1110

x+y =
$$\begin{array}{r} 1011 \\ 1110 \\ \hline 11001 \end{array}$$

Block Diagram of full adder



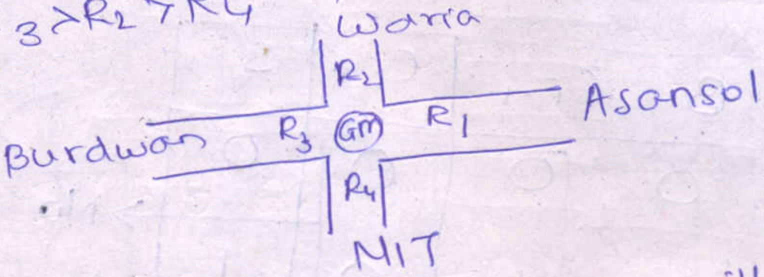
6-bit parallel Adder circu



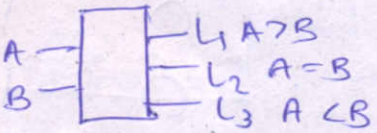
Assignment - Design a parallel adder circuit that will perform both 2's complement addition and subtraction depending on the user's choice.

Q. Design a logic ckt that will perform the following operation?
If the number of vehicles is hundred or more than 100, then that particular road will be shown green light.
If a situation comes that more than one road have more than or equal to 100 vehicles, then depending on the road priority, the green light will be shown.

$R_1 > R_3 > R_2 > R_4$ - Priority



Q. Design a logic circuit that will compare two 2-bit numbers and accordingly, output will be shown.

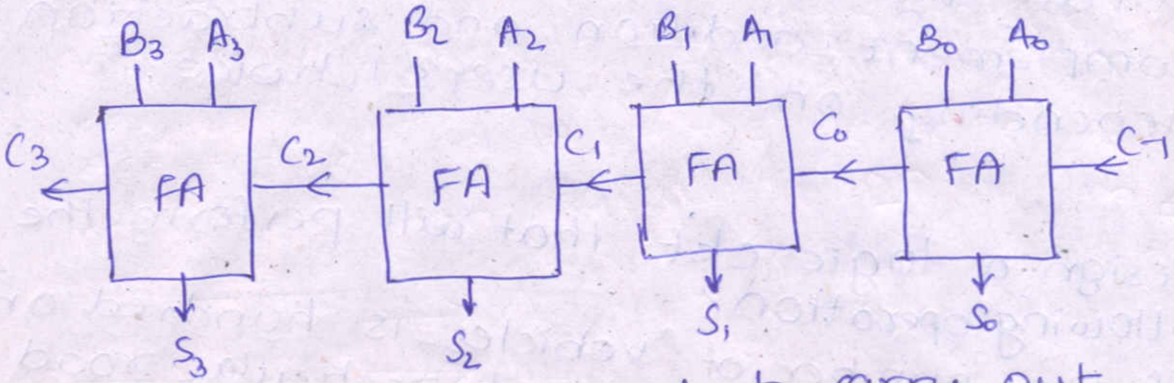


Full subtractor-

A	B	C	D	Bo
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
1	1	1	1	1

07/8/08

Parallel Adder :- (4-bit)



<u>Delay</u> is time required to carry out operation.

If delay for 1 bit operation is 't' sec

Then for 'n' bit, will be 'nt' seconds.

Carry Look Ahead Adder :-

A	B	C_{i-1}	C_i	S
0	0	0	0	0
0	0	1	0	1
0	1	0	0	1
0	1	1	1	0
1	0	0	0	1
1	0	1	1	0
1	1	0	1	0
1	1	1	1	1

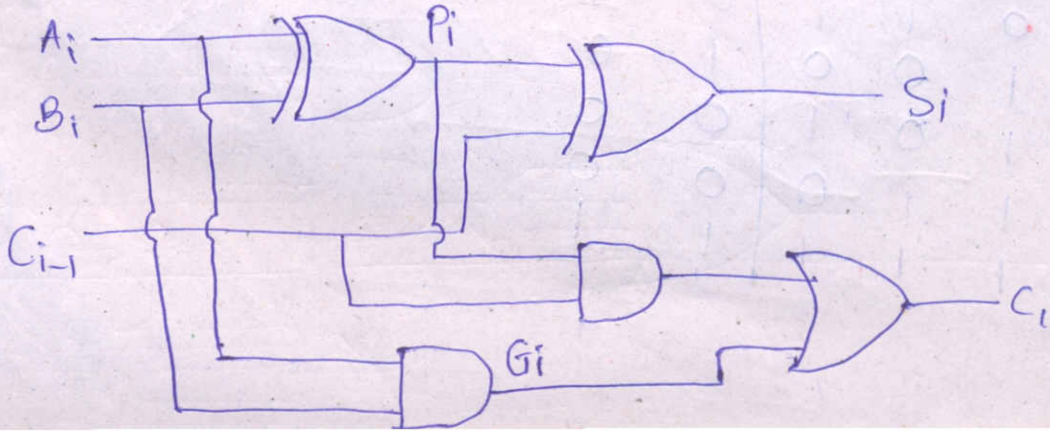
→ No carry

→ Carry propagat
or Carry(n) = Carry(

$\boxed{P_i = A_i \oplus B_i}$

→ Carry generat

$\boxed{G_i = A_i . B_i}$



So, $\boxed{S_i = P_i \oplus C_{i-1}}$

$\boxed{C_i = G_i + P_i \cdot C_{i-1}}$

where, $P_i = A_i \oplus B_i$, $G_i = A_i \cdot B_i$

$C_i = G_i + P_i \cdot C_{i-1}$

$\therefore C_1 = G_1 + P_1 . C_0$

$C_2 = G_2 + P_2 \cdot C_1$

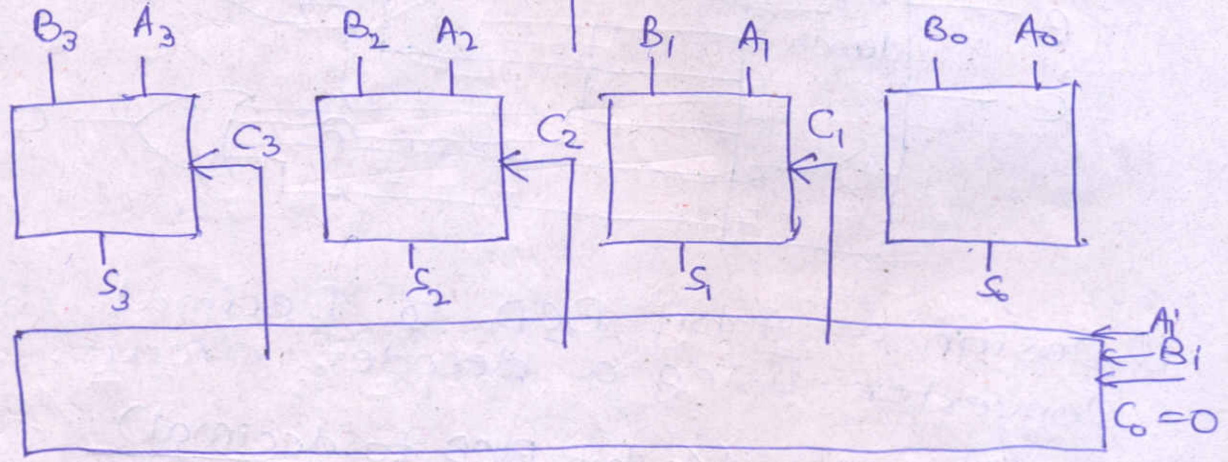
$C_3 = G_3 + P_3 \cdot C_2$

$\vdots$

$S_i = P_i \oplus C_{i-1}$

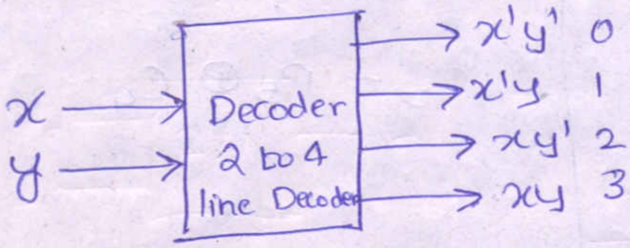
$S_5 = P_5 \oplus C_4$

$\vdots$

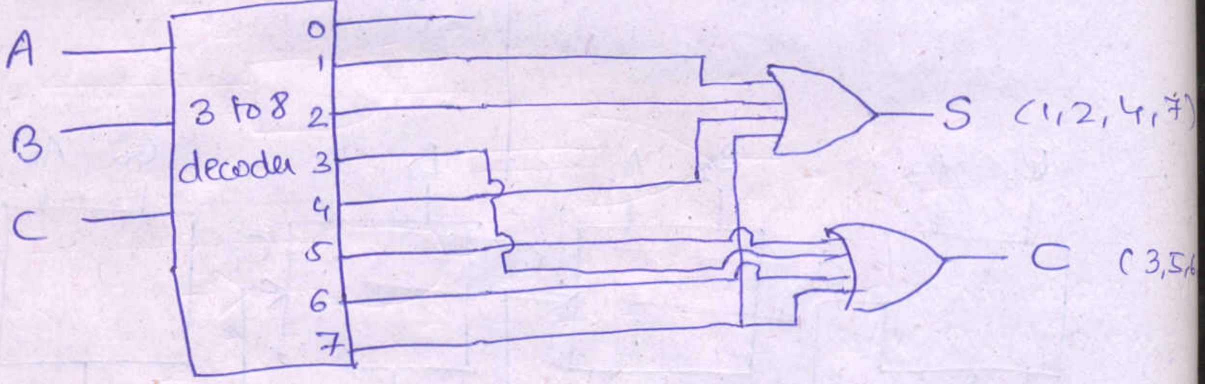


Decoder :-

It is a combinational circuit that converts n-bit binary information to 2^n no. of different combinations. Usually denoted as n to m line decoder, where m is less than equal to 2^n, i.e, $m \le 2^n$



Q. Design a full adder circuit using decoder



Q. Design a 4-bit BCD to Decimal Code Converter using a decoder circuit.

8/08/08

Decoder (4-bit BCD to decimal)

w	x	y	z	D_0	D_1	D_2	D_3	D_4	------	D_9
0	0	0	0	1	0	0	0	0	------	0
0	0	0	1	0	1	0	0	0	------	0
0	0	1	0	0	0	1	0	0	------	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	X						
1	0	1	1	X						

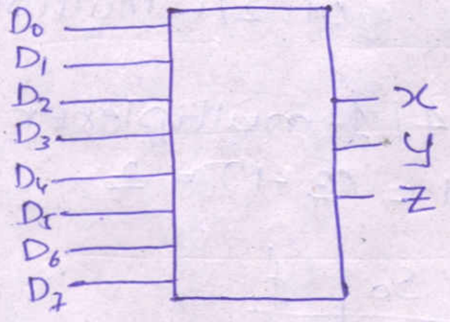
ENCODER :- 2^n to n-line

Q. Implement octal to binary conversion using an encoder.

Soln

Input line = 0 to 7 = 8 lines.

D_0	D_1	D_2	D_3	D_4	....	D_7	x	y	z
1	0	0	0	0		0	0	0	0
0	1	0	0	0		0	0	0	1
0	0	1	0	0		0	0	1	0
0	0	0	1	0		0	0	1	1
0	0	0	0	1		0	1	0	0
⋮	⋮	⋮	⋮	⋮		⋮	⋮	⋮	⋮
0	0	0	0	0		1	1	1	1



$x = D_4 + D_5 + D_6 + D_7$

$y = D_2 + D_3 + D_6 + D_7$

$z = D_1 + D_3 + D_5 + D_7$

Multiplexer :- It is a combinatorial circuit that selects a single input from all the possible input or output.

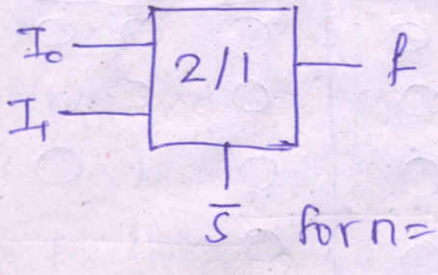
Denoted as $2^{n-1}/1$

Where 'n' is no. of variables,

no. of selection lines (S) is always (n-

If $S=0, f=I_0$

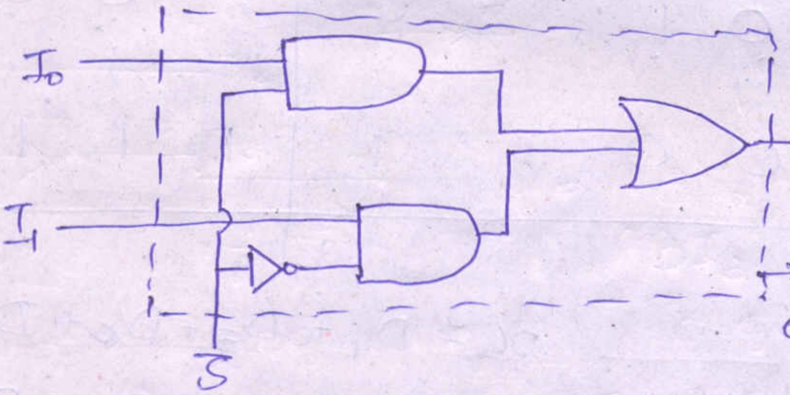
$S=1, f=I_1$



for n=2

S	f
0	I_0
1	I_1

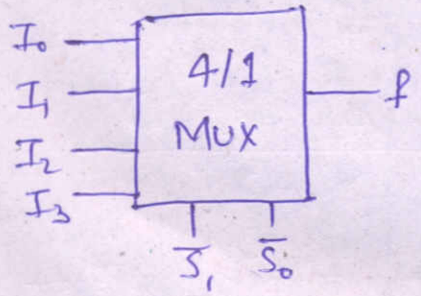
$\Rightarrow f = \bar{S}I_0 + SI_1$



→ Internal diagram of 2/1 multiplexer.

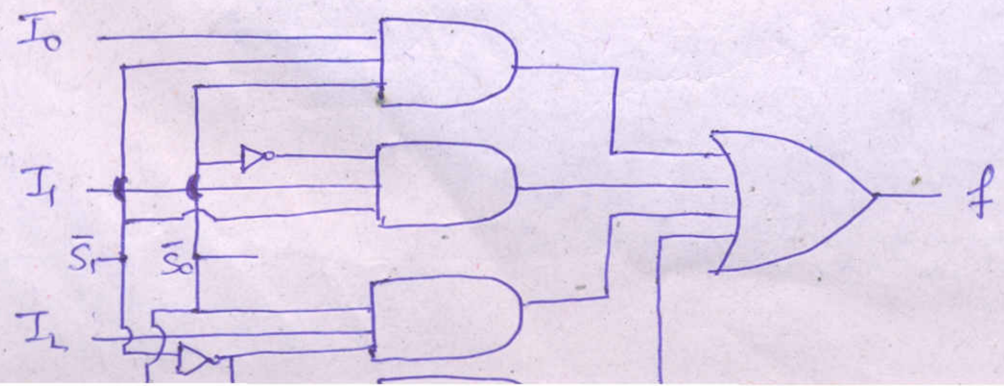
for $n=3 \Rightarrow 2^{n-1}/1 \Rightarrow 4/1$ multiplexer

Selection lines $\Rightarrow n-1 = (3-1) = 2$



S_1	S_0	f
0	0	I_0
0	1	I_1
1	0	I_2
1	1	I_3

$\therefore f = \bar{S}_1\bar{S}_0 I_0 + \bar{S}_1 S_0 I_1 + S_1\bar{S}_0 I_2 +$
$S_1 S_0 I_3$



circuit that selects a single input from all the possible input or output.

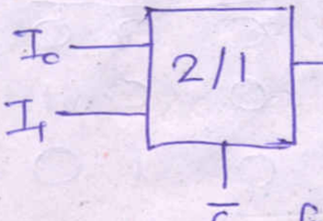
Denoted as $2^{n-1}/1$

Where 'n' is no. of variables,

no. of selection lines (S) is always (n-

If $S=0, f=I_0$

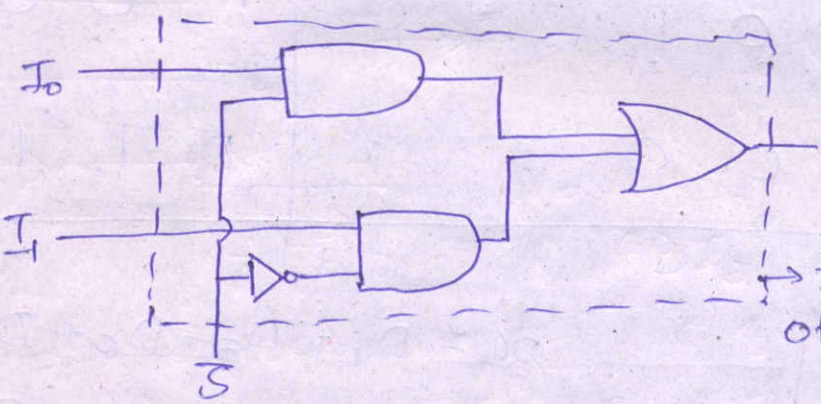
$S=1, f=I_1$



for n=2

S	f
0	I_0
1	I_1

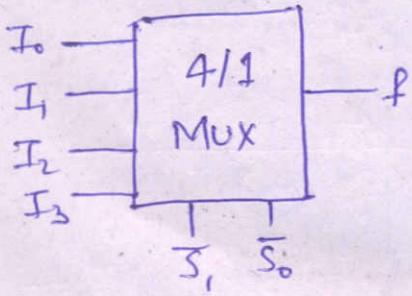
$\Rightarrow f=\bar{S}I_0+SI_1$



Internal diagram of 2/1 multiplexer.

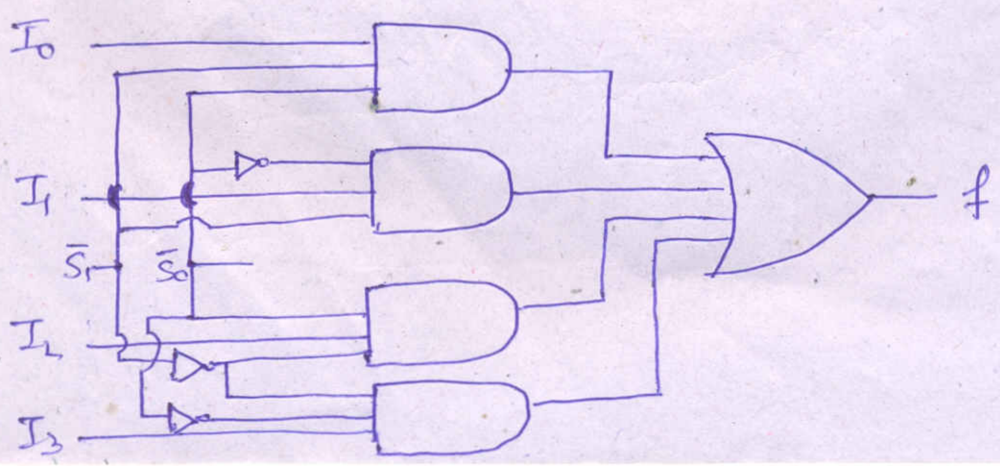
for $n=3 \Rightarrow 2^{n-1}/1 \Rightarrow 4/1$ multiplexer

selection lines $\Rightarrow n-1=(3-1)=2$



S_1	S_0	f
0	0	I_0
0	1	I_1
1	0	I_2
1	1	I_3

$\therefore f=\bar{S}_1\bar{S}_0I_0+\bar{S}_1S_0I_1+S_1\bar{S}_0I_2+$
$S_1S_0I_3$



Q. Implement the following function using multiplexer. Select B and C as selection line.

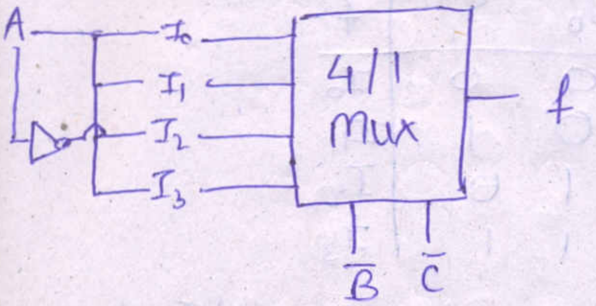
$$f = \underbrace{\bar{A}B\bar{C} + A\bar{B}C + A\bar{B}\bar{C} + ABC}_{4\ \text{inputs}}$$

<u>Soln</u>:

No. of variables, $n = 3$

Multiplexer will be $2^{3-1}/1 = 4/1$

No. of selection lines $= (3-1) = 2\ (B, C)$



Truth Table

B	C	f
0	0	I_0
0	1	I_1
1	0	I_2
1	1	I_3

$$f = \bar{B}\bar{C}I_0 + \bar{B}CI_1 + B\bar{C}I_2 + BCI_3$$

Comparing with $f = \bar{A}B\bar{C} + A\bar{B}C + A\bar{B}\bar{C} + ABC$

$I_0 = A,\ I_1 = A,\ I_2 = \bar{A},\ I_3 = A$

Q. Solve above problem by selecting A and C as the control line.

Q. Implement the function $f = XY\bar{Z} + \bar{X}\bar{Y}$. Using a multiplexer select Y and Z as selection line $(0, 0, x, 0)$.

Q. $f(A, B, C) = \Sigma(0, 2, 4, 6)$, A, B as control line.

<u>Note</u> :- 0 also implied by ground, '1' denoted by Vcc.

12/8/08

Q. Implement the function using multiplexer and 'A' 'B' as control line.

$F(A,B,C) = \Sigma(1,3,5,6)$

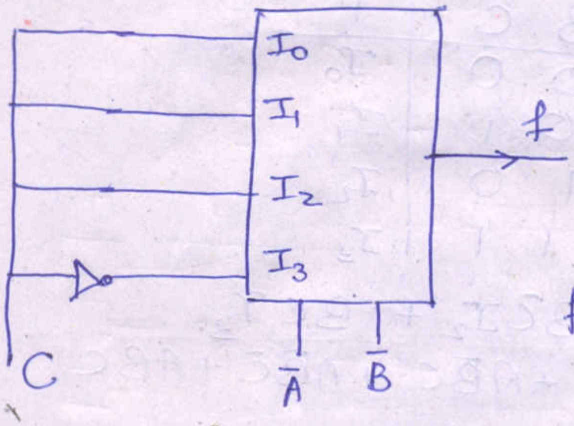
Soln

$F(A,B,C) = \Sigma(1,3,5,6)$

$= A'B'C + A'BC + AB'C + ABC'$ —(i)

$n = 3$, $n-1 = 2$ = select line

Size of multiplexer $= 2^{3-1}/1 = 4/1$.



S_0	S_1	f
0	0	I_0
0	1	I_1
1	0	I_2
1	1	I_3

$f = A'B'I_0 + A'BI_1 + AB'I_2 + ABI_3$.

Comparing with (i) we get,

$I_0 = I_1 = I_2 = C$, $I_3 = C'$

	I_0	I_1	I_2	I_3
$\bar{C}$	0	2	4	$\boxed{6}$
C	$\boxed{1}$	$\boxed{3}$	$\boxed{5}$	7
	C	C	C	C'

terms present in (i).

	A	B	C
0	0	0	0
1	0	0	1
2	0	1	0
3	0	1	1
4	1	0	0
5	1	0	1
6	1	1	0
7	1	1	1

→ If both present then '1'.

→ If none present then '0'

→ If anyone present then that one (either x or $\bar{x}$)

Q. Implement function f as $F(A,B,C,D) = \Sigma(2,4,6,8,10,11,12,14)$ using a 4/1 multiplexer. Consider A, C are the select lines.

Soln

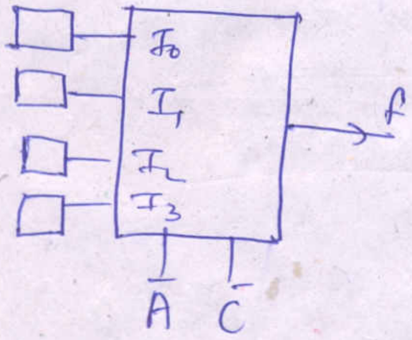
$n = 4$, Multiplexer = 8/1 Mux

But we are given 4/1 MUX.

$F(A,B,C,D) = \cancel{A'B'C'D + A'B'C}$

$= A'B'CD' + A'BC'D' + A'BCD' + AB'C'D' + AB'CD' + AB'CD + ABC'D' + ABCD'$

	A	B	C	D
0	0	0	0	0
1	0	0	0	1
2	0	0	1	0
3	0	0	1	1
4	0	1	0	0
5	0	1	0	1
6	0	1	1	0
7	0	1	1	1
8	1	0	0	0
9	1	0	0	1
10	1	0	1	0
11	1	0	1	1
12	1	1	0	0
13	1	1	0	1
14	1	1	1	0
15	1	1	1	1

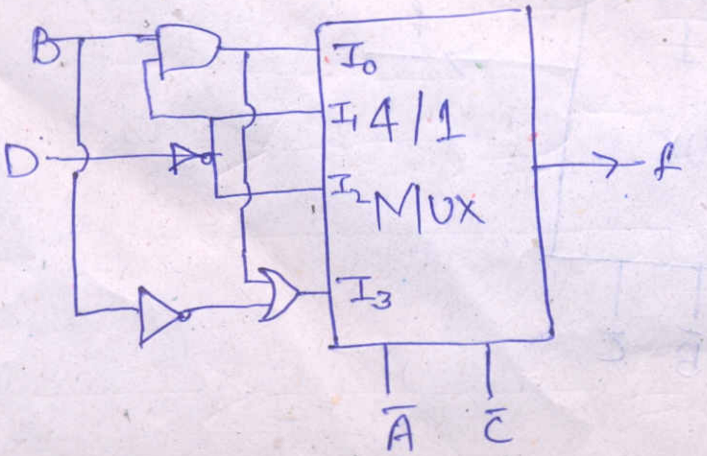


A	C	f
0	0	I_0
0	1	I_1
1	0	I_2
1	1	I_3

$f = A'C'I_0 + A'CI_1 + AC'I_2 + ACI_3$

Comparing,

$I_0 = BD'$, $I_1 = D'$, $I_2 = D'$, $I_3 = B' + BD'$



Q. Design a combinational circuit that generates the 9's complement of a 4-bit BCD number.

Soln

	a	b	c	d	w	x	y	z
	~~w~~	~~x~~	~~y~~	~~z~~	~~a~~	~~b~~	~~c~~	~~d~~
0	0	0	0	0	1	0	0	1
1	0	0	0	1	1	0	0	0
2	0	0	1	0	0	1	1	1
3	0	0	1	1	0	1	1	0
4	0	1	0	0	0	1	0	1
5	0	1	0	1	0	1	0	0
6	0	1	1	0	0	0	1	1
7	0	1	1	1	0	0	1	0
8	1	0	0	0	0	0	0	1
9	1	0	0	1	0	0	0	0
	x	x	x	x	x	x	x	x

(w x y z: 9's complement)

Simplify by K-map,

		d	d	
	1 (0)	1 (1)	0 (3)	0 (2)
b	0 (4)	0 (5)	0 (7)	0 (6)
a, b	x (12)	x (13)	x (15)	x (14)
a	0 (8)	0 (9)	x (11)	x (10)
			c	c

$w = b'c'a'$

		d	d	
	0 (0)	0 (1)	1 (3)	1 (2)
b	1 (4)	1 (5)	0 (7)	0 (6)
a, b	x (12)	x (13)	x (15)	x (14)
a	0 (8)	0 (9)	x (11)	x (10)
			c	c

$x = bc' + b'c$

$y = c$

0	0	1	1
0	0	1	1
X	X	X	X
0	0	X	X

(d over columns 3–4; b beside rows 2–3; a beside rows 3–4; c under columns 3–4)

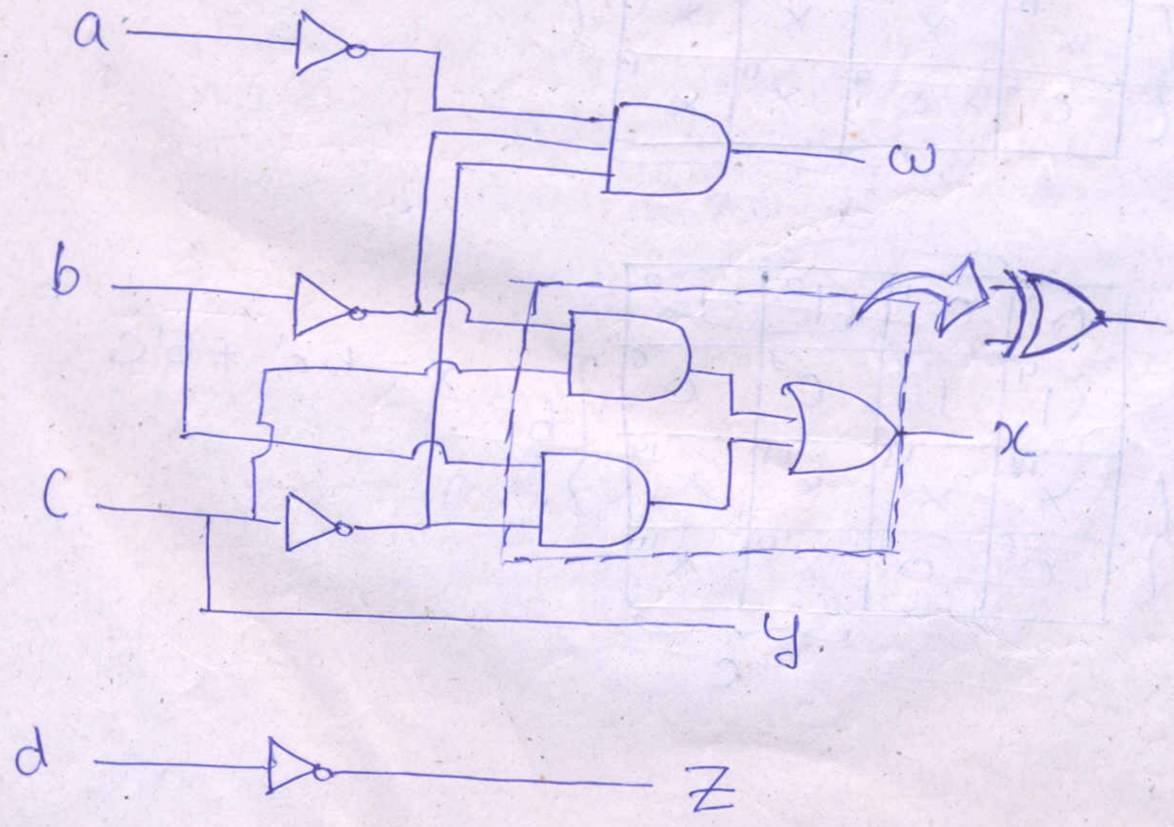
$z = d'$

1	0	0	1
1	0	0	1
X	X	X	X
1	0	X	X

(d over columns 2–3; b beside rows 2–3; a beside rows 3–4; c under columns 3–4)

$\omega = a'b'c'$, $x = bc' + b'c$, $y = c$, $z = d'$

$= b \oplus c$



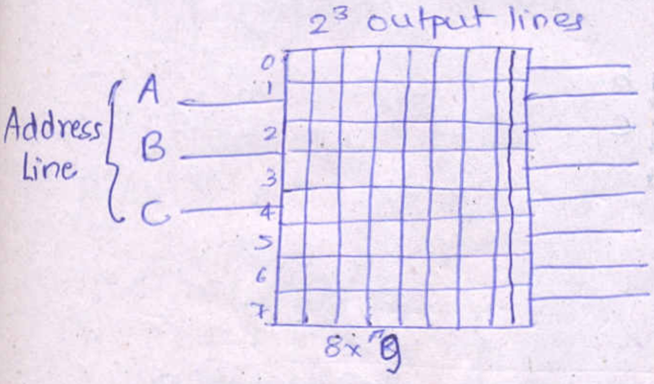
14/08

Read Only Memory (ROM)

Storage
- Primary
  - RAM
  - ROM (stores the different system information) created by manufacturer
- Secondary

Memory Addressing :-

ROM is manufactured with the concept of Decoder circuit.



	A	B	C
0	0	0	0
1	0	0	1
2	0	1	0
3	0	1	1
4	1	0	0
5	1	0	1
6	1	1	0
7	1	1	1

Here row analogy of decoder is specified by the row of inputs taken. As A=0, B=0, C=0 will be 0th row of decoder and so on.

⟹ This particular combination of A, B and C specifying ~~an address~~ a row is called Row address.

⟹ Hence, A, B and C are Address Lines.

⟹ Word :- It is a group of binary information. It depends on machine configuration.

Word length is also no. of columns in decoder.

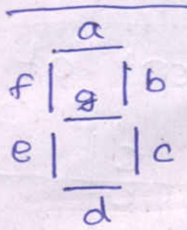
In above figure, word length = 9 bits.

Size of Row = No. of rows × word length
= 8×9 bits = 72 bits

A 32-bit system implies amount of data or row-addresses fetched at a time, here being 3

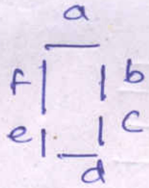
In order to read one row completely, we require output lines equal to the word length from that row.

Seven Segment Display :-

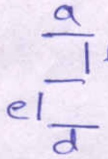


→ in calculators, platforms, digital watches etc.

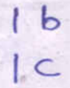
⇒ To display '0'



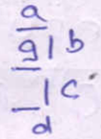
⇒ To display '2'



⇒ To display '1'



⇒ To display '3'



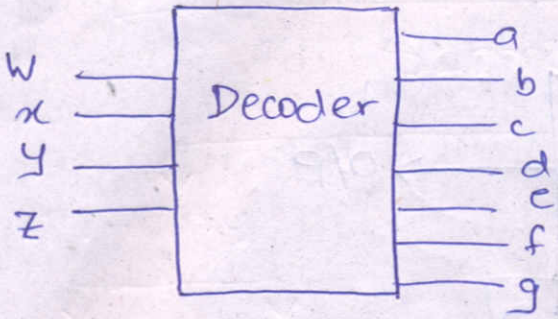
Q. Design a combinational decoder circuit that will decode the input BCD number to seven segment display.

Soln

Inputs - 4

Outputs - 7 (a, b, c, d, e, f, g)

	w	x	y	z
0	0	0	0	0
1	0	0	0	1
2	0	0	1	0
3	0	0	1	1
4	0	1	0	0
5	0	1	0	1
6	0	1	1	0
7	0	1	1	1



	a	b	c	d	e	f	g
0	1	1	1	1	1	1	0
1	0	1	1	0	0	0	0
2	1	1	0	1	1	0	0
3	1	1	1	1	0	0	1
4	0	0	1	0	0	1	1
5	1	0	1	1	0	1	1
6	1	0	1	1	1	1	1
7	1	1	1	0	0	0	0
8	1	1	1	1	1	1	1
9	1	1	1	1	0	1	1

$a = \Sigma(0, 2, 3, 5, 6, 7, 8, 9)$

		z	z	
	1^{0}	0^{1}	1^{3}	1^{2}
x	0^{4}	1^{5}	1^{7}	1^{6}
w, x	X^{12}	X^{13}	X^{15}	X^{14}
w	1^{8}	1^{9}	X^{11}	X^{10}
			y	y

$a = y + w\cancel{y'} + w'x'y'z' + w'xy'z$

$b = \Sigma(0, 1, 2, 3, 7, 8, 9)$

		z	z	
	1	1	1	1
x	0	0	1	0
w, x	X	X	X	X
w	1	1	X	X
			y	y

$b = w + w'x' + w'xyz$

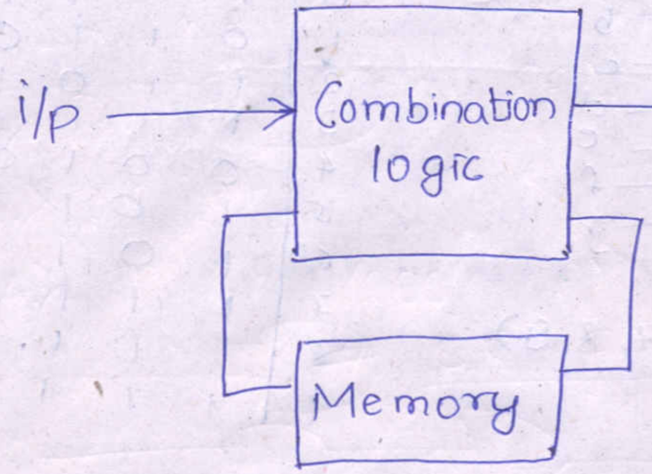
or $b = w + w'(x' + xyz)$

or, $b = w + w'(1 + (x' + yz))$

or, $b = 1 + w'(x' + yz)$

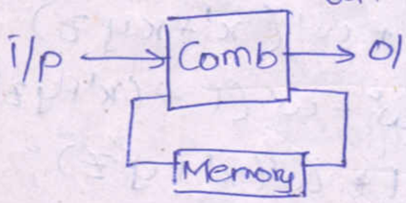
18/8/08

Sequential Circuit :-

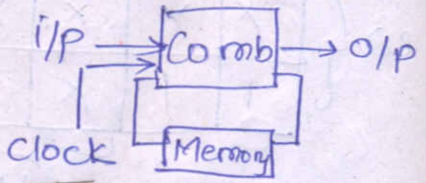


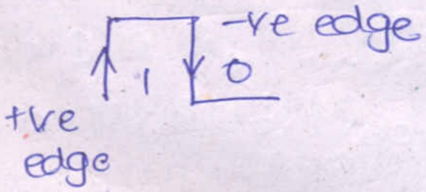
Sequential Circuit

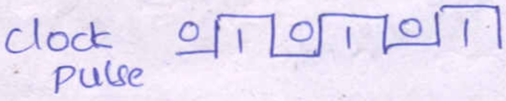
Asynchronous sequential circuit
(output changes depending upon nature of inputs)
Each module gets different clock pulse.



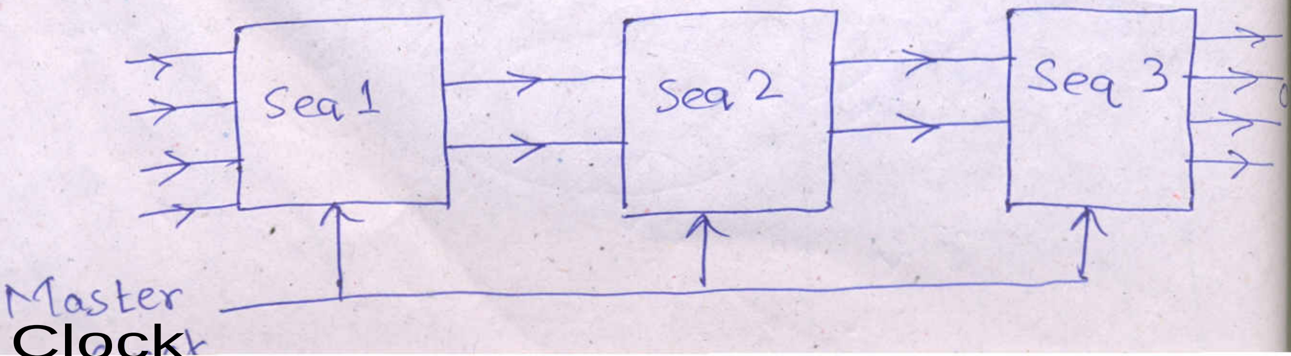
Synchronous sequential circ
(Along with i/p, and input is taken whi is clock pulse which decides how much time a particular o/p remains there)







Output changes as '1' comes and remains till again '1' comes (i.e. in ne positive half cycle)

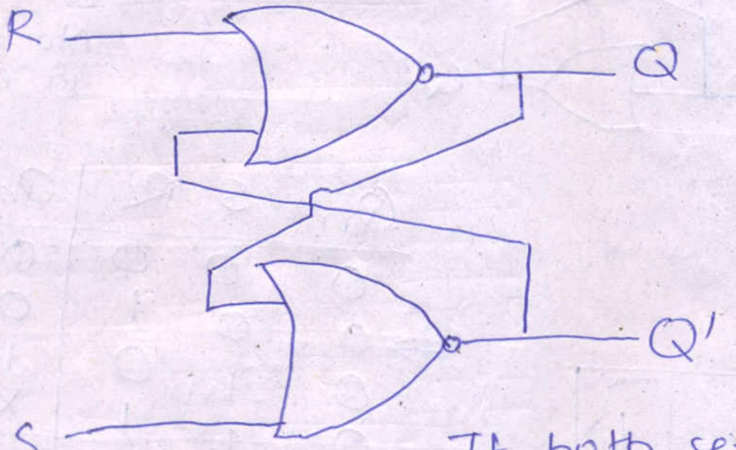


The blocks might process with different way. So, a clock called master clock is used to synchronise the pulses. But, if they have different clock speed, then they are asynchronous in nature.

FLIP-FLOPS :-

It is a sequential circuit that exhibits two stable states, one is set and the other is reset.

Verify common diagram FF -



S	R	Q	Q'
0	1	0	1
0	0	0	1
1	0	1	0
0	0	1	0
1	1	0	0

If both set (S) and reset (R) are 0, and 0, then previous value will be copied.

If ~~S=0~~, S = 1, R = 1, we get Q = 0, Q' = 0 which is not possible and hence doesn't exist.

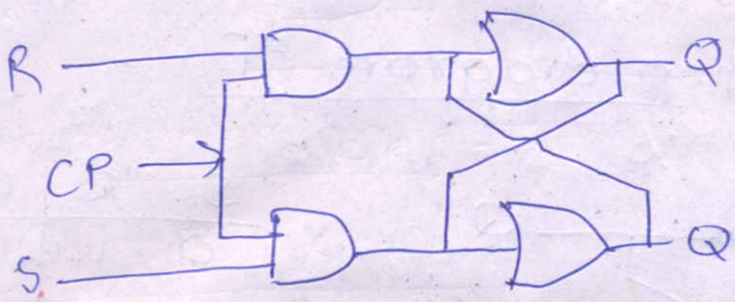
S	R	Q_{n+1}
0	0	Q_n
0	1	0
1	0	1
1	1	→

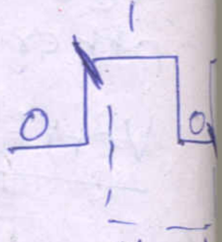
Q_n	S	R	Q_{n+1}
0	0	0	0
0	0	1	0
0	1	0	1
0	1	1	→
1	0	0	1
1	0	1	0
1	1	0	1
1	1	1	→

19/8/0 There are four types of flip-flop :-

1> SR FF (Set, Reset ff)
2> D FF (Delay ff)
3> JK FF
4> T FF (Toggle ff)

SR-FF





Q_n	S	R	Q_{n+1}
0	0	0	0
0	0	1	0
0	1	0	1
0	1	1	X
1	0	0	1
1	0	1	0
1	1	0	1
1	1	1	X

By K-map.

		X	1
1		X	1

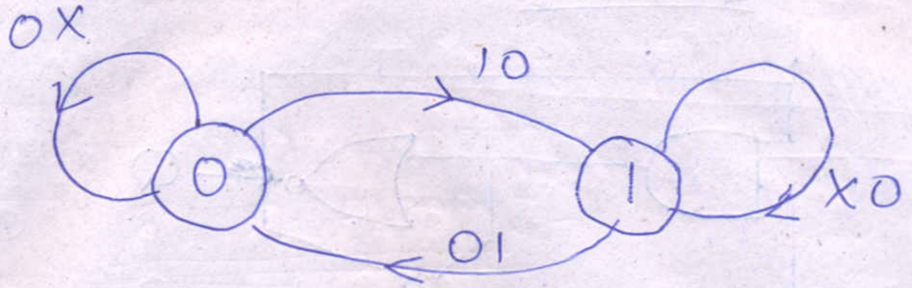
$Q_{n+1} = S + Q_n R'$ => characteristic equation

Excitation Table

Q_n	Q_{n+1}	S	R	
0	0	0	0	} 0 X
		0	1	
0	1	1	0	
1	0	0	1	
1	1	0	1	} X 0
		1	0	

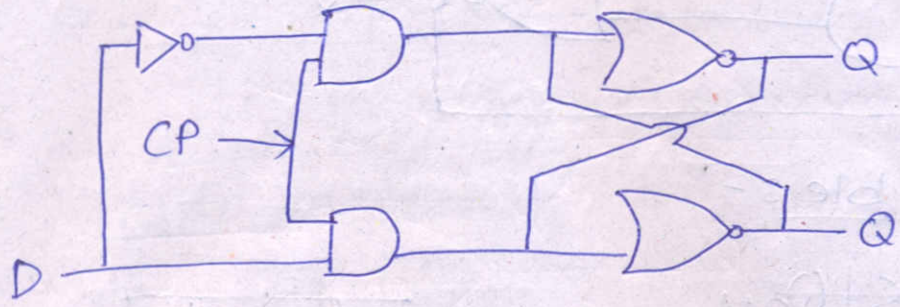
Note :- Res
is als
called
'Clear

State Diagram :-



Q = 0 ⇒ Reset state , Q = 1 , set state.

Delay FlipFlop : (DFF)



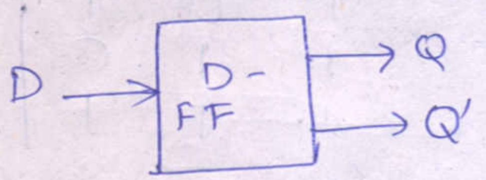
⇒ Whatever is the input, that is the output.

or, $Q_{n+1} = D$

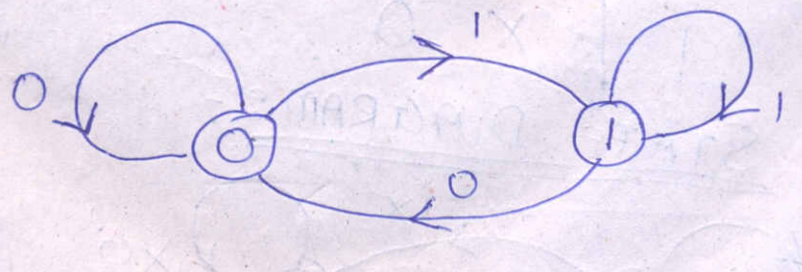
Q_n	D	Q_{n+1}
0	0	0
0	1	1
1	0	0
1	1	1

Excitation Table :-

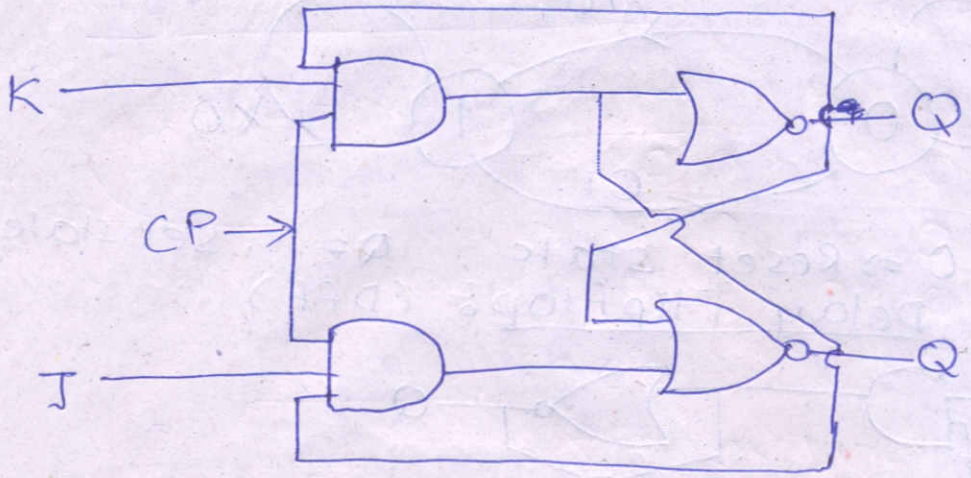
Q_n	Q_{n+1}	D
0	0	0
0	1	1
1	0	0
1	1	1



State Diagram :-



JK-FF



Truth Table:-

Q_n	J	K	Q_{n+1}
0	0	0	0
0	0	1	0
0	1	0	1
0	1	1	1
1	0	0	1
1	0	1	0
1	1	0	1
1	1	1	0

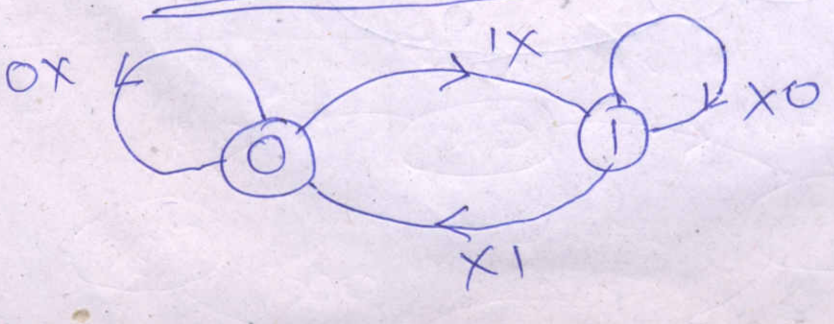
		1	1
1			1

$$\boxed{Q_{n+1} = JQ_n' + K'Q_n}$$

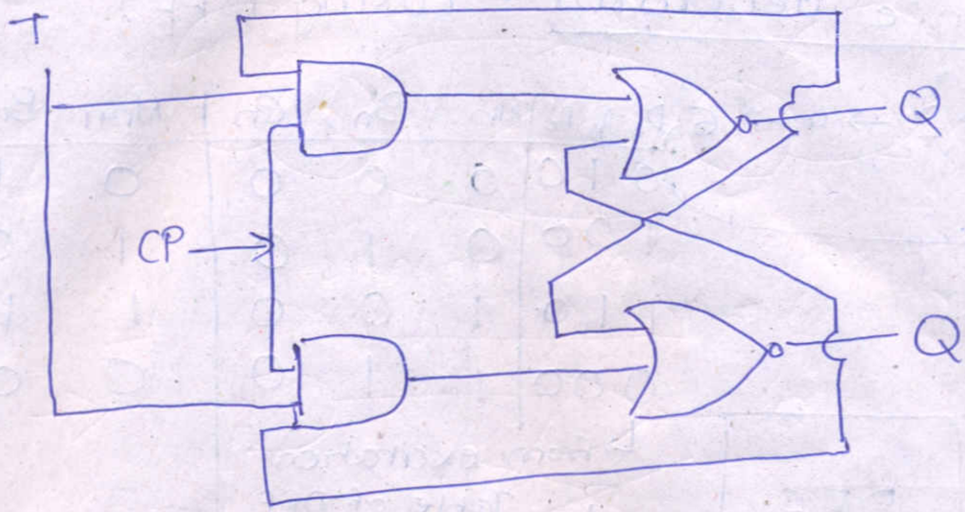
Excitation Table:-

Q_n	Q_{n+1}	J	K
0	0	0	X
0	1	1	X
1	0	X	1
1	1	X	0

STATE DIAGRAM:-



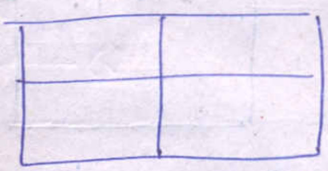
T-FF



Truth Table :-

Q_n	T	Q_{n+1}
0	0	0
0	1	1
1	0	1
1	1	0

$\Rightarrow$ same inputs give zero, else one.

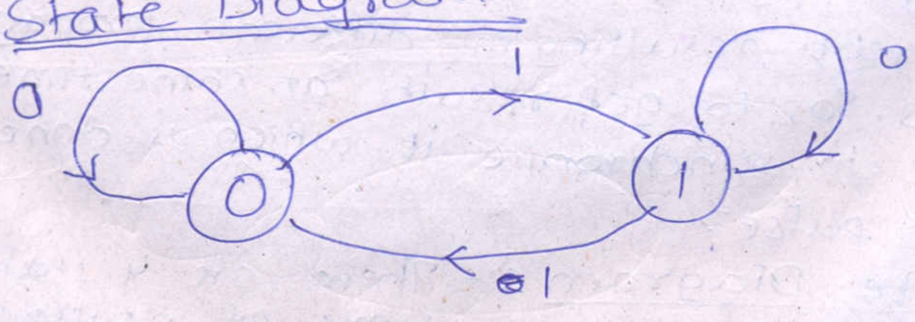


$$\boxed{Q_{n+1} = Q_n'T + Q_nT'}$$

Excitation Table -

Q_n	Q_{n+1}	T
0	0	0
0	1	1
1	0	1
1	1	0

State Diagram -



22/8/

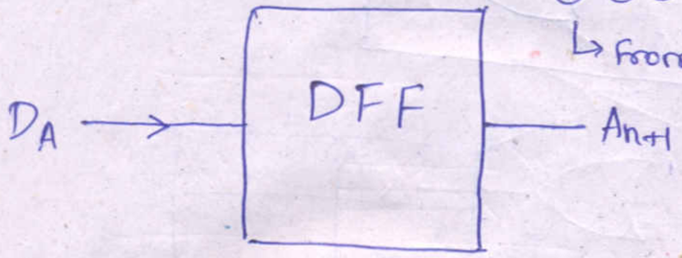
Some applications of FF :-

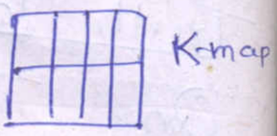
1) Sequence Generator - (Using DFF)

$0 \rightarrow 2 \rightarrow 4 \rightarrow 6$

D_A	D_B	D_C	A_n	B_n	C_n	A_{n+1}	B_{n+1}
0	1	0	0	0	0	0	1
1	0	0	0	1	0	1	0
1	1	0	1	0	0	1	1
0	0	0	1	1	0	0	0

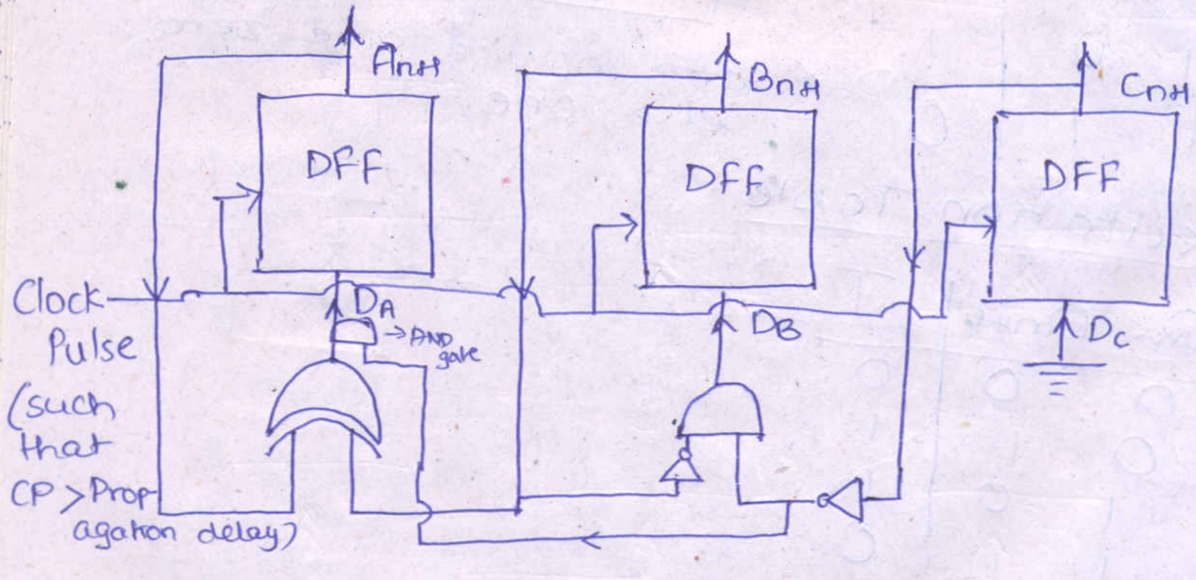
↳ from excitation Table of DFF





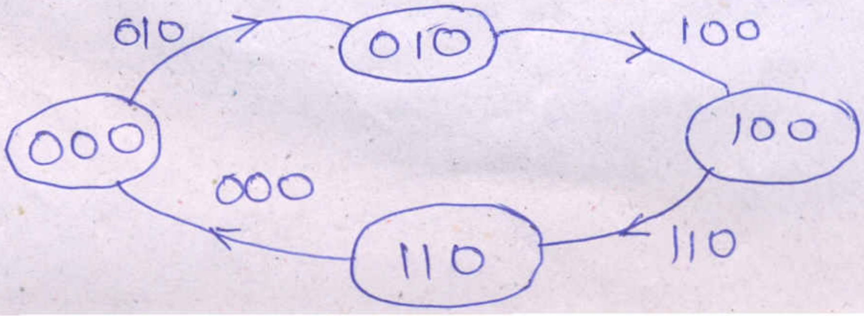
Simplifying by K-map,

$$D_A = (A_n \oplus B_n)C_n', \quad D_B = B_n' C_n' \quad D_C = 0$$



Though FF's same, but combinational logic are different resulting in different propagation times. So, to get results at same time, we have to synchronise it, which is done by clock pulse.

State Diagram :- There are 4 states as there are 4 combinations of inputs.



Q. Generate a sequence - 1 → 6 → 5 → 4 → 3 → 2 using J-K Flipflop.

Solution

A_n	B_n	C_n	A_{n+1}	B_{n+1}	C_{n+1}	J_A	K_A	J_B	K_B	J_C	K_C
0	0	1	1	1	0	1	X	1	X	X	1
1	1	0	1	0	1	X	0	X	1	1	X
1	0	1	1	0	0	X	0	0	X	X	1
1	0	0	0	1	1	X	1	1	X	1	X
0	1	1	0	1	0	0	X	X	0	X	1
0	1	0	0	0	1	0	X	X	1	1	X

Simplify by K-map,

J

C_n

	1	X	X	X
A_n	0	0	X	X

B_n

$J_A = A_n'$

CB_n

	1	X	1	0
A_n	X	X	X	X

B_n

$J_B = A_n'B_n' + A_n'C_n$

$= A_n'(B_n' + C_n)$

C_n

	X	1	1	X
A_n	X	1	X	X

B_n

$J_c = C_n$

K

C_n

	X	0	1	0
A_n	X	X	X	X

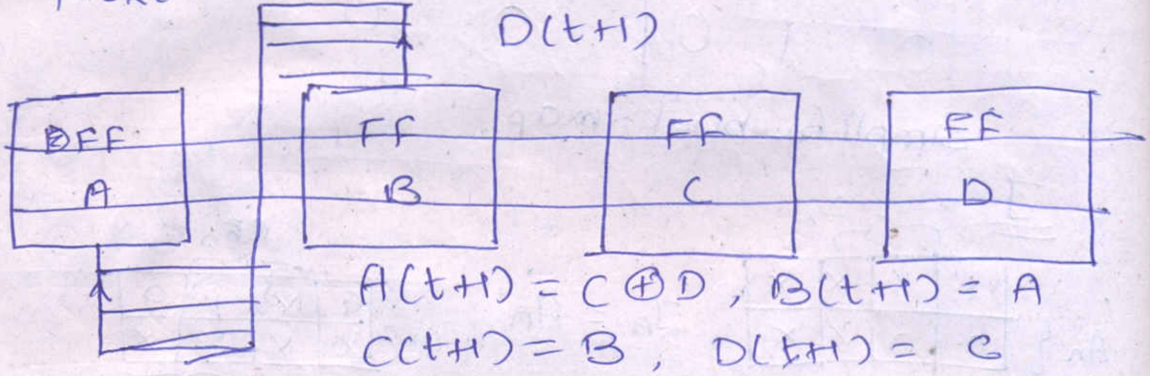
B_n

26/08/08

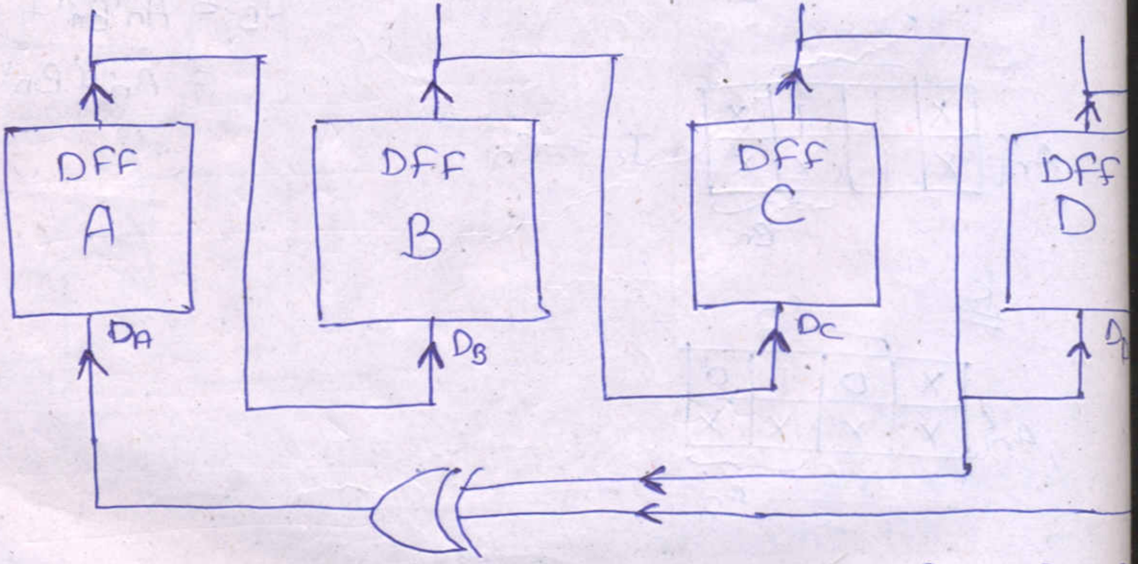
Q1) Design a sequential circuit with 4 Flipflops A, B, C, D. The next step of B, C and D are equal to the present state of A, B and C respectively. The next state of A is equal to the exclusive OR (XOR) of the present state of C and D.

Solution

let present states A, B, C, D.
Next states are A(t+1), B(t+1), C(t+1) and D(t+1)



$A(t+1) = C \oplus D$, $B(t+1) = A$
$C(t+1) = B$, $D(t+1) = C$

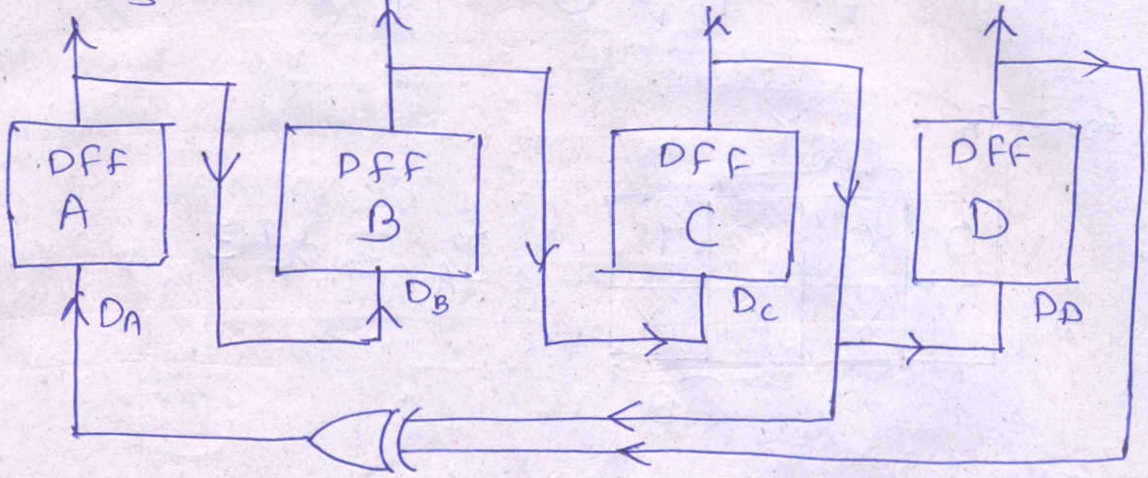


A	B	C	D	A_{n+1}	B_{n+1}	C_{n+1}	D_{n+1}	D_A	D_B	D_C	D
0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	1	1	0	0	0	1	0	0	0
0	0	1	0	1	0	0	1	1	0	0	1
0	0	1	1	0	0	0	1	0	0	0	1
0	1	0	0	0	0	1	0	0	0	1	0
0	1	0	1	1	0	1	0	1	0	1	0
0	1	1	0	1	0	1	1	1	0	1	1
0	1	1	1	0	0	1	1	0	0	1	1

A	B	C	D	A_{n+1}	B_{n+1}	C_{n+1}	D_{n+1}	D_A	D_B	D_C	D_D
1	0	0	0	0	1	0	0	0	1	0	0
1	0	0	1	1	1	0	0	1	1	0	0
1	0	1	0	1	1	0	1	1	1	0	1
1	0	1	1	0	1	0	1	0	1	0	1
1	1	0	0	0	1	1	0	0	1	1	0
1	1	0	1	1	1	1	0	1	1	1	0
1	1	1	0	1	1	1	1	1	1	1	1
1	1	1	1	0	1	1	1	0	1	1	1

So, $D_A = C \oplus D$, $D_C = B$

$D_B = A$ $D_D = C$



Q.2)

Personal state		Next state			
		x=0		x=1	
A	B	A	B	A	B
0	0	0	0	0	1
0	1	1	0	0	1
1	0	1	0	1	1
1	1	1	1	0	0

Design a sequential circuit with two J-K Flipflop A and B. Use x as an external input to the circuit. The present state and next state of the flipflop varies according to the table.

Solution

A	B	x	A(t+1)	B(t+1)	J_A	K_A	J_B	K_B
0	0	0	0	0	0	X	0	X
0	0	1	0	1	0	X	1	X
0	1	0	1	0	1	X	X	1
0	1	1	0	1	0	X	X	0
1	0	0	1	0	X	0	0	X
1	0	1	1	1	X	0	1	X
1	1	0	1	1	X	0	X	0
1	1	1	0	0	X	1	X	1

For J_A

	x	x		
	0	0	0	1
A	X	X	X	X
			B	B

$J_A = x'B$

For J_B,

	x	x		
	0	1	X	X
A	0	1	X	X
			B	B

$J_B = x$

For K_A

	x	x		
	X	X	X	X
A	0	0	1	0
			B	B

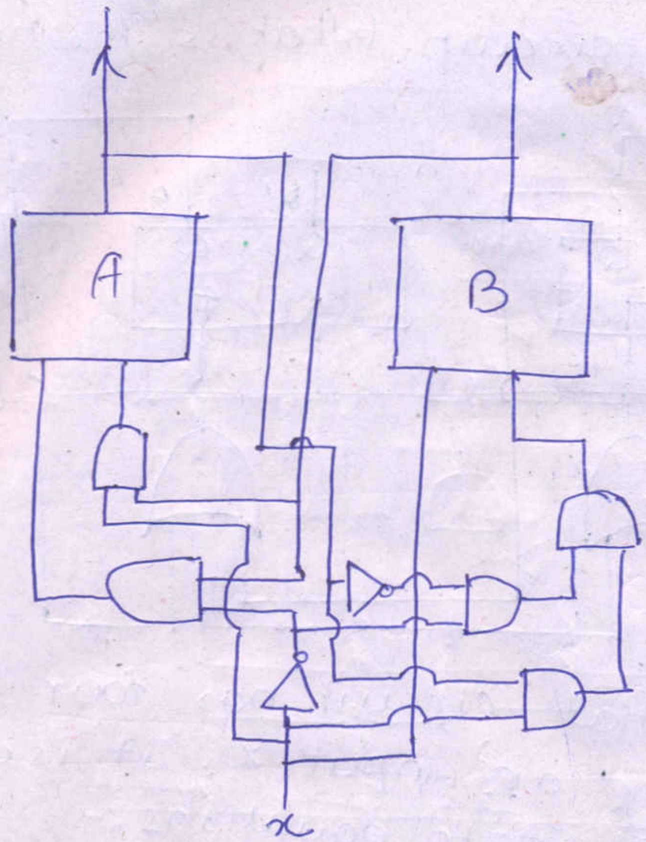
$K_A = B.x$

For K_B,

	x	x		
	X	X	0	1
A	X	X	0	0
			B	B

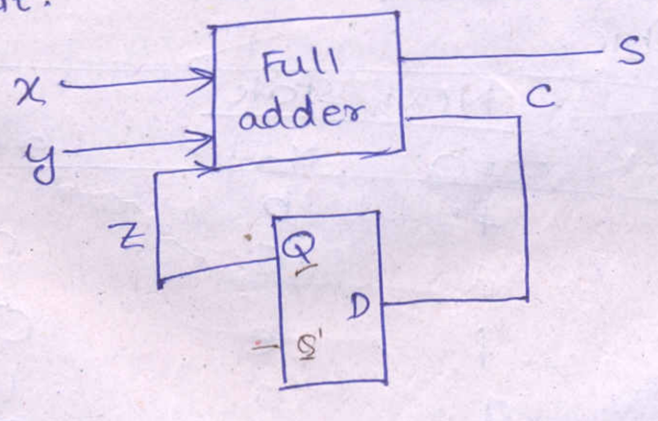
$K_B = \cancel{x'B \text{ or } B'+x'}$

$x'A' + Ax$

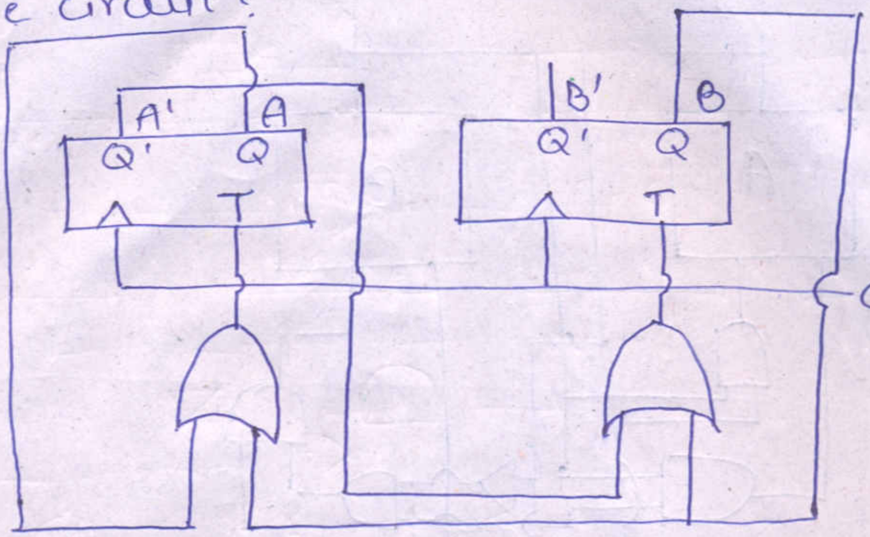


Assignment - Pg - 253 . Prob 6.10, 6.11, 6.12, 6.14, 6.27

6.10) The full-adder of fig. P6-10 receives two external inputs x and y; the third input z comes from the output of a D Flipflop. The carry output is transferred to the flipflop every clock pulse. The external S output gives the sum of x, y and z. Obtain state table and state diagram of sequential circuit.



6-11) Derive state table and state diagram of sequential circuit. What is the function of the circuit?



6-12) A sequential circuit has four flipflops A, B, C, D and an input x. It is described by following state equations:-

$A(t+1) = (CD' + C'D)x + (CD + C'D')x'$

$B(t+1) = A$

$C(t+1) = B$

$D(t+1) = C$

(a) Obtain sequence of states when $x=1$, starting from state $ABCD = 0001$.

(b) Obtain sequence of states when $x=0$, starting from $ABCD = 0000$.

6-14) Reduce the no. of states in the following state table and tabulate the reduced state table.

Present state	Next state $x=0$	Next state $x=1$	O/P $x=0$	O/P $x=1$
a	f	b	0	0
b	d	c	0	0
c	f	e	0	0
d	g	a	1	0
e	d	c	0	0
f	f	b	1	1
g	g	h	0	1
			1	0

6-27) Design a counter with the binary sequence: 0, 1, 3, 7, 6, ... Use T-flip flops.

Q. Design a sequential circuit with J-K FF Flipflop to satisfy the following equations :-

$A(t+1) = \bar{A}\bar{B}CD + \bar{A}\bar{B}C + ACD + A$ AC(bar)D(bar)

$B(t+1) = \bar{A}C + C\bar{D} + \bar{A}B\bar{C}$

$C(t+1) = B$

$D(t+1) = \bar{D}$

Solution $\Rightarrow Q_{t+1} = JQ' + K'Q$

A	B	C	D	A_{t+1}	B_{t+1}	C_{t+1}	D_{t+1}	$A_j A_k$	$B_j B_k$	$C_j C_k$	$D_j D_k$

$A_{t+1} = J_A A' + K_A' A$

By comparison, $J_A = B'CD + B'C$

$K_A' = CD + C'D' \Rightarrow K_A = C \oplus D$ $(\cancel{C \oplus D})'$

$B_{t+1} = J_B B' + K_B' B$

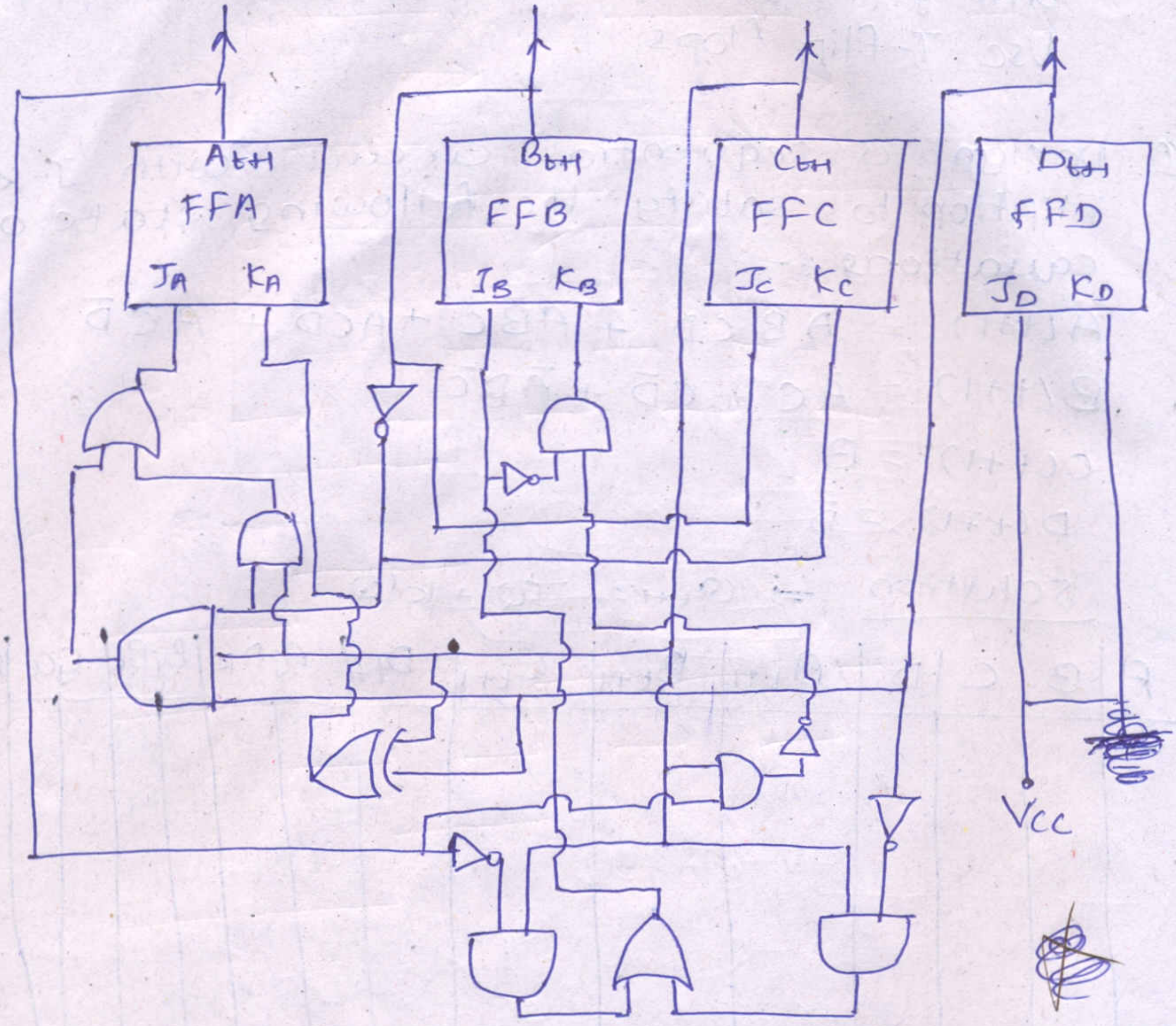
$J_B = A'C + CD'$, $K_B' = \bar{A}\bar{C} \Rightarrow K_B = (A+C)$. $(A+C)'$.

$C_{t+1} = J_C C' + K_C' C \Rightarrow J_C = B$, $K_C = B'$ $(C'+D)$

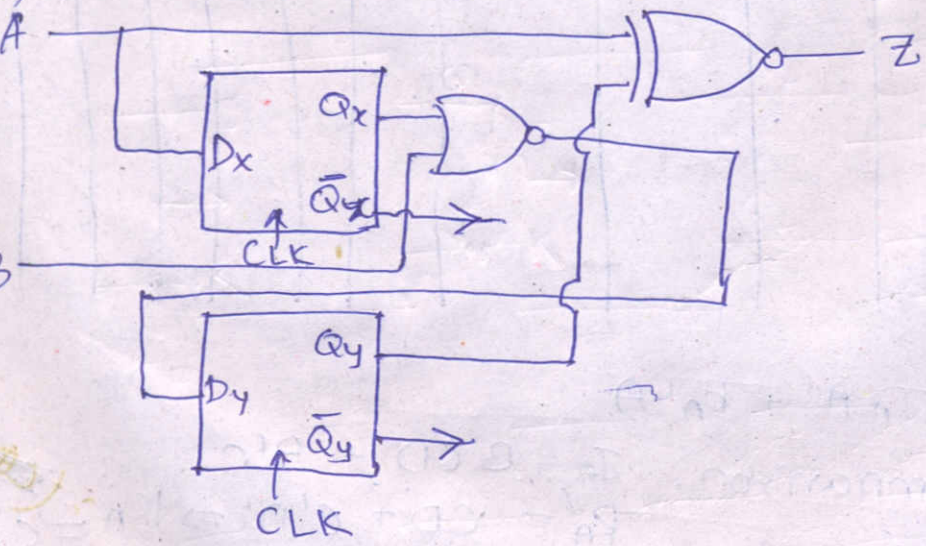
$D_{t+1} = J_D D' + K_D' D \Rightarrow J_D = 1$, $K_D' = 0$

Kd=1

28/8/08



Q) (i)



(ii) Write down the State Transition Diagram

(iii). Write Boolean equation for ckt node Z, D_x, D_y.

Solution -

(iii) $D_x = A$, $Z = (Q_y \oplus A)'$,

$D_y = (B + Q_x)'$

But, $D_x = Q_{x+1}$, $D_y = Q_{y+1}$

$D_x = A$, $D_y = (B + A)'$

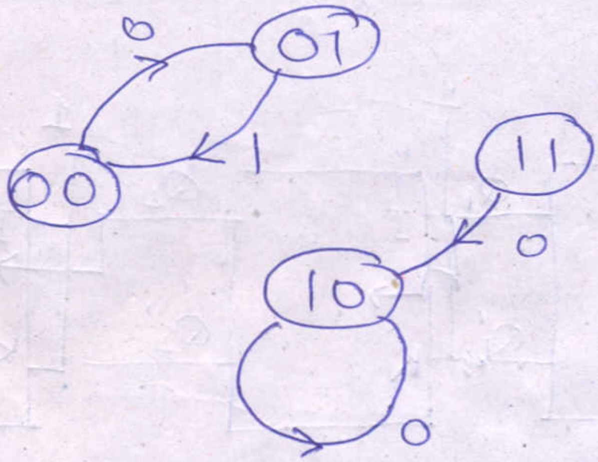
$D_y = A'B'$

$Z = (A'B' \oplus A)'$

(ii) State Transition Table :-

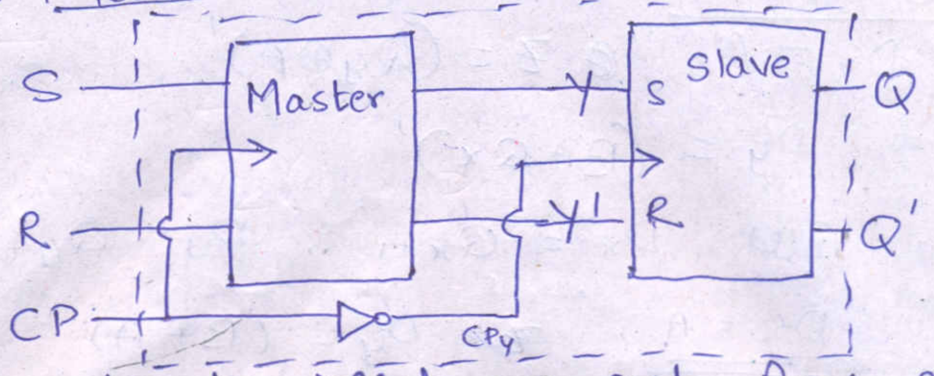
Present state A	Present state B	Next state Q_x	Next state Q_y	O/P Z
0	0	0	1	0
0	1	0	0	1
1	0	1	0	0
1	1	1	0	0

State Diagram -

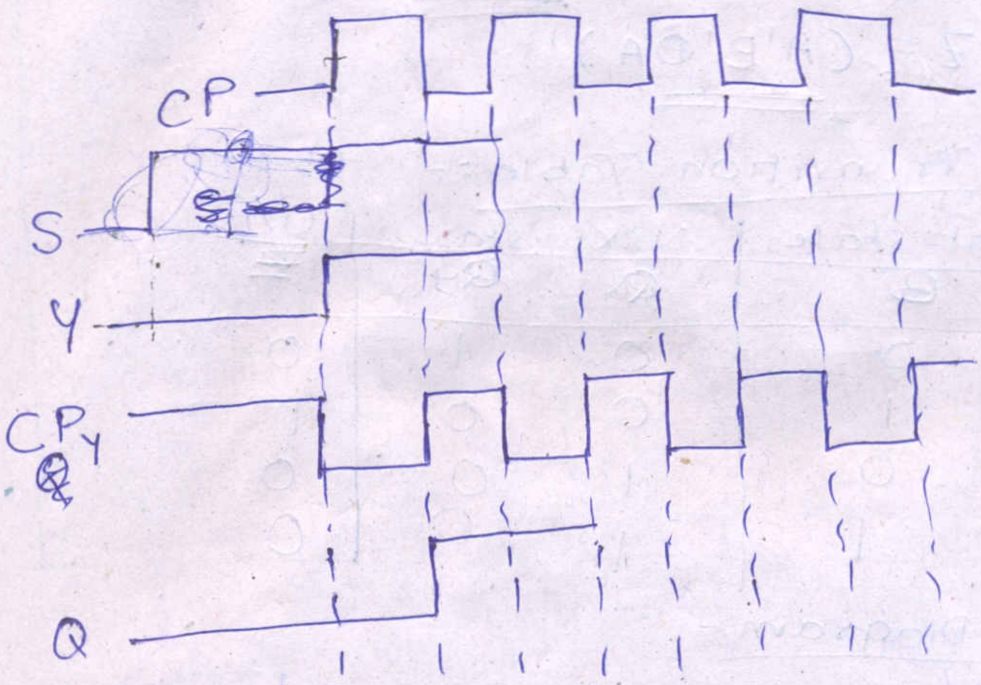


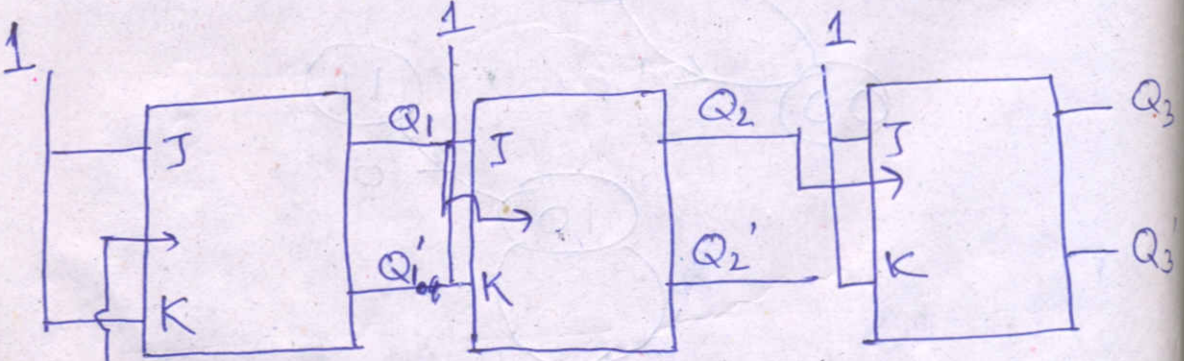
2/9/08

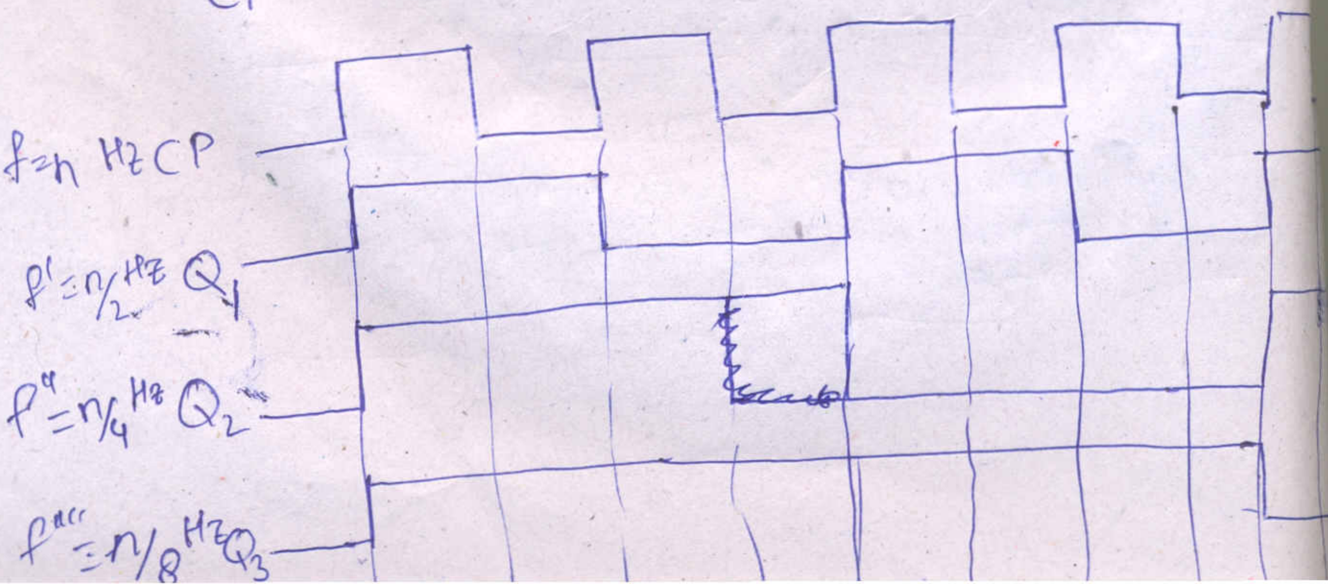
Master - Slave Flipflop :-



In order to reset we put, R=1, S=0







Q. Design a sequential ckt with two T-FF, A_1 & A_2 and one external input x ~~at~~ and output y.
The next step, an output information is obtained by :-

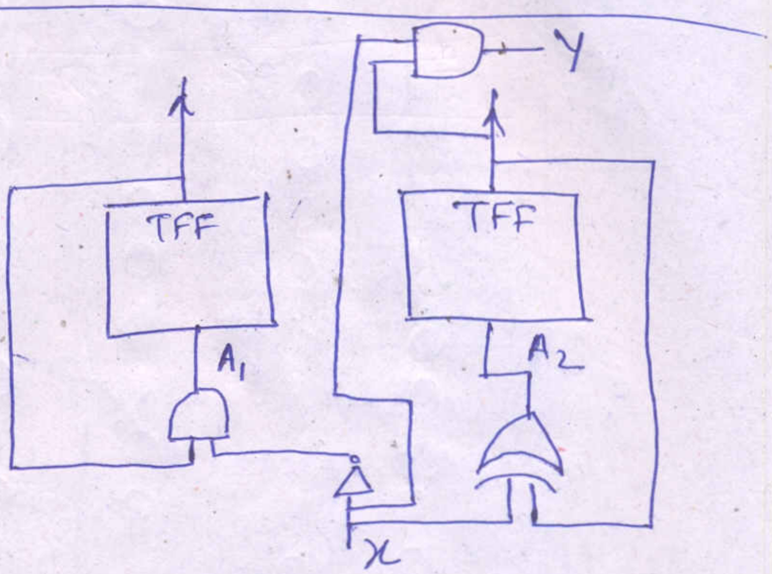
$A_1(t+1) = \Sigma(4,6)$

$A_2(t+1) = \Sigma(1,2,5,6)$

$Y(A_1, A_2, x) = \Sigma(3,7)$

Solution

A_1	A_2	x
0	0	0
0	0	1
0	1	0
0	1	1
1	0	0
1	0	1
1	1	0
1	1	1



A_1 (C over the middle columns, B over the right columns)

	0	0	0	0
A	1	0	0	1

$A_1 = Ax'$

A_2 =

0	0	1	1
		1	1

	0	1	0	1
A	0	1	0	1

B

$A_2 = Bx' + B'x = B \oplus x$

C

$Y =$

	0	0	1	0
A	0	0	1	0

B

$Y = Bx$

Soln

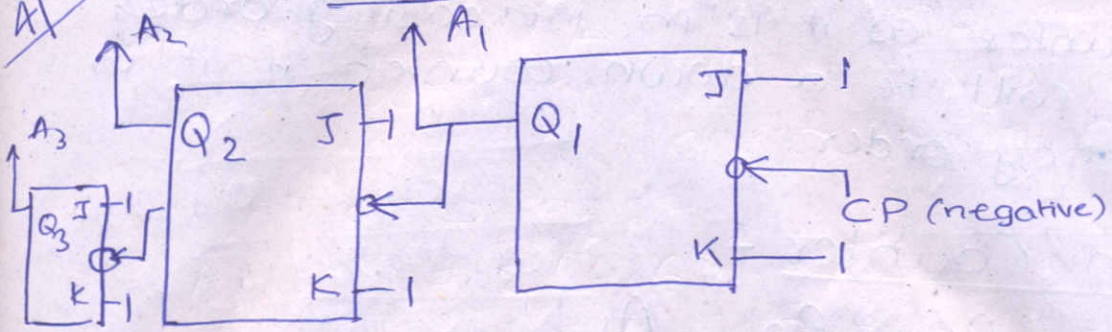
$A_1(t+1) = \Sigma(4,6)$

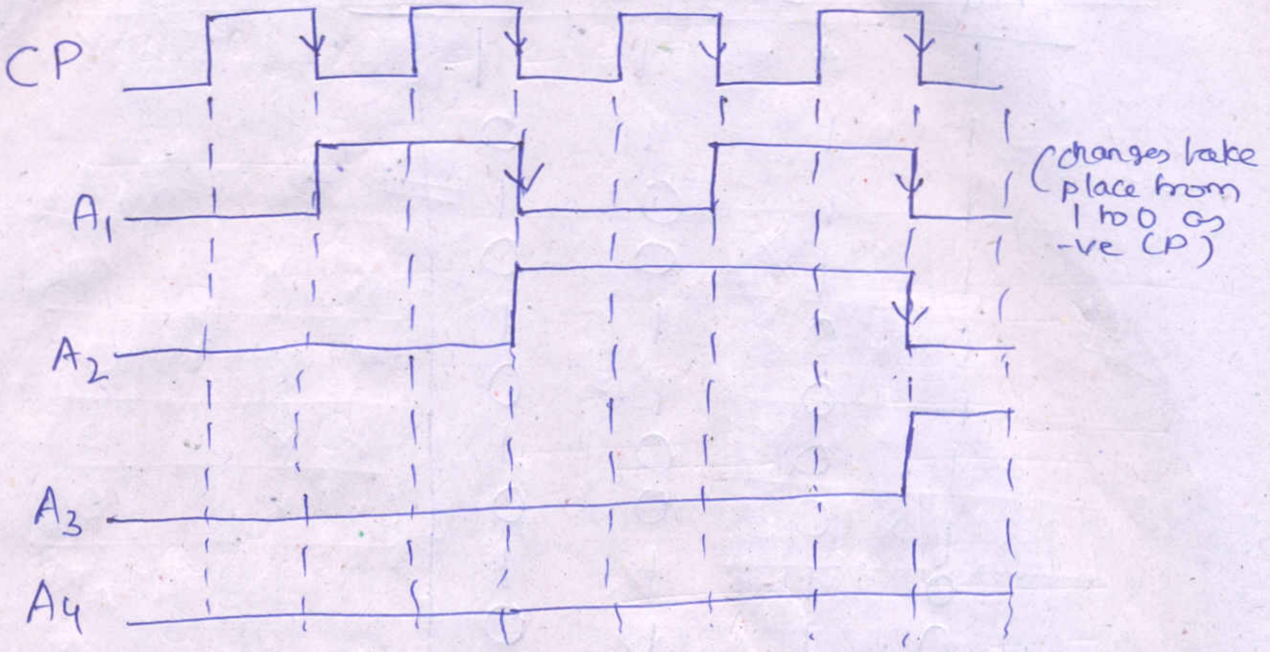
$A_2(t+1) = \Sigma(1,2,5,6)$

$Y(A_1, A_2, x)$

4/9/08

~~#Up~~ Ripple Counters :-





A_4	A_3	A_2	A_1
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
1	1	1	0

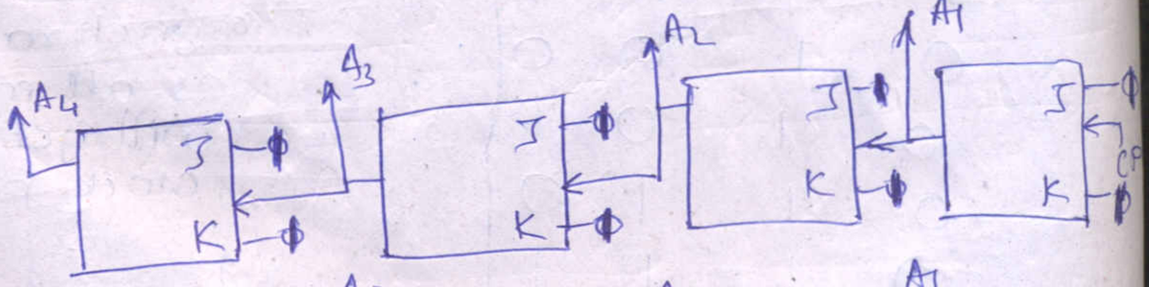
This circuit can be used as a 4-bit counter, which is asynchronous, as all modules will get diff. clock pulse.

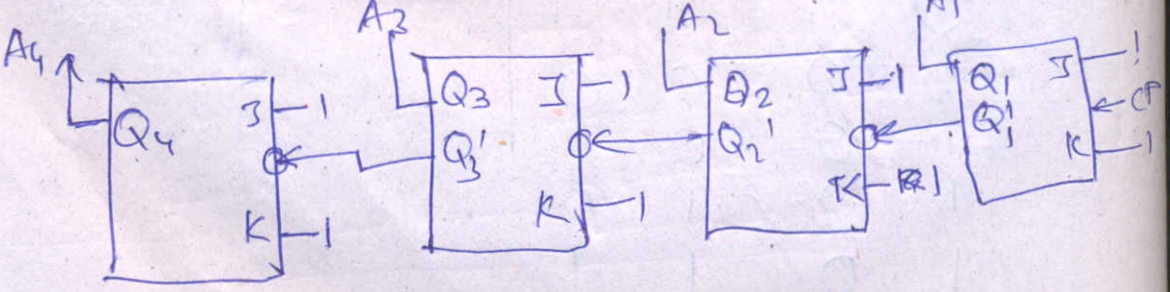
1 1 1 1

Ripple Counter :- It is an up counter as it is in increasing order else it will be a down counter if it is decreasing order.

Down Counter -

A_4	A_3	A_2	A_1
1	1	1	1
1	1	1	0
1	1	0	1
1	1	0	0
1	0	1	1
1	0	1	0
1	0	0	1
1	0	0	0
0	1	1	1
0	1	1	0
0	1	0	1
0	1	0	0
0	0	1	1
0	0	1	0
0	0	0	1
0	0	0	0





Design a 4 bit synchronous binary counter

~~A_3~~	~~A_2~~	~~A_1~~	~~A_0~~	~~A_{3n+1}~~	~~A_{2n+1}~~	~~A_{1n+1}~~	~~A_{0n+1}~~
~~0~~	~~0~~	~~0~~	~~0~~	~~0~~	~~0~~	~~0~~	~~1~~

A_3	A_2	A_1	A_0	A_{3n+1}	A_{2n+1}	A_{1n+1}	A_{0n+1}	J_{A_3}	K_{A_3}	J_{A_2}	K_{A_2}	J_{A_1}	K_{A_1}	J_{A_0}	K_{A_0}
0	0	0	0	0	0	0	1	0	X	0	X	0	X	1	X
0	0	0	1	0	0	1	0	0	X	0	X	1	X	X	1
0	0	1	0	0	0	1	1	0	X	0	X	X	0	X	X
0	0	1	1	0	1	0	0	0	X	1	X	X	1	X	1
0	1	0	0	0	1	0	1	0	X	X	0	0	X	1	X
0	1	0	1	0	1	1	0	0	X	X	0	1	X	X	1
0	1	1	0	0	1	1	1	0	X	X	0	X	0	1	X
0	1	1	1	1	0	0	0	1	X	X	1	X	1	X	1
1	0	0	0	1	0	0	1	X	0	0	X	0	X	1	X
1	0	0	1	1	0	1	0	X	0	0	X	1	X	X	1
1	0	1	0	1	0	1	1	X	0	0	X	X	0	1	X
1	0	1	1	1	1	0	0	X	0	1	X	X	1	X	1
1	1	0	0	1	1	0	1	X	0	X	0	0	X	1	X
1	1	0	1	1	1	1	0	X	0	X	0	1	X	X	1
1	1	1	0	1	1	1	1	X	0	X	0	X	0	1	X
1	1	1	1	0	0	0	0	X	1	X	1	X	1	X	1

J_A (A_0 across the top, A_3 right, A_2 left, A_1 bottom)

0	0	0	0
0	0	1	0
X	X	X	X
X	X	X	X

$J_{A_3} = A_0 A_1 A_3$

K_A (A_0 across the top, A_3 right, A_2 left, A_1 bottom)

X	X	X	X
X	X	X	X
0	0	1	0
0	0	0	0

$K_{A_3} = A_0 A_1 A_3$

J_{A_2}

$J_{A_2} = A_1 A_0$

0	0	1	0
X	X	X	X
X	X	X	X
0	0	1	0

A_0 A_1 A_3 A_2

$K_{A_2} = A_1 A_0$

X	X	X	X
0	0	1	0
0	0	1	0
X	X	X	X

A_0 A_1 A_3 A_2

J_{A_1}

0	1	X	X
0	1	X	X
0	1	X	X
0	1	X	X

A_0 A_1 A_3 A_2

$J_{A_1} = A_0$

K_{A_1}

X	X	1	0
X	X	1	0
X	X	1	0
X	X	1	0

A_0 A_1 A_3 A_2

$K_{A_1} = A_0$

J_{A_0}

1	X	X	1
1	X	X	1
1	X	X	1
1	X	X	1

A_0 A_1 A_3 A_2

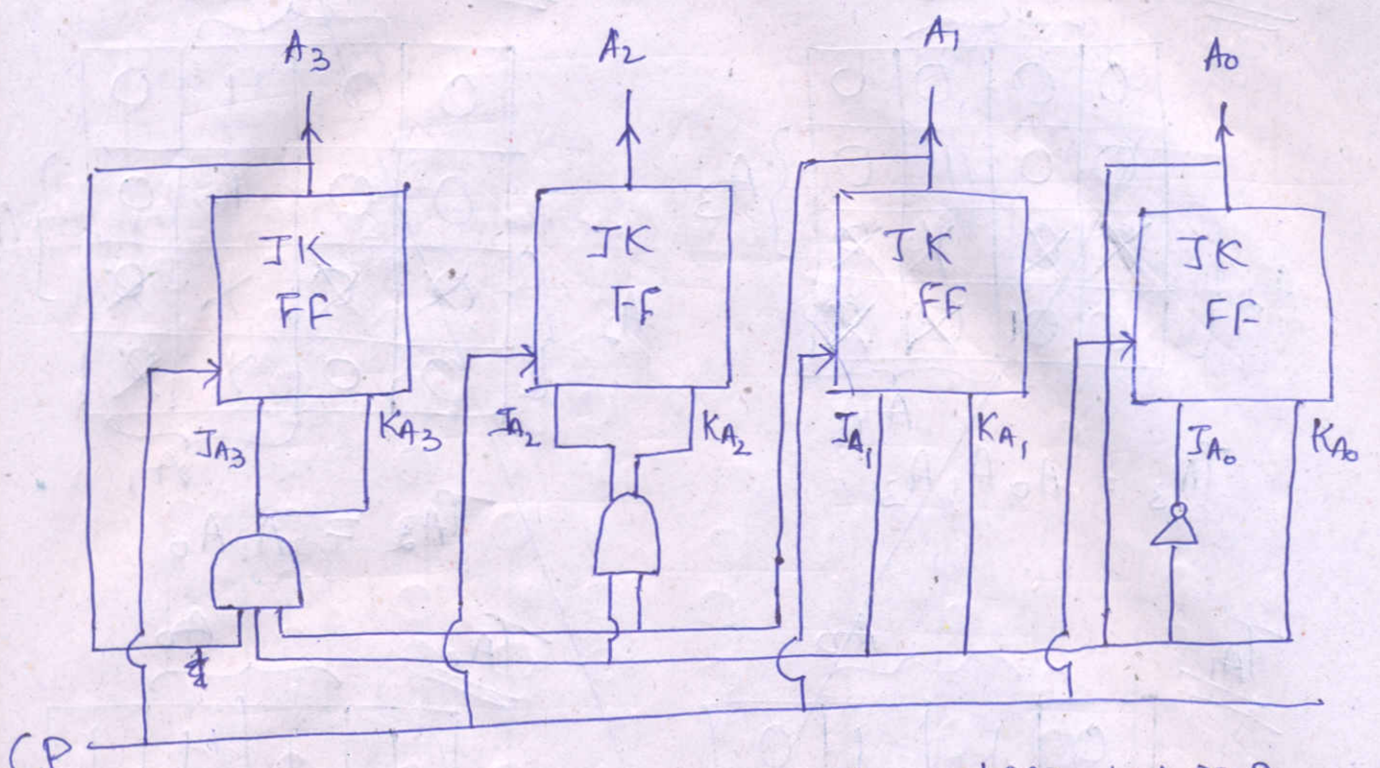
$J_{A_0} = A_0'$

K_{A_0}

X	1	1	X
X	1	1	X
X	1	1	X
X	1	1	X

A_0 A_1 A_3 A_2

$K_{A_0} = A_0$



5/9/08

Q. Design a 4-bit BCD counter using T-FF.

<u>solution</u>

	A_3	A_2	A_1	A_0	A_{3n+1}	A_{2n+1}	A_{1n+1}	A_{0n+1}	T_{A_3}	T_{A_2}	T_{A_1}	T_{A_0}
	0	0	0	0	0	0	0	1	0	0	0	1
	0	0	0	1	0	0	1	0	0	0	1	1
	0	0	1	0	0	0	1	1	0	0	0	1
	0	0	1	1	0	1	0	0	0	1	1	1
	0	1	0	0	0	1	0	1	0	0	0	1
	0	1	0	1	0	1	1	0	0	0	1	1
	0	1	1	0	0	1	1	1	0	0	0	1
	0	1	1	1	1	0	0	0	1	1	1	1
	1	0	0	0	1	0	0	1	0	0	0	1
	1	0	0	1	0	0	0	0	1	0	0	1
Dont care	1	0	1	0	1	0	1	1	0	0	0	1
	1	0	1	1	1	1	0	0	0	1	1	1
	1	1	0	0	1	1	0	1	0	0	0	1
	1	1	0	1	1	1	1	0	0	0	1	1
	1	1	1	0	1	1	1	1	0	0	0	1
	1	1	1	1	0	0	0	0	1	1	1	1

T_{A_3}

0	0	0	0
0	0	1	0
X	X	X	X
0	1	X	X

A_0, A_3, A_2, A_1

$$T_{A_3} = A_0 A_1 A_3$$

T_{A_2}

0	0	1	0
0	0	1	0
X	X	X	X
0	0	X	X

A_0, A_2, A_1

$$T_{A_2} = A_1 A_0$$

T_{A_1}

0	1	1	0
0	1	1	0
X	1	X	X
0	1	X	X

A_0, A_3, A_2, A_1

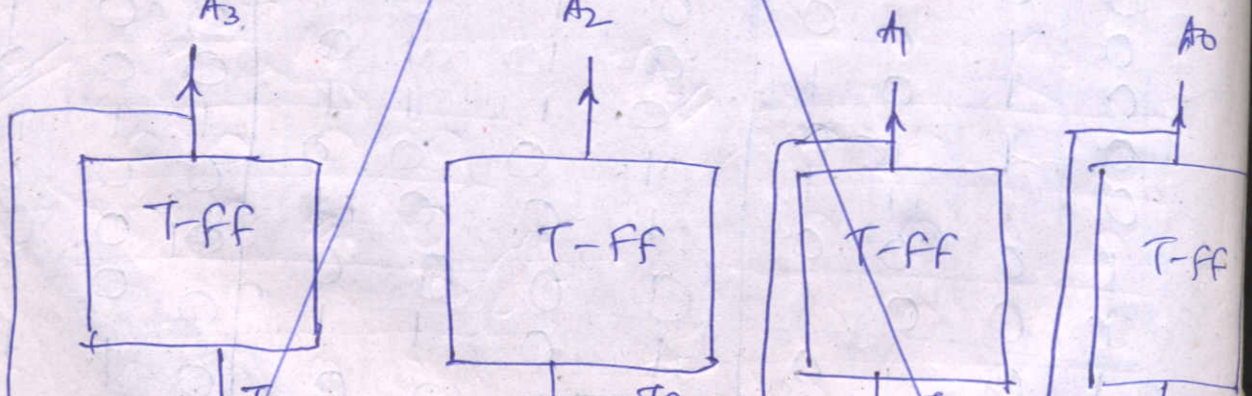
$$T_{A_1} = A_0$$

T_{A_0}

1	1	1	1
1	1	1	1
X	X	X	X
1	1	X	X

A_0, A_2, A_1

$$T_{A_0} = 1$$



T_{A3}

0	0	0	0
0	0	1	0
X	X	X	X
0	1	X	X

$T_{A_3} = A_0 A_2 + A_1 A_3 A_0$

T_{A2}

0	0	1	0
0	0	1	0
X	X	X	X
0	0	X	X

$T_{A_2} = A_1 A_0$

T_{A1}

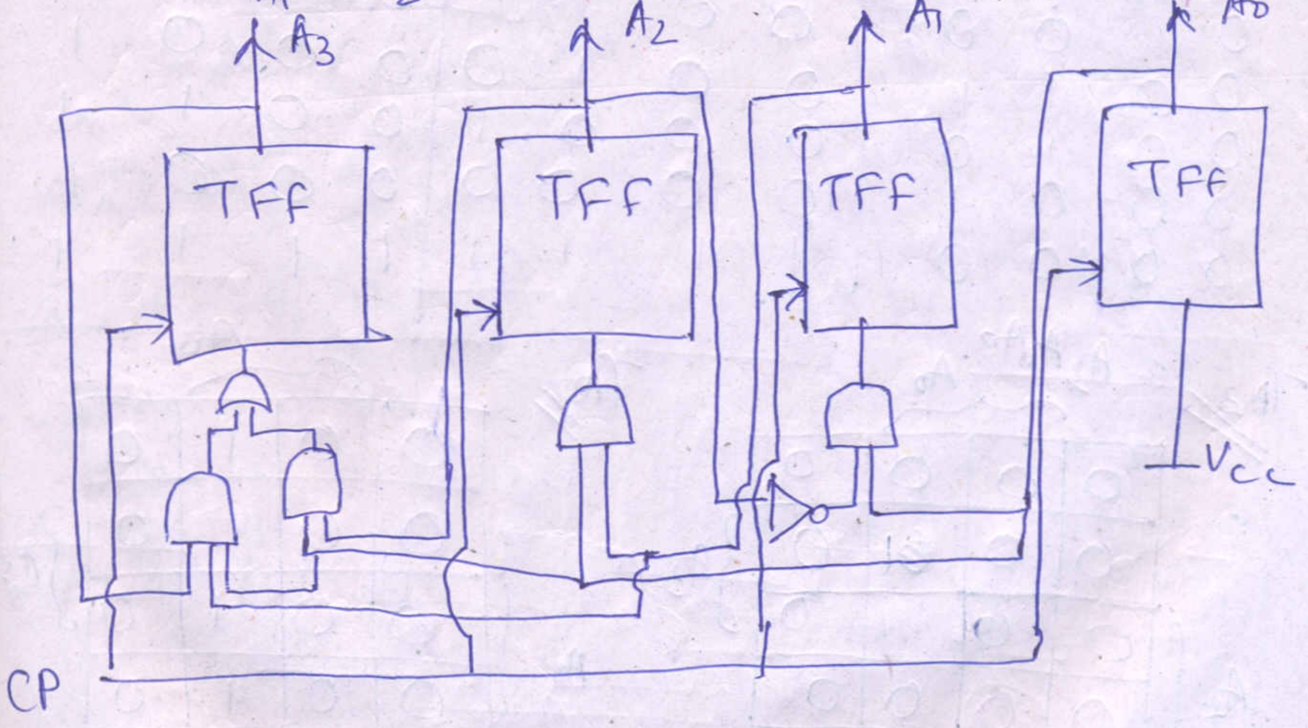
0	1	1	0
0	1	1	0
X	X	X	X
0	0	X	X

$T_{A_1} = A_2' A_0$

T_{A0}

1	1	1	1
1	1	1	1
X	X	X	X
1	1	X	X

$T_{A_0} = 1$



Design a 4 bit binary down counter using T-FF

Soln:

A_3	A_2	A_1	A_0	A_{3n+1}	A_{2n+1}	A_{1n+1}	A_{0n+1}	T_{A_3}	T_{A_2}	T_{A_1}	T_{A_0}
1	1	1	1	1	1	1	0	0	0	0	1
1	1	1	0	1	1	0	1	0	0	1	1
1	1	0	1	1	1	0	0	0	0	0	1
1	1	0	0	1	0	1	1	0	1	1	1
1	0	1	1	1	0	1	0	0	0	0	1
1	0	1	0	1	0	0	1	0	0	1	1
1	0	0	1	1	0	0	0	0	0	0	1
1	0	0	0	0	1	1	1	1	1	1	1
0	1	1	1	0	1	1	0	0	0	0	1
0	1	1	0	0	1	0	1	0	0	1	1
0	1	0	1	0	1	0	0	0	0	0	1
0	1	0	0	0	0	1	1	0	1	1	1
0	0	1	1	0	0	1	0	0	0	0	1
0	0	1	0	0	0	0	1	0	0	1	1
0	0	0	1	0	0	0	0	0	0	0	1
0	0	0	0	1	1	1	1	1	1	1	1

T_{A_3} // $= A_3 A_1 A_0$

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

A_0, A_3, A_2, A_1

T_{A_2} //

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

A_0, A_3, A_2, A_1

$T_{A_2} = A_1 A_0$

T_{A_1} //

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

A_0, A_1, A_3, A_2

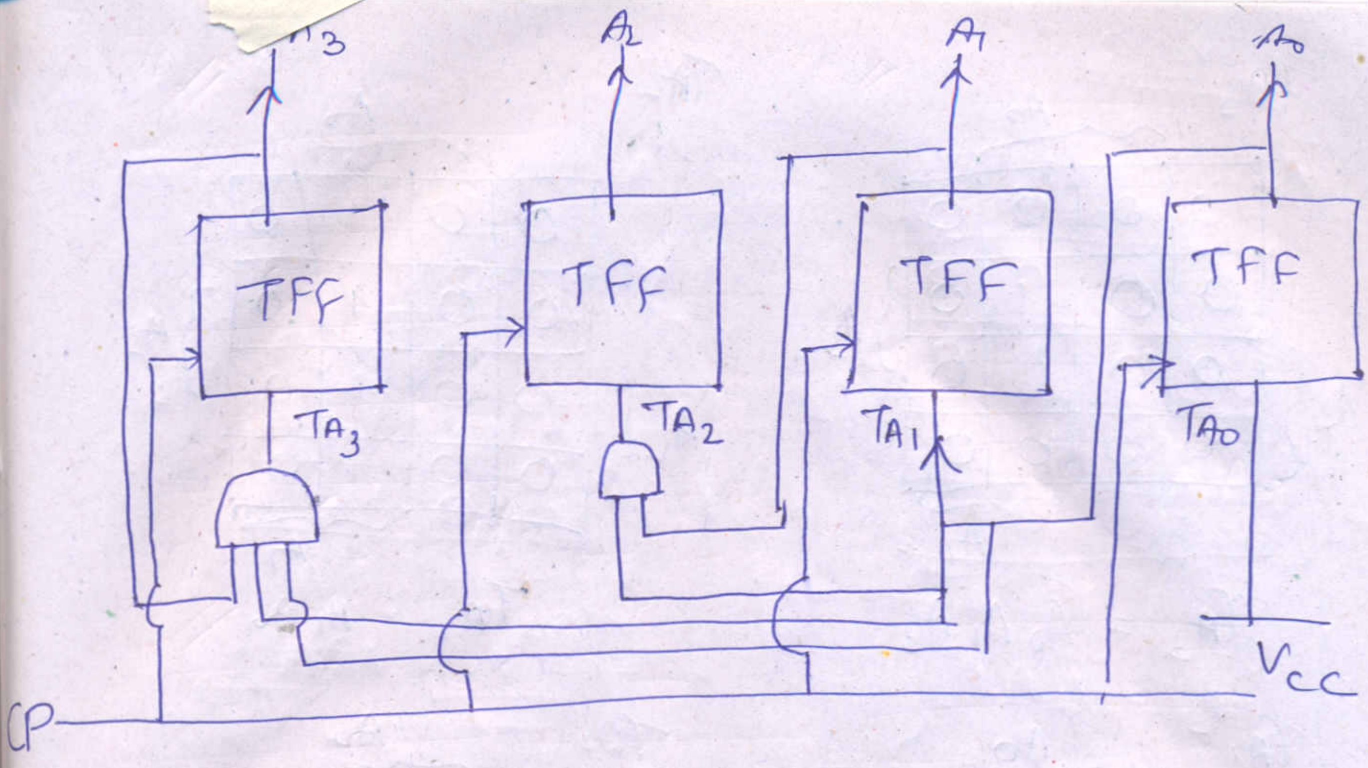
$T_{A_1} = A_0$

T_{A_0}

1	1	1	1
1	1	1	1
1	1	1	1
1	1	1	1

A_0, A_3, A_2, A_1

$T_{A_0} = 1$



Q.1. Design a 4-bit Binary up-counter using T-FF.

Q.2. Design a sequential circuit using T-FF which can act both as 4-bit binary up as well as down counter, with some external input.

1. Soln

A_3	A_2	A_1	A_0	A_{3n+1}	A_{2n+1}	A_{1n+1}	A_{0n+1}	T_{A_3}	T_{A_2}	T_{A_1}	T_{A_0}
0	0	0	0	0	0	0	1	0	0	0	1
0	0	0	1	0	0	1	0	0	0	1	1
0	0	1	0	0	0	1	1	0	0	0	1
0	0	1	1	0	1	0	0	0	1	1	1
0	1	0	0	0	1	0	1	0	0	0	1
0	1	0	1	0	1	1	0	0	0	1	1
0	1	1	0	0	1	1	1	0	0	0	1
0	1	1	1	1	0	0	0	1	1	1	1
1	0	0	0	1	0	0	1	0	0	0	1
1	0	0	1	1	0	1	0	0	0	1	1
1	0	1	0	1	0	1	1	0	0	0	1
1	0	1	1	1	1	0	0	0	1	1	1
1	1	0	0	1	1	0	1	0	0	0	1
1	1	0	1	1	1	1	0	0	0	1	1
1	1	1	0	1	1	1	1	0	0	0	1

T_{A_3}

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

$T_{A_3} = A_0 A_1 A_3$

T_{A_2}

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

$T_{A_2} = A_1 A_0$

T_{A_1}

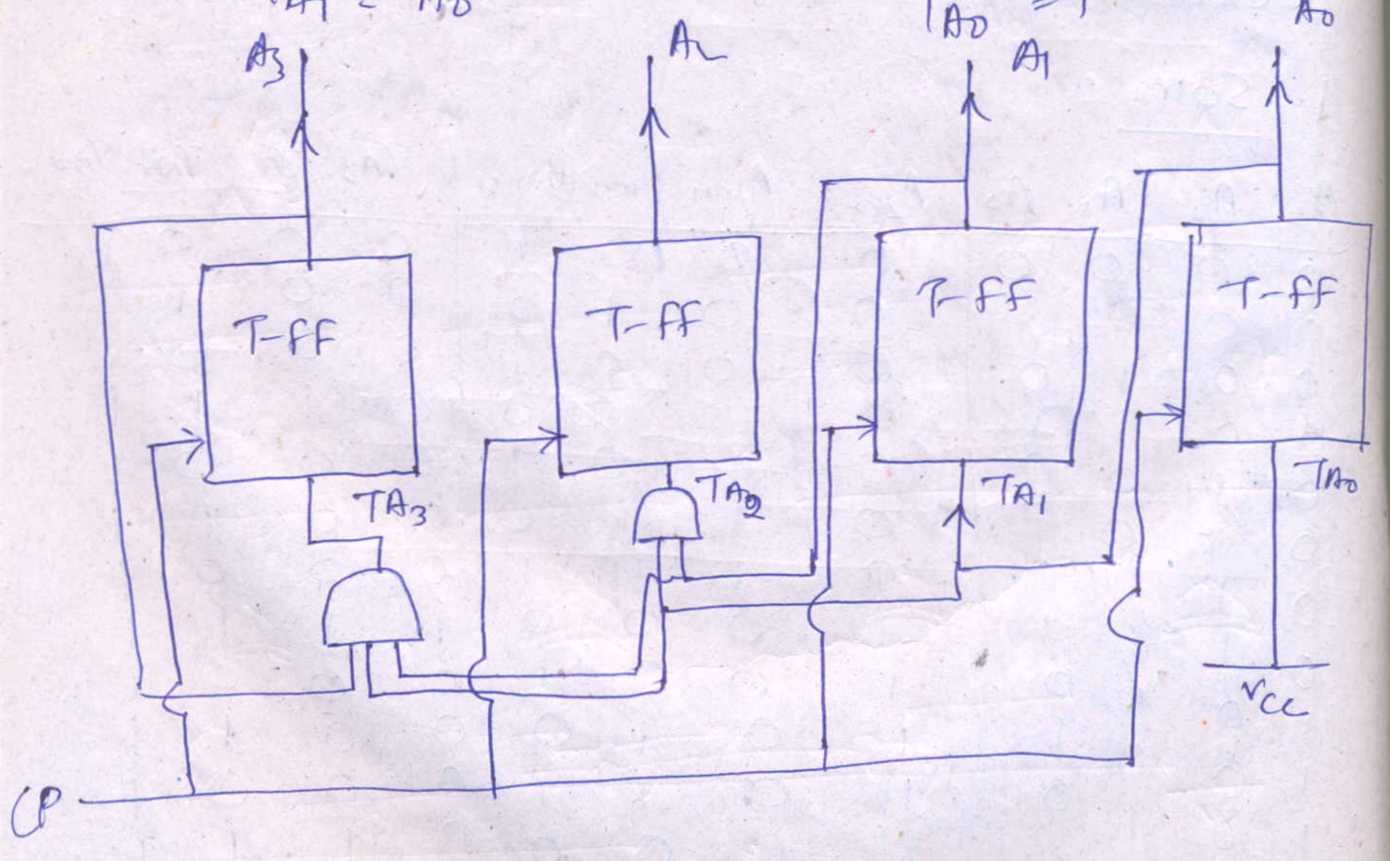
0	1	1	0
0	1	1	0
0	1	1	0
0	1	1	0

$T_{A_1} = A_0$

T_{A_0}

1	1	1	1
1	1	1	1
1	1	1	1
1	1	1	1

$T_{A_0} = 1$



2. Soln

$x = 0$

A_3	A_2	A_1	A_0	A_{3n+1}	A_{2n+1}	A_{1n+1}	A_{0n+1}	T_{A3}	T_{A2}	T_{A1}	T_{A0}
0	0	0	0	0	0	0	1	0	0	0	1
0	0	0	1	0	0	1	0	0	0	1	1
0	0	1	0	0	0	1	1	0	0	0	1
0	0	1	1	0	1	0	0	0	1	1	1
0	1	0	0	0	1	0	1	0	0	0	1
0	1	0	1	0	1	1	0	0	0	1	1
0	1	1	0	0	1	1	1	0	0	0	1
0	1	1	1	1	0	0	0	1	1	1	1
1	0	0	0	1	0	0	1	0	0	0	1
1	0	0	1	1	0	1	0	0	0	1	1
1	0	1	0	1	0	1	1	0	0	0	1
1	0	1	1	1	1	0	0	0	1	1	1
1	1	0	0	1	1	0	1	0	0	0	1
1	1	0	1	1	1	1	0	0	0	1	1
1	1	1	0	1	1	1	1	0	0	0	1
1	1	1	1	0	0	0	0	1	1	1	1

$x = 1$

A_3	A_2	A_1	A_0	A_3'	A_2'	A_1'	A_0'	T_{A3}	T_{A2}	T_{A1}	T_{A0}
1	1	1	1	1	1	1	0	0	0	0	1
1	1	1	0	1	1	0	1	0	0	1	1
1	1	0	1	1	1	0	0	0	0	0	1
1	1	0	0	1	0	1	1	0	1	1	1
1	0	1	1	1	0	1	0	0	0	0	1
1	0	1	0	1	0	0	1	0	0	1	1
1	0	0	1	1	0	0	0	0	0	0	1
1	0	0	0	0	1	1	1	1	1	1	1
0	1	1	1	0	1	1	0	0	0	0	1
0	1	1	0	0	1	0	1	0	0	1	1
0	1	0	1	0	1	0	0	0	0	0	1
0	1	0	0	0	0	1	1	0	1	1	1
0	0	1	1	0	0	1	0	0	0	0	1
0	0	1	0	0	0	0	1	0	0	1	1
0	0	0	1	0	0	0	0	0	0	0	1

For x=0

T_{A_3}

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

$T_{A_3} = A_0 A_1 A_3$

T_{A_2}

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

$T_{A_2} = A_0 A_1$

T_{A_1}

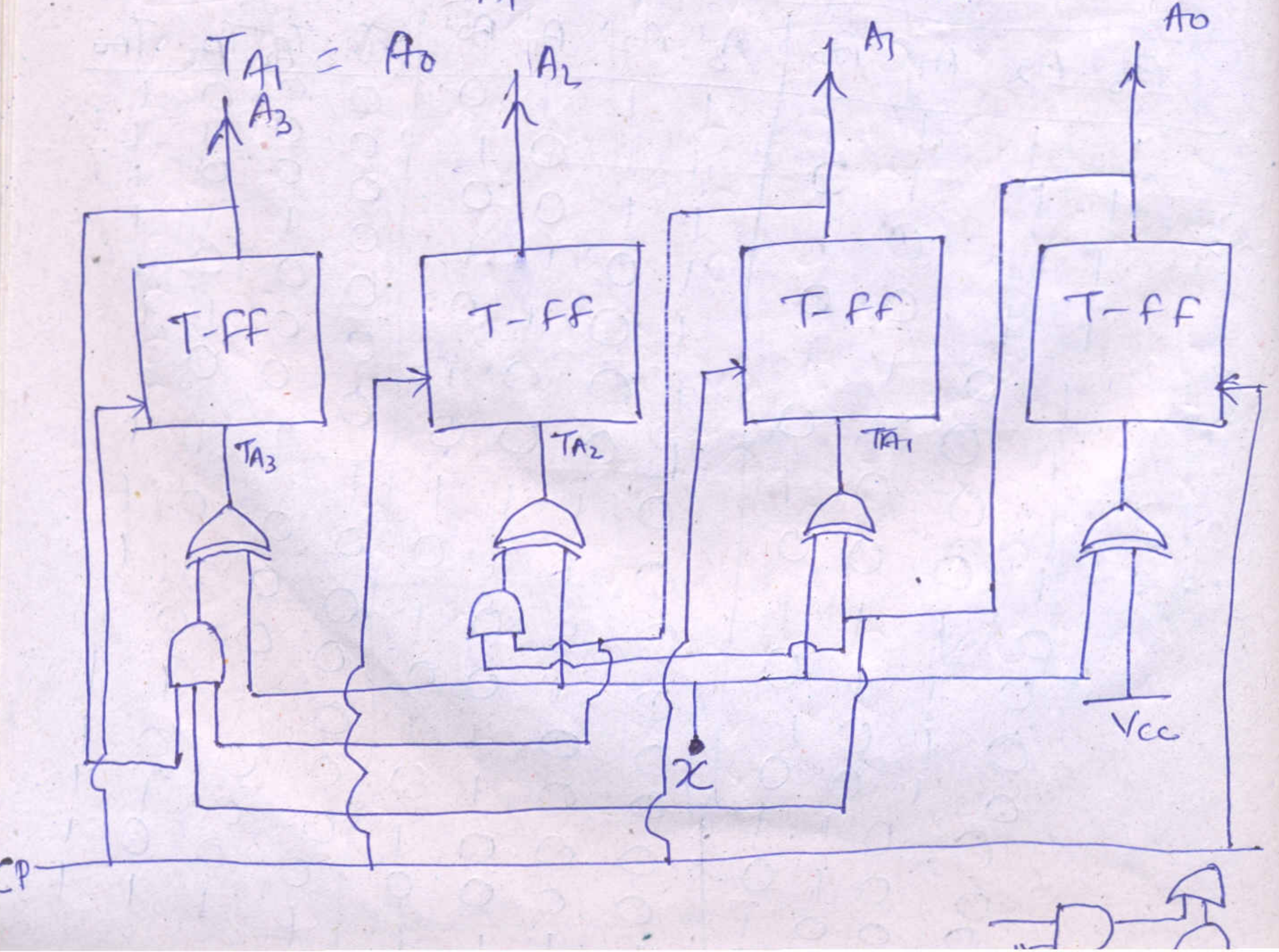
0	1	1	0
0	1	1	0
0	1	1	0
0	1	1	0

$T_{A_1} = A_0$

T_{A_0}

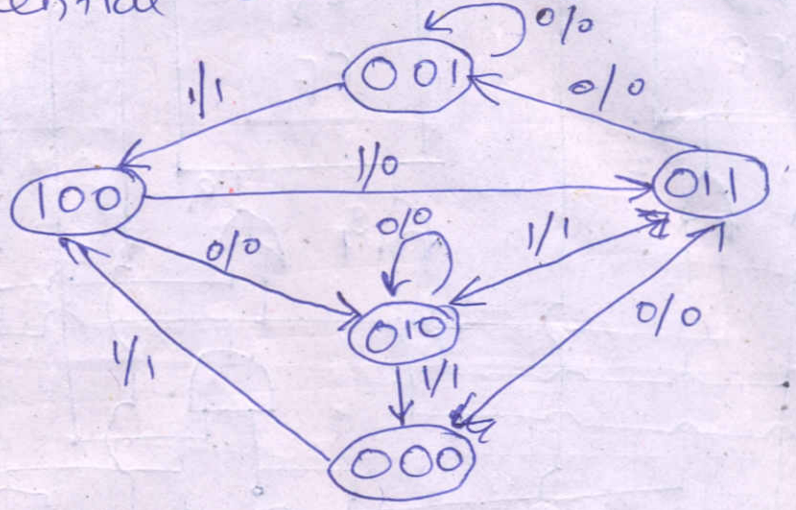
1	1	1	1
1	1	1	1
1	1	1	1
1	1	1	1

$T_{A_0} = 1$



11/9/08

Q. A sequential circuit has one input and one output. The state diagram is shown in the figure. Design the sequential circuit using R-S FF.



Soln

				i/P x=0	x=1	o/P x=0	(S,R)	x=1	(S,R)
a -	0	0	0	d	e	0	(0,X)	1	(X,0)
b -	0	0	1	b	e	0	(0,X)	1	(X,0)
c -	0	1	0	c	~~d~~ a	0	(0,X)	1	(X,0)
d -	0	1	1	b	c	0	(0,X)	1	(X,0)
e -	1	0	0	~~d~~ c	d	0	(0,X)	0	(0,1)
f -	1	0	1						
g -	1	1	0						
h -	1	1	1						

For $x = 0$

A_0	A_1	A_2	A_0	A_1	A_2	S	R
0	0	0	0	1	1	0	X
						0	X
0	0	1	0	0	1		
0	1	0	0	1	0	0	X
0	1	1	0	0	1	0	X
1	0	0	0	1	0	0	X

For $x = 1$

0	0	0	1	0	0	X	0
0	0	1	1	0	0	X	0
0	1	0	0	0	0	X	0
0	1	1	0	1	0	X	0
1	0	0	0	1	1	0	1

S =

A_2

	0	0	0	0
A_0	0	0	X	X

A_1

$S = A_0 A_1$

R =

X	X	X	X
X	X	X	X

$R = 1$

S -

A_2

X	X	X	X	
0	X	X	X	A_0

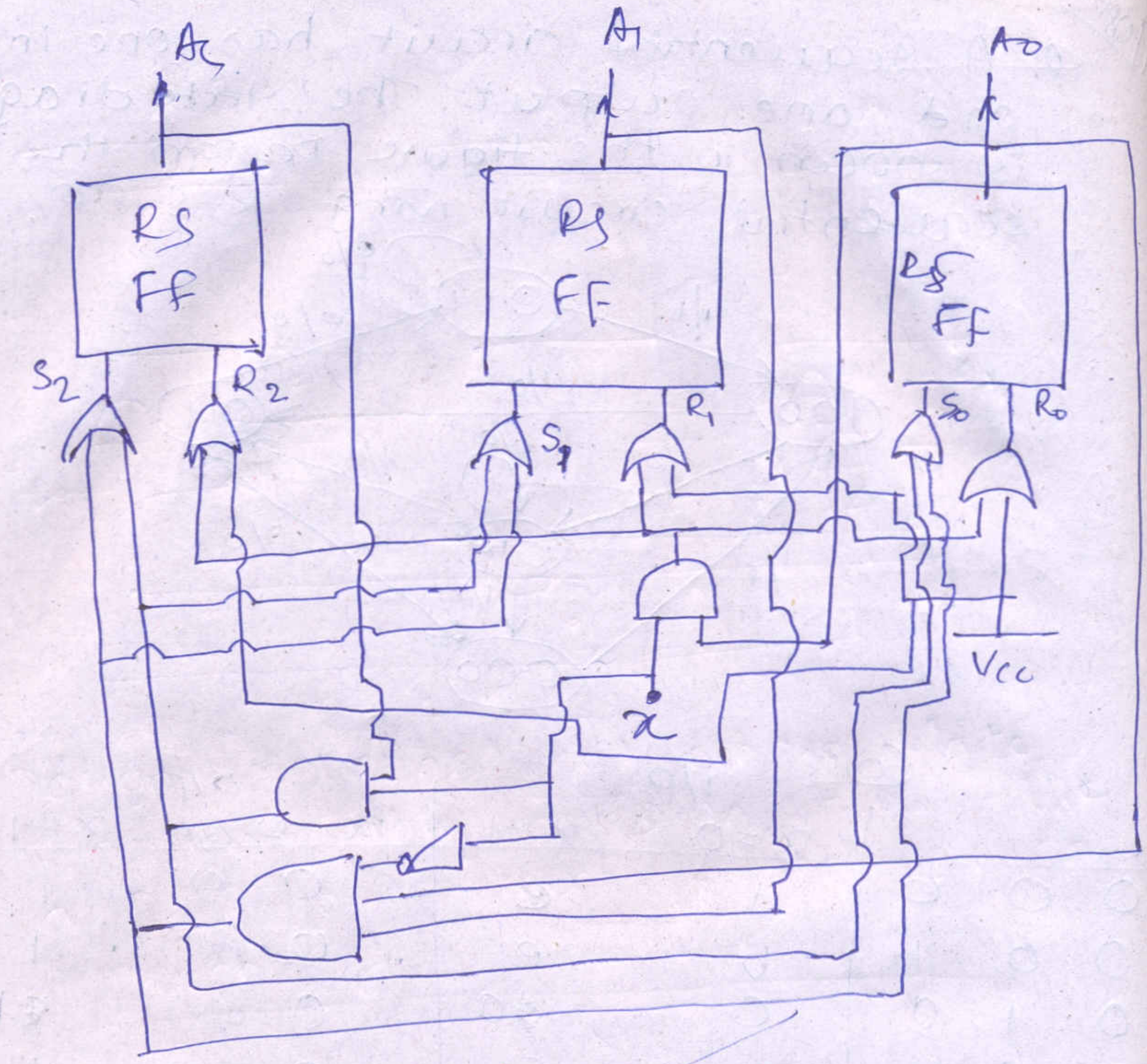
A_1

$S = A_2$

R =

0	0	0	0
1	X	X	X

$R = A_0$



ALTER :-

A_3	A_2	A_1	x	A_3	A_2	A_1	x	R_{A3}	S_{A3}	R_{A2}	S_{A2}	R_{A1}	S_{A1}
0	0	0	0	0	1	1	0	X	0	0	1	0	1
0	0	0	1	1	0	0	1	0	1	X	0	X	0
0	0	1	0	0	0	1	0	X	0	X	0	0	X
0	0	1	1	1	0	0	1	0	1	X	0	1	0
0	1	0	0	0	1	0	0	X	0	X	0	X	0
0	1	0	1	0	0	0	1	X	0	1	0	X	0
0	1	1	0	0	0	1	0	X	0	1	0	0	X
0	1	1	1	0	1	0	1	X	0	0	X	1	0
1	0	0	0	0	1	0	0	0	0	0	1	X	0
1	0	0	1	0				1	0	0	1	0	
1	0	1	0	0	1	1	0						
1	0	1	1	X	X	X	X						
1	1	0	0	X	X	X	X						
1	1	0	1										
1	1	1	0										
1	1	1	1										

R_{A_3}

A_0

X	0	0	X	
X	X	X	X	A_3
X	X	X	X	
Φ	Φ	X	X	A_2

A_1

$R_{A_3} = A_2$

S_{A_3}

A_0

0	1	1	0	
0	0	0	0	A_3
X	X	X	X	
0	0	X	X	A_2

A_1

$S_{A_3} = A_0 A_2' A_3'$

R_{A_2}

A_0

0	X	X	X	
X	1	0	1	A_3
X	X	X	X	
0	0	X	X	A_2

A_1

~~$R_{A_2} = A_2' A_3$~~

$R_{A_2} = A_1' A_2' A_0 A_3 + A_0' A_1 A_2' A_3$

S_{A_2}

A_0

1	0	0	0	
0	0	X	0	A_3
X	X	X	X	
1	1	X	X	A_2

A_1

$S_{A_2} = A_2 + A_1' A_2' A_0' A_3'$

R_{A_1}

A_0

0	X	1	0	
X	X	1	0	A_3
X	X	X	X	
X	0	X	X	A_2

A_1

$R_{A_1} = A_0 A_2'$

S_{A_1}

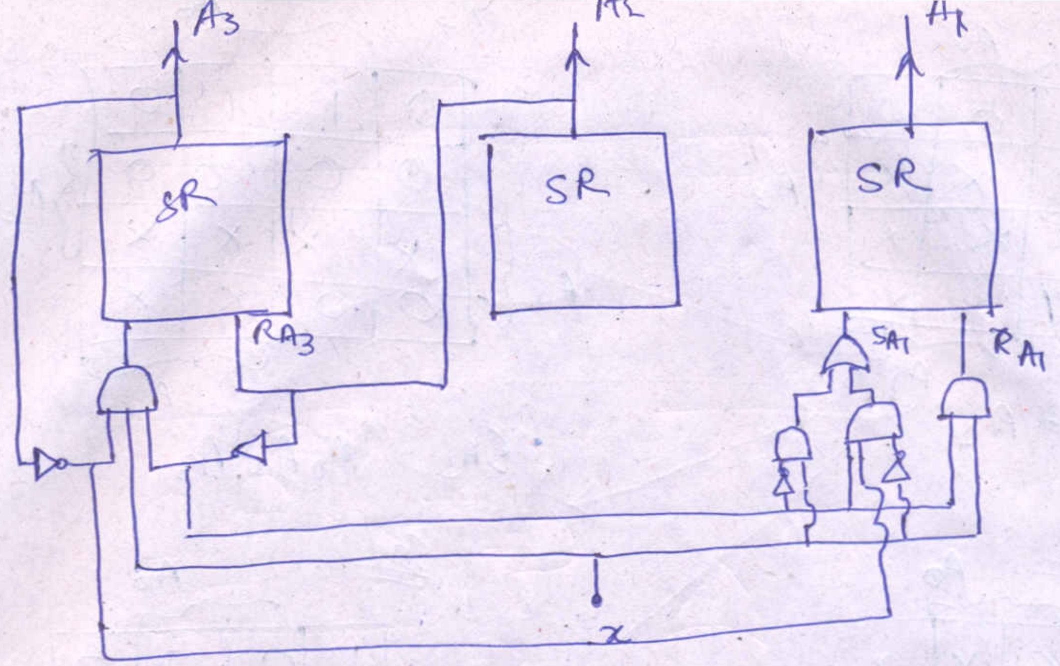
A_0

1	0	0	X	
0	0	0	X	A_3
X	X	X	X	
0	1	X	X	A_2

A_1

$S_{A_1} = A_0 A_2 + A_0' A_3' A_2'$

$A_0 = x$



Q. Design sequential ckt whose state Transition Table is shown below :- Use a D-FF to implement y_0 and J-K FF to implement y_1

PS		NS y_1, y_0, Z	
y_1	y_0	$x=0$	$x=1$
0	0	0 1 0	0 1 0
0	1	1 1 0	0 0 1
1	0	0 1 0	1 0 0
1	1	1 1 0	0 1 1

Solution

PS			NS					
y_1	y_0	x	y_1	y_0	Z	J_{y_1}	K_{y_1}	D_{y_0}
0	0	0	0	1	0	0	X	1
0	0	1	0	1	0	0	X	1
0	1	0	1	1	0	1	X	1
0	1	1	0	0	1	0	X	0
1	0	0	0	1	0	X	1	1
1	0	1	1	0	0	X	0	0
1	1	0	1	1	0	X	0	1
1	1	1	0	1	1	X	1	1

J_{y_1}

	0	0	0	1
y_1	X	X	X	X

$J_{y_1} = y_0 x'$

K_{y_1}

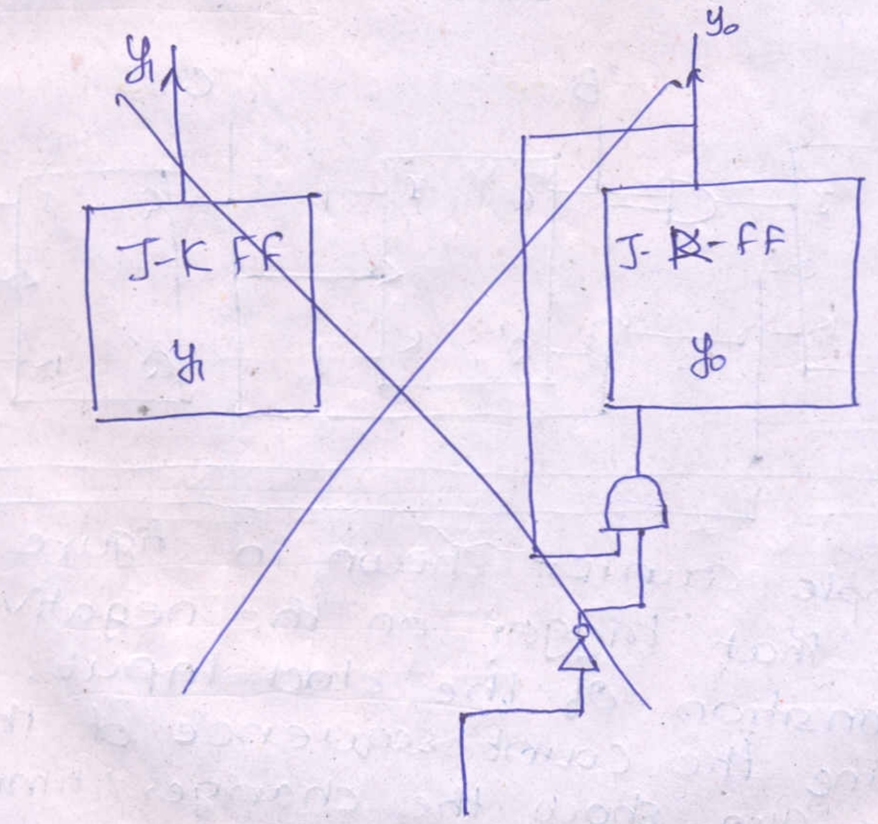
	X	X	X	X
y_1	1	0	1	0

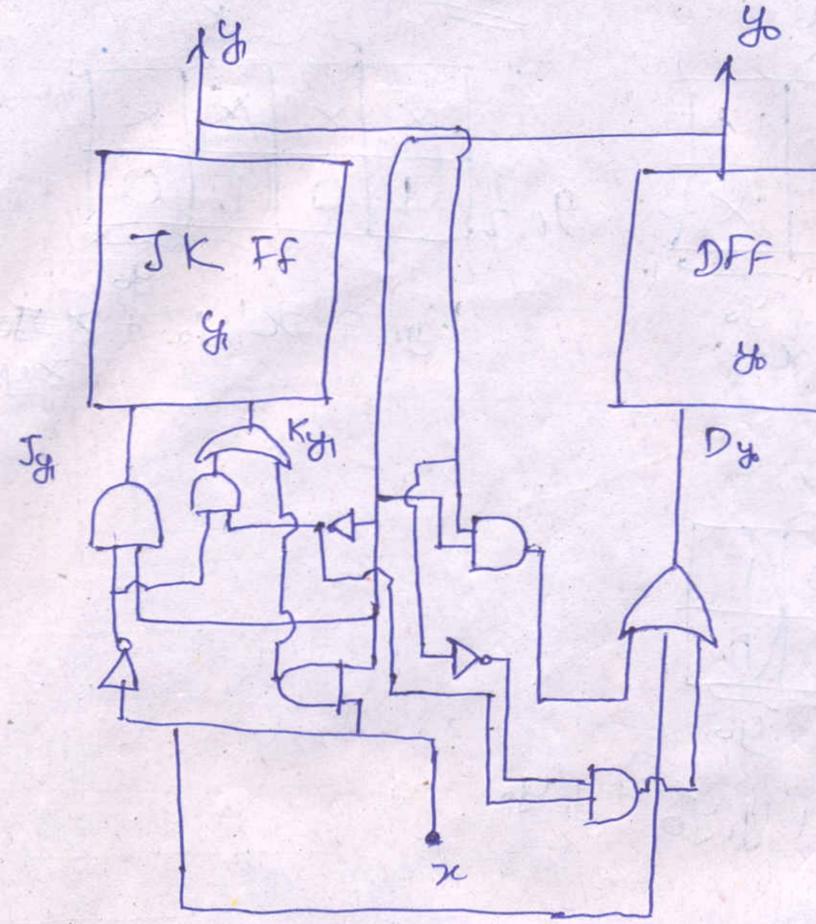
$K_{y_1} = x'y_0' + xy_0$

D_{y_1}

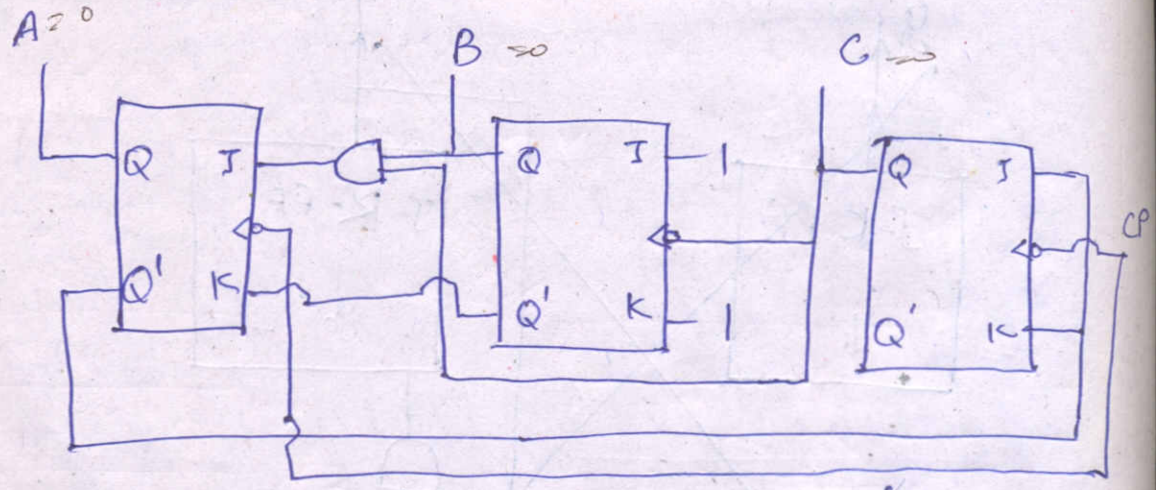
	1	1	0	1
y_1	1	0	1	1

$D_{y_1} = x' + y_1 y_0 + y_1' y_0'$



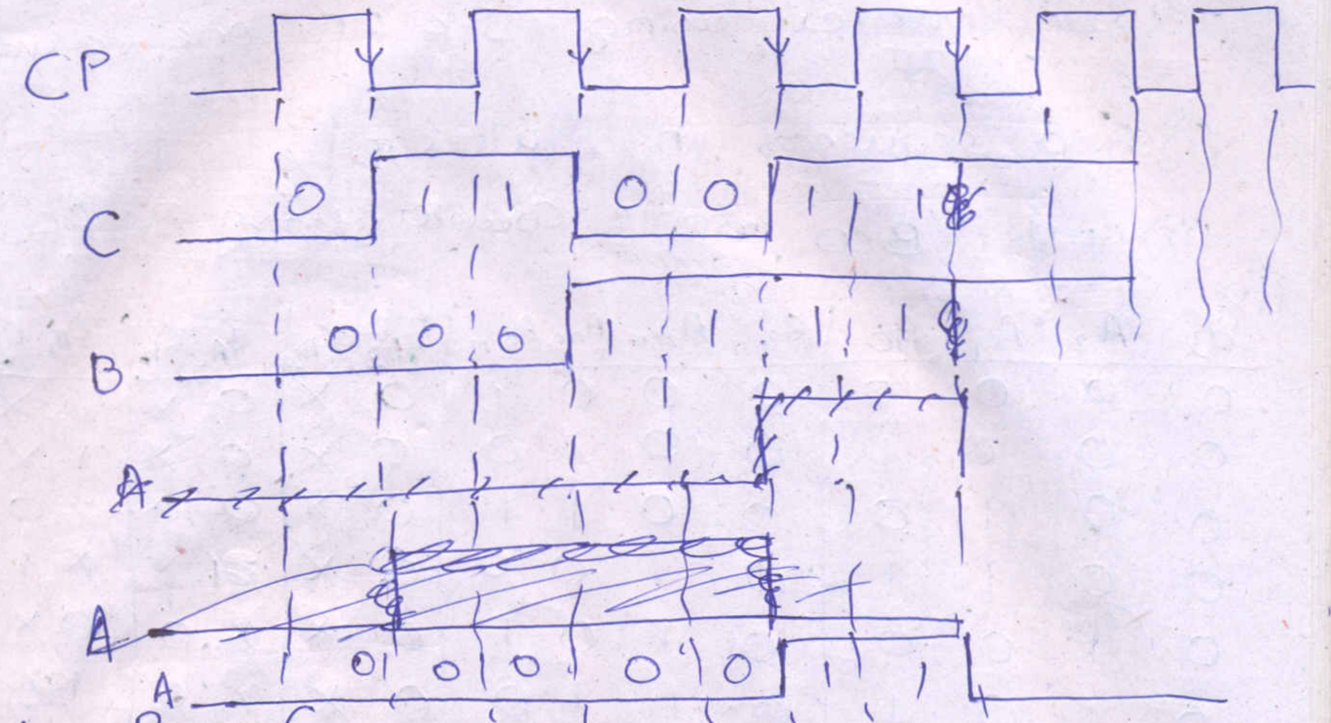


Q.



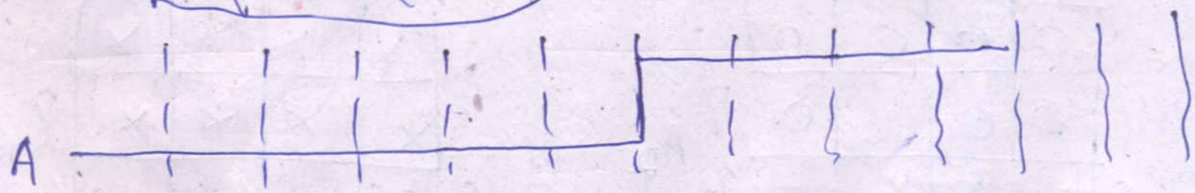
The ripple counter shown in figure uses flipflops that trigger on the negative edge transition of the clock input. Determine the count sequence of the counter. Also, show the changes (timing diagram) of A, B and C with respect to CP input.

Solution



A	B	C	
0	0	0	→ 0
1	0	1	→ 5
1	1	0	→ 6
0	1	1	→ 3
0	0	0	→ 0

$0 \rightarrow 5 \rightarrow 6 \rightarrow 3$



1) Design a 4 bit synchronous up and down counter using J-K FF, one variable x.

ii) Find sequences in Multisim

iii) 4-Bit BCD ripple Counter using JK ff

iii)

A_3	A_2	A_1	A_0	A_{3n+1}	A_{2n+1}	A_{1n+1}	A_{0n+1}	J_{A_3}	K_{A_3}	J_{A_2}	K_{A_2}	J_{A_1}	K_{A_1}	J_{A_0}
0	0	0	0	0	0	0	1	0	X	0	X	0	X	1
0	0	0	1	0	0	1	0	0	X	0	X	1	X	X
0	0	1	0	0	0	1	1	0	X	0	X	X	0	1
0	0	1	1	0	1	0	0	0	X	1	X	X	1	X
0	1	0	0	0	1	0	1	0	X	X	0	0	X	1
0	1	0	1	0	1	1	0	0	X	X	0	1	X	X
0	1	1	0	0	1	1	1	0	X	X	0	X	0	1
0	1	1	1	1	0	0	0	1	X	X	1	X	1	X
1	0	0	0	1	0	0	1	X	0	0	X	0	X	1
1	0	0	1	0	0	0	0	X	1	0	X	0	X	X
X	X	X	X	X	X	X	X							

J_{A_3} (A_0, A_1 across; A_2, A_3 down)

0	0	0	0
0	0	1	0
X	X	X	X
X	X	X	X

K_{A_3} (A_0, A_1 across; A_2, A_3 down)

X	X	X	X
X	X	X	X
X	X	X	X
0	1	X	X

$J_{A_3} = A_0 A_1 A_2$

$K_{A_3} = A_0$

J_{A_2} (A_0, A_1 across; A_2, A_3 down)

0	0	1	0
X	X	X	X
X	X	X	X
0	0	X	X

$J_{A_2} = A_0 A_1$

$K_{A_2} = A_0 A_1$ (A_0, A_1 across; A_2 down)

X	X	X	X
0	0	1	0
X	X	X	X
X	X	X	X

J_{A_1}

0	1	X	X
0	1	X	X
X	X	X	X
0	0	X	X

A_0, A_3, A_2, A_1

$J_{A_1} = A_0 A_3'$

K_{A_1}

X	X	1	0
X	X	1	0
X	X	X	X
X	X	X	X

A_0, A_2, A_3, A_1

$K_{A_1} = A_0$

J_{A_0}

1	X	X	1
1	X	X	1
X	X	X	X
1	X	X	X

$J_{A_0} = 1 = A_0'$

K_{A_0}

X	1	X	X
X	1	1	X
X	X	X	X
X	1	X	X

$K_{A_0} = 1 = A_0$

i)

JK	A_3	A_2	A_1	A_0	A_{3+}	A_{2+}	A_{1+}	A_{0+}	J_{A_3} K_{A_3}	J_{A_2} K_{A_2}	J_{A_1} K_{A_1}
1X	0	0	0	0	0	0	0	1	0 X	0 X	0 X
X1	0	0	0	1	0	0	1	0	0 X	0 X	1 X
1X	0	0	1	0	0	0	1	1	0 X	0 X	X 0
X1	0	0	1	1	0	1	0	0	0 X	1 X	~~X~~ ~~1~~
1X	0	1	0	0	0	1	0	1	0 X	X 0	~~0~~ ~~X~~
X1	0	1	0	1	0	1	1	0	0 X	X 0	~~1~~ ~~X~~
1X	0	1	1	0	0	1	1	1	0 X	X 0	X 0
X1	0	1	1	1	1	0	0	0	1 X	X 1	X 1
1X	1	0	0	0	1	0	0	1	X 0	0 X	0 X
X1	1	0	0	1	1	0	1	0	X 0	0 X	1 X
1X	1	0	1	0	1	0	1	1	X 0	0 X	X 0
X1	1	0	1	1	1	1	0	0	X 0	1 X	X 1
1X	1	1	0	0	1	1	0	1	X 0	X 0	0 X
X1	1	1	0	1	1	1	1	0	X 0	X 0	1 X
1X	1	1	1	0	1	1	1	1	X 0	X 0	X 0
X1	1	1	1	1	0	0	0	0	X 1	X 1	X 1

$J_{A0} = 1$

1	X	X	1
1	X	X	X
1	X	X	1
1	X	X	1

$K_{A0} = 1$

X	1	1	X
X	1	1	X
X	1	1	X
X	1	1	X

$J_{A1} = \cancel{A} A_0$

	0	1	X	X
	0	1	X	X
A_3	0	1	X	X
A_3	0	1	X	X

A_2 (rows 2–3), A_1 (columns 3–4)

$K_{A1} = A_0$

X	X	1	0
X	X	1	0
X	X	1	0
X	X	1	0

$J_{A2} = A_0 A_1$

0	0	1	0
X	X	X	X
X	X	X	X
0	0	1	0

$K_{A2} = A_0 A_1$

X	X	X	X
0	0	1	0
0	0	1	0
X	X	X	X

$J_{A3} = A_0 A_1 A_2$

0	0	0	0
0	0	1	0
φ	φ	φ	φ
φ	φ	φ	0

$K_{A3} = A_0 A_1 A_2$

X	X	X	X
X	X	X	X
0	0	1	0
0	0	0	0

16/9/08

Mod Counter :-

Mod N counter -

0, 1, 2, ... N-1.

Q. Design a mod 7 counter using D-FF.

Soln

A_2	A_1	A_0	A_{2n+1}	A_{1n+1}	A_{0n+1}	D_{A_2}	D_{A_1}	D_{A_0}
0	0	0	0	0	1	0	0	1
0	0	1	0	1	0	0	1	0
0	1	0	0	1	1	0	1	1
0	1	1	1	0	0	1	0	0
1	0	0	1	0	1	1	0	1
1	0	1	1	1	0	1	1	0
1	1	0	0	0	0	0	0	0
	X			X			X	

D_{A_2}

		A_0	A_0	
	0	0	1	0
A_2	1	1	X	0
			A_1	A_1

$D_{A_2} = A_1 A_0 + A_1' A_2$

D_{A_1}

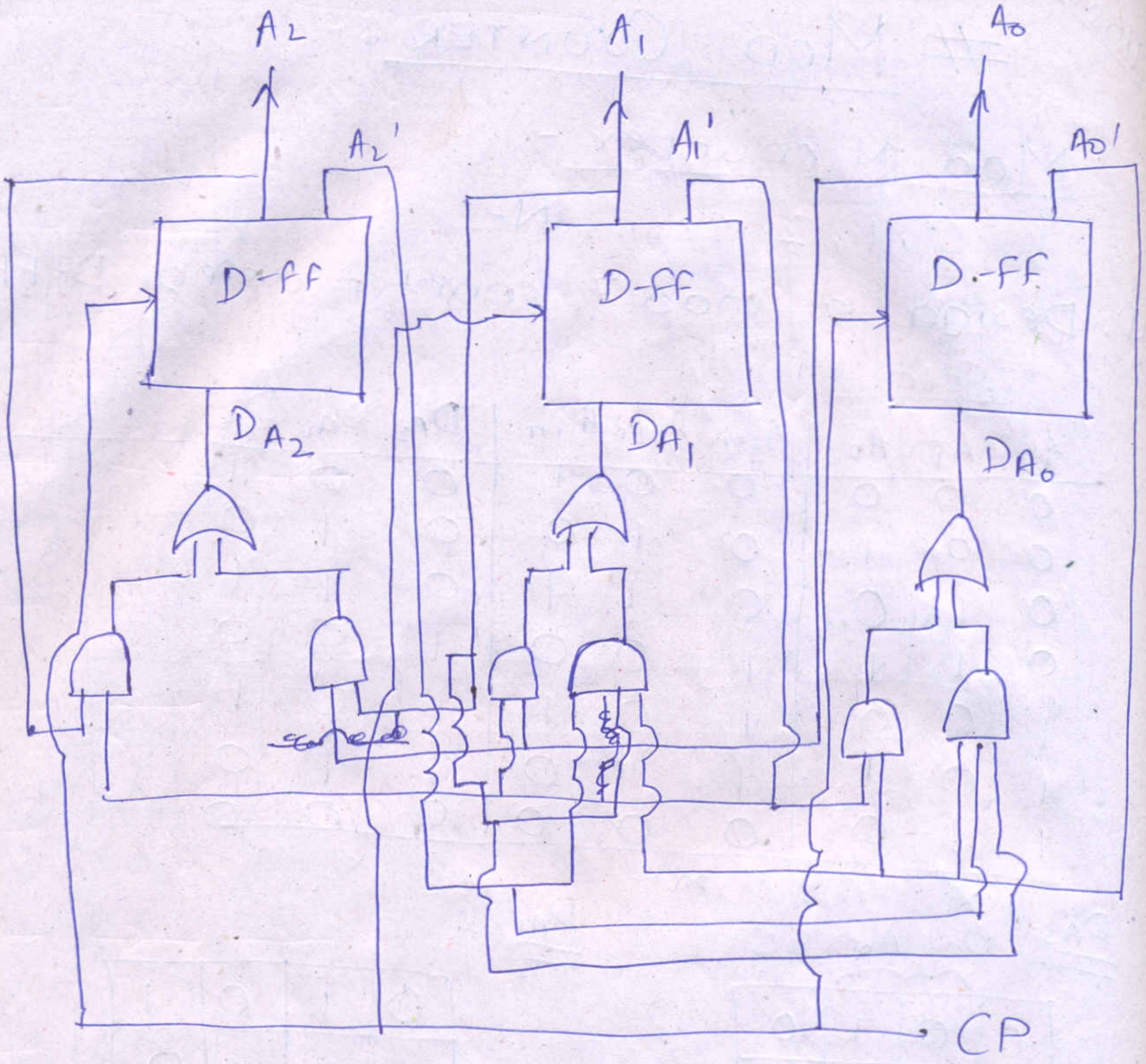
0	1	0	1
0	1	X	0

$D_{A_1} = A_0 A_1' + A_0' A_1 A_2'$

D_{A_0}

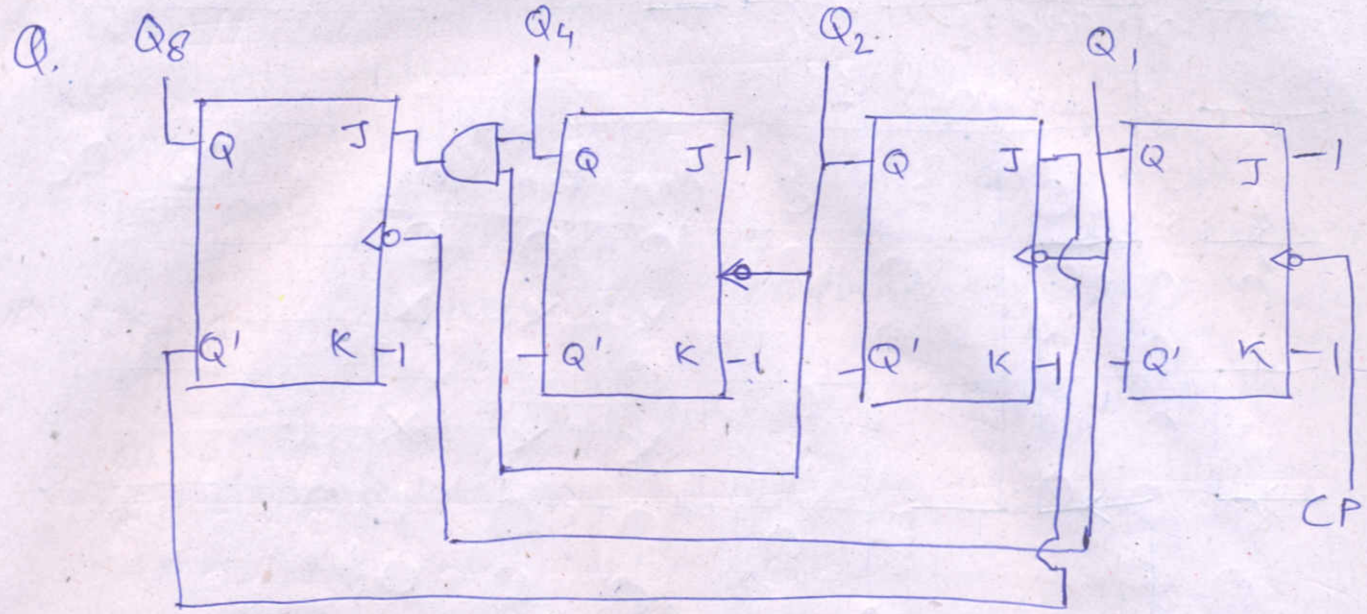
1	0	0	1
1	0	X	0

$D_{A_0} = A_0' A_1' + A_0' A_1 A_2'$



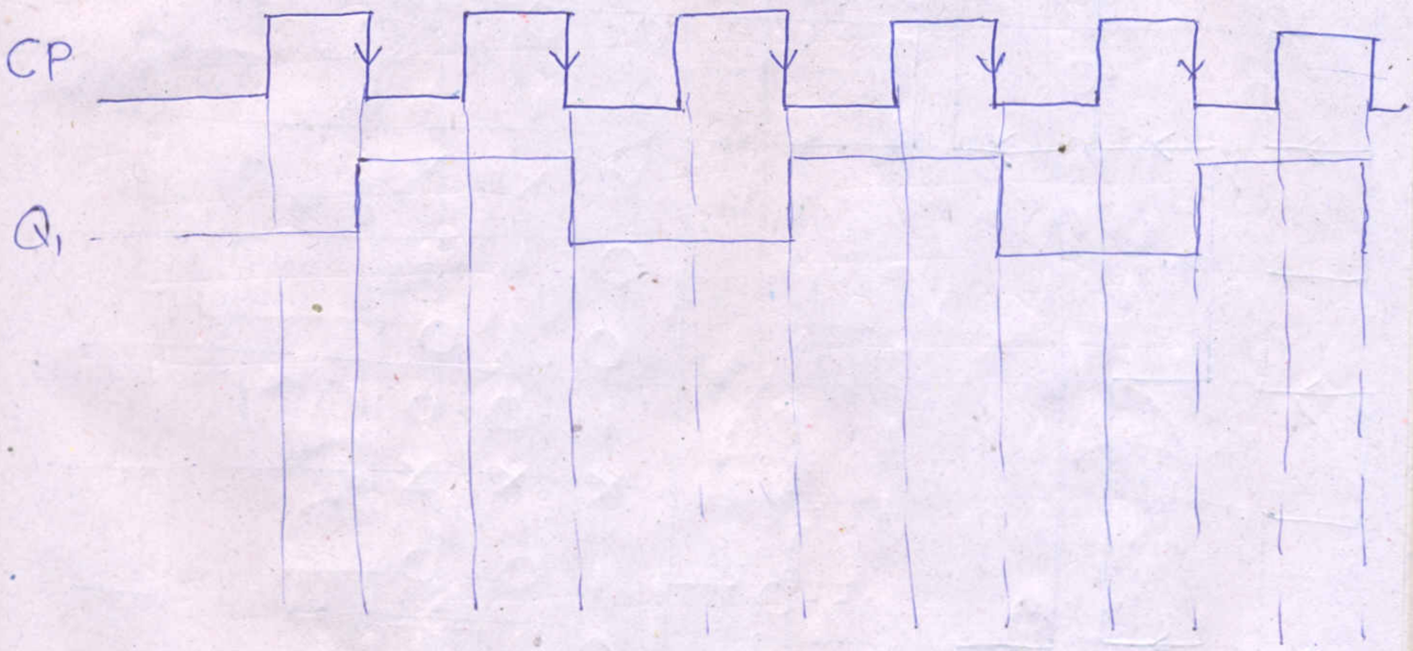
NOTE - When not explicitly defined that take any flipflop, always take D-flipflop.

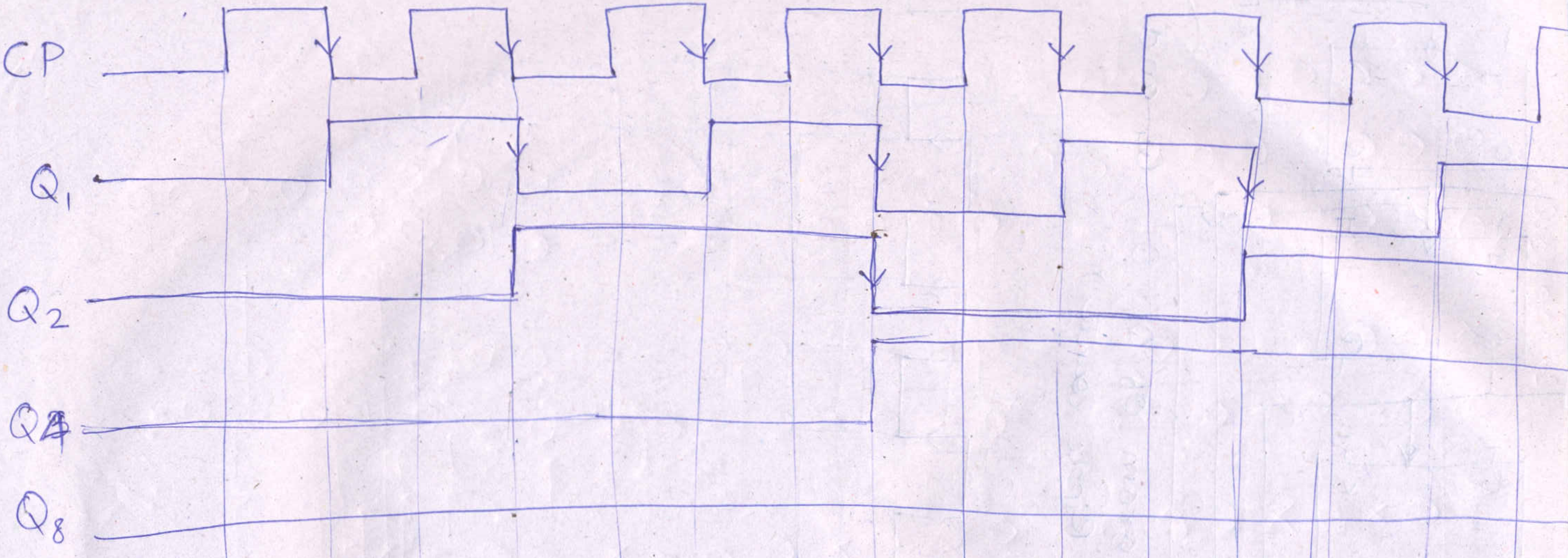
Q.



Draw timing Diagram of Q_1, Q_2, Q_4 and Q_8 wrt CP. (BCD Ripple counter)

Solution





Q_8	Q_4	Q_2	Q_1
0	0	0	0
0	0	0	1
0	0	1	0
0	1	0	0
0	1	1	0
0	1	1	1

Q. Draw a sequential ckt which maintains a 3-bit value in their flip-flops, with two external inputs:- "add one" and "subtract one". The "add one" operation is like a normal counter, ie, the output of the counter increases by one. "Subtract one" performs the opposite function, ie, like 6 will change to five. These two external input lines cannot be performed simultaneously.

<u>Soln</u>

If A_0, A_1 and A_2 be the three D-flipflops.

A_0	A_1	A_2	A = 1 D = 0			A = 0 D = 0			A = 0 D = 1		
0	0	0	0	0	1	0	0	0	X	X	X
0	0	1	0	1	0	0	0	1	0	0	0
0	1	0	0	1	1	0	1	0	0	0	1
0	1	1	1	0	0	0	1	1	0	1	0
1	0	0	1	0	1	1	0	0	0	1	1
1	0	1	1	1	0	1	0	1	1	0	0
1	1	0	1	1	1	1	1	0	1	0	1
1	1	1	X	X	X	1	1	1	1	1	0

x : Add 1

y : Subtract 1

	A	B	C	x	y	A_{n+1}	B_{n+1}	C_{n+1}	D_A	D_B	D_C
0	0	0	0	0	0	0	0	0			
1	0	0	0	0	1	X	X	X			
2	0	0	0	1	0	0	0	1			
3	0	0	0	1	1	X	X	X			
4	0	0	1	0	0	0	0	1			
5	0	0	1	0	1	0	0	0			
6	0	0	1	1	0	0	1	0			
7	0	0	1	1	1	X	X	X			
8	0	1	0	0	0	0	1	0			
9	0	1	0	0	1	0	0	1			
10	0	1	0	1	0	0	1	1			
11	0	1	0	1	1	X	X	X			
12	0	1	1	0	0	0	1	1			
13	0	1	1	0	1	0	1	0			
14	0	1	1	1	0	1	0	0			
15	0	1	1	1	1	X	X	X			
16	1	0	0	0	0	1	0	0			
17	1	0	0	0	1	0	1	1			
18	1	0	0	1	0	1	0	1			
19	1	0	0	1	1	X	X	X			
20	1	0	1	0	0	1	0	1			
4	1	0	1	0	1	1	0	0			
22	1	0	1	1	0	1	1	0			
23	1	0	1	1	1	X	X	X			
24	1	1	0	0	0	1	1	0			
25	1	1	0	0	1	1	0	1			
	1	1	0	1	0	1	1	1	X	X	X

	A	B	C	x	y	A_{n+1}	B_{n+1}	C_{n+1}
28	1	1	1	0	0	1	1	1
29	1	1	1	0	1	1	1	0
30	1	1	1	1	0	X	X	X
31	1	1	1	0	0	X	X	X

<u>D_A</u>

C

(0) 0	(1) X	(3) X	(2) 0	(6) 0	(7) X	(5) 0	(4) 0
(8) 0	(9) 0	(11) X	(10) 0	(14) 1	(15) X	(13) 0	(12) 0
(24) 1	(25) 1	(27) X	(26) 1	(30) X	(31) X	(29) 1	(28) 1
(16) 1	(17) 0	(19) X	(18) 1	(22) 1	(23) X	(21) 1	(20) 1

B

A

y x y

$$D_A = AB + \cancel{Ay' + xCB}$$
$$+ AC + Ay' + BCx$$

<u>D_B</u>

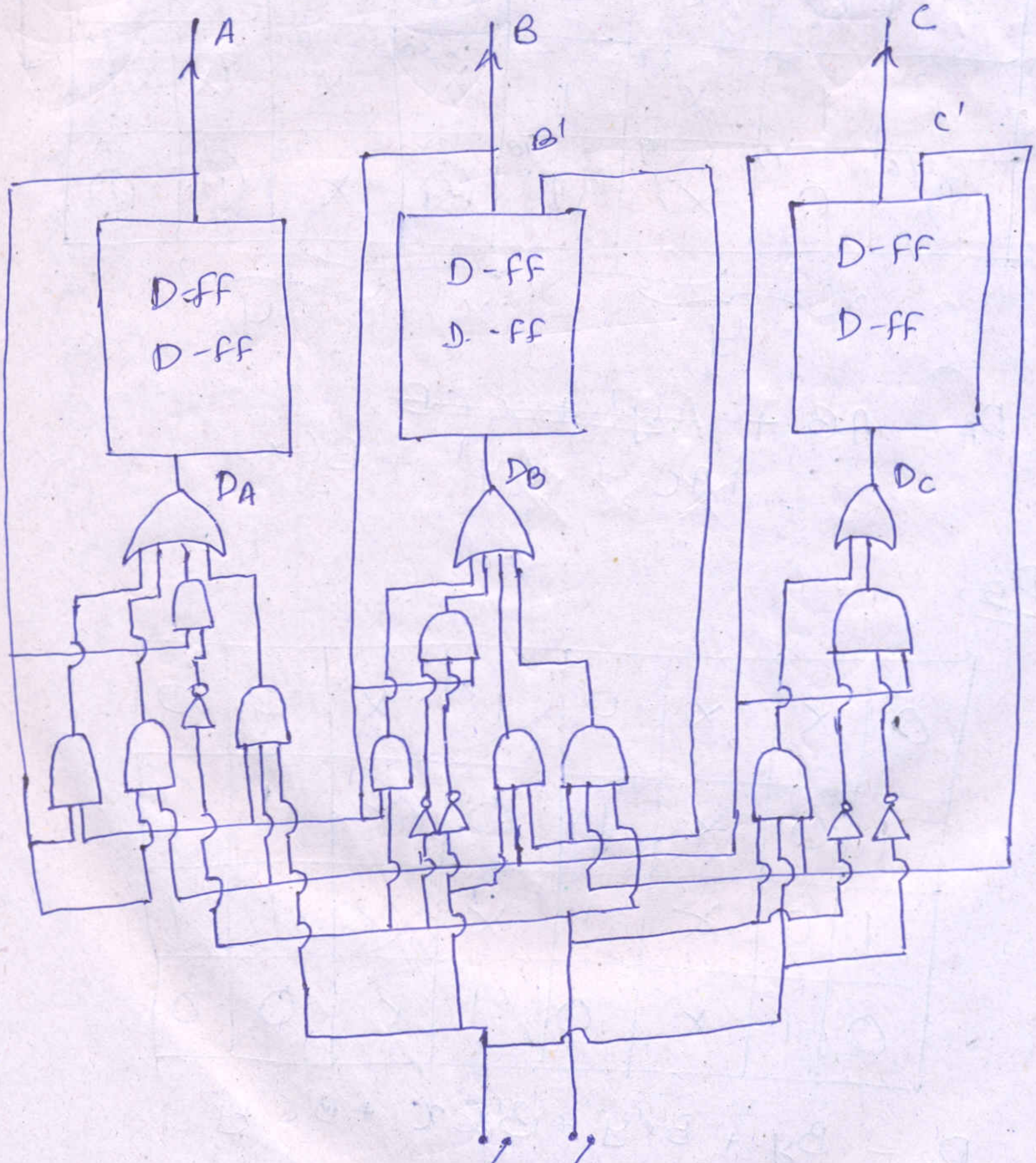
0	X	X	0	1	X	0	0
1	0	X	1	0	X	1	1
1	0	X	1	X	X	1	1
0	1	X	0	1	X	0	0

$$D_B = By + Bx'y' + B'Cx + B'C'y$$

D_C

0	X	X	1	0	X	0	1
0	1	X	1	0	X	0	1
0	1	X	1	X	X	0	1
0	1	X	1	0	X	0	1

$D_C = C'xy + Cx'y'$

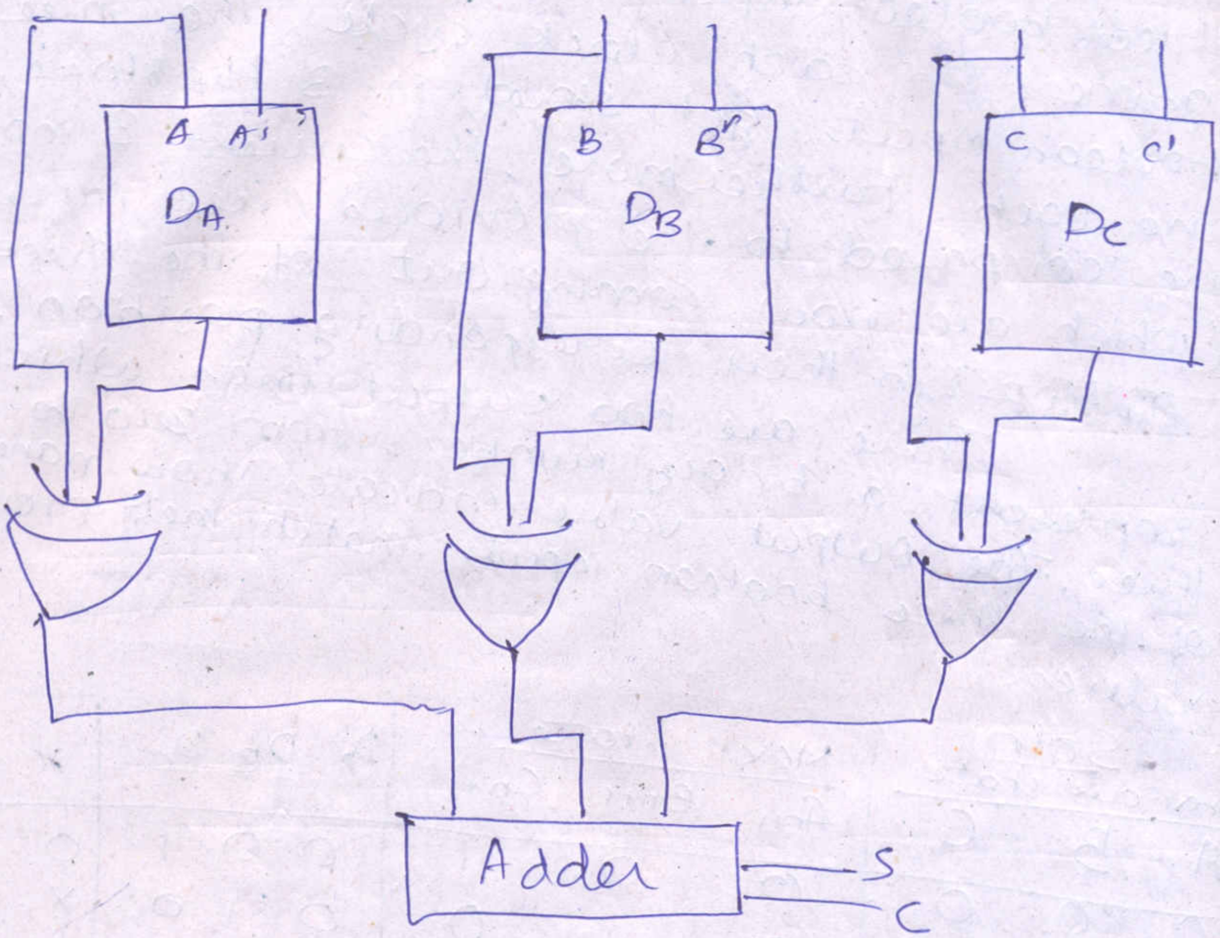


Q. Design a sequential circuit with 3 boolean inputs and a clock input. The three boolean inputs represent choices in a game. On each clock cycle, the three boolean inputs get stored in 3 flipflops, one each. Furthermore, the current 3 inputs are compared to the previous three inputs (which are now coming out of the three flipflops) in their corresponding position.

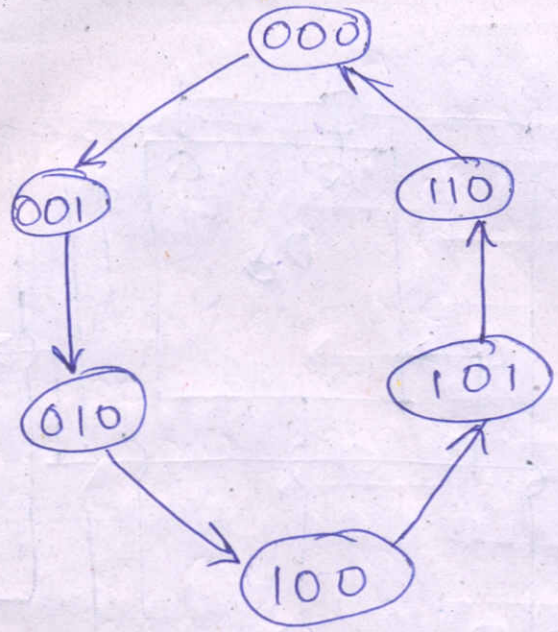
There are two output lines which represent a binary number from zero to three. The output value indicates how many of the three boolean inputs match their previous values.

Soln

Previous state			Next state							
A	B	C	A_{n+1}	B_{n+1}	C_{n+1}	D_A	D_B	D_C	x	y
0	0	0	0	0	1	0	0	1	0	0
0	0	1	0	1	0	0	1	0	X	X
0	1	0	0	1	1	0	1	1	0	1
0	1	1	1	0	0	1	0	0	X	X
1	0	0	1	0	1	1	0	1	1	0
1	0	1	1	1	0	1	1	0	X	X
1	1	0	1	1	1	1	1	1	1	1
1	1	1	0	0	0	0	0	0	X	X


A A'
DA
B B'
DB
C C'
DC
Adder
S
C

20/9/08



Q. State diagram of a sequential circuit is shown in figure. Design sequential circuit.

	A	B	C	A_{t+1}	B_{t+1}	C_{t+1}	D_A	D_B	D_C
0	0	0	0	0	0	1	0	0	1
1	0	0	1	0	1	0	0	1	0
2	0	1	0	1	0	0	1	0	0
3	0	1	1	x	x	x	x	x	x
4	1	0	0	1	0	1	1	0	1
5	1	0	1	1	1	0	1	1	0
6	1	1	0	0	0	0	0	0	0
7	1	1	1	x	x	x	x	x	x

For D_A

B

			x	1
A	1	1	x	

C

$D_A = AB' + BA'$

$= A \oplus B$

D_B

B

		1	x	
A		1	x	

C

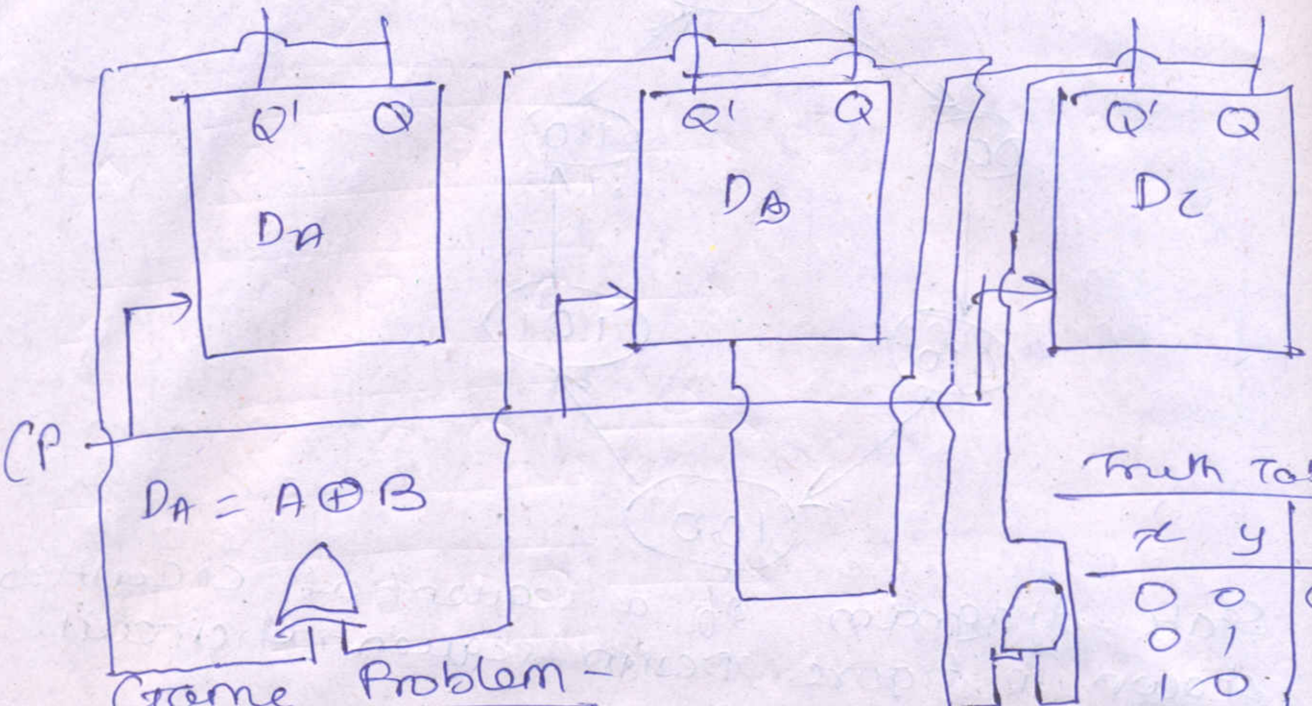
$D_B = C$

B

	1		x	
A	1		x	

C

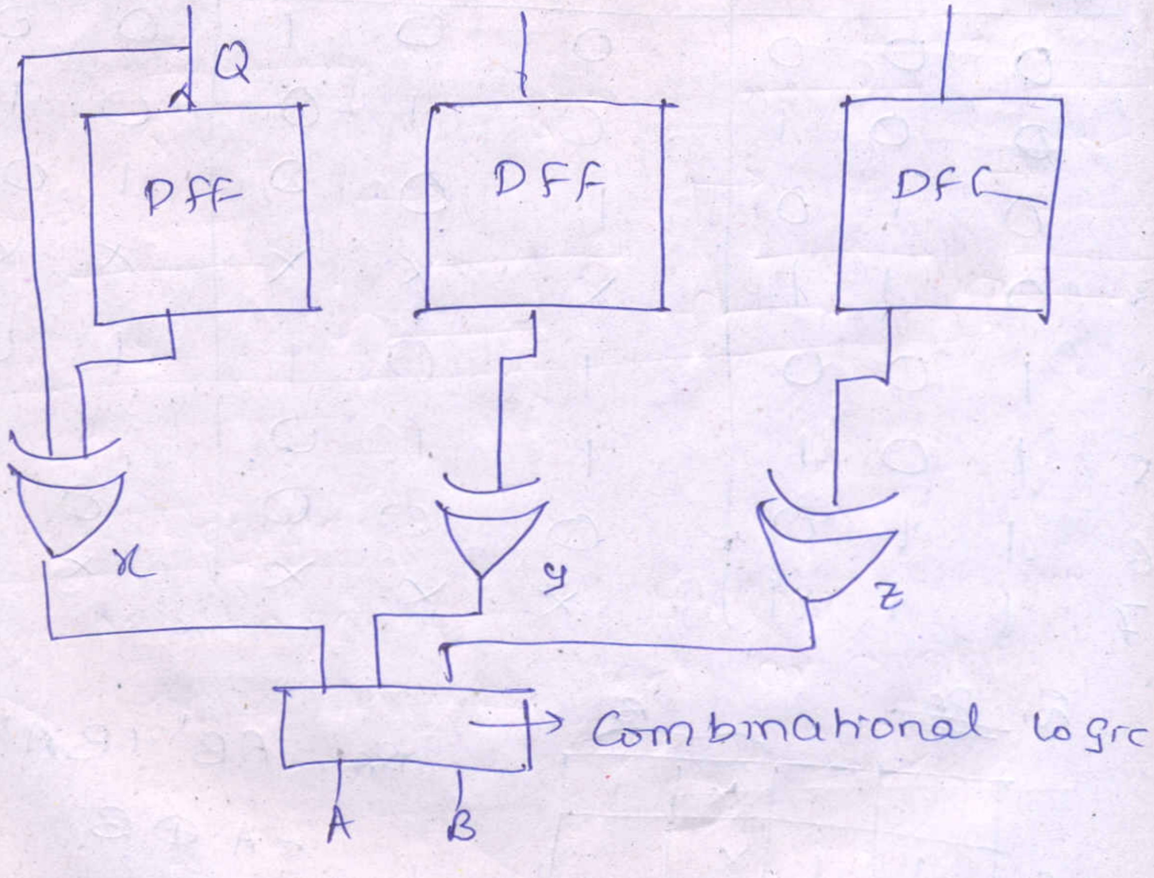
$D_C = B'C'$



Truth Table

x	y	z
0	0	0
0	1	1
1	0	1
1	1	0

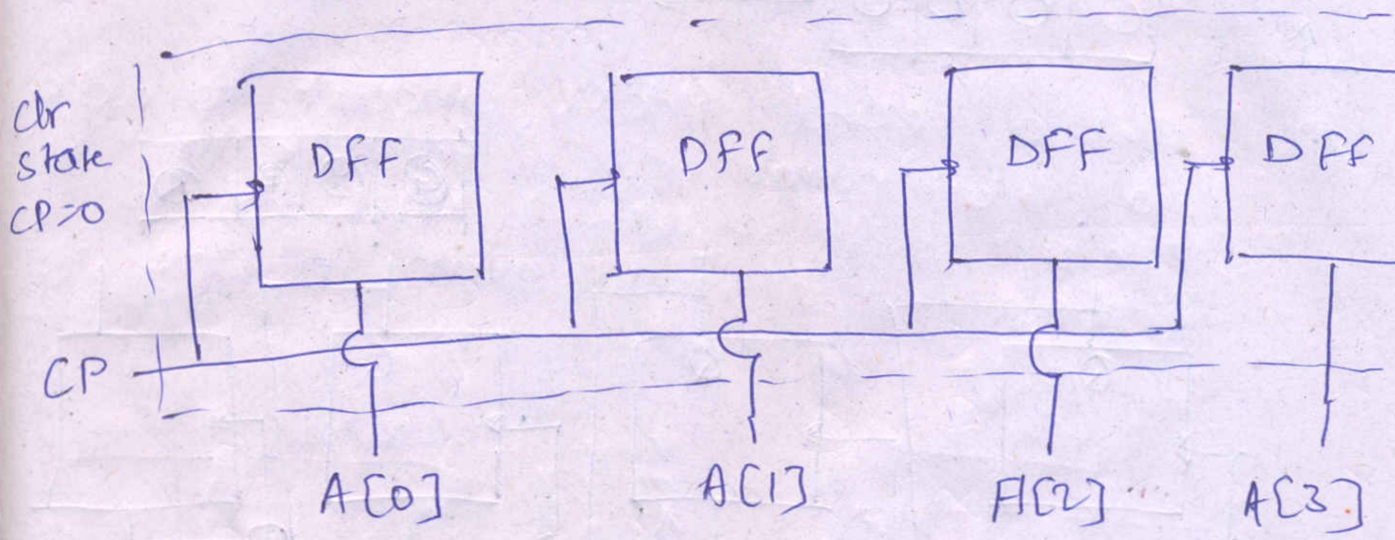
Game Problem -



x	y	z	A	B
0	0	0	1	1
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	0	0

Note - D-ff can be used as 1-bit storage value

Registers:-



Register is volatile in nature - once the power will be off, values are removed

1 byte → 8 bit.

$\boxed{1\ 0\ 1\ 1}$ → o/p will be like

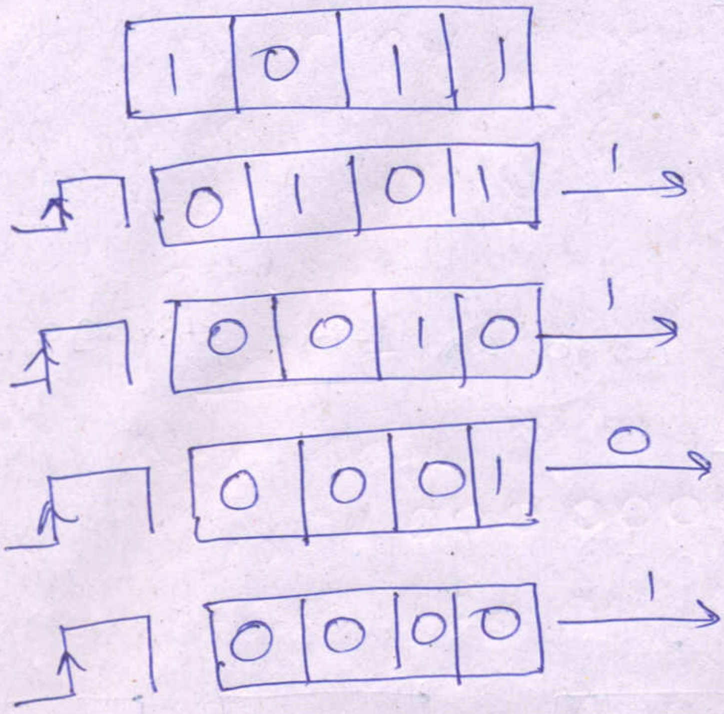
1

1

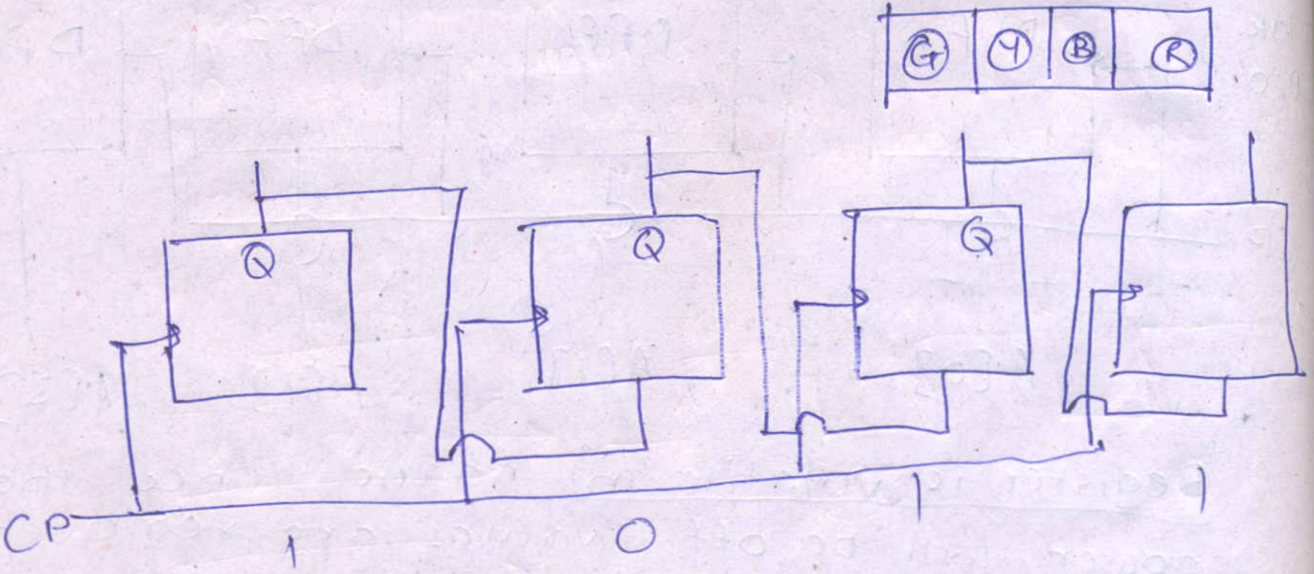
0

1

Output of one FF will be input of other.



By shifting

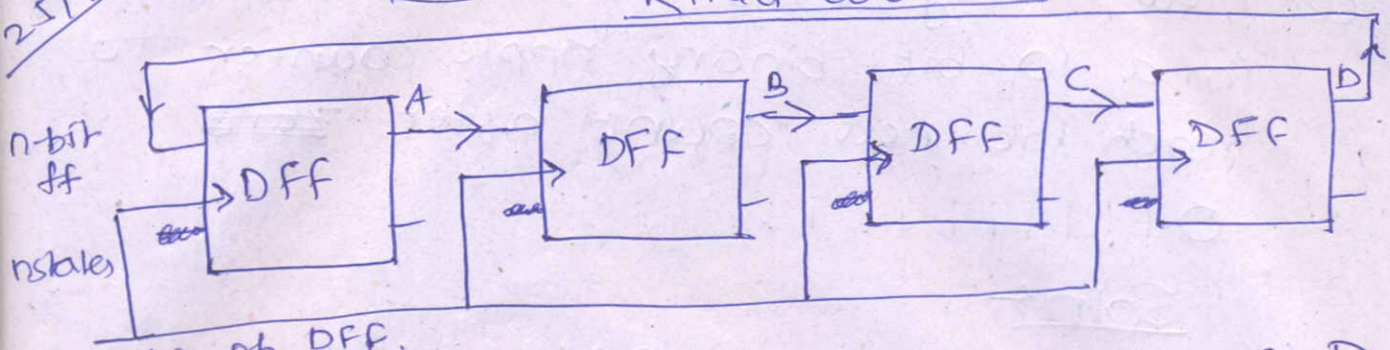


Shift Register -

Ux: Multiplication of Binary register.
shifting may be left or right.

25/9/08

4-bit shift Register - RING COUNTER -



n → no. of DFF.

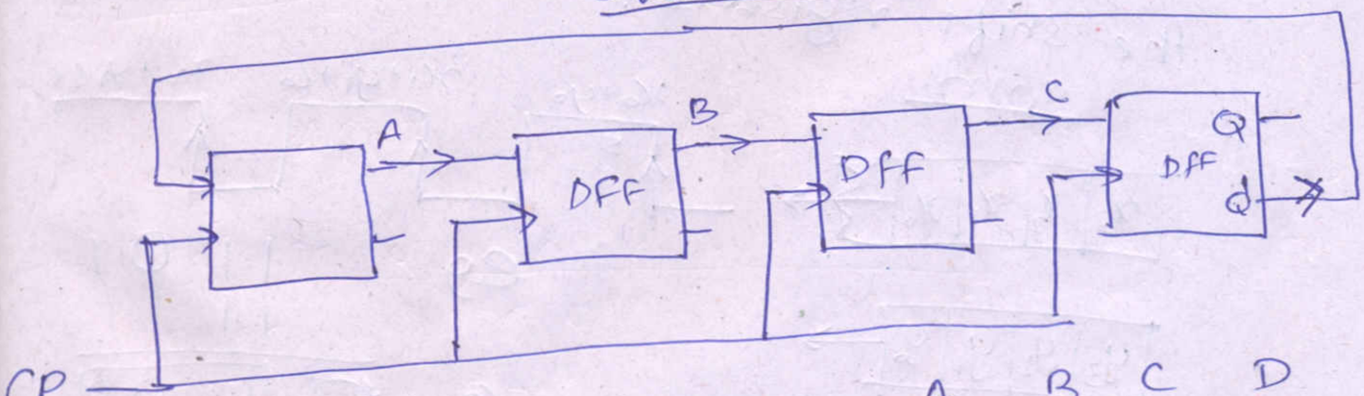
1 0 1 1

CP → 0 1 0 1

→ 0 0 1 0

A	B	C	D
1	0	0	0
0	1	0	0
0	0	1	0
0	0	0	1
1	0	0	0

To convert 'n' states to '2n' states use Switch tail Ring counter:-



A	B	C	D
1	0	0	0
1	1	0	0
1	1	1	0
1	1	1	1
0	1	1	1
0	0	1	1
0	0	0	1
0	0	0	0

'2n' states

Q. How many FF's must be complemented in a 10-bit binary ripple counter to reach the next count after ~~zero~~ 0111111111.

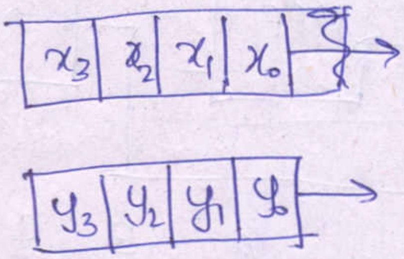
Soln:

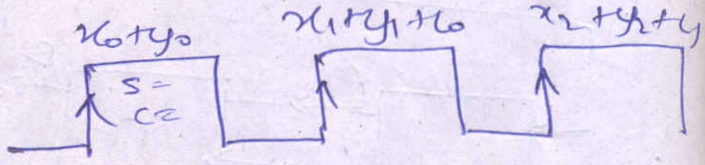
Next stage after 0111111111 is 1000000000

So, 10 FFs are required.

Q. Draw a serial Adder circuit to add two 4-bit numbers. (which are in two different shift registers). Add the two numbers and store the sum in one of the shift registers.

Soln





eg. 1 1 0 1
1 1 1 1

Carry ← 1 $\boxed{1100}$ ↓ sum