SEQUENTIAL CIRCUITS – II

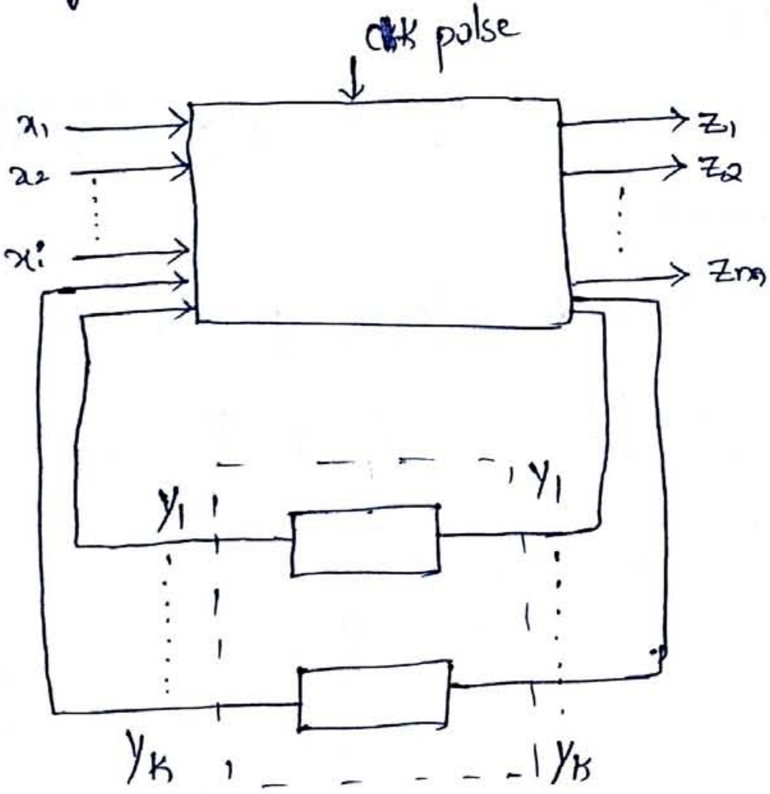
Finite State Machines –

— A Sequential circuit is referred to as fsm. It is an abstract model that describes the synchronous seq machine.

∵ In a seq ckt o/p depends on present i/p as well as on past o/ps. Since a m/c might have infinite no. of past o/p's, it would need an infinite capacity for storing them. Since it is impossible to implement m/c which have infinite storage capabilities, we consider only finite state machines.

— Fsm are seq ckts whose past histories can affect their future behaviour in only a finite no. of ways.

— The classes of input histories are referred to as the finite no. of memory devices.



Capabilities and limitations of fsm:-

1. periodic Seq of finite states:- With n-state m/c, we can generate a periodic Seq of n states or Smaller than n states.

Ex:- In a 6-state m/c, we can have a max periodic Seq as 0, 1, 2, 3, 4, 5, 0, 1 ...

2. No. infinite Sequence:-

Consider an infinite Seq Such that o/p is 1 when and only when no. of i/ps received So far is Equal to $P(P+1)/2$ for $P = 1, 2, 3, \ldots$ i.e, desired input - output seq has following form:

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

Such an infinite Seq Cannot be produced by a fsm.

3. limited Memory:-

fsm has limited memory and due to limited memory it cannot produce Certain o/p's

*→ finite state machines are of two types: They differ in the way the o/p is generated.

1. mealy:- In this the o/p is a fn of p.s and present input.
2. moore:- In this, the o/p is a fn of present state only.

Mathematical Representation of Synchronous Seq m/c: ②

— W.K.T, next state of a Seq m/c depends upon the P.S and present input.

The relation b/w the present state $s(t)$, present input $x(t)$, and next state $s(t+1)$ can be given as

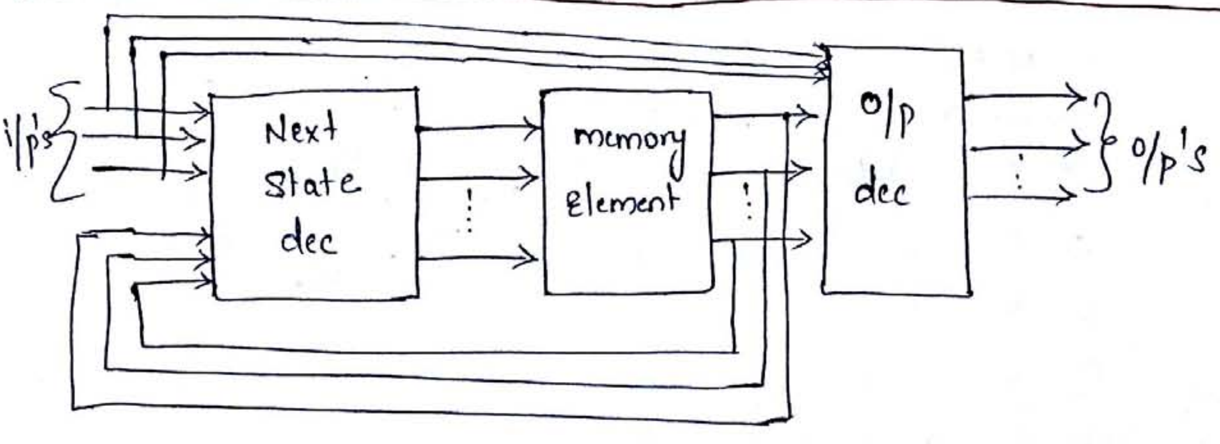
$$S(t+1) = f\{s(t), x(t)\}.$$

The value of o/p $z(t)$ can be given as

$$z(t) = g\{s(t), x(t)\} \quad \text{for mealy model}$$

$$z(t) = g\{s(t)\} \quad \text{for moore model.}$$

moore machine	mealy machine
1. Its o/p is a fn of P.S only $z(t) = g\{s(t)\}$	Its o/p is a fn of P.S as well as present i/p. $z(t) = g\{s(t), x(t)\}$
2. Input changes does not affect the o/p	2. I/p changes may affect the o/p of ckt
3. It requires more no. of states for implementing same fn	3. It requires less no. of status for implementing same fn.



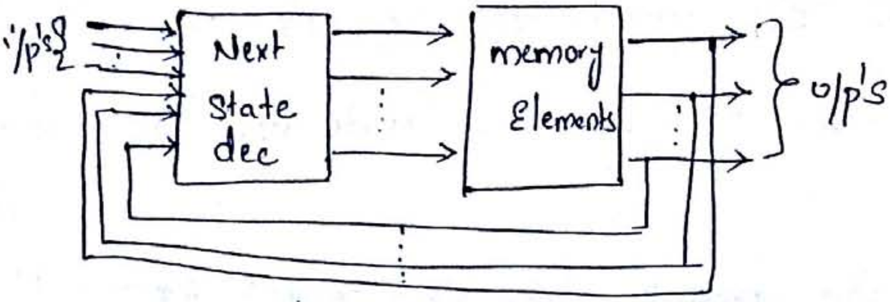
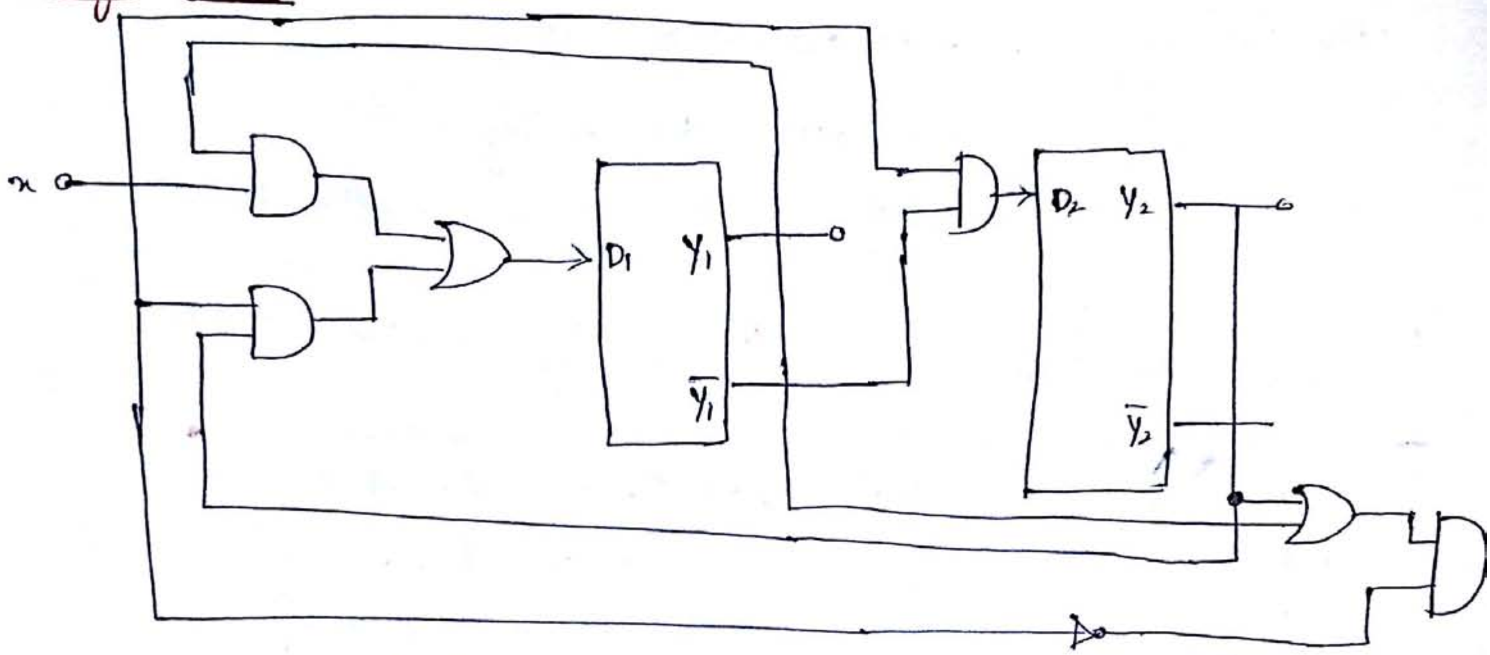


fig: moore model.

Mealy model:-



$$y_1(t+1) = x(t) \cdot y_1(t) + y_2(t) \cdot x(t)$$

$$y_2(t+1) = x(t) \cdot \bar{y}_1(t)$$

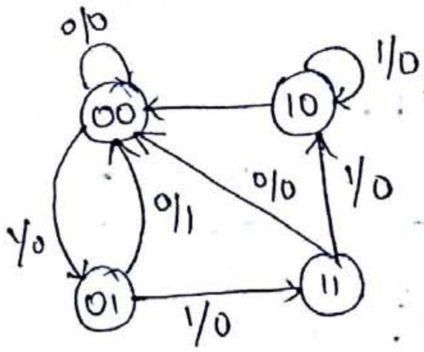
and o/p $z(t) = \bar{x}(y_2 + y_1)$

$$\Rightarrow y_1(t+1) = Y_1 = y_1 x_1 + y_2 x$$

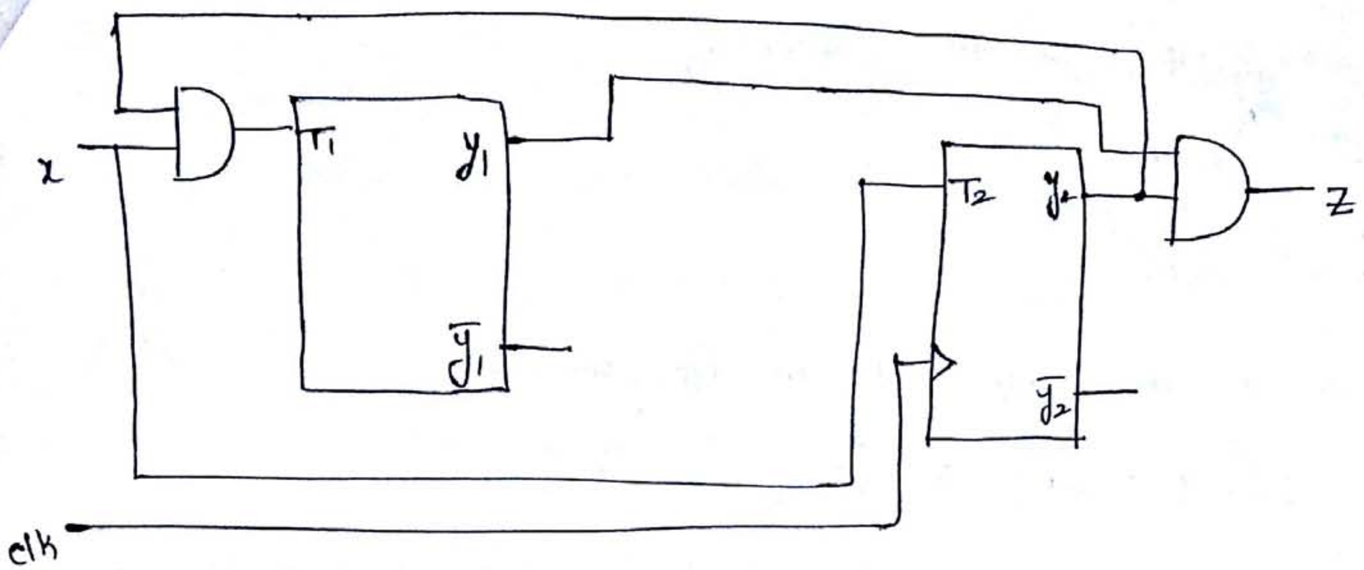
$$y_2(t+1) = Y_2 = \bar{y}_1 \cdot x$$

$$z = (y_2 + y_1)\bar{x}$$

P.S		N.S				O/P	
		x=0		x=1		x=0	x=1
y_1	y_2	Y_1	Y_2	Y_1	Y_2	x=0 z	x=1 z
0	0	0	0	0	1	0	0
0	1	0	0	1	1	1	0
1	0	0	0	1	0	1	0
1	1	0	0	1	0	1	0



Moore model:-



The C.E for T-F/F is

$$Q(t+1) = T\bar{Q} + \bar{T}Q = T \oplus Q$$

$$y_1(t+1) = x(t) \cdot y_2(t)$$

$$y_2(t) = x(t)$$

$$y_1(t+1) = Y_1 = (y_2 x) \oplus y_1 = (\overline{y_2 x}) y_1 + (y_2 x)\bar{y}_1$$

$$= y_1(\bar{y}_2 + \bar{x}) + y_2 x \cdot \bar{y}_1$$

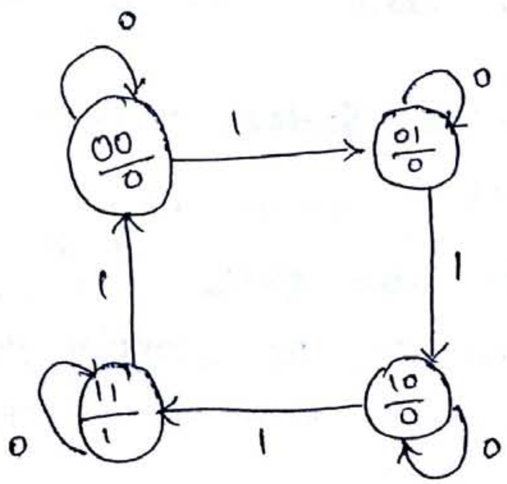
$$= y_1\bar{y}_2 + y_1\bar{x} + \bar{y}_1 y_2 x$$

$$y_2(t+1) = Y_2 = x \oplus y_2$$

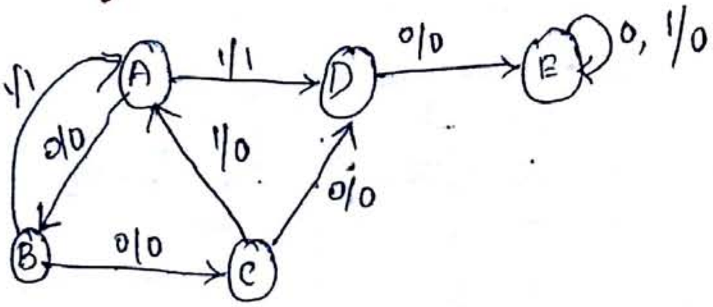
$$= x \cdot \bar{y}_2 + \bar{x} y_2$$

$$Z = y_1 y_2$$

P.S		N.S				O/P
		x=0		x=1		
y_1	y_2	Y_1	Y_2	Y_1	Y_2	Z
0	0	0	0	0	1	0
0	1	0	1	1	0	0
1	0	1	0	1	1	0
1	1	1	1	0	0	1



FSM Definitions :-



- It is a 5 state m/c with one i/p, one o/p.

$S\{A, B, C, D, E\}$, $I = \{0, 1\}$, $O = \{0, 1\}$.

Successor :- from fig, we say that when P.S is A and i/p is 1, next state D. In other words this condition is specified as D is 1- Successor of A.

Similarly we can say that A is 1 Successor of B&C,

D is 11 Successor of B&C

C is 00 Successor of A, B is 000 Successor of A.

E is 10 Successor of A or 0000 Successor of A.

and Soon.

In general, we can say that if an i/p Seq X takes a m/c from State S_i to S_j then S_j Said to be X Successor of S_i.

Terminal State :- A State is said to be terminal when

(i) there are no outgoing arcs which starts from it and terminate in other states.

(ii) There are no incoming arcs which starts from other states and terminate in it. E is called terminal state (in above fig)

State Equivalence and Machine Minimisation:-

State Equivalence theorem:-

It states that two states S_1 and S_2 are Equivalent if for Every possible i/p Seq. applied, the m/c goes to Same next State and generates same o/p.

i.e if $S_1(t+1) = S_2(t+1)$ and $Z_1 = Z_2$ then $S_1 = S_2$

→ Distinguishable states and Distinguishing Sequences:-

Two States S_A, S_B of Seq m/c are distinguishable if and only if there Exists at least one finite i/p Seq which when applied to Sequential m/c Causes different o/p Sequence depending on whether S_A or S_B is initial state. The Seq which distinguishes these states is called distinguishing Seq of pair (S_A, S_B)

E.g

P.S	NS, Z (X=0)	NS, Z (X=1)
A	C, 0	F, 0
B	D, 1	F, 0
C	E, 0	B, 0
D	B, 1	E, 0
E	D, 0	B, 0
F	D, 1	B, 0

→ Consider States A & B. When i/p x=0, o/p's are 0,1 respectively and therefore states A & B are called 1- distinguishable.

Now consider A & E

$x = 0 \begin{cases} A \longrightarrow C, 0 \text{ and } E \rightarrow D, 0 \text{ . o/p's Same} \\ C \longrightarrow E, 0 \text{ and } D \longrightarrow B, 1 \text{ o/p are different.} \end{cases}$

Here o/p's are different after (two) 2. State transitions and hence

A & E are 2-distinguishable.

→ Again consider states A and C

$x = 0 \begin{cases} A \rightarrow C, 0 \text{ and } C \rightarrow E, 0 \text{ . o/p's Same} \\ C \rightarrow E, 0 \text{ and } E \rightarrow D, 0 \text{ o/p's Same.} \\ E \rightarrow D, 0 \text{ and } D \rightarrow B, 1 \text{ o/p's are different.} \end{cases}$

∴ A & C are 3-distinguishable.

→ In general if two states have a distinguishable Sequence of length k, the states are said to be k-distinguishable

∴ States that are not k distinguishable are said to be k-Equivalent.

Minimization of Completely Specified m/c using portion technique:-

Eg:-①

P.S	NS, Z x = 0	NS, Z x = 1
A	E, 0	D, 1
B	F, 0	D, 0
C	E, 0	B, 1
D	F, 0	B, 0
E	C, 0	F, 1
F	B, 0	C, 0

1:- partition the states into Subsets Such that all states in the Same Subsets are 1- Equivalent.

i.e P_1 Can be obtained by placing those, states having the Same o/p's under all i/p's in Same block.

(i) (A, C, E) : their o/p's are under 0 & 1, i/p's are 0, 1 Respectively

(ii) (B, D, F) : o/p's are under 0 & 0, i/ps are 0, 1 Respectively.

$\therefore P_1 = (A, C, E)\ (B, D, F)$

step2: partition the States into Subsets Such that all the States in the Subsets are 2- Equivalent

a) The 0 & 1 - Successor of (A, C, E) are (C, E) and (B, D, F) Respectively. Since both are Contained in common blocks of P_1, the States (A, C, E) are Said to be 2- Equivalents.

$\therefore$ (A, C, E) Constitutes a block in P_2.

b) The 0- Successor of (B, D, F) is (B, F) which Contain in P_1, however 1 - Successor of (B, D, F) is (D, B, C) in which (D, B) and (C) are not contained in Single block of P_1

$\therefore$ (B, D, F) must be Split into (BD) & (F)

$P_2 = (A, C, E),\ (B, D)\ (F)$

step3:- partition the states into Subsets Such that all States in the Same Subsets are 3 - Equivalent.

We can split block (A, C, E) of P_2 into (A, C) and (E) since 1- Successor of (A, C, E) is (D, B, F) which is not in single block of P_2.

$P_3 = (A, C)\ (E)\ (B, D)\ (F)$

further partitioning of states is not possible because 0 & 1 - successors of blocks in P_3 are in common block of P_3.

- The states in common block of P_3 are Equivalent

∴ states A=C and B=D.

P.S	NS,Z	
	x=0	x=1
A	E,0	B,1
B	F,0	B,0
C	A,0	F,1
D	B,0	A,0

Eg2: Determine the minimized state table.

P.S	NS,Z	
	x=0	x=1
A	C,0	F,0
B	D,1	F,0
C	E,0	B,0
D	B,1	E,0
E	D,0	B,0
F	D,1	B,0

Step1: 1- Equivalent
under x=0 o/p's are same for ACE and BDF.

Step2: 2- Equivalent
0- Successor of (A,C,E) are (C,E,D). They are in different blocks of P_1. So the blocks (A,C,E) must split into (A,C) and (E)

1 - Successor of (B, D, F) are (F, E, B): They are in different blocks of P_1, ⑥

∴ (B, D, F) must split into (B, F) and (D)

∴ P_2 = (A, C) (E) (B, F) (D)

→ 1- Successor of (A, C, E) are (F, B, B). They are in Same block of P_1

→ 0- Successor of (B, D, F) are (D, B, D). They are in Same block are Equivalent. ~~for~~ this consider states which are 2-Equivalent. of P_1.

→ So no partitioning is possible.

Step3:- partitioning states into Subsets such that all states in same block are Equivalent. for this consider states which are 2-Equiv-alent.

→ 0- Successor of (A, C) are (C, E). They are in different blocks in P_2. So partition (A, C) into (A) and (C)

→ 1- Successor of (A, C) are (F, B). They are in Same block of P_2.

∴ P_3 = (A) (E) (C) (B, F) (D)

Further partitioning is not possible because 0 & 1 - Successor of (B, F) P_3 (D, D) & (B, F) are in Same block of P_3

∴ B = F

	NS, Z x=0	x=1
A	C, 0	F, 0
B	D, 1	F, 0
C	E, 0	B, 0
D	B, 1	E, 0
E	D, 0	B, 0

Eg 3:-

PS	NS, Z x=0	NS, Z x=1
A	B,0	E,0
B	E,0	D,0
C	D,1	A,0
D	C,1	E,0
E	B,0	D,0

Step 1:- under x=0 o/ps are same for

(A,B,E) (C,D)

$\therefore P_1 = (A,B,E)(C,D)$

Step 2:- 0- Successor of (A,B,E) are (B,E,D) ~~different~~. Same block so no partition.

1- Successor of (A,B,E) are (E,D,D). different. block so partition. (A) (B,E) $\Rightarrow \therefore P_2 = (A)(B,E)(C,D)$

Step 3:- 0- Successor of (C,D) is (D,C) → no. partition ~~different~~.

1- Successor of (C,D) is (A,E) → different in P_2 So partition.

$\therefore P_3 = (A)(B,E)(C),(D).$

Step 4:- 0 & 1 - Successor of (B,E) are (E,B) & (D,D) → so no partition

$\therefore P_4 = (A)(B,E)(C)(D) \Rightarrow B = E$

	NS, Z x=0	NS, Z x=1
A	B,0	B,0
B	B,0	D,0
C	D,1	A,0
D		

Q5: for the m/c below obtain

a) Corresponding reduced m/c table.

b) find a min length that distinguishable state B from state C.

P.S	NS, Z x=0	NS, Z x=1
A	A, 0	E, 1
B	A, 1	E, 1
C	B, 1	F, 1
D	B, 1	F, 1
E	C, 0	G, 0
F	C, 0	G, 0
G	D, 0	H, 0
H	D, 0	H, 0

Step1:- P_1 = (A) (B, C, D) (E, F, G, H)

Step2: 0- Successor of (B, C, D) are (A, B, B)

$\Rightarrow$ (A) (B) (C, D) (E, F, G, H)

Step3:- 0&1 - Successor of (C, D) are (B, B) & (F, F)

0&1 - Successor of (E, F, G, H) are (C, C, D, D) & (G, G, H, H)

no partition

$\therefore$ P_3 = (A)(B) (C, D) (E, F, G, H)

C = D ✓ E = F = G = H

P.S	NS, Z x=0	NS, Z x=1
A	A, 0	E, 1
B	A, 1	E, 1
C	B, 1	E, 1
D	C, 0	E, 0

Eg4: Find reduced m/c table.

P.S	NS, Z X=0	X=1
A	F, 0	B, 0
B	D, 0	C, 0
C	F, 0	E, 0
D	G, 1	A, 0
E	D, 0	C, 0
F	F, 1	B, 1
G	G, 0	H, 0
H	G, 1	A, 0

Step1:- (A, B, C, E, G) (D, H), (F)

Step2:- 1- Successor of (A, B, C, E, G) is (B, C, E, C, H) ⇒ (A, B, C, E) (G)

0- Successor of (A, B, C, E, G) is (F, D, F, D, G)

are in different blocks. So partition (A, B, C, E, G) into (A, C) (B, E) and (G)

∴ P_2 = (A, C) (B, E) (G) (D, H) (F)

Step3:- 0- Successor of (A, C) is (F, F)
1- Successor of (A, C) is (B, E) } no partition

0- Successor of (B, E) is (D, D)
1- Successor of (B, E) is (C, C) } no partition

0- Successor of (D, H) is (G, G)
1- Successor of (D, H) is (A, A) } no partition.

∴ P_3 = (A, C) (B, E) (D, H) (G) (F)

A = C, B = E and D = H

D C, 1 D, 0

⑤ min length of Sequence

$$x=0 \begin{cases} B \to A,1 & C \to B,1 & \text{— o/p's are Same} \\ A \to A,0 & B \to A,1 & \text{— o/p are different.} \end{cases}$$

$$x=1 \begin{cases} B \to E,1 & C \to F,1 & \text{o/p Same} \\ E \to G,1 & F \to G,0 & \text{o/p Same} \\ G \to H,0 & G \to H,0 & \text{o/p different.} \end{cases}$$

∴ min length of Sequence that distinguishes State B from State C is 2.

→ Eg6

P.S	N.S, Z x=0	x=1
q_1	$q_2,0$	$q_8,1$
q_2	$q_6,0$	$q_4,1$
q_3	$q_4,1$	$q_5,1$
q_4	$q_3,1$	$q_6,1$
q_5	$q_4,1$	$q_5,1$
q_6	$q_3,0$	$q_5,1$
q_7	$q_3,0$	$q_4,1$
q_8	$q_3,1$	$q_1,0$

Sol $P_1 = (q_1, q_2) q_5, q_6, q_7) (q_3, q_4, q_8)$

$P_2 = (q_1, q_2) (q_5, q_6) (q_7) (q_3, q_4, q_8)$

$P_3 = (q_1)(q_2) (q_5, q_6) (q_7) (q_3, q_4) (q_8)$

$P_4 = (q_1) (q_2) (q_5, q_6) (q_7) (q_3, q_4) (q_8)$

∴ min length from q_1 to q_2 is 3 $x=0 \rightarrow$ (3 states), $x=1 \longrightarrow$ (4 - states)

Simplification of incompletely Specified m/cs :-

- In designing Combinational logic ckts, we often Encountered Situations where the T.T was incompletely Specified, which resulted in don't care terms.
- I'ly in Sequential ckts the o/p's and state transitions are not completely Specified.
- When reducing incompletely Specified state table we use term state Compatability instead of state Equivalence.

** → States Si & Sj are Said to be Compatible if and only if for every i/p Sequence that affects two states, the Same o/p Sequence occurs whenever both the o/ps are Specified and regardless of whether Si or Sj is initial state.

Merger chart methods :-

→ merger graphs :- It is a state reducing tool used to reduce states in incompletely Specified m/c.

→ It is defined as

i) Each State in state table is represented by Vertex in merger graph. So it contains Same no. of vertices as the state table contains states.

ii) each Compatible state pair is indicated by an unbroken line drawn b/n two State Vertices.

(iii) Every potentially compatible state pair with non-conflicting o/p's but with different next states is Connected by broken line. The implied states are written in line break b/n the two potential Compatible states.

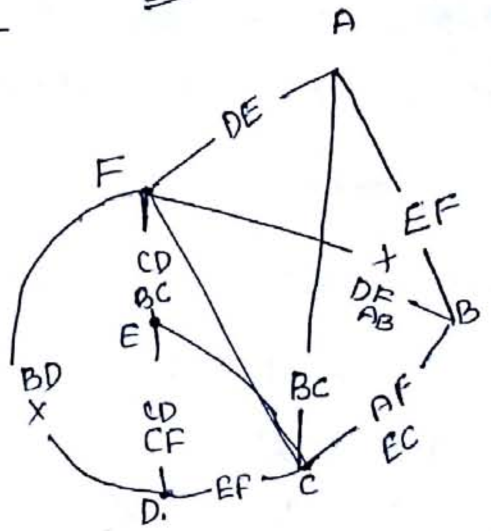
(iv) If two states are incompatible no Connecting line.

→

Draw the merger graph and obtain set of maximal compatibilities (9) for incompletely specified sequential m/c.

P.S	N.S	Z
	I_1	I_2
A	E, 0	B, 0
B	F, 0	A, 0
C	E, -	C, 0
D	F, 1	D, 0
E	C, 1	C, 0
F	D, -	B, 0

Sol



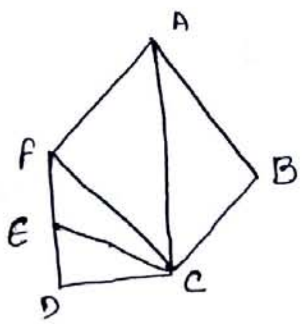
→ In the graph B & D are not connected so pair (B, D) is not compatible.

→ (D, F) is compatible only if implied pair is compatible since (B, D) is not compatible So (D, F) is not compatible.

→ (B, F) is compatible only if implied pairs (D, F) & (A, B) are compatible. Since (D, F) is not compatible, (B, F) is also not compatible.

→ Removing broken lines corresponding to non-compatible pairs i.e (D, F) & (B, F) and replacing broken lines of other pairs by unbroken lines.

→ merger graph is redrawn as



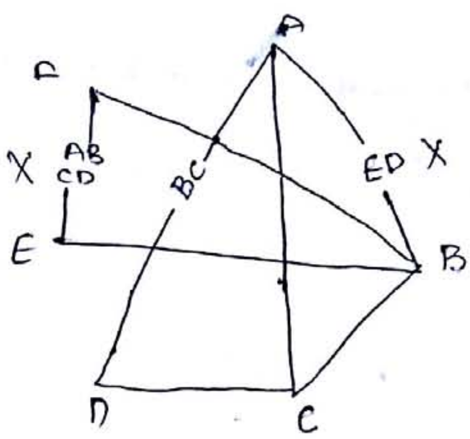
→ After checking all possibilities of incompatibility, merger graph the following a compatible pairs.

(A,B), (A,C), (A,F), (B,C), (C,D), (C,E), (C,F), (D,E), (E,F)

→ From merger graph, we can find the maximal compatibles corresponding to triangles as (A,B,C), (A,C,F), (C,D,E), (C,E,F)

→ Draw the merger graph and obtain the set of maximal compatibles for incompletely specified m/c.

P.S	NS, Z			
	I_1	I_2	I_3	I_4
A	–	E,1	B,1	–
B	–	D,1	–	F,1
C	F,1	–	–	–
D	–	–	C,1	–
E	C,0	–	A,0	F,1
F	D,0	A,1	B,0	–



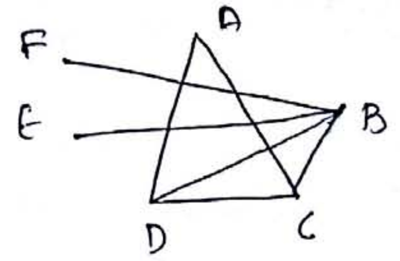
→ AE, AF are not related, so not compatible

→ CE, CF are not related, so not compatible

→ DE are not related, so not compatible

So AB also not compatible
EF also not compatible.

∴ merger graph as follows.



After checking all possibilities of incompatibility, merger graph gives 7 compatibility pairs (A,C), (A,D), (B,C), (B,D), (B,E), (C,D), (B,F)

→ Set of maximal compatibles for this is (A,C,D), (B,C,D), (B,E), (B,F)

→ merger table :-

It is also called paull-unger method or implication chart method.

— This more convenient than merger graph to find compatible pairs and implicants while performing state reduction of m/c having large no. of states.

→ obtain the set of maximal compatibles for sequential m/c whose state table is shown using merger table.

P.S	Ns, Z I_1	I_2	I_3	I_4
A	—	C, 1	E, 1	B, 1
B	E, 0	—	—	—
C	F, 0	F, 1	—	—
D	—	—	B, 1	—
E	—	F, 0	A, 0	D, 1
F	C, 0	—	B, 0	C, 1

sol

	A	B	C	D	E
B	✓				
C	CF	EF			
D	BE	✓	✓		
E	X	✓	X	X	
F	X	X	✓ CF	X	AB CD

X → In the cells corresponding to pairs (A,F), (A,E), (C,E), (D,E) and (D,F) because they are non-compatible.

✓ → is put in cells corresponding to pairs (A,B), (B,D), (B,E), (C,D) (C,F) because they are compatible.

→ In other words cells implied pairs are written since pair (C,E). is not compatible put a X in cell (B,F) which has this implied pair

→ merger tables give following set of maximal compatibles

column E : (E,F)

D: (E,F)

C: (C,D)(C,F) (E,F)

B: (B,C,D) (B,E) (C,F) (E,F)

A : (A,B,C,D) (B,E) (C,F) (E,F)

→ Right most column is E. It indicates that E is compatible with F resulting in a compatible pair (E,F).

→ Column D indicates no compatible pair. So at column D we have only (E,F)

→ Column C indicates C is compatible with D&F. So add new pairs (C,D), (C,F) (E,F).

Column B has 3 compatibilities. Since it is compatible with both C and new compatible pair (B,E). So at column B we have (B, C, D), (B, E), (C, F), (E, F) (11)

→ In Column A, state A has Compatibility with states (B, C, D). Since (B, C, D) is already grouped, we can form a bigger group (A, B, C, D). So at column A we have (A, B, C, D) (B, E) (C, F) (E, F) as the Set of maximal Compatibles.

→ (2) obtain Set of maximal Compatibles for state table given using merger table.

P.S	NS, Z 00	01	10	11
A	B, –	D, –	–	C, –
B	F, –	I, –	–	–
C	–	–	G, –	H, –
D	B, –	A, –	F, –	E, –
E	–	–	–	F, –
F	A, 0	–	B, –	–, 1
G	E, 1	B, –	–	–
H	E, –	–	–	A, 0
I	E, –	C, –	–	–

Sol

B	BF DI							
C	CH	✓						
D	CE ×	BF AI ×	FG EH ×					
E	CF	✓	FH ×	EF				
F	AB	AF	BG	AB BF	✓			
G	BE BD ×	EF BI	✓	BE AB	✓	×		
H	BE AC	EF	AH	BE AE	AF	×	✓	
I	× BE CD	EF CI	✓	BE AC	✓	AE	BC	✓
	A	B	C	D	E	F	G	H

The maximal compatibles are obtained as follows column

H : (H, I)

G : (G, H, I)

F : (F, I) (G, H, I)

E : (E, G, H, I) (E, F, I)

D : (D, E, G, H, I) (D, E, F, I)

C : (D, E, G, H, I) (C, F) (C, G) (C, H) (C, I) (D, E, F, I)

B : (D, E, G, H, I) (B, C, F) (B, C, H) (B, C, G) (B, E, F) (D, F) (B, C, I), (D, E, F, I)

A : (D, E, G, H, I) (B, C, I) (A, B, C, H) (B, C, G) (A, B, C, F) (D, E, F, I)

Eg3: obtain the set of maximal compatibles for the sequential m/c whose state table is given below using

a) merger table b) merger graph.

P.S	NS, Z I_1	I_2	I_3
A	C, 0	E, 1	—
B	C, 0	E, —	—
C	B, —	E, 0	A, —
D	B, 0	C, 0	E, —
E	—	E, 0	A, —

Solution

B	✓			
C	X	CE		
D	X	BC CE	AE X	
E	X	✓	✓	AE X CE
	A	B	C	D

from the table we see that states (A, E) are not compatible. So cross all cells which contain AE. So states (D, E) and (C, D) are also not compatible

∴ maximal compatible pairs are

Column D: Nil

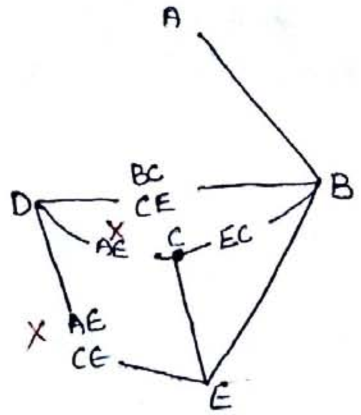
C: (C, E)

B: (B, C, E) (B, D)

A: (A, B) (B, C, E) (B, D)

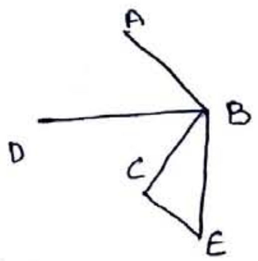
∴ Set of maximal compatibles are (A, B) (B, C, E) (B, D)

b) Merger graph



A is not connected C, D, E so AC, AD, AE not compatible => CD, DE, also not compatible.

∴ merger graph is as follows.



Set of compatible pairs are: (A,B) (B,C) (B,D) (B,E) (C,E)

= (B,C,E) (A,B) (B,D)

Eg4: Obtain set of maximal compatibles for seq m/c whose state table is given using a) merger table b) merger chart.

P.S	NS, Z			
	00	01	11	10
A	C, 0	—	C, 0	—
B	A, —	B, 1	D, —	—
C	—	E, 1	—, 0	D, 0
D	E, 0	—	F, 1	C, 0
E	F, 0	—	B, 1	A, 1
F	—	B, 1	—, 0	C, 0

  By — P.Jayaratham and B.kalpana (Asst. Prof.)

merger table

B	AC CD ~~X~~				
C	✓	BE X			
D	X	AE DF X	X		
E	CF BC X	AF BD X	X	EF BF AC X	
F	✓	✓	BE CD X	X	X
	A	B	C	D	E

→ from table (E,F) are not Compatible. So States (D,E) are also not Compatible.

→ If (C,F) are not Compatible then states (A,E) are not Compatible.

→ states (D,F) are not Compatibles. So (B,D) are also not Compatible.

→ So the compatible pair (A,C) (A,F) (B,F)

column E: Nil

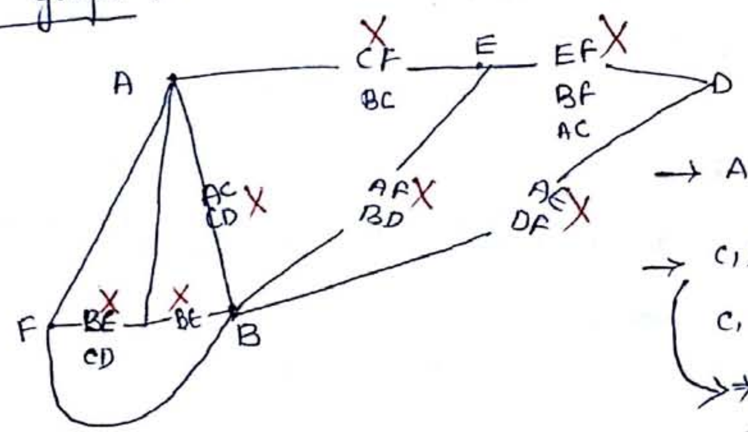
D: Nil

C: Nil

B: (B,F)

A: (A,C) (A,F) (B,F)

b) merger graph :-



→ A,D is not Connected

→ C,D not Connected

C, E " "

⇒ AB not Compatible

⇓

AC, CD not compatible

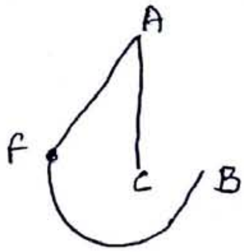
⇓

CF ⇒ AE ⇒ BD

⇓

BE ⇒ BC

Simplified merger graph:-



Compatible pairs are (A,C) (A,F) (B,F)

Minimal Cover table:-

Compatability graph:-

It is directed graph whose vertices correspond to all compatible pairs, and arc leads from vertex (S_i, S_j) to (S_p, S_q) only if (S_i, S_j) implies (S_p, S_q). It can be easily constructed from merger graph or merger table.

Subgraph of compatibility graph:-

Any part of compatibility graph is called subgraph of the compatibility graph.

closed Graph:-

A subgraph of a compatibility graph is said to be closed if for every vertex in the subgraph all outgoing arcs and terminating vertices also belong to subgraph. Each vertex in the subgraph belongs to one state.

minimal cover table:-

It is a table which consists of the states of minimal state machine.

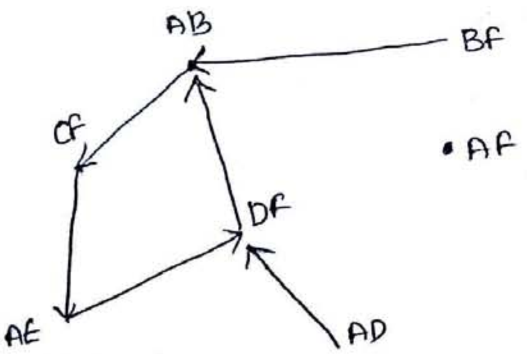
Construct the compatibility graph and obtain minimal cover table for sequential m/c described by state table.

P.S	NS, Z: I_1	NS, Z: I_2
A	—	F, 0
B	B, 0	C, 0
C	E, 0	A, 1
D	B, 0	D, 0
E	F, 1	D, 0
F	A, 0	—

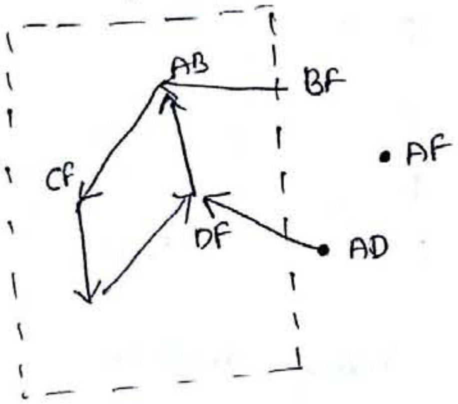
Soln merger table:-

	A	B	C	D	E
B	CF				
C	X	X			
D	DF	~~CD~~	X		
E	DF	X	X	X	
F	✓	AB	AE	AB	X

from the merger table, compatibility graph is constructed



Compatibility graph



Closed subgraph.

from closed Covering we obtain the minimal Cover table. There are 4 states (AB), (CF), (AE), (DF) in the subgraph.

Assign then

(AB) → P , (AE) → Q (CF) → R, (DF) → S

minimal cover table

P.S		NS, Z I_1	NS, Z I_2
AB	→ P	P, 0	R, 0
AE	→ Q	R/S, 1	S, 0
CF	→ R	Q, 0	P/Q, 1
DF	→ S	P, 0	S, 0

Converting from mealy m/c to Corresponding moore m/c

P.S	NS, Z x=0	NS, Z x=1
A	B, 0	E, 0
B	E, 0	D, 0
C	D, 1	A, 0
D	C, 1	E, 0
E	B, 0	D, 0

Sol Given mealy machine

In next state Column, identify the no. of o/ps associated with Each state.

A has o/p '0' ; B has o/p '0'

C has o/p '0' & '1' ; D has o/p '0' & '1' ;

E has o/p '0'

i.e D&C has got two o/p 0 & 1. So we have two states C_0 and C_1 for C & D_0, D_1 for D.

moore machine o/p depends only on P.S. So Equivalent moore m/c as

P.S	NS $x=0$	$x=1$	o/p
A	B, 0	E	0
B	E	D	0
C_0	D_1	A	0
C_1	D_1	A	1
D_0	C_1	E	0
D_1	C_1	E	1
E	B	D_0	0

→ C_0 has to be written if o/p is '0'
→ C_1 has to be written if o/p is '1'
→ D_0 has to be written if o/p is '0'
→ D_1 has to be written if o/p is '1'