Minimization of Finite Automata:- It is always describe to use the least amount of Computer memory whether it is length of code or programming data (stored in variables). If we wish to simulate a particular DFA by using some programming language. we always try to deal with the DFA with minimum no of states.

Two finite automata M and M' are said to be equivalent if and only if both accept the same language or same set of strings. If automaton M consist n_1 states and there exists another automaton M' equivalent to M' containing n_2 state with $n_2 \le n_1$ then it is desirable to use M'.

Unreachable States:- These states are those states of a DFA, which are not reachable from the initial state of DFA on any possible i/p. The unreachable or non-reachable states are useless states in an automation and these should be eliminated immediately.

Dead State:- These states are those states of a DFA whose transmission on every i/p symbol terminate on itself. The Dead State in finite automata are useless state and these should be eliminated immediately.

Method I for Constructing of Minimum State Automata:-

Step1:- We will create a set Π_0 as $\Pi_0 = \{Q_1^0, Q_2^0\}$ where Q_1^0 is a set of all final states and $Q_2^0 = Q - Q_1^0$ where Q is a set of all the states in DFA.

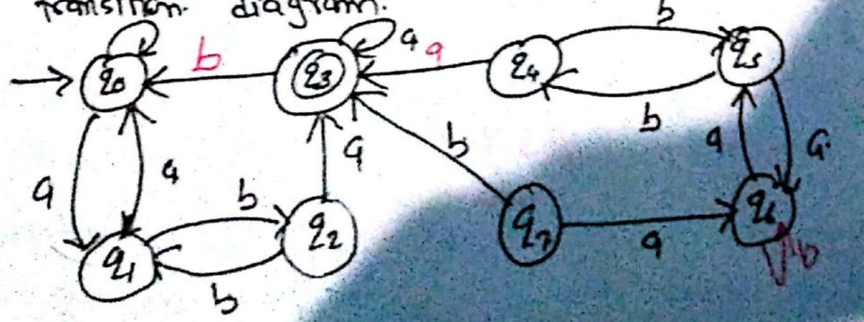
Step2:- Now we will construct Π_{K+1} from Π_K. Let Q_1^K be any subset in Π_K. If q_1 and q_2 are in Q_1^K they are (K+1) equivalent provided $\delta(q_1,a)$ and $\delta(q_2,a)$ are K equivalent. Find out whether $\delta(q_1,a)$ and $\delta(q_2,a)$ are residing in the same equivalence class Π_K. Then it is said that q_1 and q_2 are (K+1) equivalent. Thus Q_1^K is further divided into (K+1) equivalence classes.

Repeat step 2 for every Q_1^K in Π_K and obtain all the elements of Π_{K+1}.

Step3:- Construct Π_n for $n = 1, 2, \ldots$ until $\Pi_n = \Pi_{n+1}$

Step4:- Then replace all the equivalent state in one equivalence class by representative state. This helps in minimization the given DFA.

Question:- Construct the minimum state automata for the following transition diagram.



This question is solved using method I

I/P State	a	b
$\to q_0$	q_1	q_0
q_1	q_0	q_2
q_2	q_3	q_1
q_3	q_3	q_0
q_4	q_3	q_5
q_5	q_6	q_4
q_6	q_5	q_6
q_7	q_6	q_3

There is only one final state i.e q_3. Hence we can partition Q as Q_1^0 and Q_2^0

$Q_1^0 = F = \{q_3\}$ as $\{q_3\}$ cannot be partitioned further.

$Q_2^0 = Q - Q_1^0$
$= \{q_0, q_1, q_2, q_4, q_5, q_6, q_7\}$.

Now we will compare q_0 with q_2.

	a	b
q_0	q_1	q_0
q_2	q_3	q_1

Under a column of q_0 and q_2 we get q_1 and q_3 respectively. But q_1 and q_3 lie in different states. Hence they are not 1-equivalent.

Similarly consider q_0 and q_4

	a	b
q_0	q_1	q_0
q_4	q_3	q_5

q_0 is not 1 equivalent to q_4.
similarly q_0 is not 1 equivalent to q_7

$Q_2^1 = \{q_0, q_1, q_5, q_6\}$

q_2 is one-equivalent to q_4. Hence
$Q_3^1 = \{q_2, q_4\}$

$Q_4^1 = \{q_7\}$

$\Pi_1 = \{\{q_3\}, \{q_0, q_1, q_5, q_6\}, \{q_2, q_4\}, \{q_7\}\}$.

Now q_0 is 2 equivalent to q_6 because.

$\delta(q_0, a) = q_1$ | $\delta(q_6, a) = q_5$
$\delta(q_0, b) = q_0$ | $\delta(q_6, b) = q_6$

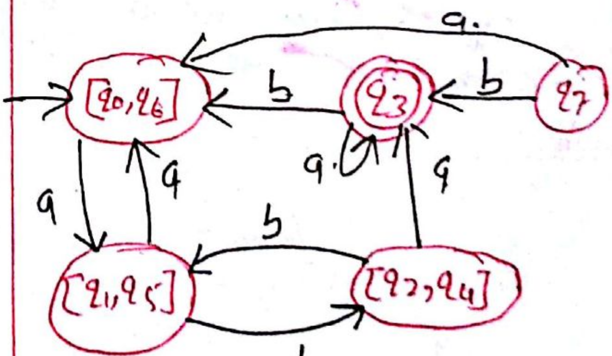
Same class

$Q_2^2 = \{q_0, q_6\}$.

$Q_3^2 = \{q_1, q_5\}$.

$\Pi_2 = \{\{q_3\}, \{q_0, q_6\}, \{q_1, q_5\}, \{q_2, q_4\}, \{q_7\}\}$.

I/P State	a	b
$\to [q_0, q_6]$	$[q_1, q_5]$	$[q_0, q_6]$
$[q_1, q_5]$	$[q_0, q_6]$	$[q_2, q_4]$
$[q_2, q_4]$	$[q_3]$	$[q_1, q_5]$
$[q_3]$	$[q_3]$	$[q_0, q_6]$
$[q_7]$	$[q_0, q_6]$	$[q_3]$



Here q_7 is unreachable state. Simply remove it.

Minimization of Finite Automata :- Using IInd method (GO)

Q-1

Present state	Next state a	Next state b
$\rightarrow q_1$	q_2	q_1
q_2	q_1	q_3
q_3	q_4	q_2
q_4	q_4	q_1
q_5	q_4	q_6
q_6	q_7	q_5
q_7	q_6	q_7
q_8	q_7	q_4

Construct the minimum state of DFA equivalent to the DFA given by table.

Solun :- On the basis of final state and non-final state.

$\pi_0 \Rightarrow$ Class 1 = $\{q_4\}$

Class 2 = $\{q_1, q_2, q_3, q_5, q_6, q_7, q_8\}$

for given a input symbols on class 2

$$\pi_1 \Rightarrow \left\{ \underbrace{\{q_4\}}_{class 1}, \underbrace{\underbrace{\{q_1, q_2, q_6, q_7\}}_{class 4}, \underbrace{\{q_3, q_5, q_8\}}_{class 3}}_{Class 2} \right\}$$

for given a same I/P on class 3

$$\pi_2 = \left\{ \underbrace{\{q_4\}}_{1}, \overbrace{\underbrace{\underbrace{\{q_3, q_5\}}_{5}, \underbrace{\{q_8\}}_{6}}_{class 3}, \underbrace{\{q_1, q_2, q_6, q_7\}}_{Class 4}}^{class 2} \right\}$$

for given a same I/P on class 4

$\delta(q_1, a) = q_2$ $\quad$ $\delta(q_7, a) = q_6$ $\quad$ $\delta(q_7, b) = q_7$

$\delta(q_2, a) = q_1$ $\quad$ $\delta(q_1, b) = q_1$

$\delta(q_6, a) = q_7$ $\quad$ $\left.\begin{array}{l}\delta(q_2, b) = q_3 \\ \delta(q_6, b) = q_5\end{array}\right]$ class 5

$\pi_3 = \{ \{q_4\}, \{q_2, q_6\}, \{q_1, q_7\}, \{q_3, q_5\}, \{q_8\} \}$ (11)

for given I/P symbol on class

$\delta(q_4, a) = q_4$

$\delta(q_4, b) = \{q_1, q_7\}$

$\delta(\{q_2, q_6\}, a) = \{q_1, q_7\}$

$\delta(\{q_2, q_6\}, b) = \{q_3, q_5\}$

$\delta(\{q_1, q_7\}, a) = \{q_2, q_6\}$

$\delta(\{q_1, q_7\}, b) = \{q_1, q_7\}$

$\delta(\{q_3, q_5\}, a) = \{q_4\}$

$\delta(\{q_3, q_5\}, b) = \{q_2, q_6\}$

$\delta(\{q_8\}, a) = \{q_1, q_7\}$

$\delta(q_8, b) = \{q_4\}$

Transistion Table for minimization ⇒

P S	Next state a	Next state b
$\rightarrow \{q_1, q_7\}$	$\{q_2, q_6\}$	$\{q_1, q_7\}$
$\{q_2, q_6\}$	$\{q_1, q_7\}$	$\{q_3, q_5\}$
$\{q_3, q_5\}$	$\{q_4\}$	$\{q_2, q_6\}$
$\{q_4\}$	$\{q_4\}$	$\{q_1, q_7\}$
$\{q_8\}$	$\{q_1, q_7\}$	$\{q_4\}$

∵ q_8 is a unreachable state because q_8 state doesn't exist in any next state so it can be minimized.

Ques-2 Construct the minimum state of DFA equivalent to the DFA given by table.

Present state	Next state a	Next state b
$\rightarrow q_0$	q_1	q_5
q_1	q_6	q_2
q_2	q_0	q_2
q_3	q_2	q_6
q_4	q_1	q_5
q_5	q_2	q_6
q_6	q_6	q_4
q_7	q_6	q_2

(P-123) 72

Solu^n:- Step-1:- On the basis of final state & non-final state

$$\Pi_0 \Rightarrow \text{class } 1 = \{q_2\}$$
$$\text{class } 2 = \{q_0, q_1, q_3, q_4, q_5, q_6, q_7\}$$

for given a I/P symbol on class 2 on the basis of final state

$\delta(q_0, a) = q_1$
$\delta(q_0, b) = q_5$
$\delta(q_1, a) = q_6$
$\delta(q_1, b) = q_2$
$\delta(q_3, a) = q_2$
$\delta(q_3, b) = q_6$

$\delta(q_4, a) = q_1$
$\delta(q_4, b) = q_5$
$\delta(q_5, a) = q_2$
$\delta(q_5, b) = q_6$
$\delta(q_6, a) = q_6$
$\delta(q_6, b) = q_4$

$\delta(q_7, a) = q_6$
$\delta(q_7, b) = q_2$

$$\Pi_1 = \Big\{ \underbrace{\{q_2\}}_{\text{class1}}, \underbrace{\underbrace{\{q_3, q_5, q_7\}}_{\text{Class 3}} \; \underbrace{\{q_0, q_1, q_4, q_6\}}_{\text{Class4}}}_{\text{Class 2}} \Big\}$$

On the basis of same I/P & same next state at class3

$\checkmark \delta(q_3, a) = q_2$ $\quad \delta(q_5, b) = q_6$
$\delta(q_3, b) = q_6$ $\quad \delta(q_7, a) = q_6$
$\checkmark \delta(q_5, a) = q_2$ $\quad \delta(q_7, b) = q_2$

$$\Pi_2 = \Big\{ \underbrace{\{q_2\}}_{1}, \underbrace{\underbrace{\overbrace{\{q_3, q_5\}}^{5} \; \overbrace{\{q_7\}}^{6}}_{\text{class 3}}, \underbrace{\{q_0, q_1, q_4, q_6\}}_{\text{class 4}}}_{\text{class 2}} \Big\}$$

similiarly for given a I/P symbol on class 4

$$\Pi_3 = \Big\{ \underbrace{\{q_2\}}_{1} \; \underbrace{\underbrace{\overbrace{\{q_3, q_5\}}^{5} \; \overbrace{\{q_7\}}^{6}}_{\text{class3}}, \underbrace{\overbrace{\{q_0, q_4\}}^{7} \; \overbrace{\{q_1\}}^{8}_{9}, \{q_6\}_{10}}_{\text{class 4}}}_{\text{class 2}} \Big\}$$

because
$\delta(q_0, a) = q_1$ $\quad \delta(q_1, a) = q_6$ but $\delta(q_1, b) = q_2$
$\delta(q_4, a) = q_1$ $\quad \delta(q_6, a) = q_6$ $\quad \delta(q_6, b) = q_4$

Transistion Table for new DFA

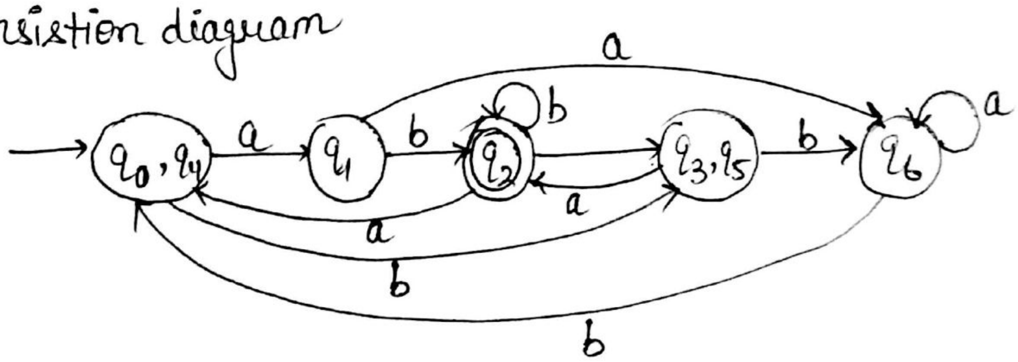
P.S	Next state	
	a	b
$\rightarrow \{q_0, q_4\}$	q_1	q_5
$\{q_1\}$	q_6	q_2
$\{q_2\}$	q_0	q_2
$\{q_3, q_5\}$	q_2	q_6
$\{q_6\}$	q_6	q_4
$\{q_7\}$	q_6	q_2

$\because$ q_7 is an unreachable state so it can be neglected.

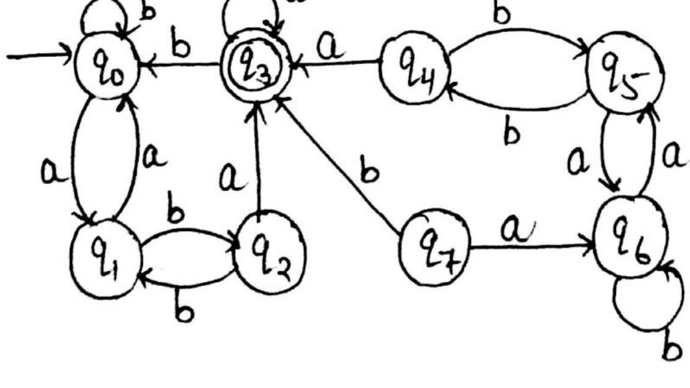
New Transistion Table for DFA :-

P.S	N.S	
	a	b
$\rightarrow \{q_0, q_4\}$	q_1	q_5
$\{q_1\}$	q_6	q_2
$\{q_2\}$	q_0	q_2
$\{q_3, q_5\}$	q_2	q_6
$\{q_6\}$	q_6	q_4

Transistion diagram



Q=3 Construct the minimization of DFA equivalent to given DFA.



Transition Table :-   (P-125)   (24)

Present state	Next state a	Next state b
$\rightarrow q_0$	q_1	q_0
q_1	q_0	q_2
q_2	q_3	q_1
q_3	q_3	q_0
q_4	q_3	q_5
q_5	q_6	q_4
q_6	q_5	q_6
q_7	q_6	q_3

Step 1 :- On the basis of final state and non-final state

$$\pi_0 \Rightarrow \text{class } 1 = \{q_3\}$$
$$\text{Class } 2 = \{q_0, q_1, q_2, q_4, q_5, q_6, q_7\}$$

for given a I/P symbol on class 2

$\delta(q_0, a) = q_1$ $\quad \delta(q_2, a) = q_3$ $\quad \delta(q_5, a) = q_6$ $\quad \delta(q_7, a) = q_6$

$\delta(q_0, b) = q_0$ $\quad \delta(q_2, b) = q_1$ $\quad \delta(q_5, b) = q_4$ $\quad \delta(q_7, b) = q_3$

$\delta(q_1, a) = q_0$ $\quad \delta(q_4, a) = q_3$ $\quad \delta(q_6, a) = q_5$

$\delta(q_1, b) = q_2$ $\quad \delta(q_4, b) = q_5$ $\quad \delta(q_6, b) = q_6$

$$\pi_1 = \left\{ \underbrace{\{q_3\}}_{1} \; \underbrace{\underbrace{\{q_2, q_4, q_7\}}_{\text{Class } 3} \; \underbrace{\{q_0, q_1, q_5, q_6\}}_{\text{Class } 4}}_{\text{Class } 2} \right\}$$

Step 2 :- for given a I/P symbol on class 3
similarly find out the all transition function for each state

$\delta(q_2, a) = q_3$ $\quad \delta(q_4, a) = q_3$ $\quad \delta(q_7, a) = q_6$

$\delta(q_2, b) = q_1$ $\quad \delta(q_4, b) = q_5$ $\quad \delta(q_7, b) = q_3$

q_2 & q_4 has same next state at a input so they belongs to same class.

$$\pi_2 = \left\{ \underbrace{\{q_3\}}_{\text{class } 1}, \underbrace{\underbrace{\overbrace{\{q_2, q_4\}}^{5} \; \overbrace{\{q_7\}}^{6}}_{\text{Class } 3} \; \overbrace{\{q_0, q_1, q_5, q_6\}}^{4}}_{\text{Class } 2} \right\}$$

again applying the I/P symbol on class 4

$$\pi_3 = \left\{ \underbrace{\{q_3\}}_{1}, \overbrace{\overbrace{\underbrace{\{q_2, q_4\}}_{5}, \underbrace{\{q_7\}}_{6}}^{3}, \underbrace{\underbrace{\{q_1, q_5\}}_{7}, \underbrace{\{q_0, q_6\}}_{8}}_{4}}^{\text{class } 2} \right\}$$

(P-126) (75)

because $\left.\begin{array}{l}\delta(q_0, a) = q_1 \\ \delta(q_6, a) = q_5\end{array}\right\}$ belongs to same class $\left.\begin{array}{l}\delta(q_1, b) = q_2 \\ \delta(q_5, b) = q_4\end{array}\right\}$ belongs to same class.

so final Transistion Table of DFA

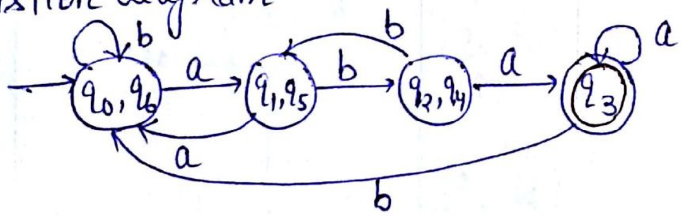
P.S	Next state	
	a	b
$\rightarrow \{q_0, q_6\}$	$\{q_1, q_5\}$	$\{q_0, q_6\}$
$\{q_1, q_5\}$	$\{q_0, q_6\}$	$\{q_2, q_4\}$
$\{q_2, q_4\}$	q_3	$\{q_1, q_5\}$
$\{q_3\}$	q_3	q_0
$\{q_7\}$	q_6	q_3

$\because$ q_7 is an unreachable state so it can be neglected.

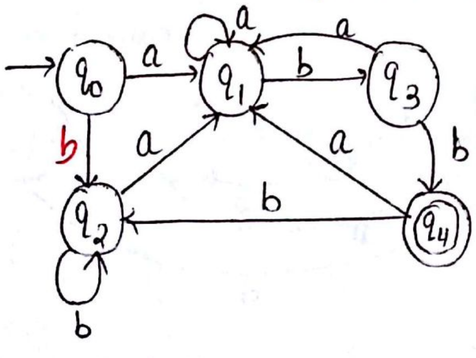
Then new Transistion Table for minimization of DFA.

Present state	Next state	
	a	b
$\rightarrow \{q_0, q_6\}$	$\{q_1, q_5\}$	$\{q_0, q_6\}$
$\{q_1, q_5\}$	$\{q_0, q_6\}$	$\{q_2, q_4\}$
$\{q_2, q_4\}$	q_3	$\{q_1, q_5\}$
q_3	q_3	q_0

Transistion diagram



Case = 4



construct the minimization of DFA equivalent to given DFA.

Transistion Table:-

Present state	Next state	
	a	b
$\rightarrow q_0$	q_1	q_2
q_1	q_1	q_3
q_2	q_1	q_2
q_3	q_1	q_4
q_4	q_1	q_2

Scanned by CamScanner

Step-1 :- On the basis of final state & non-final state

$\pi_0 \Rightarrow$ Class 1 = $\{q_4\}$

Class 2 = $\{q_0, q_1, q_2, q_3\}$

Step-2 :- for given a I/P symbol on class 2

✓ $\delta(q_0, a) = q_1$	✓ $\delta(q_2, a) = q_1$
✓ $\delta(q_0, b) = q_2$	✓ $\delta(q_2, b) = q_2$
$\delta(q_1, a) = q_1$	$\delta(q_3, a) = q_1$
$\delta(q_1, b) = q_3$	$\delta(q_3, b) = q_4$

because (q_0, q_2) have same next state at a and b input.
but q_3 & q_1 have different next state at b input

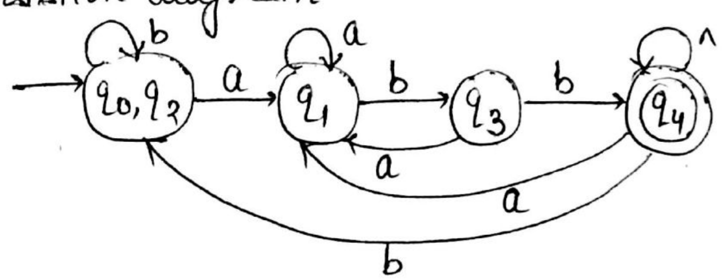
so

$$\pi_1 = \Big\{ \underbrace{\{q_4\}}_{\text{Class 1}}, \underbrace{\underbrace{\{q_0, q_2\}}_{\text{Class 3}}, \underbrace{\overbrace{\{q_1\}}^{5} \; \overbrace{\{q_3\}}^{6}}_{\text{Class 4}}}_{\text{Class 2}} \Big\}$$

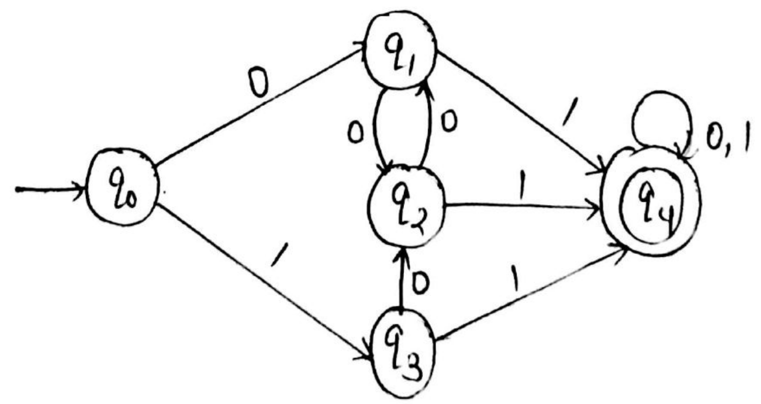
Then transistion Table for DFA

P.S	N.S a	N.S b
$\rightarrow \{q_0, q_2\}$	q_1	$\{q_0, q_2\}$
$\{q_1\}$	q_1	q_3
q_3	q_1	q_4
q_4	q_1	$\{q_0, q_2\}$

Transistion diagram



Ques = 5



Transition Table :-

P.S	N.S 0	N.S 1
$\rightarrow q_0$	q_1	q_3
q_1	q_2	q_4
q_2	q_1	q_4
q_3	q_2	q_4
q_4	q_4	q_4

Step-1 :- on the basis of final state & non-final state

$\pi_0 \Rightarrow$ Class 1 = $\{q_4\}$
Class 2 = $\{q_0, q_1, q_2, q_3\}$

Step-2 :- for given a I/P symbol on class 2

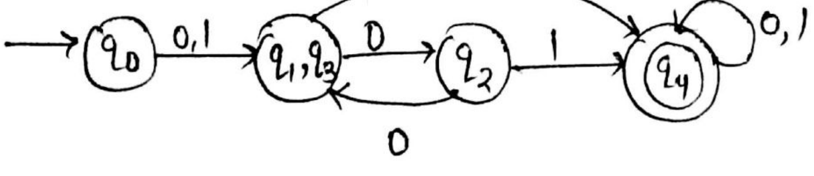
$\delta(q_0, 0) = q_1$ $\delta(q_0, 1) = q_3$
$\delta(q_1, 0) = q_2$ $\delta(q_1, 1) = q_4$
$\delta(q_2, 0) = q_1$ $\delta(q_2, 1) = q_4$
$\delta(q_3, 0) = q_2$ $\delta(q_3, 1) = q_4$

$$\pi_1 = \Big\{ \underbrace{\{q_4\}}_{\text{Class1}} \; \underbrace{\underbrace{\{q_1, q_3\}}_{3} \; \underbrace{\overbrace{\{q_2\}}^{5} \; \overbrace{\{q_0\}}^{6}}_{4}}_{\text{Class2}} \Big\}$$

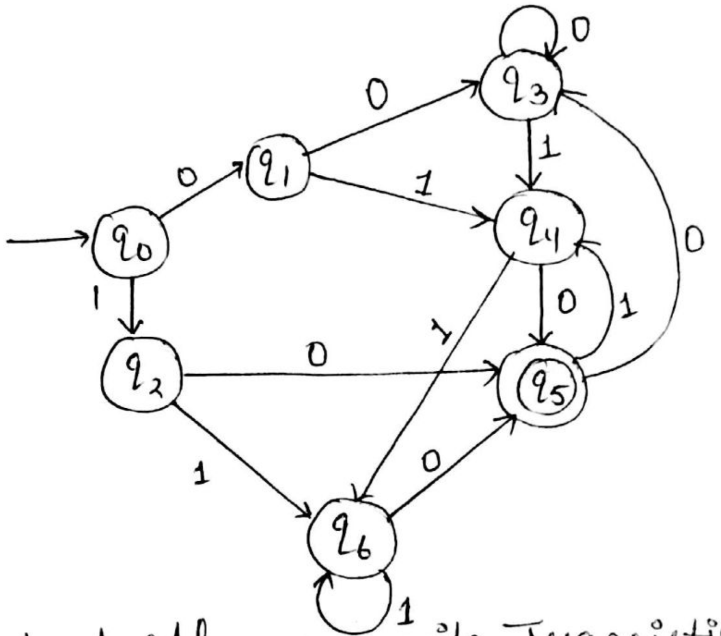
Transition Table for New DFA :-

P.S	N.S 0	N.S 1
$\rightarrow q_0$	q_1	$\{q_3, q_4\}$
$\{q_1, q_3\}$	q_2	q_4
$\{q_2\}$	q_1	q_4
q_4	q_4	q_4

Transition diagram



Ques-6 construct the minimization of DFA equivalent to given DFA.



Solution:- first of all we write Transition Table

Present state	N.S 0	N.S 1
$\rightarrow q_0$	q_1	q_2
q_1	q_3	q_4
q_2	q_5	q_6
q_3	q_3	q_4
q_4	q_5	q_6
q_5	q_3	q_4
q_6	q_5	q_6

Step 1:- On the basis of final state and non-final state

$\Pi_0 \Rightarrow$ Class 1 = $\{q_5\}$

Class 2 = $\{q_0, q_1, q_2, q_3, q_4, q_6\}$

Step-2:- for given a I/P symbols on class 2

$\delta(q_0,0) = q_1$ $\qquad \delta(q_0,1) = q_2$

$\delta(q_1,0) = q_3$ $\qquad \delta(q_1,1) = q_4$

$\delta(q_2,0) = q_5$ $\qquad \delta(q_2,1) = q_6$

$\delta(q_3,0) = q_3$ $\qquad \delta(q_3,1) = q_4$

$\delta(q_4,0) = q_5$ $\qquad \delta(q_4,1) = q_6$

$\delta(q_6,0) = q_5$ $\qquad \delta(q_6,1) = q_6$

(P 136)

$$\Pi_1 = \left\{ \underbrace{\{q_5\}}_{\text{Class 1}} \underbrace{\overbrace{\{q_2, q_4, q_6\}}^{3} \overbrace{\{q_0, q_1, q_3\}}^{4}}_{\text{Class 2}} \right\}$$ On the basis of final state for each state.

step-3 :- for given a Input symbol on class 3

$\delta(q_2,0) = q_5$ $\delta(q_2,1) = q_6$

$\delta(q_4,0) = q_5$ $\delta(q_4,1) = q_6$

$\delta(q_6,0) = q_5$ $\delta(q_6,1) = q_6$

all next states are same so q_2, q_4, q_6 are combined together in an one class.

$$\Pi_2 = \left\{ \underbrace{\{q_5\}}_{\text{Class 1}} \underbrace{\overbrace{\{q_2, q_4, q_6\}}^{3} \overbrace{\{q_0, q_1, q_3\}}^{4}}_{\text{Class 2}} \right\}$$

step 4 :- for given a I/P symbol on class 4

Then $\delta(q_0,0) = q_1$ $\delta(q_0,1) = q_2$

$\delta(q_1,0) = q_3$ $\delta(q_1,1) = q_4$

$\delta(q_3,0) = q_3$ $\delta(q_3,1) = q_4$

q_1, q_3 belongs to same class so.

$$\Pi_3 = \left\{ \underbrace{\{q_5\}}_{\text{Class 1}}, \underbrace{\underbrace{\{q_2, q_4, q_6\}}_{\text{Class 3}}, \underbrace{\overbrace{\{q_0\}}^{5}, \overbrace{\{q_1, q_3\}}^{6}}_{\text{Class 4}}}_{\text{Class 2}} \right\}$$

Transistion table can be drawn by

$\delta(q_5,0) = q_3$ $\delta(q_0,0) = q_1$

$\delta(q_5,1) = q_4$ $\delta(q_0,1) = q_2$

$\delta(\{q_2,q_4,q_6\},0) = q_5$ $\delta(\{q_1,q_3\},0) = q_3$

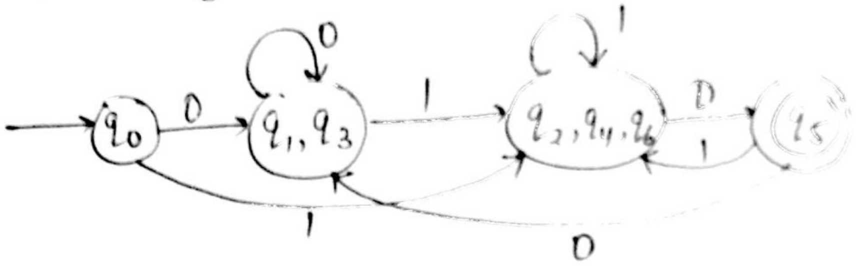
$\delta(\{q_2,q_4,q_6\},1) = q_6$ $\delta(\{q_1,q_3\},1) = q_4$

Transistion Table for new DFA

P.S	0	1
$\rightarrow q_0$	$\{q_1, q_3\}$	$\{q_2, q_4, q_6\}$
$\{q_1, q_3\}$	$\{q_1, q_3\}$	$\{q_2, q_4, q_6\}$
$\{q_2, q_4, q_6\}$	q_5	$\{q_2, q_4, q_6\}$
q_5	$\{q_3, q_1\}$	$\{q_2, q_4, q_6\}$

$\delta(\{q_2,q_4,q_6\},1) = \{q_2,q_4,q_6\}$ because q_2, q_4 & q_6 belongs to same class and q_1 & q_3 belongs to same class

Transition diagram :-



Finite Automata with Output :-

Finite automata is a collection of $(Q, \Sigma, \delta, q_0, F)$ where

Q = set of state including q_0 as a start state

Σ = set of terminal

δ = Transistion function $Q \times \Sigma \rightarrow Q$ (state function)

q_0 = starting state

F = final state

In finite automata, after reading the I/P string if we get final state, the string is said to be acceptable if we don't get final state it is said that string is rejected. That means there is no need O/P for the finite automata. But if, there is a need for specifying the output other than yes or no then in such a case we require finite automata along with O/P. There are two types of F.A with O/P.

F.A with O/P
- → Moore machine
- → Mealy machine

Moore machine :- is a finite state machine in which the next state is decided by the current state and current I/P symbol. The O/P symbol at a given time depends only on the present state of the machine. The formal definition of moore machine.

" Moore machine is a six tuple $(Q, \Sigma, \Delta, \delta, \lambda, q_0)$ where.

Q is a finite set of state

Σ is a finite set of terminal

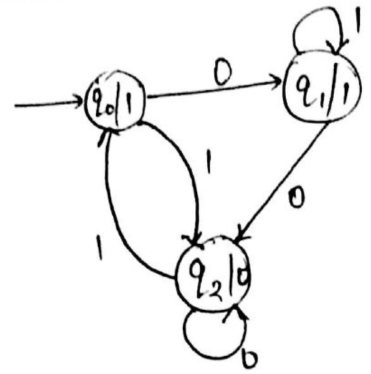
Δ is an Output alphabet

δ is a transistion function

$Q \times \Sigma \rightarrow Q$ (state function)

λ is O/P function : $Q \rightarrow \lambda$ (also called Machine function)

q_0 is called initial state of the machine.

Ex:- 

Transition Table:-

P.S \ I.P	N.S 0	N.S 1	O/P
$\rightarrow q_0$	q_1	q_2	1
q_1	q_2	q_1	1
q_2	q_2	q_0	0

Mealy machine:- Mealy machine is a machine in which O/P symbols depends upon the present I/P symbol & present state of the machine. The mealy machine can be defined as —

Mealy machine is a six tuple $(Q, \Sigma, \Delta, \delta, \lambda, q_0)$

where, Q is a finite set of states

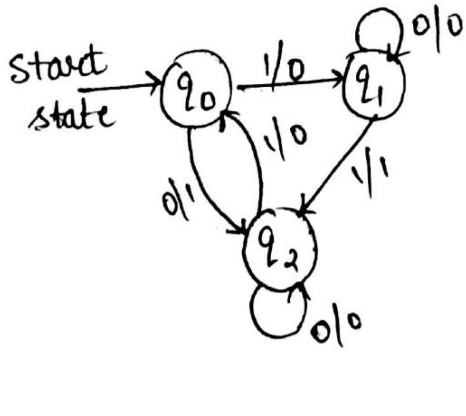
Σ is a finite set of I/P symbols

Δ is an output alphabet

δ is a transition function such that $Q \times \Sigma \rightarrow Q$ (state function)

λ is an output function. $Q \rightarrow \lambda$ (also called Machine function)

q_0 is an initial state of machine.

Example:- 

Transistion Table of mealy machine:-

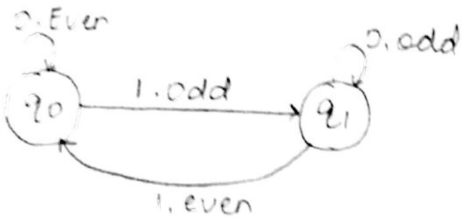
(P-134) (80)

I/P P.S	N.S a = 0 N.state	a = 0 O/P	a = 1 N.state	a = 1 O/P
q_0	q_2	1	q_1	0
q_1	q_1	0	q_2	1
q_2	q_2	0	q_0	0

(P. 135) (82)

Ques. Construct a Mealy machine which can output EVEN, ODD according as the total no. of 1's encounted is even or odd. The input symbols are 0 and 1.

Solution:



Transition table for above Mealy machine.

Present state	Next state			
	$a = 0$		$a = 1$	
	state	Output	state	Output
q_0	q_0	Even	q_1	Odd
q_1	q_1	Odd	q_0	even

MEALY AND MOORE MODELS

Ques: consider the Mealy machine describe by the transition table given by table. construct a Moore machine which is equivalent to the Mealy machine.

Present state	Next state			
	a = 0		a = 1	
	state	Output	state	Output
→ q_1	q_3	0	q_2	0
q_2	q_1	1	q_4	0
q_3	q_2	1	q_1	1
q_4	q_4	1	q_3	0

Solution

Step I →

Present state	Next state			
	a = 0		a = 1	
	state	Output	state	Output
→ q_1	q_3	0	q_{20}	0
q_{20}	q_1	1	q_{40}	0
q_{21}	q_1	1	q_{40}	0
q_3	q_{21}	1	q_1	1
q_{40}	q_{41}	1	q_3	0
q_{41}	q_{41}	1	q_3	0

Step II -

Present state	Next state		Output
	a = 0	a = 1	
$\rightarrow q_1$	q_3	q_{20}	1
q_{20}	q_1	q_{40}	0
q_{21}	q_1	q_{40}	1
q_3	q_{21}	q_1	0
q_{40}	q_{41}	q_3	0
q_{41}	q_{41}	q_3	1

Step III -

Present state	Next state		Output
	a = 0	a = 1	
$\rightarrow q_0$	q_3	q_{20}	0
q_1	q_3	q_{20}	1
q_{20}	q_1	q_{40}	0
q_{21}	q_1	q_{40}	1
q_3	q_{21}	q_1	0
q_{40}	q_{41}	q_3	0
q_{41}	q_{41}	q_3	1

P-138

35

Ques. Construct a Mealy Machine which is equivalent to the Moore machine given by table.

Present state	Next state		Output
	$a=0$	$a=1$	
$\rightarrow q_0$	q_3	q_1	0
q_1	q_1	q_2	1
q_2	q_2	q_3	0
q_3	q_3	q_0	0

Solution:

Step-I:

Present state	Next state			
	$a=0$		$a=1$	
	state	Output	state	Output
$\rightarrow q_0$	q_3	0	q_1	1
q_1	q_1	1	q_2	0
q_2	q_2	0	q_3	0
q_3	q_3	0	q_0	0

Ques: Consider the Moore machine described by the transition table given by table. construct the corresponding Mealy machine.

Present state	Next state $a=0$	Next state $a=1$	Output
$\rightarrow q_1$	q_1	q_2	0
q_2	q_1	q_3	0
q_3	q_1	q_3	1

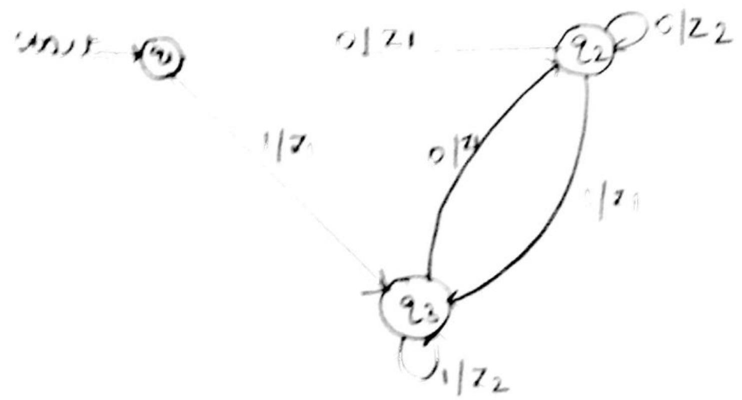
Solution: construct the transition table associating the output with the transitions

Present state	Next state			
	$a=0$		$a=1$	
	state	Output	state	Output
$\rightarrow q_1$	q_1	0	q_2	0
q_2	q_1	0	q_3	1
q_3	q_1	0	q_3	1

The rows corresponding to q_2 & q_3 are identical so, we can delete one of two states.
$\therefore$ q_3 is replaced by q_2.

Present state	Next state			
	$a=0$		$a=1$	
	state	Output	state	Output
$\rightarrow q_1$	q_1	0	q_2	0
q_2	q_1	0	q_2	1

Ques: Consider a Mealy machine by fig given below. Construct a Moore machine equivalent to this Mealy machine.



Solution: Firstly, convert the transition diagram into the transition table

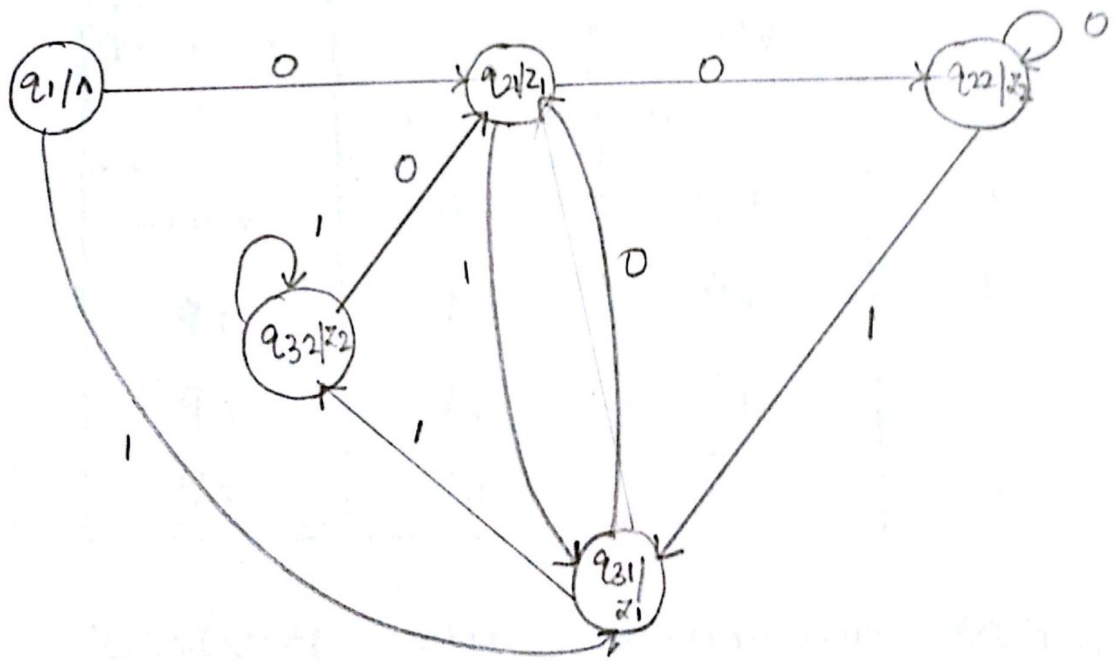
Present state	Next state			
	a = 0		a = 1	
	State	Output	State	Output
$\rightarrow q_1$	q_2	Z_1	q_3	Z_1
q_2	q_2	Z_2	q_3	Z_1
q_3	q_2	Z_1	q_3	Z_2

Transition table for Moore machine.

Present state	Next state		Output
	a = 0	a = 1	
$\rightarrow q_1$	q_{21}	q_{31}	$\wedge$
q_{21}	q_{22}	q_{31}	Z_1
q_{22}	q_{22}	q_{31}	Z_2
q_{31}	q_{21}	q_{32}	Z_1
q_{32}	q_{21}	q_{32}	Z_2

(P-141) (82)

- The transition diagram of the required Moore Machine.



Scanned by CamScanner

Ques: construct a Mealy Machine which is equivalent to the Moore machine define by table.

Present state	Next state		Output
	a=0	a=1	
→ q_0	q_1	q_2	1
q_1	q_3	q_2	0
q_2	q_2	q_1	1
q_3	q_0	q_3	1

solution: construct the transition table by associating the output with the transitions.

Present state	Next state			
	a = 0		a = 1	
	state	Output	state	Output
→ q_0	q_1	0	q_2	1
q_1	q_3	1	q_2	1
q_2	q_2	1	q_1	1
q_3	q_0	1	q_3	1

(P-143) 90

Ques. Construct a Moore machine equivalent to the Mealy machine M defined by table.

Present state	Next state			
	a = 0		a = 1	
	state	Output	state	Output
$\rightarrow q_1$	q_1	1	q_2	0
q_2	q_4	1	q_4	1
q_3	q_2	1	q_3	1
q_4	q_3	0	q_1	1

Solution: Construct a transition table for the new states

Present state	Next state			
	a = 0		a = 1	
	state	Output	state	Output
$\rightarrow q_1$	q_1	1	q_{20}	0
q_{21}	q_4	1	q_4	1
q_{20}	q_4	1	q_4	1
q_{30}	q_{21}	1	q_{31}	1
q_{31}	q_{21}	1	q_{31}	1
q_4	q_{30}	0	q_1	1

(P-144) (91)

step-II: Transition table for moore machine

Present state	Next state		Output
	a=0	a=1	
→ q_1	q_1	q_{20}	1
q_{21}	q_4	q_4	1
q_{20}	q_4	q_4	0
q_{30}	q_{21}	q_{31}	0
q_{31}	q_{21}	q_{31}	1
q_4	q_{30}	q_1	1

step-III: In the above table we observes that q_1 is associated with output 1. This means that with input ^ the O/P is 1, if machine starts with state q_1. Thus, this moore machine accepts a null sequence which is not accepted by Mealy machine. To overcome this situation we must add a new starting state q_0.

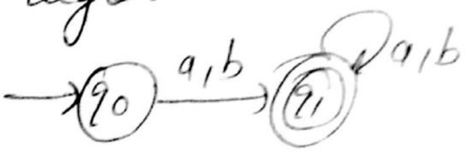
Present state	Next state		Output
	a=0	a=1	
→ q_0	q_1	q_{20}	0
q_1	q_1	q_{20}	1
q_{21}	q_4	q_4	1
q_{20}	q_4	q_4	0
q_{30}	q_{21}	q_{31}	0
q_{31}	q_{21}	q_{31}	1
q_4	q_{30}	q_1	1

Scanned by CamScanner

Miscellaneous Questions.

Q4. Construct NFA for $\begin{bmatrix} a \\ b \end{bmatrix}^+$ and derive DFA through subset construction algorithm. [UPTU 2007]

Sol: • NFA for $\begin{bmatrix} a \\ b \end{bmatrix}^+$, we get



state	input a	input b
$\rightarrow q_0$	q_1	q_1
q_1	q_1	q_1

⇒ This is also DFA.

Construction of DFA with ε-Moves

Q1. Consider the following NFA with ε-moves

$\epsilon\text{-closure}(q_0) = \{q_0, q_1, q_2\}$
$\epsilon\text{-closure}(q_1) = \{q_1, q_2\}$
$\epsilon\text{-closure}(q_2) = \{q_2\}$

The final states of DFA are those new states which contain final state of NFA with ε-moves as member

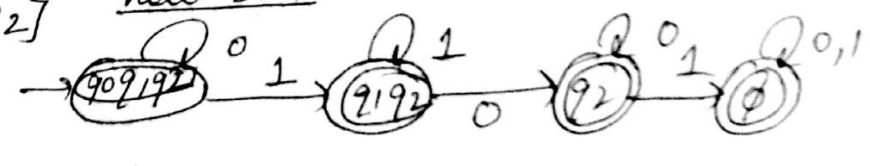
⇒ So $[q_0, q_1, q_2]$, $[q_1, q_2]$ and $[q_2]$ all are final states

$\delta'([q_0, q_1, q_2], 0) = \epsilon\text{-closure}(\delta(q_0, 0) \cup \delta(q_1, 0) \cup \delta(q_2, 0))$
$= \epsilon\text{-closure}(\{q_0\} \cup \phi \cup q_2)$
$= \epsilon\text{-closure}(q_0, q_2)$
$= \{q_0, q_1, q_2\}$ state.

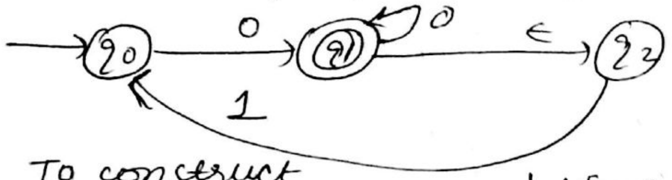
$\delta'([q_0, q_1, q_2], 1) = \epsilon\text{-closure}(\delta(q_0, 1) \cup \delta(q_1, 1) \cup \delta(q_2, 1))$
$= \epsilon\text{-closure}(\phi \cup \{q_1\} \cup \phi)$
$= \epsilon\text{-closure}(q_1)$
$= \{q_1, q_2\}$ state

$\delta'([q_1, q_2], 0) = [q_2]$
$\delta'([q_1, q_2], 1) = [q_1, q_2]$
$\delta'([q_2], 0) = [q_2]$
$\delta'([q_2], 1) = \phi$

now DFA :-



Q2. Construct DFA equivalent to following NFA.



Solution: To construct DFA:

$\epsilon\text{-closure}(q_0) = \{q_0\}$

$\epsilon\text{-closure}(q_1) = \{q_1, q_2\}$

$\epsilon\text{-closure}(q_2) = \{q_2\}$

$\delta'([q_0], 0) = \epsilon\text{-closure}(\delta[q_0], 0))$
$= \epsilon\text{-closure}(q_1)$
$= [q_1, q_2]$ state

$\delta'([q_0], 1) = \epsilon\text{-closure}(\delta[q_0], 1))$
$= \epsilon\text{-closure}(\phi)$
$= \phi$

$\delta'([q_1, q_2], 0) = \epsilon\text{-closure}(\delta[q_1, q_2], 0))$
$= \epsilon\text{-closure}(\delta(q_1, 0) \cup \delta(q_2, 0))$
$= \epsilon\text{-closure}(q_1 \cup \phi)$
$= \epsilon\text{-closure}(q_1)$
$= [q_1, q_2]$

$\delta'([q_2], 0) = \epsilon\text{-closure}(\delta([q_2], 0))$
$= \epsilon\text{-closure}(\phi)$
$= \phi$

$\delta'([q_2], 1) = \epsilon\text{-closure}(\delta[q_2], 1)$
$= \epsilon\text{-closure}(\phi)$
$= \phi$

$\delta'([q_0], 2) = \epsilon\text{-closure}(\delta[q_0], 2)$
$= \epsilon\text{-closure}(\phi)$
$= \phi$

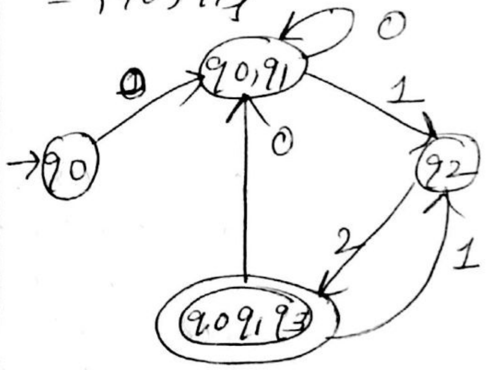
$\delta'([q_0, q_1], 2) = \epsilon\text{-closure}(\delta[q_0, q_1], 2))$
$= \epsilon\text{-closure}(\delta(q_0, 2) \cup \delta(q_1, 2))$
$= \epsilon\text{-closure}(\phi \cup \phi)$
$= \phi$

$\delta'([q_2], 2) = \epsilon\text{-closure}(\delta([q_2], 2))$
$= \epsilon\text{-closure}(q_3) = \{q_0, q_1, q_3\}$
$= \{q_0, q_1, q_3\}$ as new state

$\delta'([q_0, q_1, q_3], 1) = \epsilon\text{-closure}(\delta[q_0, q_1, q_3], 1))$
$= \epsilon\text{-closure}(\delta(q_0, 1) \cup \delta(q_1, 1) \cup \delta(q_3, 1))$
$= \epsilon\text{-closure}(\phi \cup q_2 \cup \phi)$
$= \epsilon\text{-closure}(q_2)$
$= [q_2]$ state

$\delta'([q_0, q_1, q_3], 2) = \epsilon\text{-closure}(\delta[q_0, q_1, q_3], 2))$
$= \epsilon\text{-closure}(\delta(q_0, 2) \cup \delta(q_1, 2) \cup \delta(q_3, 2))$
$= \epsilon\text{-closure}(\phi \cup \phi \cup \phi)$
$= \epsilon\text{-closure}(\phi) = \phi$

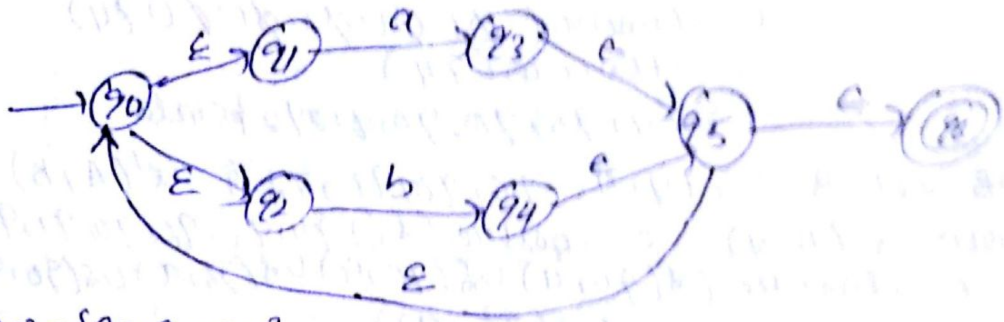
$\delta'([q_0, q_1, q_3], 0) = \epsilon\text{-closure}(\delta[q_0, q_1, q_3], 0))$
$= \epsilon\text{-closure}(\delta(q_0, 0) \cup \delta(q_1, 0) \cup \delta(q_3, 0))$
$= \epsilon\text{-closure}(q_1 \cup \phi \cup \phi)$
$= \epsilon\text{-closure}(q_1)$
$= \{q_0, q_1\}$



(P-114)

Q3. Construct NFA for $(a/b)^+$ and derive DFA through subset construction algorithm.

Solution: let NFA be $M = (Q, \Sigma, \delta, q_0, F)$



ϵ-closure$(q_0) = \{q_0, q_1, q_2\}$
ϵ-closure$(q_1) = \{q_1\}$
ϵ-closure$(q_2) = \{q_2\}$
ϵ-closure$(q_3) = \{q_3, q_5, q_6, q_0, q_1, q_2\}$
ϵ-closure$(q_4) = \{q_4, q_5, q_6, q_0, q_1, q_2\}$
ϵ-closure$(q_5) = \{q_5, q_6, q_0, q_1, q_2\}$
ϵ-closure$(q_6) = \{q_6\}$

$\delta'([q_0, q_1, q_2], a) = \epsilon\text{-closure}(\delta[q_0, q_1, q_2], a))$
$= \epsilon\text{-closure}(\delta(q_0, a) \cup \delta(q_1, a) \cup \delta(q_2, a))$
$= \epsilon\text{-closure}(\phi \cup q_3 \cup \phi)$
$= \epsilon\text{-closure}(q_3)$
$= \{q_3, q_5, q_6, q_0, q_1, q_2\}$
$= \{q_3, q_5, q_6, q_0, q_1, q_2\}$ state

let $S = [q_0, q_1, q_2]$ and $A = [q_3, q_5, q_6, q_0, q_1, q_2]$ then $\delta'(S, a) = A$

$\delta'(S, b) = \epsilon\text{-closure}(\delta[q_0, q_1, q_2], b))$
$= \epsilon\text{-closure}(\delta(q_0, b) \cup \delta(q_1, b) \cup \delta(q_2, b))$
$= \epsilon\text{-closure}(\phi \cup \phi \cup q_4)$
$= \epsilon\text{-closure}(q_4)$
$= \{q_4, q_5, q_6, q_0, q_1, q_2\}$

let $B = [q_4, q_5, q_6, q_0, q_1, q_2]$ $\delta'(S, b) = B$

$\delta'(A, a) = \epsilon\text{-closure}(\delta([q_3, q_5, q_6, q_0, q_1, q_2], a))$
$= \epsilon\text{-closure}(\delta(q_3, a) \cup \delta(q_5, a) \cup \delta(q_6, a) \cup \delta(q_0, a) \cup \delta(q_1, a) \cup \delta(q_2, a))$
$= \epsilon\text{-closure}(\phi \cup \phi \cup \phi \cup \phi \cup q_3 \cup \phi)$
$= \epsilon\text{-closure}(q_3)$
$= \{q_3, q_5, q_6, q_0, q_1, q_2\} = A$ state

$\delta'(A, a) = A$

Scanned by CamScanner

$\delta'(A,b) = \varepsilon\text{-closure}(\delta([q_3,q_5,q_6,q_0,q_1,q_2],b))$

$= \varepsilon\text{-closure}(\delta(q_3,b) \cup \delta(q_5,b) \cup \delta(q_6,b) \cup \delta(q_0,b) \cup \delta(q_1,b) \cup \delta(q_2,b))$

$= \varepsilon\text{-closure}(\phi \cup \phi \cup \phi \cup \phi \cup \phi \cup q_4)$

$= \varepsilon\text{-closure}(q_4)$

$= \{q_4, q_5, q_6, q_0, q_1, q_2\}$ state

let $B = \{q_4, q_5, q_6, q_0, q_1, q_2\}$ $\quad \delta'(A,b) = B$

Now $\delta'(B,a) = \varepsilon\text{-closure}(\delta([q_4,q_5,q_6,q_0,q_1,q_2],a))$

$= \varepsilon\text{-closure}(\delta(q_4,a) \cup \delta(q_5,a) \cup \delta(q_6,a) \cup \delta(q_0,a) \cup \delta(q_1,a) \cup \delta(q_2,a))$

$= \varepsilon\text{-closure}(\phi \cup \phi \cup \phi \cup \phi \cup q_3 \cup \phi)$

$= \varepsilon\text{-closure}(q_3)$

$= A$ state

i.e., $\delta'(B,a) = A$

$\delta'(B,b) = \varepsilon\text{-closure}(\delta([q_4,q_5,q_6,q_0,q_1,q_2],b))$

$= \varepsilon\text{-closure}(\delta(q_4,b) \cup \delta(q_5,b) \cup \delta(q_6,b) \cup \delta(q_0,b) \cup \delta(q_1,b) \cup \delta(q_2,b))$

$= \varepsilon\text{-closure}(\phi \cup \phi \cup \phi \cup \phi \cup \phi \cup q_4)$

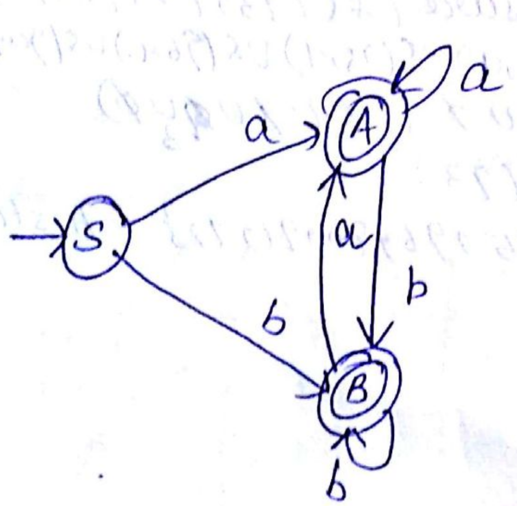
$= \varepsilon\text{-closure}(q_4)$

$= B$ state

i.e., $\delta'(B,b) = B$

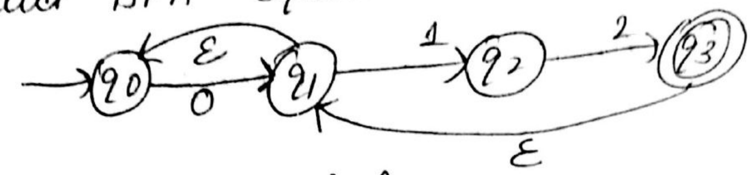
transition table for DFA is

state	input a	input b
$\rightarrow S$	A	B
(A)	A	B
(B)	A	B



(P-145)

Q4. construct DFA equivalent to following NFA



solution: ε-closure$(q_0) = \{q_0\}$

ε-closure$(q_1) = \{q_0, q_1\}$

ε-closure$(q_2) = \{q_2\}$

ε-closure$(q_3) = \{q_3, q_1, q_0\}$

Thus, the initial state in DFA is $\{q_0\}$.

$\delta'([q_0], 0) = \varepsilon$-closure $(\delta([q_0], 0))$
$= \varepsilon$-closure$(q_1) = \{q_0, q_1\}$
$= [q_0, q_1]$ state

$\delta'([q_0], 1) = \varepsilon$-closure $(\delta[q_0], 1)$
$= \varepsilon$-closure(ϕ)
$= \phi$

$\delta'([q_0, q_1], 0) = \varepsilon$-closure $(\delta([q_0, q_1], 0))$
$= \varepsilon$-closure $(\delta(q_0, 0) \cup \delta(q_1, 0))$
$= \varepsilon$-closure$(q_1 \cup \phi)$
$= \varepsilon$-closure$(q_1) = \{q_0, q_1\}$
$[q_0, q_1]$ which already exist.

$\delta'([q_0, q_1], 1) = \varepsilon$-closure $(\delta([q_0, q_1], 1))$
$= \varepsilon$-closure $(\delta(q_0, 1) \cup \delta(q_1, 1))$
$= \varepsilon$-closure $(\phi \cup q_2) = \varepsilon$-closure$(q_2) = \{q_2\}$
$= [q_2]$ state

$\delta'([q_1, q_2], 1) = \varepsilon$-closure $(\delta([q_1, q_2], 1))$
$= \varepsilon$-closure $(\delta(q_1, 1) \cup \delta(q_2, 1))$
$= \varepsilon$-closure $(\phi \cup q_0)$
$= \varepsilon$-closure (q_0)
$= [q_0]$.

Now, for $[q_2]$ state

$\delta'([q_2], 0) = \varepsilon$-closure $(\delta[q_2], 0)$
$= \varepsilon$-closure (ϕ)
$= \phi$

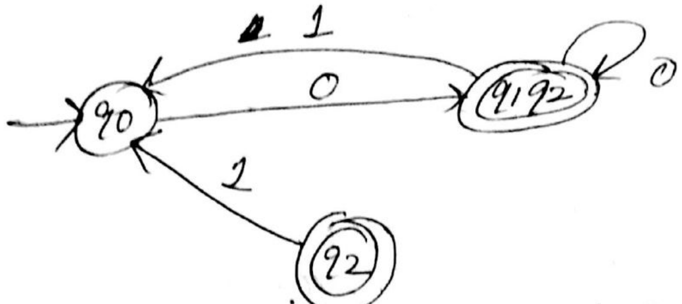
$\delta'([q_2], 1) = \varepsilon$-closure $(\delta[q_2], 1)$
$= \varepsilon$-closure $(q_0) = [q_0]$.

transition table for δ':

state	input a	input b
$\rightarrow [q_0]$	$[q_1, q_2]$	ϕ
$[q_1, q_2]$	$[q_1, q_2]$	$[q_0]$
$[q_2]$	ϕ	$[q_0]$

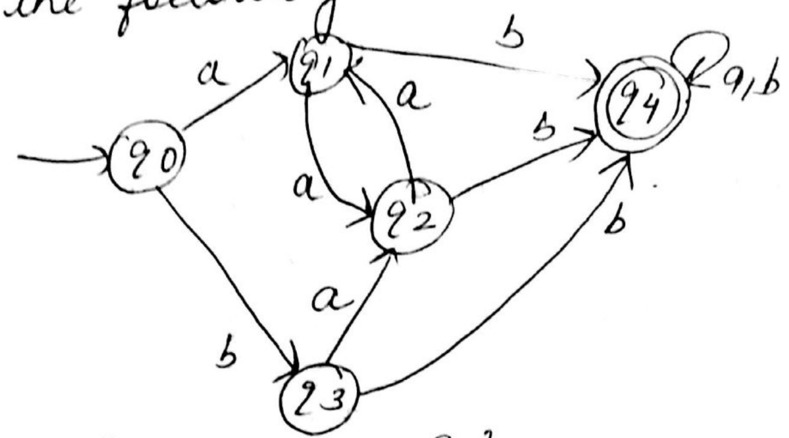
(P.150) (12)

final DFA is →



Minimization of finite automata

Q1. Consider the following automata.



here $Q = \{q_0, q_1, q_2, q_3, q_4\}$
we partition Q into two groups

$\boxed{q_4}$ $\boxed{q_0\, q_1\, q_2\, q_3}$
1 2

further division of group 1 is not possible so apply the method in group 2.

$\delta(q_0,a) = q_1$ $\delta(q_0,b) = q_3$
$\delta(q_1,a) = q_2$ $\delta(q_1,b) = q_4$
$\delta(q_2,a) = q_1$ $\delta(q_2,b) = q_4$
$\delta(q_3,a) = q_2$ $\delta(q_3,b) = q_4$

Now, we get partition of group 2 for input 'a' is not possible because all members go to the states which are in same group i.e. group 2 but for input 'b' we get state q_3 which is distinguishable from rest of states because all others go to group 1 but q_3 in group 2. hence, we divide group 2 into two groups.

$\boxed{q_4}$ $\boxed{q_1\, q_2\, q_3}$ $\boxed{q_0}$
1 2 3

Now, apply the method for group 2,

$\delta(q_1,a) = q_2$
$\delta(q_2,a) = q_1$
$\delta(q_3,a) = q_2$

Scanned by CamScanner

Now apply the method for group 2,

$\delta(q_1, a) = q_2$
$\delta(q_2, a) = q_1$
$\delta(q_3, a) = q_2$

Partition of group 2 for input 'a' not possible since all the members go to states which are in same group.

Now for input 'b', we get

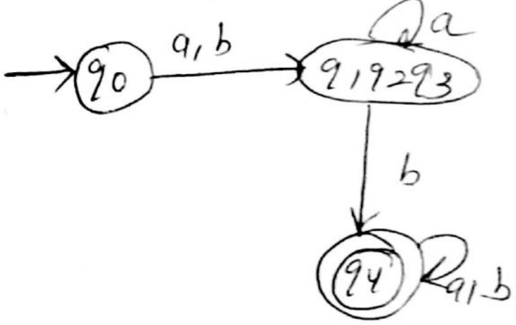
$\delta(q_1, b) = q_4$
$\delta(q_2, b) = q_4$
$\delta(q_3, b) = q_4$

no partition is possible so again we get,

$\boxed{q_4}$ $\boxed{q_0}$ $\boxed{q_1 q_2 q_3}$

i.e. q_1, q_2 and q_3 are equivalent states. we can merge these states i.e. new states is $[q_1 q_2 q_3]$ instead of $[q_1], [q_2]$ and $[q_3]$ states.

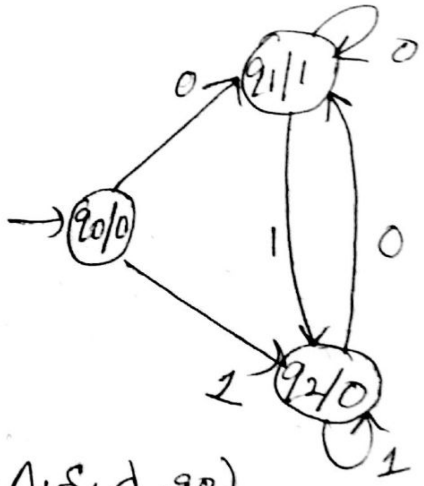
State / Σ	a	b
→ q_0	$[q_1 q_2 q_3]$	$[q_1 q_2 q_3]$
$[q_1, q_2, q_3]$	$[q_1 q_2 q_3]$	$[q_4]$
$[q_4]$	$[q_4]$	$[q_4]$



Finite Automata with Outputs

Q. Design a Moore machine to generate 1's complement of given binary number

Present state	Next state 0	Next state 1	Output (λ)
→ q_0	q_1	q_2	0
q_1	q_1	q_2	1
q_2	q_2	q_2	0



Thus Moore machine $M = (Q, \Sigma, \Delta, \delta, \lambda, q_0)$

where $Q = \{q_0, q_1, q_2\}$ $\Sigma = \{0, 1\}$ $\Delta = \{0, 1\}$, δ & λ are functions shown in table.

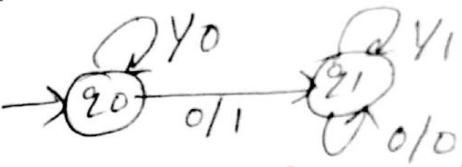
Q2. Design a Mealy machine which will increment the given binary number by 1.

Solution: In this, we will read the binary number from LSB one bit at a time and we will replace each 1 by 0 until we get first 0. Once we get first 0 we will replace it by 1 and then keep remaining bits as it is e.g's

11011 ← Read from this side bit by bit.

110100

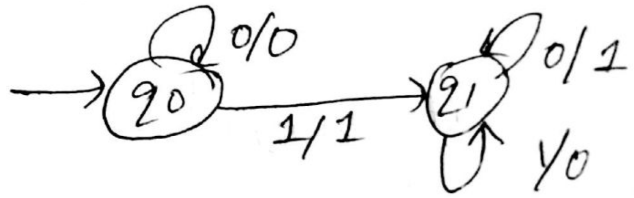
Mealy machine as:-



Q3. Design a Mealy machine to find 2's complement of a given binary number.

Solution: To find 2's complement of a binary number we assume that input is read from LSB to MSB. We will keep the binary number as it is until we read first 1. Keep this 1 as it is and then change the remaining 1's by 0's and 0's by 1's.

Thus, 2's complement of 10011 is 01101.



Q4. Construct a Moore machine to determine residue mod 3 for binary number.

Solution: This is also called remainder 3 tester. In this machine we will get remainder 0, remainder 1 and remainder 2.

To interpret the given binary number in its decimal value, we consider n as a number. If 0 is written after n then its value becomes 2n, and if 1 is written after n then its value becomes 2n+1.

for eg: If n = 0 then its decimal value is 0

$01 = 2n+1 = 2\times0+1 = 1$

$011 = 2n+1 = 2\times1+1 = 3$

$0111 = 2n+1 = 2\times3+1 = 7$

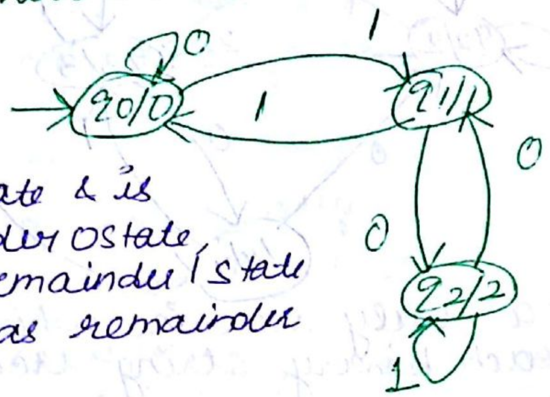
If $n=1$ then its decimal value is 1, then

$10 = 2n = 2\times1 = 2$

$101 = 2n+1 = 2\times2+1 = 4+1 = 5$

$1010 = 2n = 2\times5 = 10$.

so moore machine as:



here, q_0 is start state & is considered as remainder 0 state, similarily q_1 as remainder 1 state & q_2 is considered as remainder 2 state.

Q5. Give Moore machine for $\Sigma = \{0,1,2\}$ print the residue modulo 5 of input treated as a ternary number.

solution: The ternary number is made up of 0, 1 & 2.

(I) If we write '0' after 'n' then number becomes $3\times n$.

(II) If we write '1' after 'n' then number becomes $3n+1$.

(III) If we write '2' after 'n' then the number becomes $3n+2$.

for eg; if $n=5$, then its value is $5\times3^0 = 5$

If we write 0 after 5 that is

$50 = 3\times n$ or $5\times3^1 + 0\times3^0$

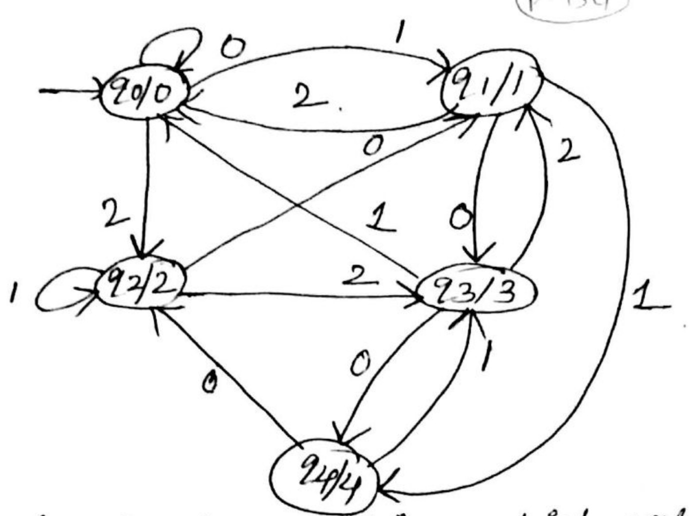
$3\times5 = 15$ $= 15+0 = 15$.

If we write 1 after 5 then

$51 = 3n+1$
$= 15+1 = 16$.

similarily $52 = 3n+2$
$= 15+2 = 17$.

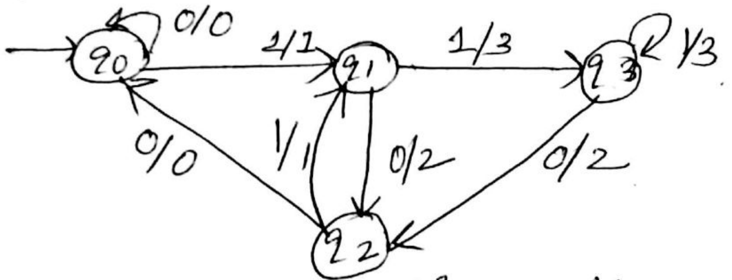
now, for residue modulo 5, we get remainder 0, 1, 2, 3, 4.
Then we assume various states for these remainders as q_0 as remainder 0 state; q_1 as remainder 1 state; q_2 as remainder 2 state; q_3 as remainder 3 state; q_4 as remainder 4 state.



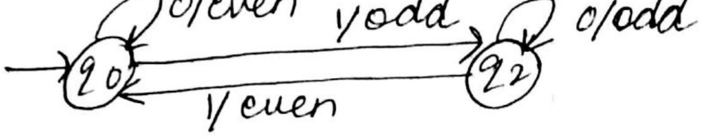
Q16. Construct a Mealy machine which calculates residue mod 4 for each binary string treated as binary integer.

solution: Let Mealy machine be $M = (Q, \Sigma, \Delta, \delta, \lambda, q_0)$

$Q = \{q_0, q_1, q_2, q_3\}$ $\Sigma = \{0, 1\}$ and $\Delta = \{0, 1, 2, 3\}$



Q17. Construct a Mealy machine which can output EVEN, ODD according to total number of 1's encountered is even or odd. The input symbols are 0 and 1.

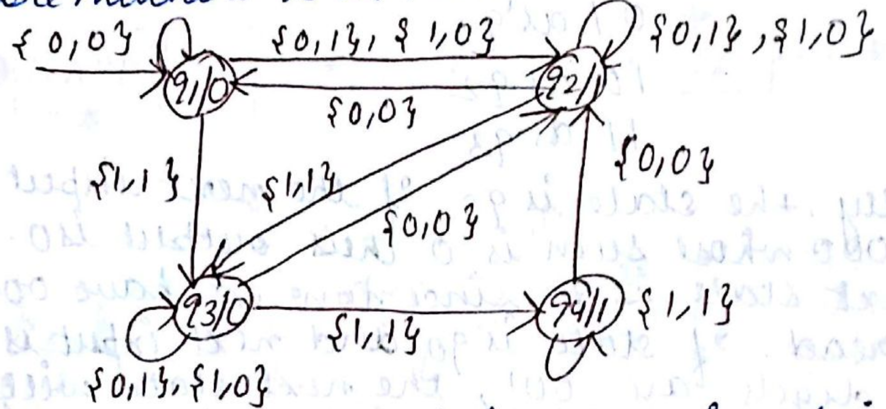


Q. construct a Moore machine which adds binary number.

Solution: In two binary number addition, we start from the least significant bit of both number and add them. If carry occurs then we propogate it to next significant bit and add next significant bit of both numbers along with carry. we have four possible cases:

1. 0+0=0 o/p → 0 carry=0
2. 0+1=1, 1+0=1 o/p → 1 carry=0
3. 1+1=10 o/p → 0 carry=1
4. 1+1+1=11 o/p → 1 carry=1

So the Moore machine is as :-



Here we show input as (i, j) where i and j are the bits to be added.

Let us discuss the transition from state q_1. clearly, if the state q_1 corresponds the output 0 and carry=0 When the state is q_1, and we add 0+0 the output=0 carry=0 which shows the same state q_1. If we add 1+0 or 0+1 then output is 1 and carry=0 which corresponds state q_2, & if we add 1+1 then output=0 and carry=1 which corresponds the state q_3. Therefore, we have drawn the edges labelled (0,0) as self loop in q_1, labelled (0,1) and (1,0) from q_1 to q_2 and labelled (1,1) from q_1 to q_3. In the similar manner we can show all possible transitions in Moore machine.

Scanned by CamScanner

Q. Design a Mealy machine which has input as binary number and output is sum of three consecutive symbols. (output is in ~~sym~~ decimal).

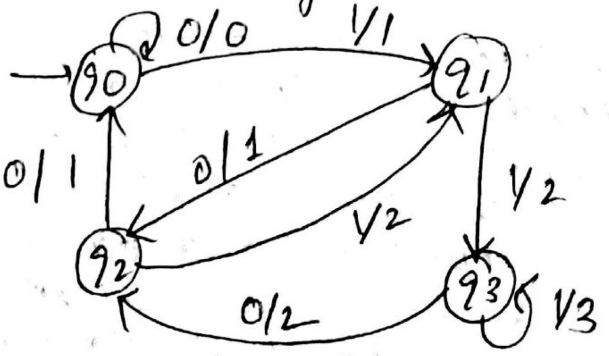
Solution: $\Sigma = \{0,1\}$ $\Delta = \{0,1,2,3\}$ since the maximum possible sum of three consecutive symbols in binary is 3. (all digits are 111).

Input 1 0 1 1 0 0 1 0 1 1 1 0 1 0 1 1 1 1 1 0
Output 1 1 2 2 2 1 1 1 2 2 3 2 2 1 2 2 3 3 3 2

we define states as the last two digits read.

00 as q_0
01 as q_1
10 as q_2
11 as q_3

Initially, the state is q_0. If the next input is 0, we have 000 whose sum is 0 thus output is 0. Again, the next state is q_0 since now we have 00 as two digits read. If state is q_0 and next input is 1, then the three digits are 001, the next state will be q_1, since last two digits are 01 and output is 1.



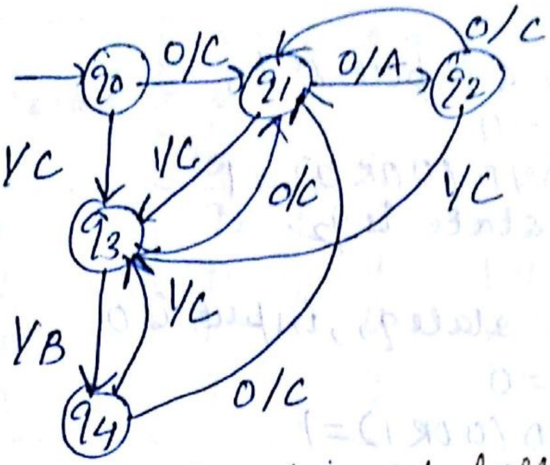
Q. Design a Mealy machine that scans sequence of inputs of 0 & 1 and generates output 'A' if the input string terminates in 00, output 'B' if string terminates in 11 and output 'C' otherwise.

sol: Let Mealy machine be

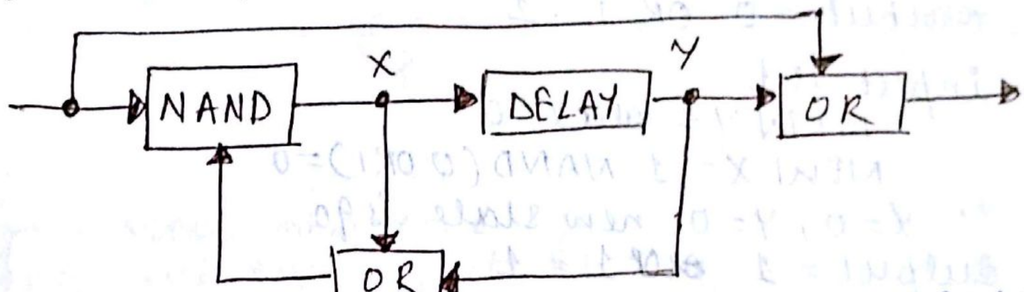
$$M = (Q, \Sigma, \Delta, \delta, \lambda, q_0)$$

where $Q = \{q_0, q_1, q_2, q_3, q_4\}$ $\Sigma = \{0,1\}$ & $\Delta = \{A, B, C\}$

(P-157)



Q. Design Mealy Machine of following circuit.



The meaning of NAND and OR as usual. The DELAY delays the transmission of the signal along the wire by one clock pulse. DELAY is sometimes called D flip-flop. We identify here, we assume all signals in form 0 & 1. We identify four status based on whether signal at X and Y in the circuit are 0 or 1.

q_0 as $X=0$ $Y=0$

q_1 as $X=0$ $Y=1$

q_2 as $X=1$ $Y=0$

q_3 as $X=1$ $Y=1$

It is clear from circuit that operation is as follows:

New Y = old X

New X = (input) NAND (old X OR old Y)

output = (input) OR (old Y)

Suppose, initially we are at state q_0 and we receive the input signal 0:

NEW Y = old X = 0

NEW X = (input) NAND (old X OR old Y)

= 0 NAND (0 OR 0)

= 0 NAND 0

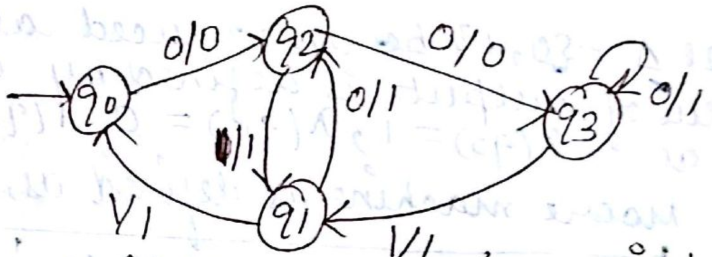
= 1

i.e. $X=1, Y=0$ the state is q_2

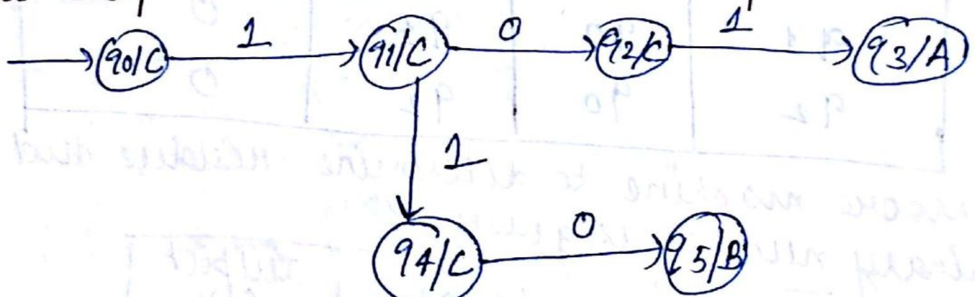
output = 0 OR 0 = 0
If we are in state q_0 & input is 1,
NEW Y = old X = 0
NEW X = 1 NAND (0 OR 0) = 1
i.e., X = 1, Y = 0, New state is q_2.
output = 1 OR 0 = 1
similarly, if we are in state q_1, input is 0
NEW Y = old X = 0
NEW X = 0 NAND (0 OR 1) = 1
i.e., X = 0, Y = 1, new state is q_2.
output = 0 OR 1 = 1
If input is 1
NEW Y = old X = 0
NEW X = 1 NAND (0 OR 1) = 0
X = 0, Y = 0 new state is q_0.
output = 1 OR 1 = 1
If we are in state q_2, & input is 0
NEW Y = old X = 1
NEW X = 0 NAND (1 OR 0) = 1
X = 1 Y = 1, new state is q_3.
output = 0 OR 0 = 0
If we are in state q_2 & input is 1
NEW Y = old X = 1
NEW X = 1 NAND (1 OR 0) = 0
X = 0, Y = 1, NEW state is q_1.
output = 1 OR 0 = 1
If we are in state q_3, and input is 0
NEW Y = old X = 1
NEW X = 0 NAND (1 OR 1) = 1
X = 1, Y = 1, New state is q_3.
output = 0 OR 1 = 1
If we are in state q_3, & input is 1
New Y = old X = 1
New X = 1 NAND (1 OR 1) = 0
X = 0, Y = 1, new state is q_1.
output = 1 OR 1 = 1.

Present state	Next state			
	input 0		Input 1	
	state	output	state	output
q_0	q_2	0	q_2	1
q_1	q_2	1	q_0	1
q_2	q_3	0	q_1	1
q_3	q_3	1	q_1	1

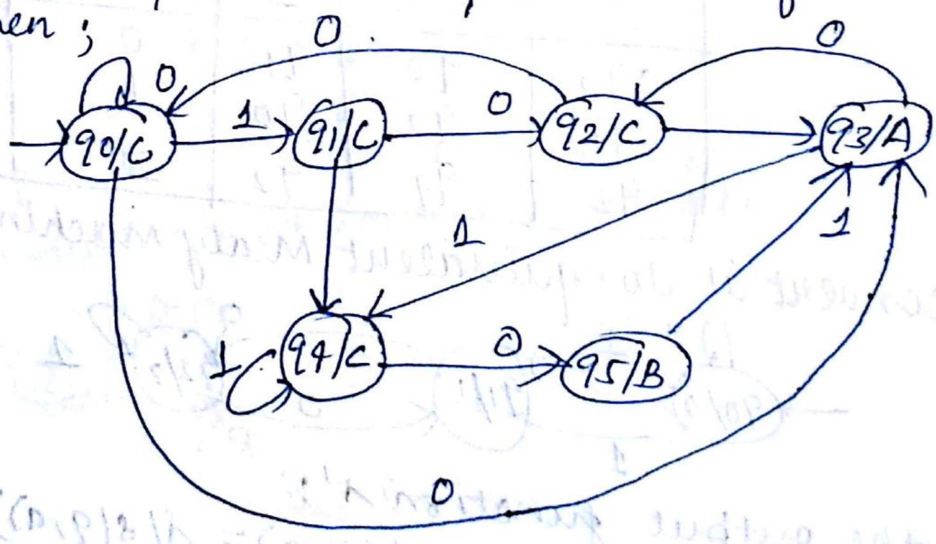
transition diagram:



Q. Design a Moore machine for binary input sequence such that if it has a substring 101 the machine outputs A if input has substring 110 it outputs B otherwise it outputs C.



we will insert possibilities of 1's and 0's for each state then ;



(P-160)

Q. Consider the FA given by table. Convert this FA into a Moore machine.

State	Next state	
	$a=0$	$a=1$
$\rightarrow q_0$	q_2	q_0
q_1	q_0	q_1
q_2	q_0	q_2

Sol: Let $\Delta = \{0,1\}$ be introduced alphabet and the value of output is defined by output function $\lambda(q)$ as $\lambda(q_0)=1$, $\lambda(q_1)=0$ $\lambda(q_2)=0$

The Moore machine is defined as,

State	Next state		output
	$a=0$	$a=1$	
$\rightarrow q_0$	q_2	q_0	1
q_1	q_0	q_1	0
q_2	q_0	q_2	0

Q. The Moore machine to determine residue mod 3 for binary number is given as

P.S	N.S		Output (λ)
	0	1	
$\rightarrow q_0$	q_0	q_1	0
q_1	q_2	q_0	1
q_2	q_1	q_2	2

Convert it to equivalent Mealy Machine.

Sol:

The output function λ':

$$\lambda'(q,a) = \lambda(\delta(q,a))$$

Scanned by CamScanner

$\lambda'(q_0, 0) = \lambda(\delta(q_0, 0))$
$= \lambda(q_0)$ i.e, output of q_0.

$\boxed{\lambda'(q_0, 0) = 0}$.

$\lambda'(q_0, 1) = \lambda(\delta(q_0, 1))$
$= \lambda(q_1)$ i.e. output of q_1.

$\boxed{\lambda'(q_0, 1) = 1}$

$\lambda'(q_1, 0) = \lambda(\delta(q_1, 0))$
$= \lambda(q_2)$

$\boxed{\lambda'(q_1, 0) = 2}$

$\lambda'(q_1, 1) = \lambda(\delta(q_1, 1))$
$= \lambda(q_0)$

$\boxed{\lambda'(q_1, 1) = 0}$

$\lambda'(q_2, 0) = \lambda(\delta(q_2, 0))$
$= \lambda(q_1)$

$\boxed{\lambda'(q_2, 0) = 1}$

$\lambda'(q_2, 1) = \lambda(\delta(q_2, 1))$
$= \lambda(q_2)$

$\boxed{\lambda'(q_2, 1) = 2}$

P.S	NS			
	0		1	
	state	O/P	state	O/P
q_0	q_0	0	q_1	1
q_1	q_2	2	q_0	0
q_2	q_1	1	q_2	2

transition diagram of Mealy machine is

Scanned by CamScanner

Q. Construct a Mealy machine which is equivalent to the Moore machine

PS	Next state a = 0	Next state a = 1	output
$\rightarrow q_0$	q_1	q_2	1
q_1	q_3	q_2	0
q_2	q_2	q_1	1
q_3	q_0	q_3	1

Sol: The output function λ'

$$\lambda'(q,a) = \lambda(\delta(q,a))$$

for every transition corresponding to input symbol

$\lambda'(q_0,0) = \lambda(\delta(q_0,0))$
$\lambda(q_1)$ i.e. output of q_1
$= 0$

$\lambda'(q_0,1) = \lambda(\delta(q_0,1))$
$= \lambda(q_2) = 1$

$\lambda'(q_1,1) = \lambda(\delta(q_1,1))$
$\lambda(q_2) = 1$

$\lambda'(q_1,0) = \lambda(\delta(q_1,0))$
$= \lambda(q_3) = 1$

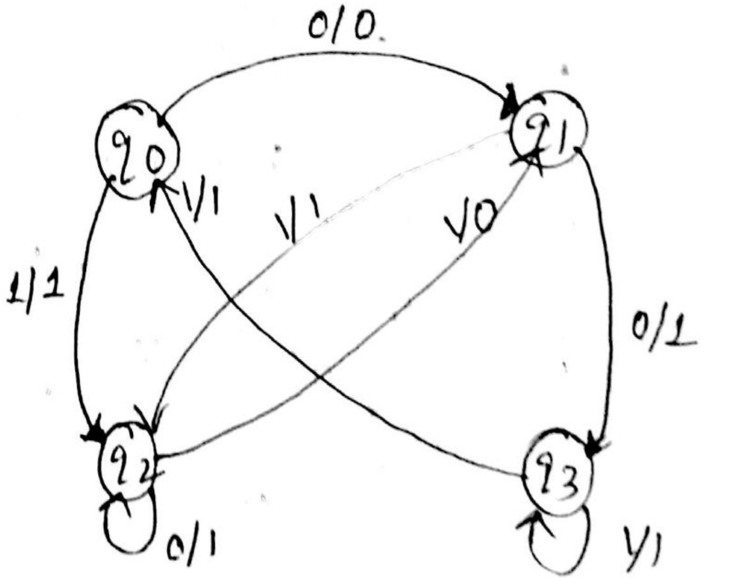
$\lambda'(q_2,0) = \lambda(\delta(q_2,0))$
$= \lambda(q_2) = 1$

$\lambda'(q_2,1) = \lambda(\delta(q_2,1))$
$= \lambda(q_1) = 0$

$\lambda'(q_3,0) = \lambda(\delta(q_3,0))$
$= \lambda(q_0) = 1$

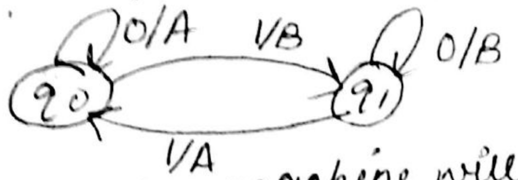
$\lambda'(q_3,1) = \lambda(\delta(q_3,1))$
$= \lambda(q_3) = 1$

P.S	Next state: input 0 state	input 0 O/P	input 1 state	input 1 O/P
q_0	q_1	0	q_2	1
q_1	q_3	1	q_2	1
q_2	q_2	1	q_1	0
q_3	q_0	1	q_3	1



(P.163)

Q Convert the following Mealy machine into equivalent Moore machine.



Sol: The states for Moore machine will be Q X A

[q0, A], [q0, B], [q1, A], [q1, B]

we calculate δ' and λ' as follows:

$\delta'([q_0, A], 0) = [\delta(q_0, 0), \lambda(q_0, 0)]$
$= [q_0, A]$

$\lambda'([q_0, A]) = A$

$\delta'([q_0, B], 0) = [\delta(q_0, 0), \lambda(q_0, 0)]$
$= [q_0, A]$

$\lambda'([q_0, B]) = B$

$\delta'([q_0, A], 1) = [\delta(q_0, 1), \lambda(q_0, 1)]$
$= [q_1, B]$

$\lambda'([q_0, A]) = A$

$\delta'([q_0, B], 1) = [\delta(q_0, 1), \lambda(q_0, 1)]$
$= [q_1, B]$

$\lambda'([q_0, B]) = B$

$\delta'([q_1, A], 0) = [\delta(q_1, 0), \lambda(q_1, 0)]$
$= [q_1, B]$

$\lambda'([q_1, A]) = A$

$\delta'([q_1, A], 1) = [\delta(q_1, 1), \lambda(q_1, 1)]$
$= [q_0, A]$

& $\lambda'([q_1, A]) = A$

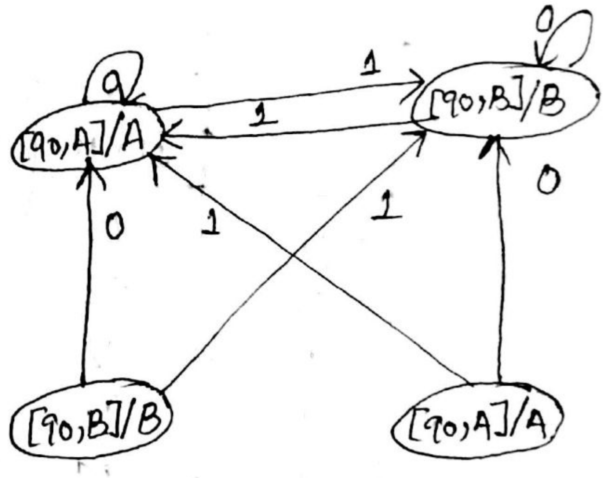
$\delta'([q_1, B], 0) = [\delta(q_1, 0), \lambda(q_1, 0)]$
$= [q_1, B]$

$\lambda'([q_1, B]) = B$

$\delta'([q_1, B], 1) = [\delta(q_1, 1), \lambda(q_1, 1)]$
$= [q_0, A]$

$\lambda'([q_1, B]) = A$

PS	NS 0	NS 1	O/P (λ')
[q0, A]	[q0, A]	[q1, B]	A
[q0, B]	[q0, A]	[q1, B]	B
[q1, A]	[q1, B]	[q0, A]	A
[q1, B]	[q1, B]	[q0, A]	B



Q: Consider the mealy machine given by following table:

PS	Next state			
	a = 0		a = 1	
	state	O/P	state	O/P
→ q_1	q_3	1	q_2	0
q_2	q_1	1	q_4	1
q_3	q_2	0	q_1	1
q_4	q_4	1	q_3	0

construct a Moore machine equivalent to this given mealy machine.

Sol: In this, look onto next column for those states which are associated with more than one output. There is only one state q_3 which is associated with two output 0 & 1. So we split q_3 into q_{30} and q_{31}. Now we will construct a transition table by introducing new states q_{30} and q_{31} in place of q_3.

P.S	NS			
	a = 0		a = 1	
	state	O/P	state	O/P
→ q_1	q_{31}	1	q_2	0
q_2	q_1	1	q_4	1
q_{31}	q_2	0	q_1	1
q_{30}	q_2	0	q_1	1
q_4	q_4	1	q_{30}	0

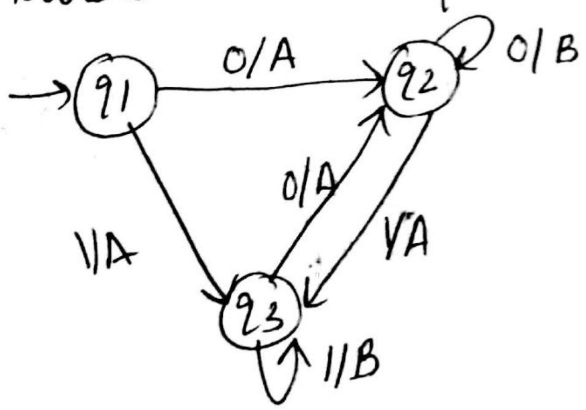
Now, Mealy machine has 5 states q_1, q_2, q_{31}, q_{30} & q_4 & we transform this table onto Moore machine by eliminating outputs within next state column. Here, we introduce a new 'output' column adjacent to Next state column & the values in this output column are corresponding to 'present state' column.

PS	NS		O/P
	a = 0	a = 1	
→ q_1	q_{31}	q_2	1
q_2	q_1	q_4	0
q_{31}	q_2	q_1	1
q_{30}	q_2	q_1	0
q_4	q_4	q_{30}	1

This table gives the Moore machine. Here we observe that the initial state q_1 is associated with output 1. This means that with output $\wedge$, we get an output of 1. Thus this Moore machine accepts an empty string which is not accepted by the Mealy Machine. To overcome this situation, either we must neglect the response of a Moore machine to input $\wedge$ or we must add a new starting state q_0, whose state transitions are identical with those of q_1 but whose output is 0

PS	Next state		output
	a = 0	a = 1	
→ q_0	q_{31}	q_2	0
q_1	q_{31}	q_2	1
q_2	q_1	q_4	0
q_{31}	q_2	q_1	1
q_{30}	q_2	q_1	0
q_4	q_4	q_{30}	1

Q. Construct a Moore machine equivalent to Mealy Machine given as



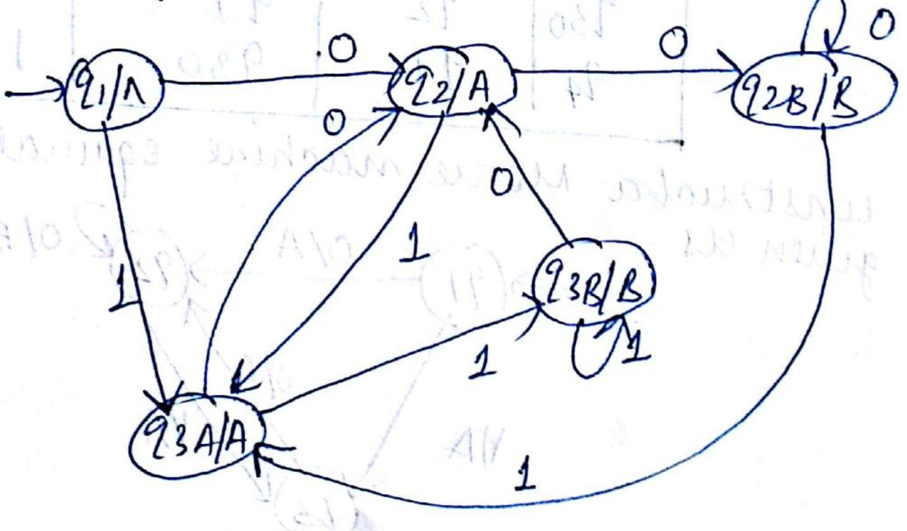
The transition table for given Mealy machine is

PS	Next state			
	a = 0		a = 1	
	state	O/P	state	O/P
→ q_1	q_2	A	q_3	A
q_2	q_2	B	q_3	A
q_3	q_2	A	q_3	B

here, q_2 is associated with two different outputs A & B and q_3 is associated with two different outputs A & B. Thus we must split q_2 into q_{2A} & q_{2B} with A & B outputs respectively and q_3 into q_{3A} & q_{3B} with outputs A & B respectively.

P.S	Next state		O/P
	a = 0	a = 1	
q_1	q_{2A}	q_{3A}	A
q_{2A}	q_{2B}	q_{3A}	A
q_{2B}	q_{2B}	q_{3A}	B
q_{3A}	q_{2A}	q_{3B}	A
q_{3B}	q_{2A}	q_{3B}	B

The transition diagram:



Equivalence of DFA:-

Two finite automata M_1 and M_2 are said to be equivalent if they accepts the same language.

i.e. $\boxed{L(M_1) = L(M_2)}$

Two finite automata M_1 and M_2 are not equivalent over Σ if there exists a w such that:

$$w \in L(M_1) \text{ and } w \notin L(M_2)$$
or
$$w \notin L(M_1) \text{ and } w \in L(M_2)$$

That is one DFA reaches a final state on application of w and other reaches a non final state.

Thus, if two DFAs are equivalent, then for every $w \in \Sigma^*$.

- On application of w, either both M_1 and M_2 will be in a final state or both M_1, and M_2 will be in non final state, simultaneously.

Tesing of Equivalence of two DFA's:

Equivalence of two DFA's can be established by constructing a combined DFA for two DFA M_1 and M_2. Let the combined DFA for M_1 and M_2 be M_3.

where $M_1 = (S, \Sigma, \delta_1, S_0, F_1)$

and $M_2 = (Q, \Sigma, \delta_2, q_0, F_2)$ The combined machine M_3 can be described in terms M_1 and M_2.

$$M_3 = (S \times Q, \Sigma, \delta_3, \langle S_0, q_0 \rangle, F_3)$$

- A state of M_3 is of the form $\langle s_i, q_j \rangle$.
  where $s_i \in S$ of M_1 and $q_i \in Q$ of M_2.

The transition function δ_3 is defined as:

$$\delta_3(\langle s_j, q_i \rangle, a) = \langle \delta_1(s_j, a), \delta_2(q_i, a) \rangle$$

δ_3 is a function from $S \times Q$ to $S \times Q$.

The algorithm for generation of transition behaviour δ_3 of M_3 is given below;

1. Add the state $\langle s_0, q_0 \rangle$ to M_3.
2. for every state $\langle s_i, q_i \rangle$ in M_3.

{ if (< s_i, q_i > has not been expanded).

{ for each alphabet $a_i \in \Sigma$.

{ add a transition < $\delta_1(s_i, a_i)$, $\delta(q_j, a_i)$ to M_3

}

}

}

M_1 and M_2 are equivalent if:

for every state < s_i, q_i > in M_3.

{ both s_i and q_i are final states

or

both s_i and q_i are non final state.

}.

M_1 and M_2 are not equivalent if:

for any state < s_i, q_i > in M_3

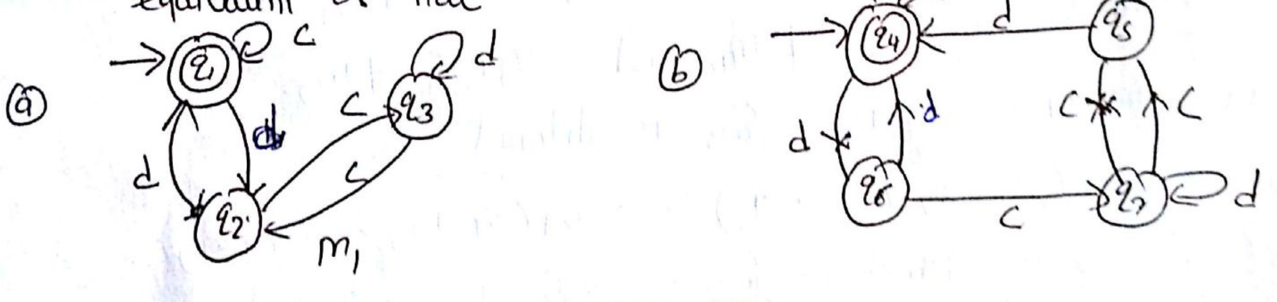
{ s_i is a final state and q_i is a non-final state.

or

s_i is a non-final state and q_i is a final state.

}

Ques:-> Show whether the automata M_1 and M_2 in given fig (a) and (b) are equivalent or not



Solution:-> Construction of Combined DFA M_3:->

Step1:- we start with the combined state < q_1, q_4 >, where q_1 is start state of M_1 and q_4 is the start state of M_2 - < q_1, q_4 > is expanded and necessary transition added.