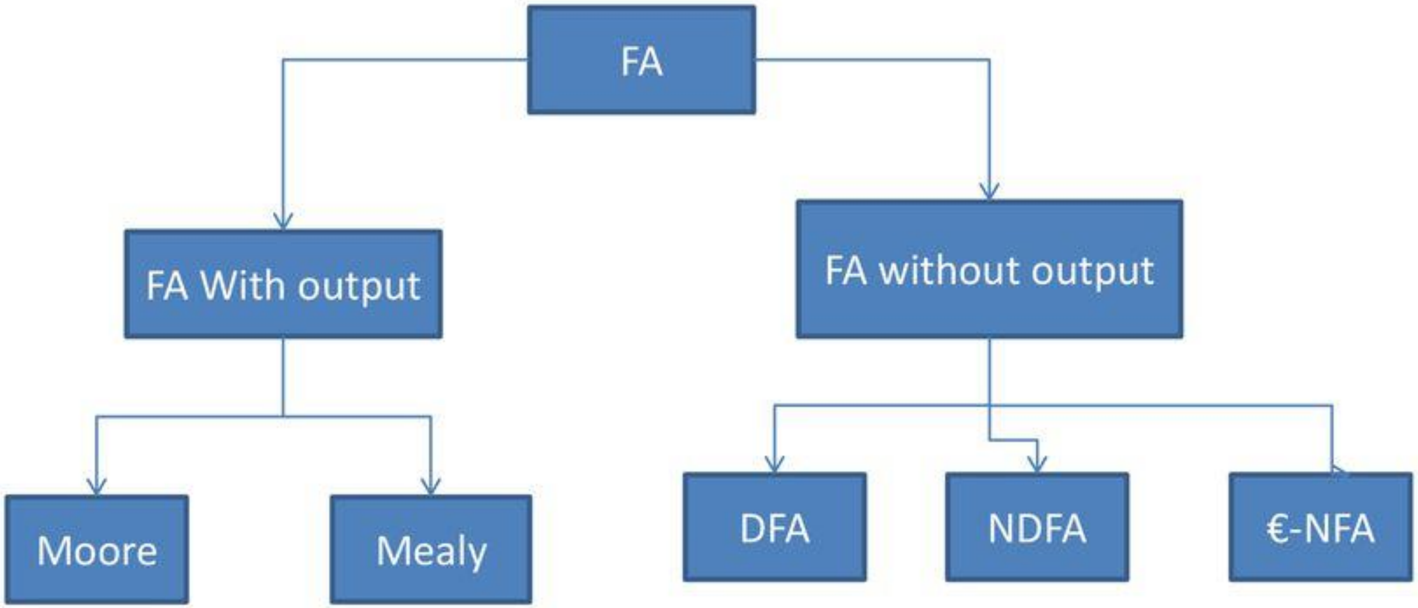
FA With Output… Moore Machine

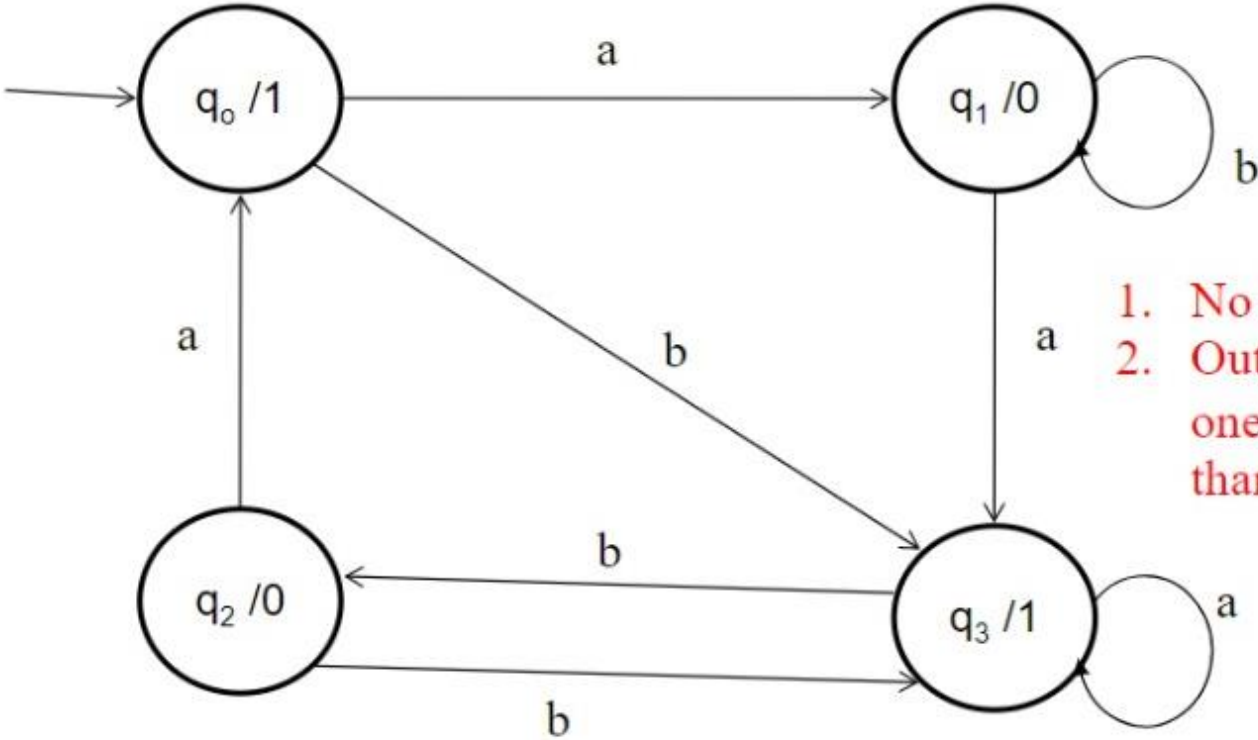
-- Sakshi Surve


FA
FA With output
FA without output
Moore
Mealy
DFA
NDFA
€-NFA

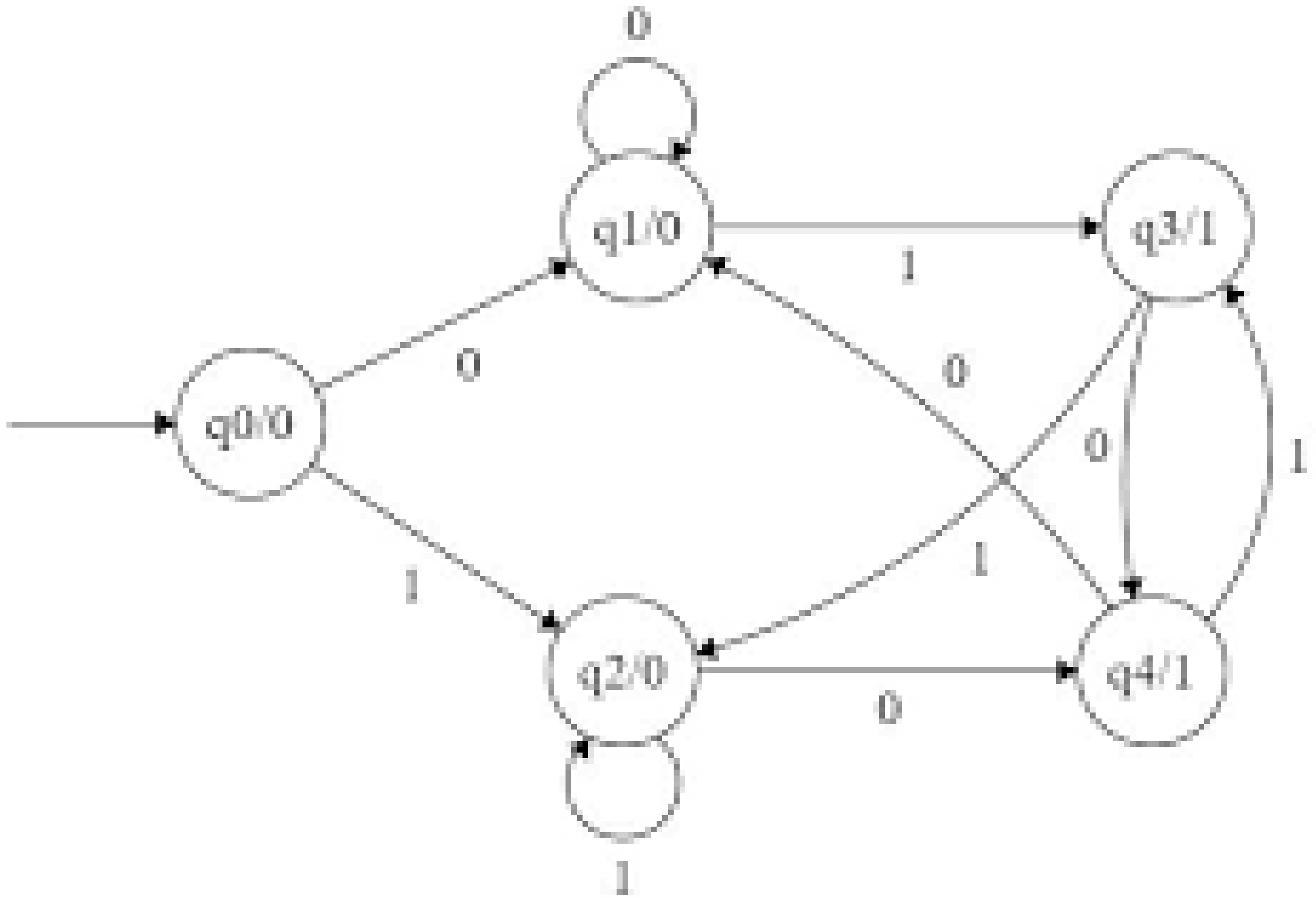
Moore machine – example

Input: ~~a~~ ~~b~~ ~~b~~ ~~a~~

Output: 1 0 0 0 1



1. No final state
2. Output is always one letter more than the input


0
q1/0
1
q3/1
0
q0/0
0
0
1
1
1
q2/0
0
q4/1
1

Moore Machine

_Moore Machine is six-tuple $(Q,\sum,\Delta,\delta,\lambda,q_0)$:

(i) Q is a finite set of states

(ii) $\sum$ is the input alphabet

(iii) Δ is the output alphabet

(iv) δ is the transition function from $\sum$ X Q into Q

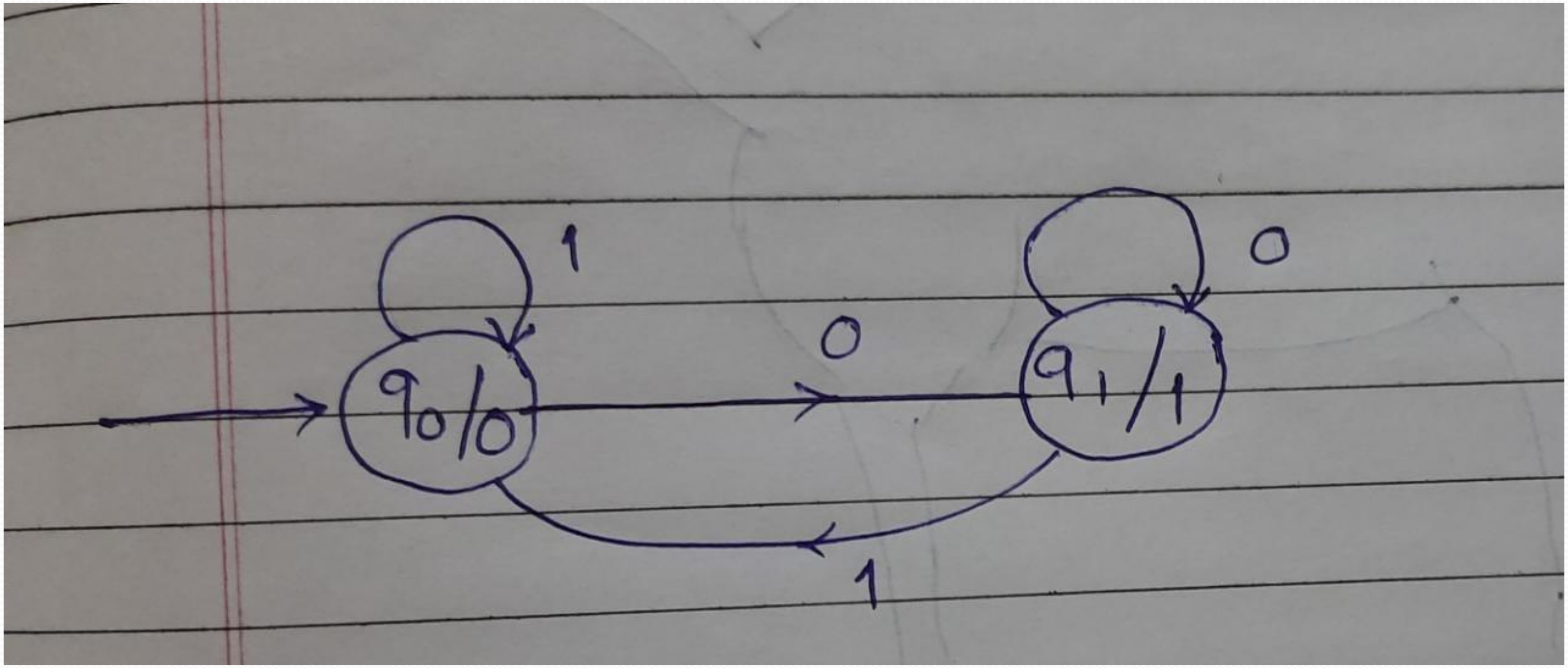
(v) λ is the output function mapping Q into Δ and

(vi) q_0 is the initial state

- **Mapping Functions :**

Current State	Next State (δ)		Output(λ)
	0	1	
q_0	q_1	q_2	1
q_1	q_2	q_1	1
q_2	q_2	q_0	0

- Design Moore Machine to find one's complement of a binary input string



- Transition Mapping Function

δ	0	1
q_0	q_1	q_0
q_1	q_1	q_0

- Output Mapping Function

λ	Δ
q_0	0
q_1	1

Example

1 0 1 0

Applying δ to i/p string

$\Rightarrow \delta(1 \quad 0 \quad 1 \quad 0)$

$q_0 \quad q_0 \quad q_1 \quad q_0 \quad q_1$

Now, leaving the initial state & Applying 'λ' to resultant states, we get -

$\lambda(q_0 \quad q_1 \quad q_0 \quad q_1)$

$\Rightarrow \quad 0 \quad 1 \quad 0 \quad 1$

∴ I/p string - 1 0 1 0

O/p string - 0 1 0 1.

0 0 0

$\delta(0 \quad 0 \quad 0)$

$q_0 \quad q_1 \quad q_1 \quad q_1$

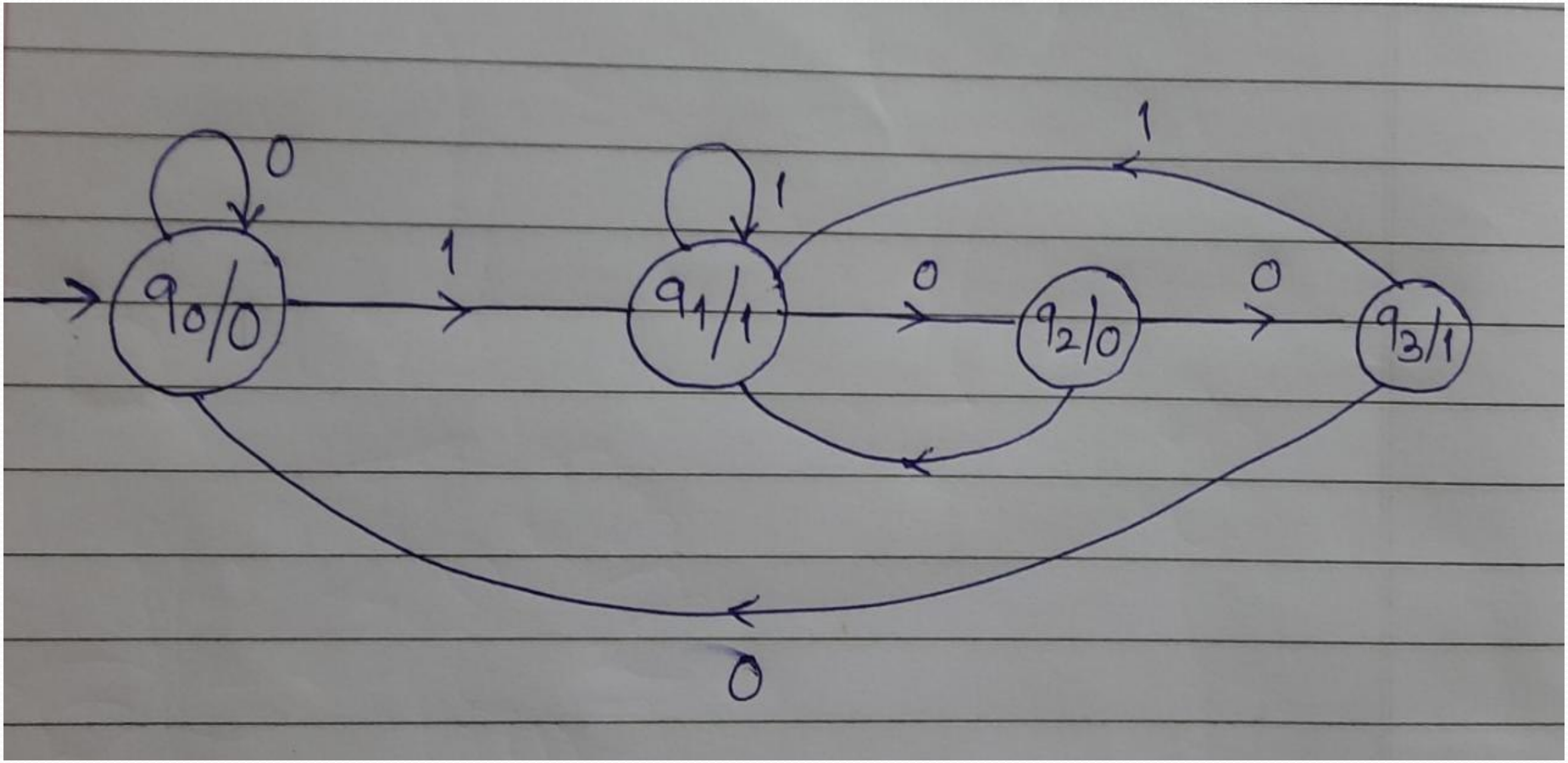
$\lambda(q_1 \quad q_1 \quad q_1)$

$\Rightarrow \quad 1 \quad 1 \quad 1$

I/p string - 0 0 0

O/p string - 1 1 1

- Design Moore Machine to convert every occurrence of 100 to 101



- Transition Mapping Function

δ	0	1
q_0	q_0	q_1
q_1	q_2	q_1
q_2	q_3	q_1
q_3	q_0	q_1

- Output Mapping Function

λ	Δ
q_0	0
q_1	1
q_2	0
q_3	1

Example -

0 1 0 0

$$\delta(0\ 1\ 0\ 0)$$

$q_0 \quad q_0 \quad q_1 \quad q_2 \quad q_3$

Now, leaving the initial state & Applying λ to resultant states, we get,

$$\lambda(q_0\ q_1\ q_2\ q_3)$$

$\Rightarrow$ 0 1 0 1

$\therefore$ i/p string - 0 1 0 0
o/p string - 0 1 0 1

Example -

1 0 0 1

$$\delta(1\ 0\ 0\ 1)$$

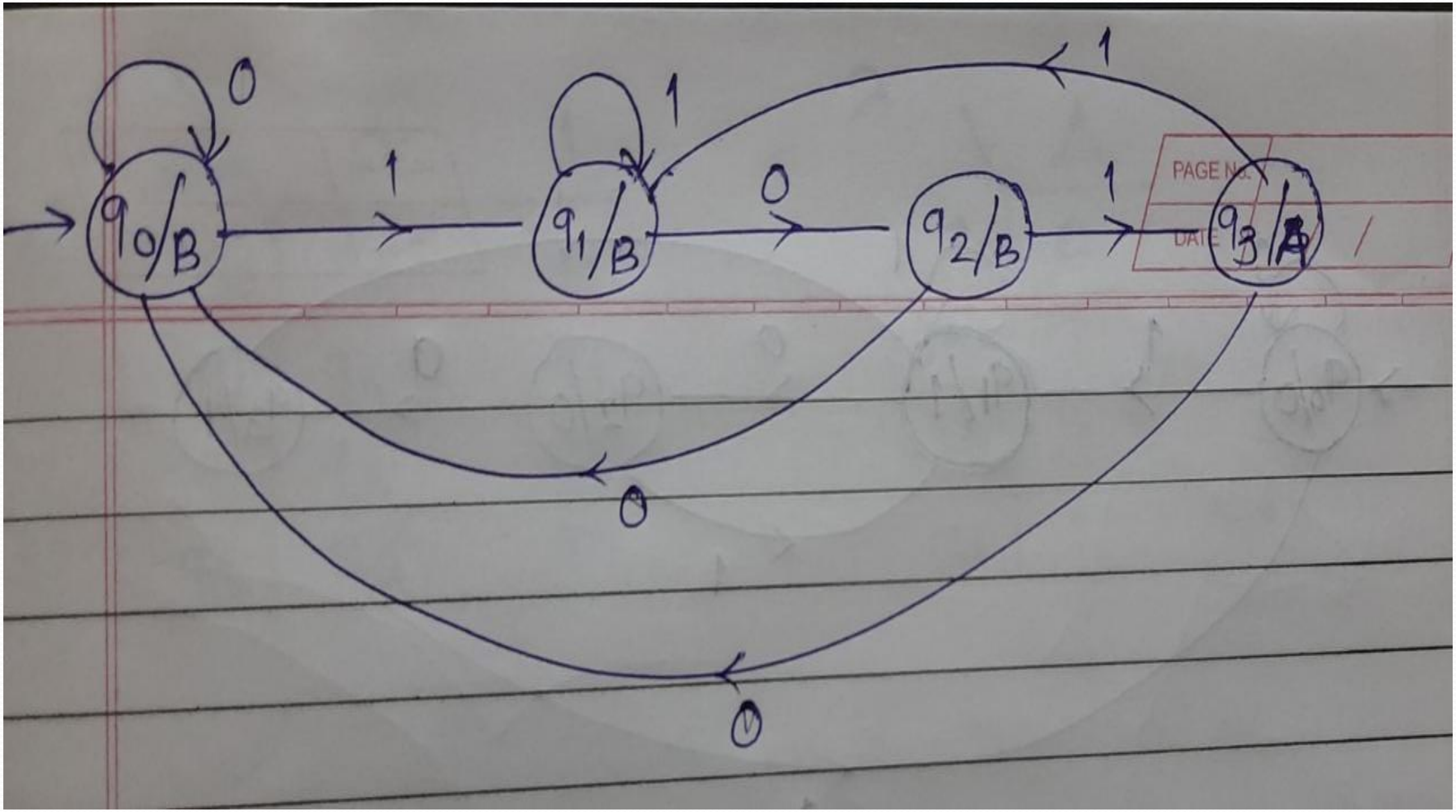
$q_0 \quad q_1 \quad q_2 \quad q_3 \quad q_1$

$$\lambda(q_1\ q_2\ q_3\ q_1)$$

$\Rightarrow$ 1 0 1 1

$\therefore$ I/p string - 1 0 0 1
o/p string - 1 0 1 1

- Design Moore Machine which outputs ‘A’ if ‘101’ is recognized , otherwise outputs ‘B’



- Transition Mapping Function

$\delta \Rightarrow$	0	1
q_0	q_0	q_1
q_1	q_2	q_1
q_2	q_0	q_3
q_3	q_0	q_1

- Output Mapping Function

$\lambda \Rightarrow$	Δ
q_0	B
q_1	B
q_2	B
q_3	A

Example -

1 0 1 1

$\delta(1\ 0\ 1\ 1)$

$q_0\ \ q_1\ \ q_2\ \ q_3\ \ q_4$

$\lambda(q_1\ q_2\ q_3\ q_1)$

$\Rightarrow$ B B A B

$\therefore$ I/P :- 1 0 1 1

O/P :- B B A B

1 1 1 1

$\delta(1\ 1\ 1\ 1)$

$q_0\ \ q_1\ \ q_1\ \ q_1\ \ q_1$

$\lambda(q_1\ q_1\ q_1\ q_1)$

$\Rightarrow$ B B B B

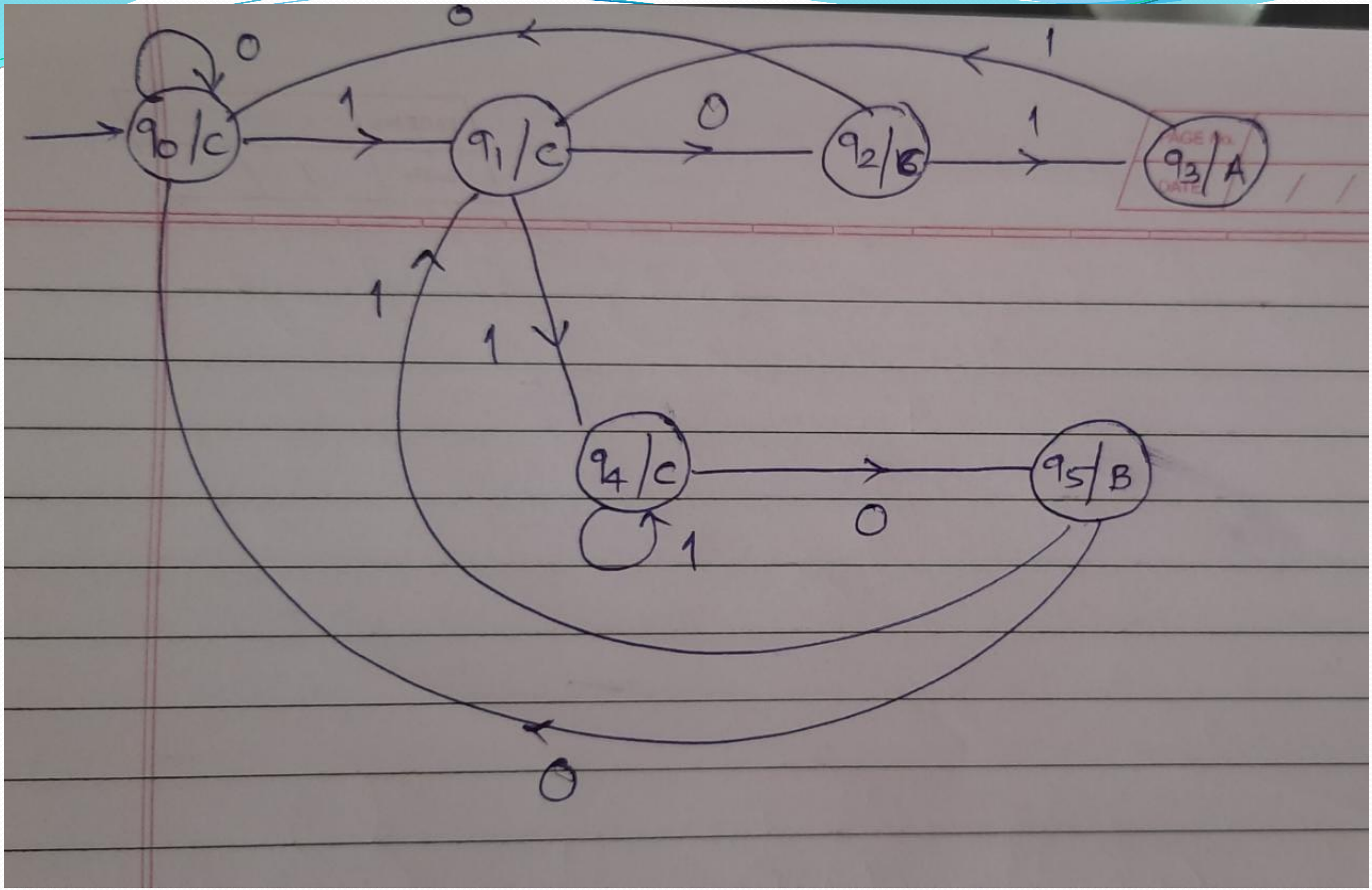
$\therefore$ I/P - 1 1 1 1

O/P - B B B B

Homework

- Design Moore Machine which outputs 'A' if '101' is recognized , outputs 'B' if '110' is recognized, and 'C' otherwise

- Design a Moore machine to convert every occurrence of '121' to '122' over $\sum = \{1,2,3\}$



q0/c
q1/c
q2/B
q3/A
q4/c
q5/B
0
1

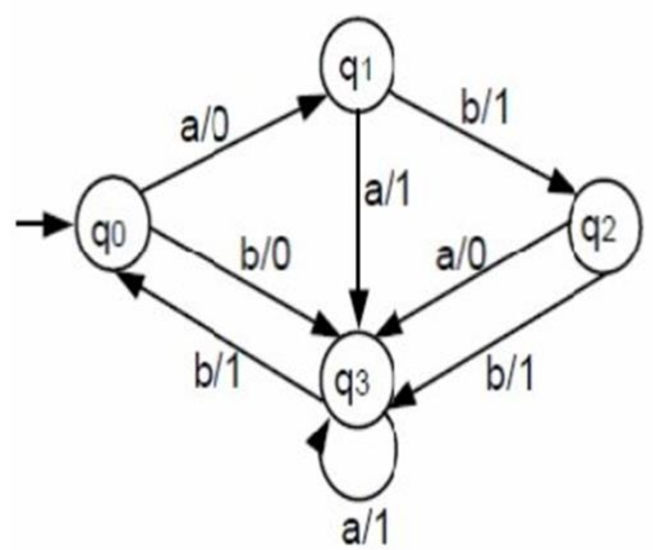
FA with Output … Mealy Machine

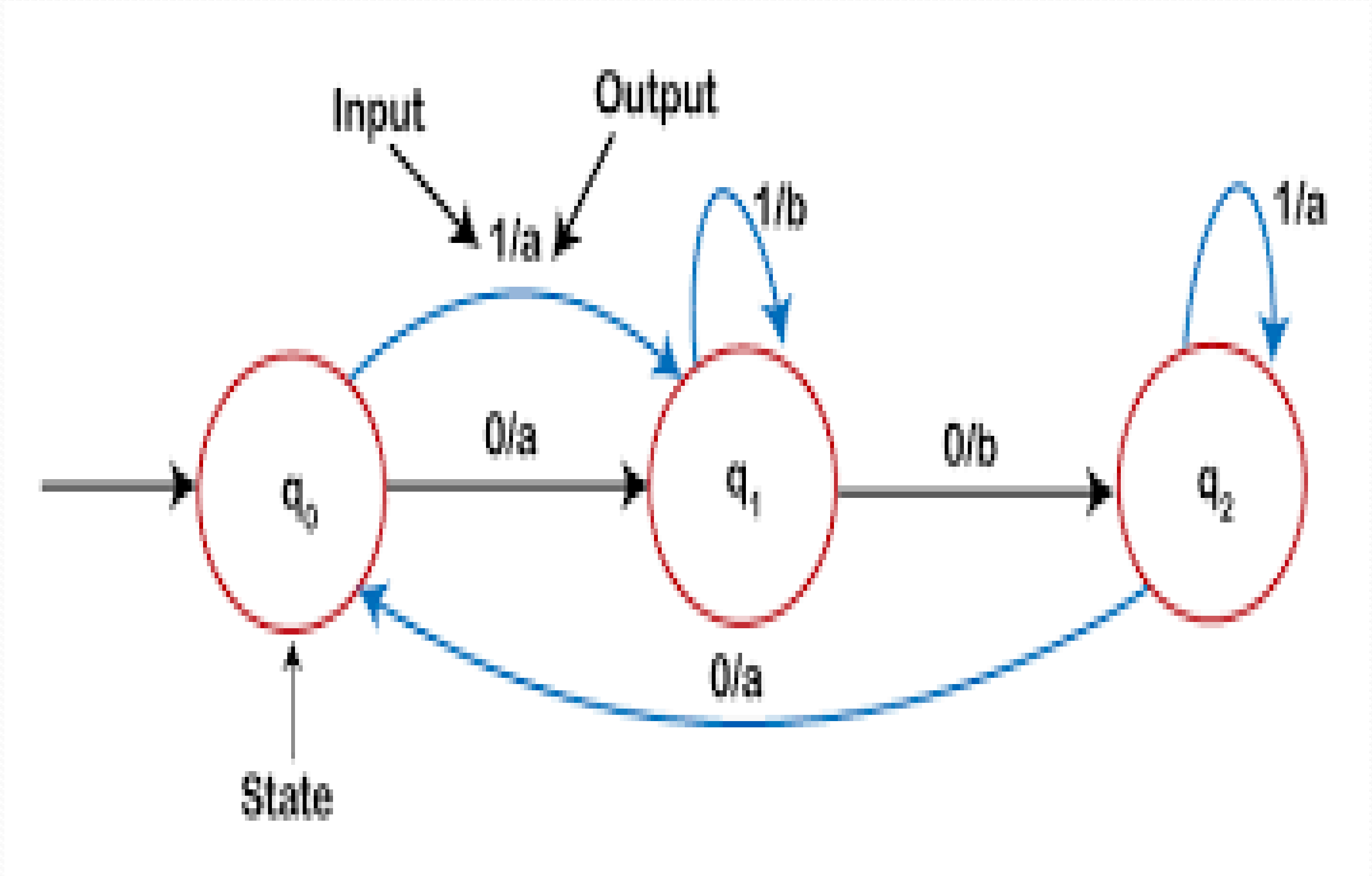
--- **Sakshi Surve**

- **FA with Output associated with a State :**
  - Moore Machine
- **FA with Output associated with a Transition :**
  - Mealy Machine

FA With Output- Mealy Machine

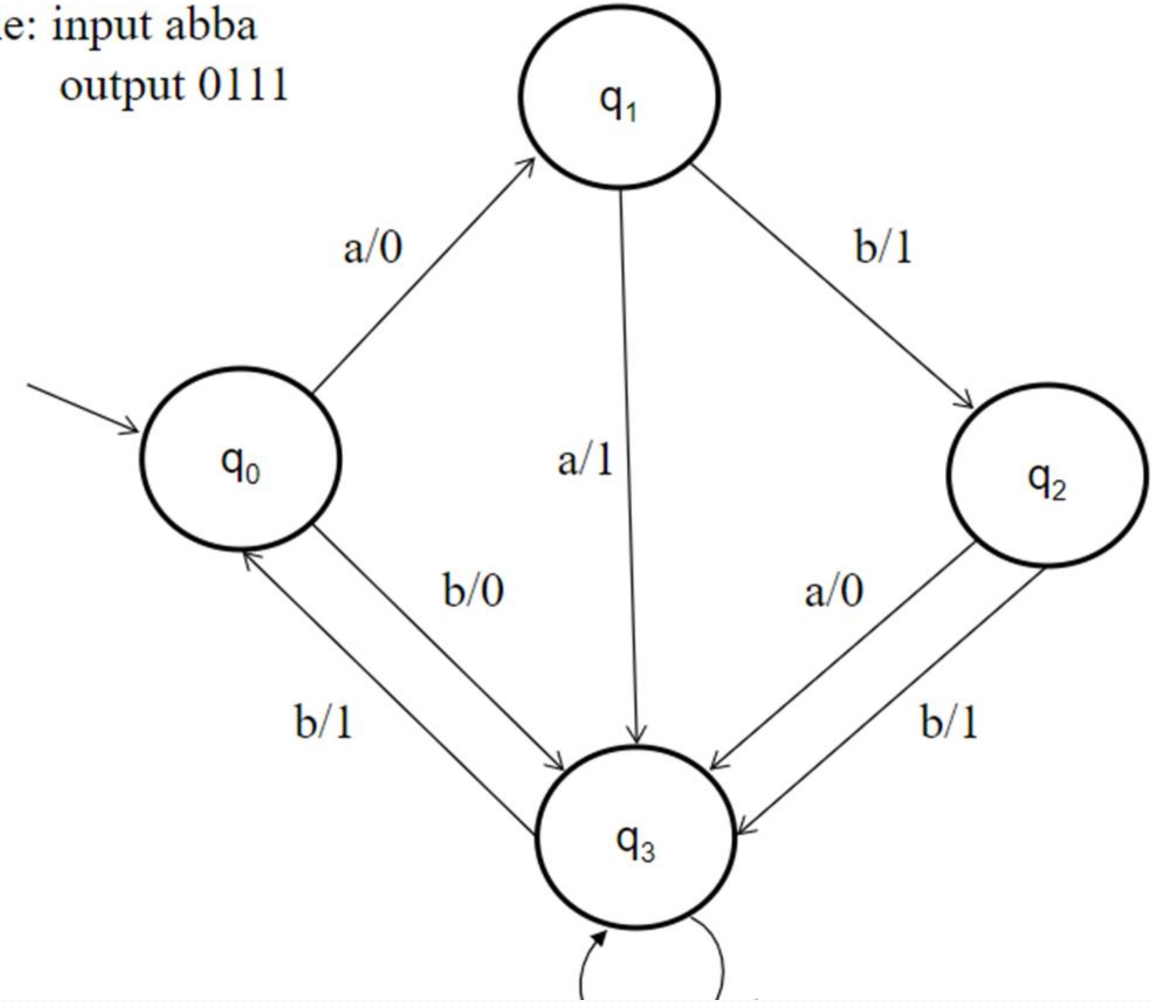
- output on edge
- same number of input to output
- Input: a a a b b
- Output: 0 1 1 1 0





Input
Output
1/a
1/b
1/a
0/a
0/b
q0
q1
q2
0/a
State

Example: input abba
output 0111



Definition:

Mealy Machine

A Mealy machine is defined by the sextuple

$$M = (Q, \Sigma, r, \delta, \theta, q_o)$$

Where

Q is a finite set of internal states,

Σ is the input alphabet,

R is the output alphabet,

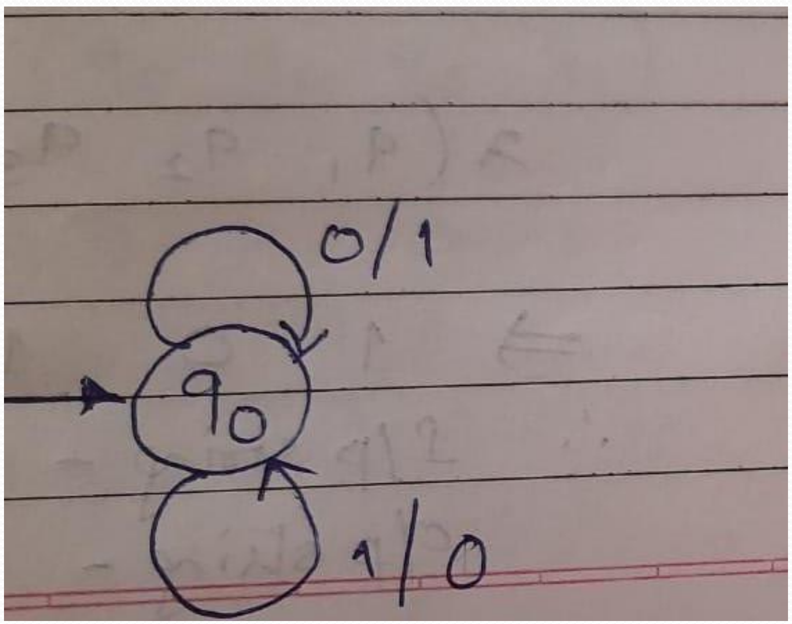
$\delta : Q \times \Sigma \rightarrow Q$ is the transition function.

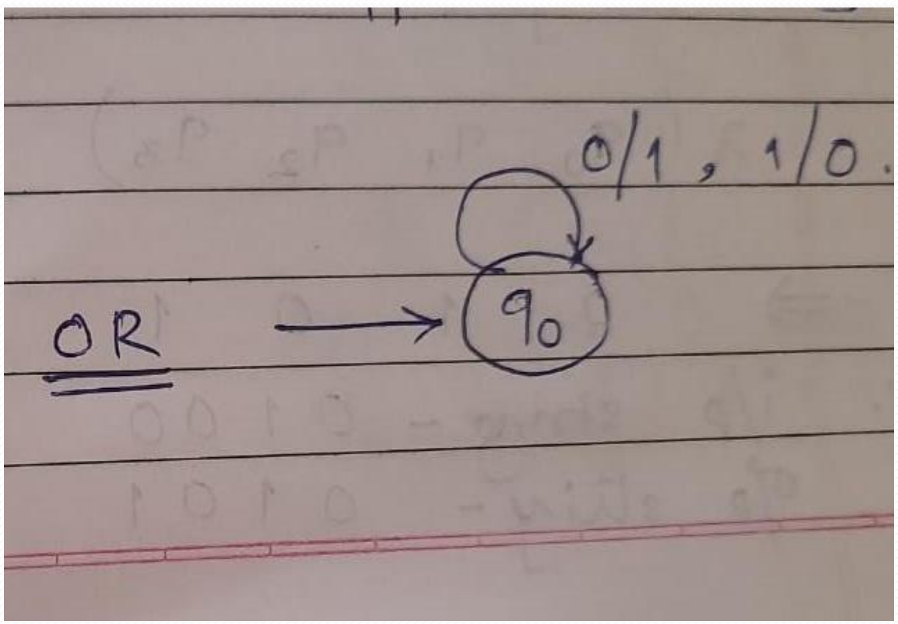
$\theta : Q \times \Sigma \rightarrow r$ is the output function.

$q_o \in Q$ is the initial state of M.

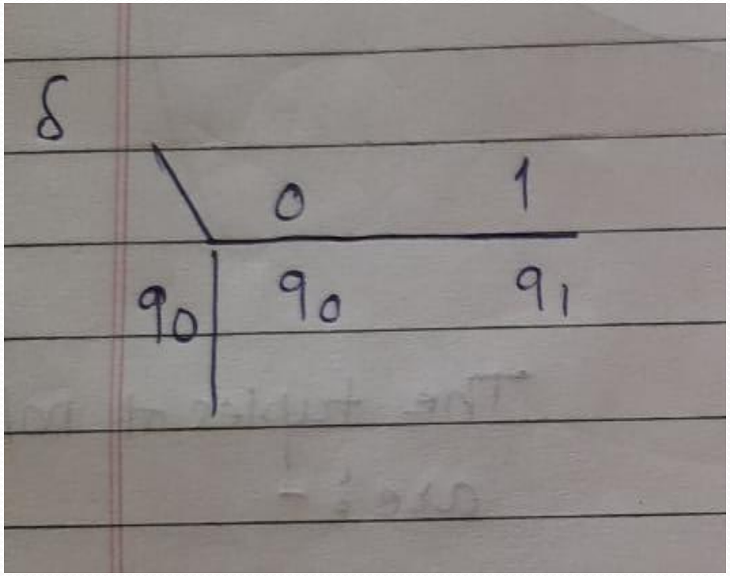
State	Next State			
	Input = 0		Input = 1	
	State	Output	State	Output
→ q0	q2	b	q1	b
q1	q3	b	q2	b
q2	q2	b	q2	b
q3	q2	b	q0	a

- Design Mealy Machine to find one's complement of a binary input string



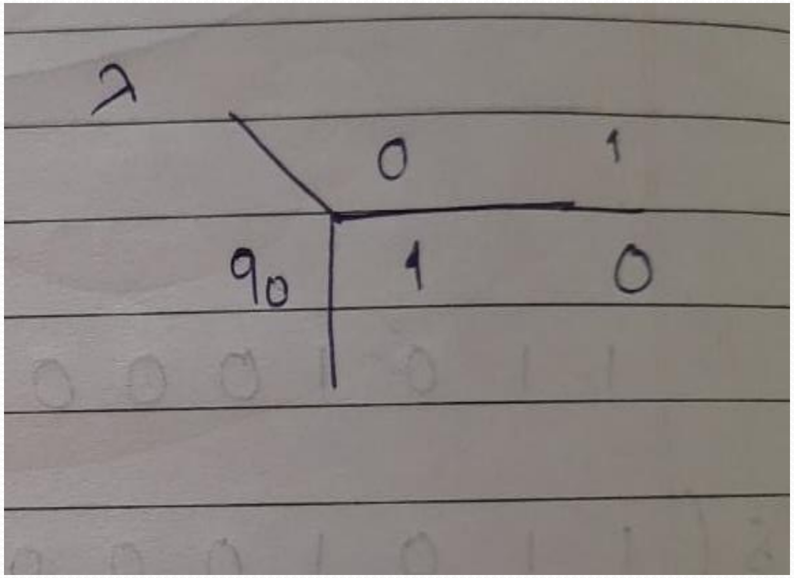


- Transition Mapping Function
- Output Mapping Function



δ

	0	1
q_0	q_0	q_1



λ

	0	1
q_0	1	0

- Examples :

1 0 1 0

Applying 'δ' to input

$\delta(1\ 0\ 1\ 0)$

$q_0\quad q_0\quad q_0\quad q_0\quad q_0$

Now, excluding the last state, applying λ to the resultant states, we get -

$\cancel{\lambda(q_0\ q_0\ q_0\ q}$

$\lambda(q_0, 1) = 0$

$\lambda(q_0, 0) = 1$

$\lambda(q_0, 1) = 0$

$\lambda(q_0, 0) = 1.$

∴ I/p string = 1 0 1 0

O/p string = 0 1 0 1.

0 0 0

$\delta(0\ 0\ 0)$

$q_0\quad q_0\quad q_0\quad q_0$

$\lambda(q_0, 0) = 1$

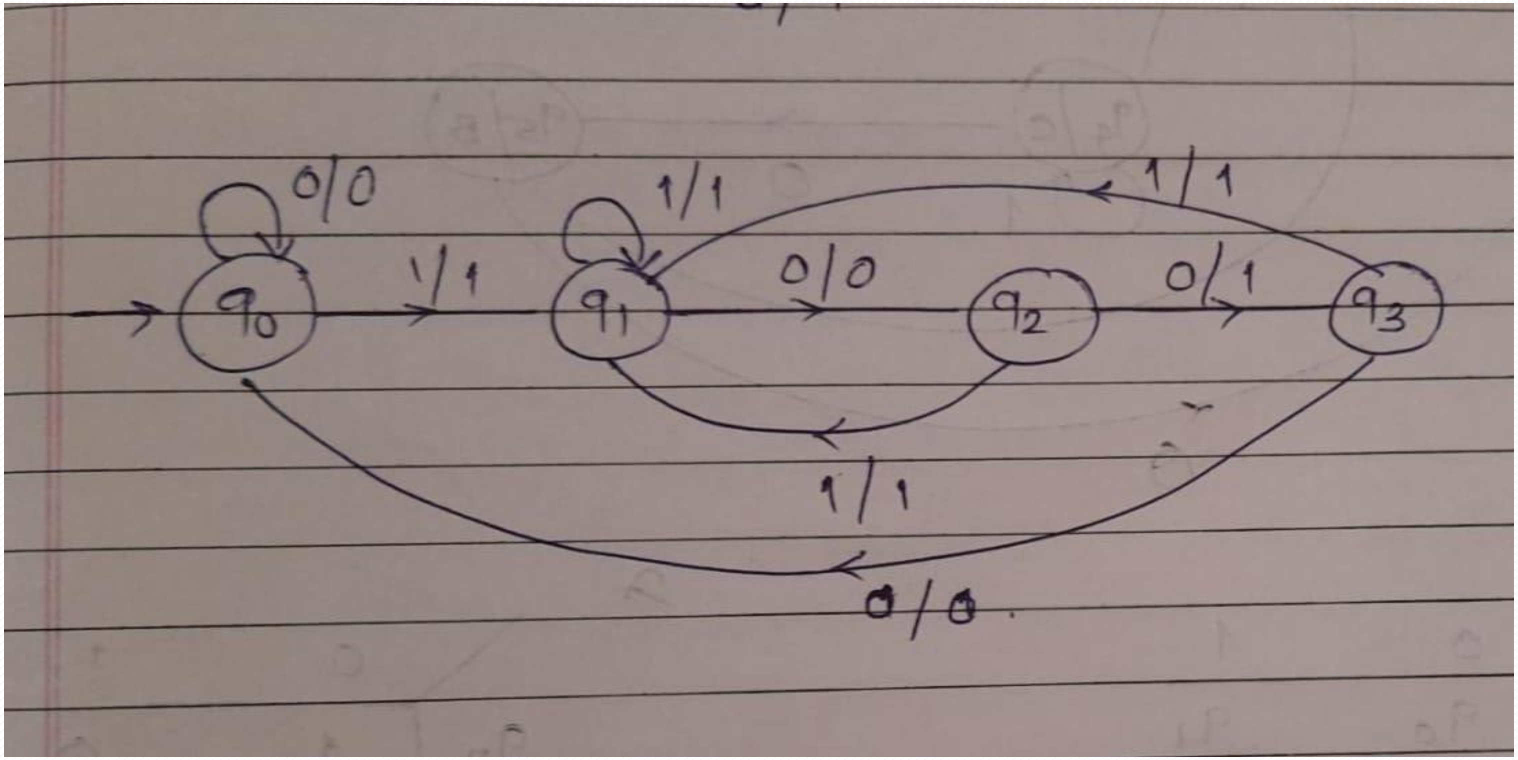
$\lambda(q_0, 0) = 1$

$\lambda(q_0, 0) = 1.$

∴ I/p string = 0 0 0

O/p string = 1 1 1

- Design Mealy Machine to convert every occurrence of 100 to 101



- Transition Mapping Function

$\delta \Rightarrow$	0	1
q_0	q_0	q_1
q_1	q_2	q_1
q_2	q_3	q_1
q_3	q_0	q_1

- Output Mapping Function

$\lambda \Rightarrow$	0	1
q_0	0	1
q_1	0	1
q_2	1	1
q_3	0	0

1 1 0 1 0 0 0 1

δ (1 1 0 1 0 0 0 1)

q_0 q_1 q_1, q_2 q_1, q_2 q_3 q_0 q_1

$\lambda(q_0, 1) = 1$

$\lambda(q_1, 1) = 1$

$\lambda(q_1, 0) = 0$

$\lambda(q_2, 1) = 1$

$\lambda(q_1, 0) = 0$

$\lambda(q_2, 0) = 1$

$\lambda(q_3, 0) = 0$

$\lambda(q_0, 1) = 1$

$\therefore$ I/p $\Rightarrow$ 1 1 0 1 0 0 0 1

O/p $\Rightarrow$ 1 1 0 1 0 1 0 1

The tuples of m/c are :-

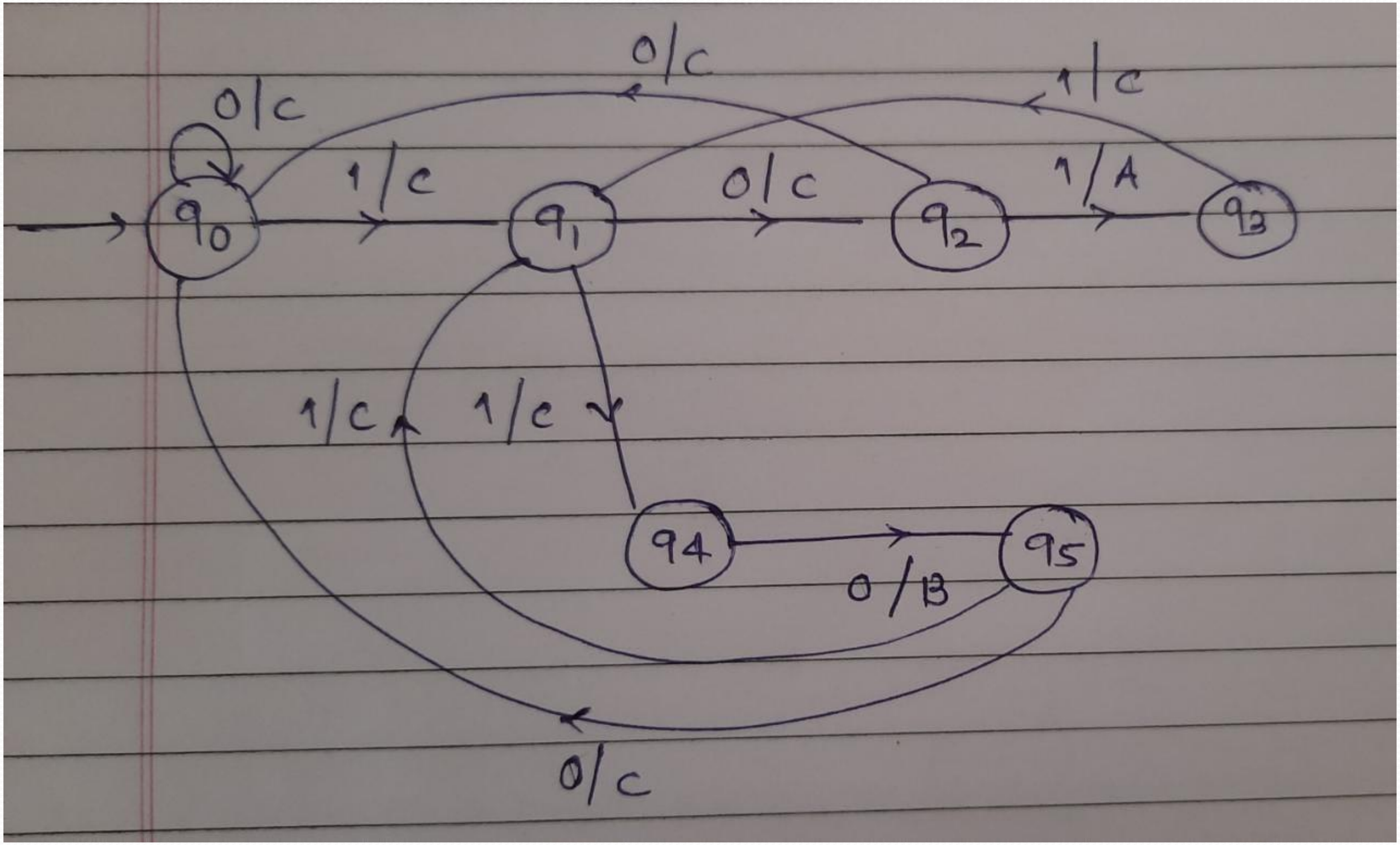
$Q = \{q_0, q_1, q_2, q_3\}$

$\Sigma = \{0, 1\}$

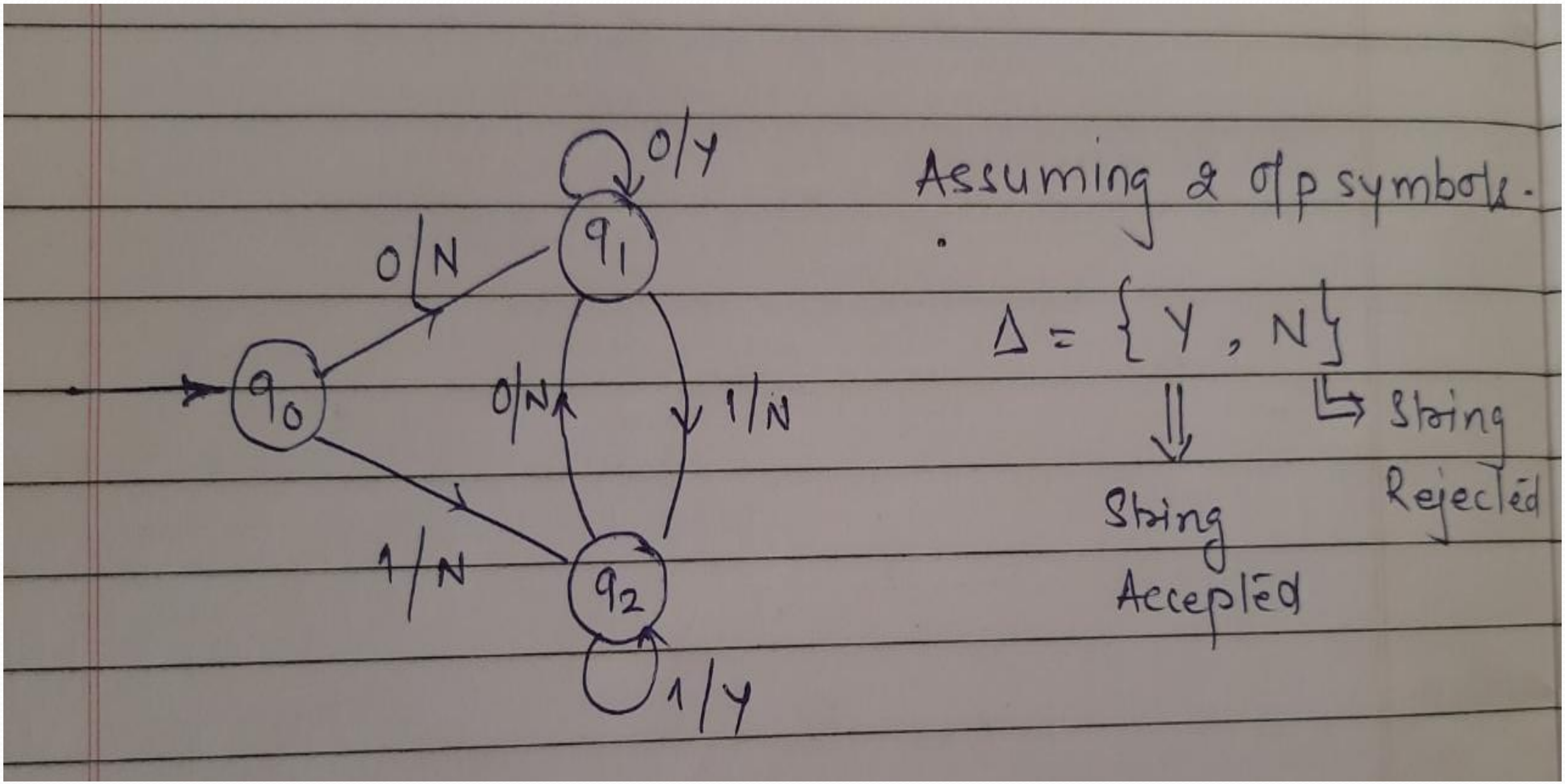
$q_0 = q_0$

$\Delta = \{0, 1\}$

- Design Mealy Machine which outputs ‘A’ if ‘101’ is recognized , outputs ‘B’ if ‘110’ is recognized, and ‘C’ otherwise



- Design a Mealy Machine to accept a binary string if it ends with ‘00’ or ‘11’



- Design Moore and Mealy Machine to determine residue modulo 3 of a binary input

  - Residue Modulo n = Remainder after dividing by n

  - The expected answer is remainder when a binary number is divided by 3 ....i.e. 0, 1, 2

$\lambda \Rightarrow$

	Δ
q_0	0
q_1	1
q_2	2

	0	1
q_0	0	1
q_1	2	0
q_2	1	2

	0	1
q_0	q_0	q_1
q_1	q_2	q_0
(10) q_2	q_1	q_2

Example - (Moore Machine)

$$(1 \quad 1 \quad 1)_2$$

$$\delta\,(1 \quad 1 \quad 1)$$

$$q_0 \quad q_1 \quad q_0 \quad q_1$$

$$\lambda\,(q_1 \quad q_0 \quad q_1)$$

$$\Rightarrow \quad 1 \quad 0 \quad \boxed{1}$$

The Last digit represents remainder

$$(1 \quad 0 \quad 1)_2$$

$$\delta\,(1 \quad 0 \quad 1)$$

$$q_0 \quad q_1 \quad q_2 \quad q_2$$

$$\lambda\,(q_1 \quad q_2 \quad q_2)$$

$$\Rightarrow \quad 1 \quad 2 \quad \boxed{2}$$

$\therefore\ 5 \% 3 = 2.$

$(101)_2 \% 3 = 2$

Example (Mealy M/c)

$(1\ 1\ 1)_2 \% 3$

$$\delta\,(1 \quad 1 \quad 1)$$

$$q_0 \quad q_1 \quad q_0 \quad q_1$$

$\lambda\,(q_0, 1) = 1$

$\lambda\,(q_1, 1) = 0$

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

The last digit of answer represents remainder.

$\therefore\ (111)_2 \% 3$

$= 1.$

$$(1 \quad 0 \quad 1)_2$$

$$\delta\,(1 \quad 0 \quad 1)$$

$$q_0 \quad q_1 \quad q_2 \quad q_2$$

$\lambda(q_0, 1) = 1$

$\lambda(q_1, 0) = 2$

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

$\therefore\ (101)_2 \% 3$

$= 2.$

Homework

- Design a Moore Machine to accept a binary string if it ends with ‘00’ or ‘11’
- Design Mealy Machine which outputs ‘A’ if ‘101’ is recognized , otherwise outputs ‘B’
- Design a Mealy machine to convert every occurrence of ‘121’ to ‘122’ over $\sum = \{1,2,3\}$

Mealy Machine	Moore Machine
Output depends both upon the present state and the present input	Output depends only upon the present state.
Generally, it has fewer states than Moore Machine.	Generally, it has more states than Mealy Machine.
The value of the output function is a function of the transitions and the changes, when the input logic on the present state is done.	The value of the output function is a function of the current state and the changes at the clock edges, whenever state changes occur.
Mealy machines react faster to inputs. They generally react in the same clock cycle.	In Moore machines, more logic is required to decode the outputs resulting in more circuit delays. They generally react one clock cycle later.