Moore and Mealy Machine

- Both moore and mealy machine are special case of DFA
- Both acts like o/p producers rather than language acceptors
- In moore and mealy machine no need to define the final states
- No concepts of dead states and no concepts of final states
- Mealy and Moore Machines are equivalent in power.

George H Mealy

Edward F Moore

Moore Machine

A Moore machine is a six-tuple (Q, Σ, Δ, δ, λ, q_0), where

- Q is a finite set of states:
- Σ is the input alphabet:
- Δ is the output alphabet.
- δ is the transition function $Q \times \Sigma$ into Q
- λ is the output function mapping Q into Δ and
- q_0 is the initial state.

Examples: The below table shows the transition table of a Moore Machine.

Present state	Next state δ		Output
	$a = 0$	$a = 1$	λ
$\rightarrow q_0$	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

- In moore machine for every state output is associated.
- If the length of i/p string is n, then length of o/p string will be n+1
- Moore machine response for empty string $\in$

Q construct a Moore machine take all the string of a's and b's as i/p and counts the no of a's in the i/p string in terms of 1, $\sum = \{a, b\}$, $\Delta = \{0, 1\}$?

Q construct a Moore machine take all the string of a's and b's as i/p and counts the no of occurrence of sub-string 'ab' in terms of 1, $\sum = \{a, b\}$, $\Delta = \{0, 1\}$?

Q construct a Moore machine where $\Sigma = \{0, 1\}$, $\Delta = \{a, b, c\}$, machine should give o/p a, if the i/p string ends with 10, b if i/p string ends with 11, c otherwise?

Mealy Machine

- Mealy machine is a six-tuple $(Q, \Sigma, \Delta, \delta, \lambda, q_0)$, where all the symbols except λ have meaning as in the Moore machine. λ is the output function mapping $Q \times \Sigma$ into Δ.
- In case of mealy machine, the output symbol depends on the transition.

Example: The below table shows the transition table of a Mealy Machine.

Present state	Next state			
	$a = 0$		$a = 1$	
	state	output	state	output
$\rightarrow q_1$	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

- If the length of i/p string is n, then length of o/p string will be n
- Mealy machine do not response for empty string $\in$

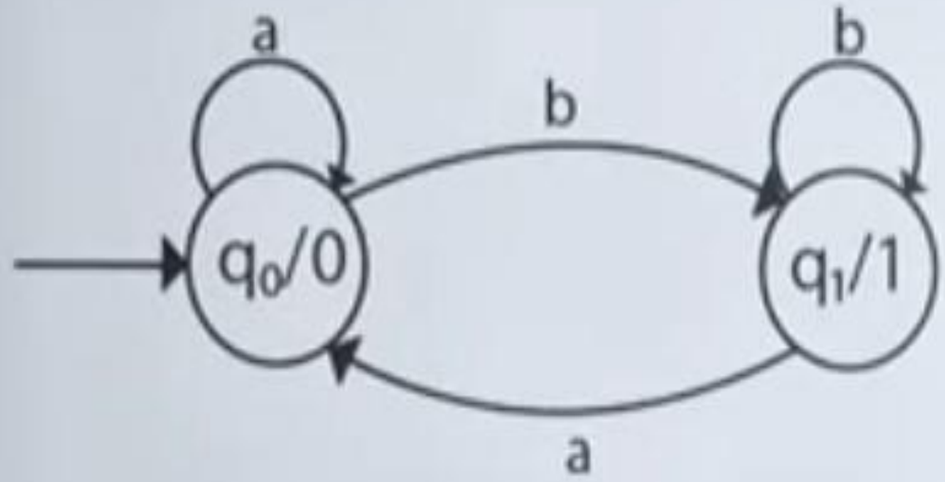
Q construct a Mealy machine take all the string of a's and b's as i/p and counts the no of a's in the i/p string in terms of 1, $\Sigma = \{a, b\}$, $\Delta = \{0, 1\}$?

Q construct a Mealy machine take all the string of a's and b's as i/p and counts the no of occurrence of sub-string 'ab' in terms of 1, $\sum = \{a, b\}$, $\Delta = \{0, 1\}$?

Q construct a Mealy machine where $\sum = \{0, 1\}$, $\Delta = \{a, b, c\}$, machine should give o/p a, if the i/p string ends with 10, b if i/p string ends with 11, c otherwise?

CONVERSION OF MOORE TO MEALY MACHINE

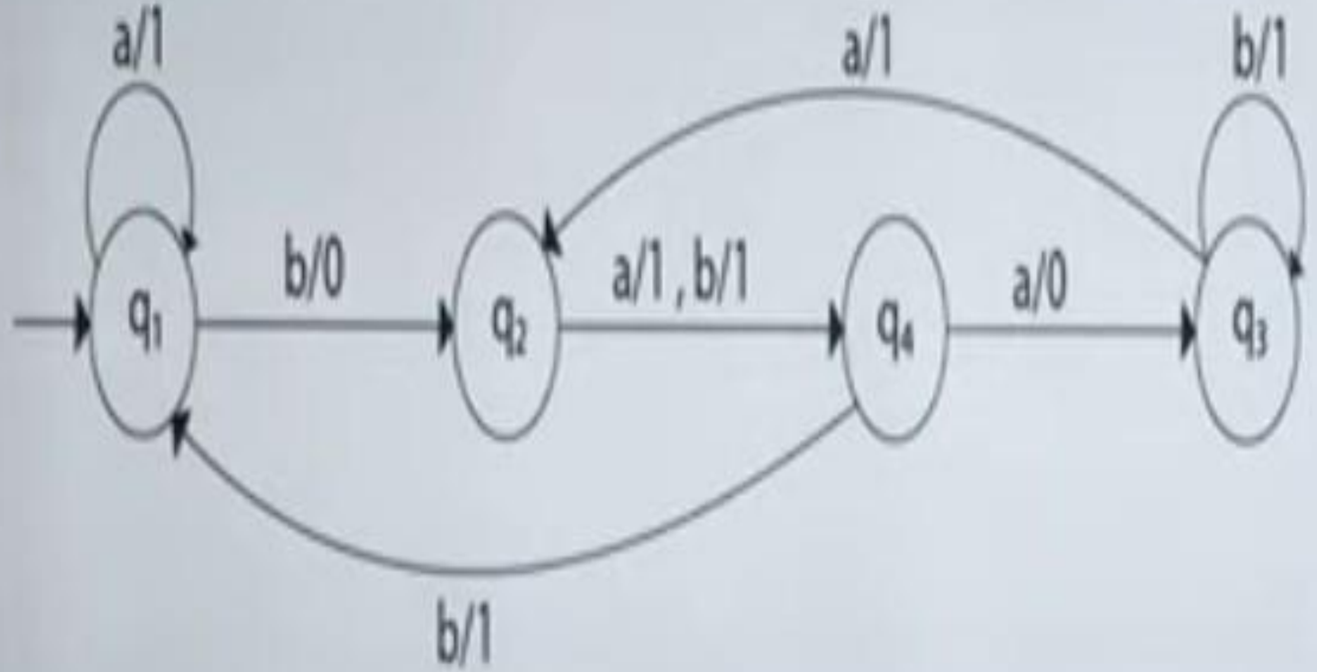
- Let us take an example to understand the conversion:
- Convert the following Moore machine into its equivalent Mealy machine.



Q	a	b	Output(λ)
q0	q0	q1	0
q1	q0	q1	1

PROCEDURE FOR TRANSFORMING A MEALY MACHINE INTO A MOORE MACHINE

Consider the Mealy Machine:



Present State	Next State			
	a		b	
	State	O/P	State	O/P
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

Regular Expressions:

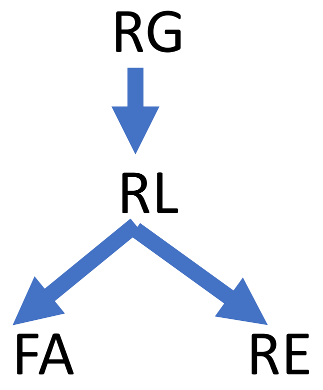
- A way of representing Regular Language.
- A Language is said to be Regular Language if their exist a Regular Grammar.
- A Language is said to be Regular Language if there exist a Finite Automata to accept it.
- A Language is said to be Regular Language if there exist a Regular Expression to represent it. Regular Expression is Like a mathematics in which we use string in place of operands and some special different operators in place of +, -,*,/.
- Expression of String & Operators

Like (1) * (Kleen Closure) [a*]

(2) + (Positive Closure) [a+]

(3) . (Concatination) [a.b]

(4) + (Union) [a+b]



Regular Expressions:

- Regular Expression is said to be valid if it can be derived from the primitive Regular Expression by a finite number of application of the rule r*, r+, r1.r2, r1+r2
- $\varepsilon\ (sigma)$ is a given albhabet then,

$\emptyset, \frac{\epsilon\ (epsilon)}{\lambda}, a\epsilon\varepsilon\ are\ primitive\ regular\ expression.$

RE		RL
Ø	=	{ } or Ø
$\epsilon\ (epsilon)$	=	$\{\epsilon\}$

Identities of Regular Expression

1) $\emptyset + R = R$

2) $\emptyset R + R\emptyset = \emptyset$

3) $\epsilon R = R\epsilon = R$

4) $\epsilon^* = \epsilon$ and $\emptyset^* = \epsilon$

5) $R + R = R$

6) $R^*R^* = R^*$

7) $RR^* = R^*R$

8) $(R^*)^* = R^*$

9) $\epsilon + RR^* = \epsilon + R^*R = R^*$

10) $(PQ)^*P = P(QP)^*$

11) $(P + Q)^* = (P^* Q^*)^* = (P^* + Q^*)^*$

12) $(P + Q) R = PR + QR$ and

$$R(P + Q) = RP + RQ$$

1. $r = \phi$, $L(r) = \{\ \} / \phi$
2. $r = \epsilon$, $L(r) = \{\epsilon\}$
3. $r = a$, $L(r) = \{a\}$
4. $r = a+b$, $L(r) = \{a, b\}$
5. $r = a \cdot b$, $L(r) = \{ab\}$
6. $r = a+b+c$, $L(r) = \{a, b, c\}$
7. $r = (ab+a) \cdot b$, $L(r) = \{abb, ab\}$
8. $r = a^+$, $L(r) = \{a^1, a^2, a^3, \ldots\}$

$a^2 = a \cdot a$

$a \cdot b \neq ba$

$a^2 = a \times a$