CSE 4205
Digital Logic Design

Sequential Circuit Model

Course Teacher: Md. Hamjajul Ashmafee

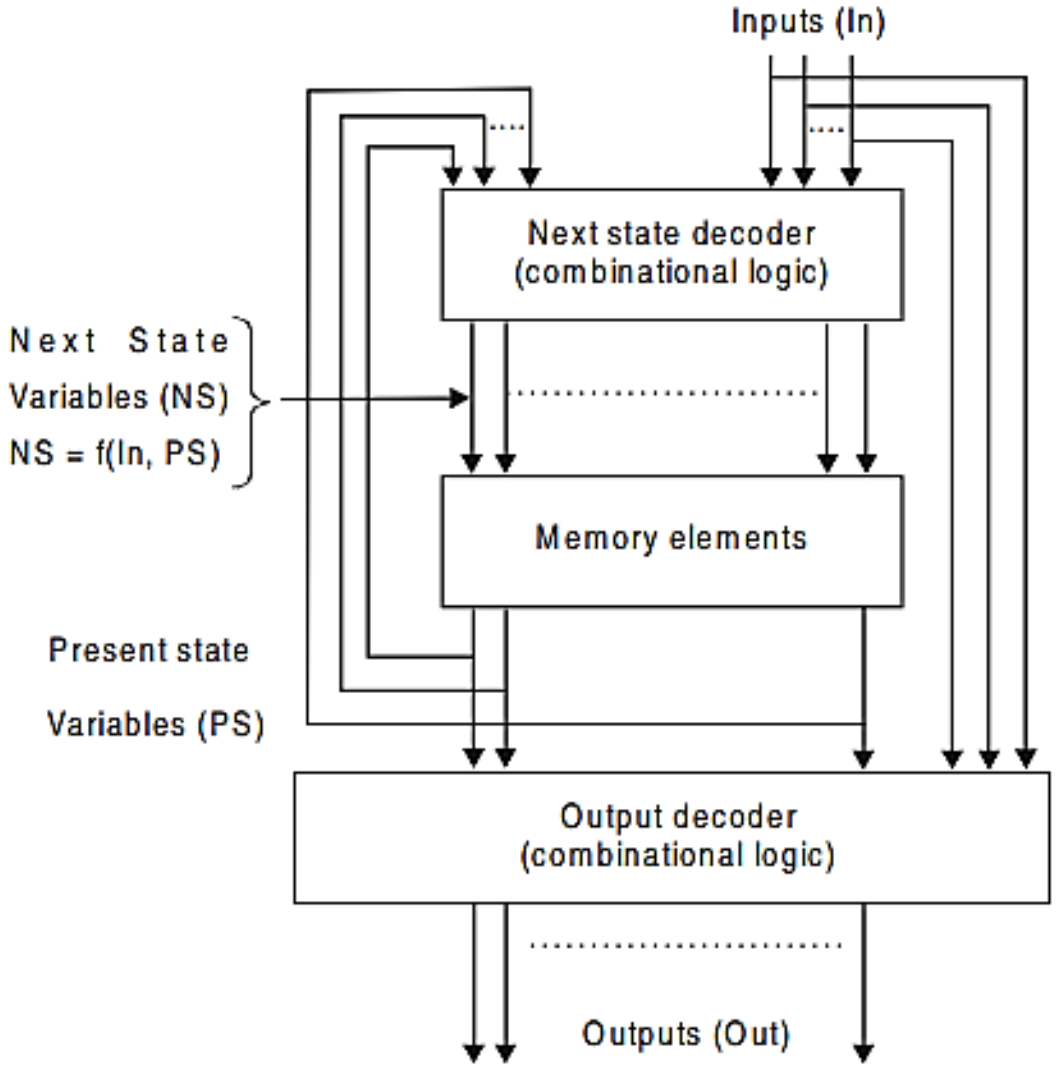
Lecturer, CSE, IUT

Email: ashmafee@iut-dhaka.edu

Sequential Circuit Model

- The present state of the circuit is stored in the memory element
  - This memory element can be any device to store enough information to specify the state of the circuit
- There are **two** **logic circuits** in the SC model:
  - **Next state decoder:**
  - **Output decoder:**

Sequential Circuit Model: Block Diagram

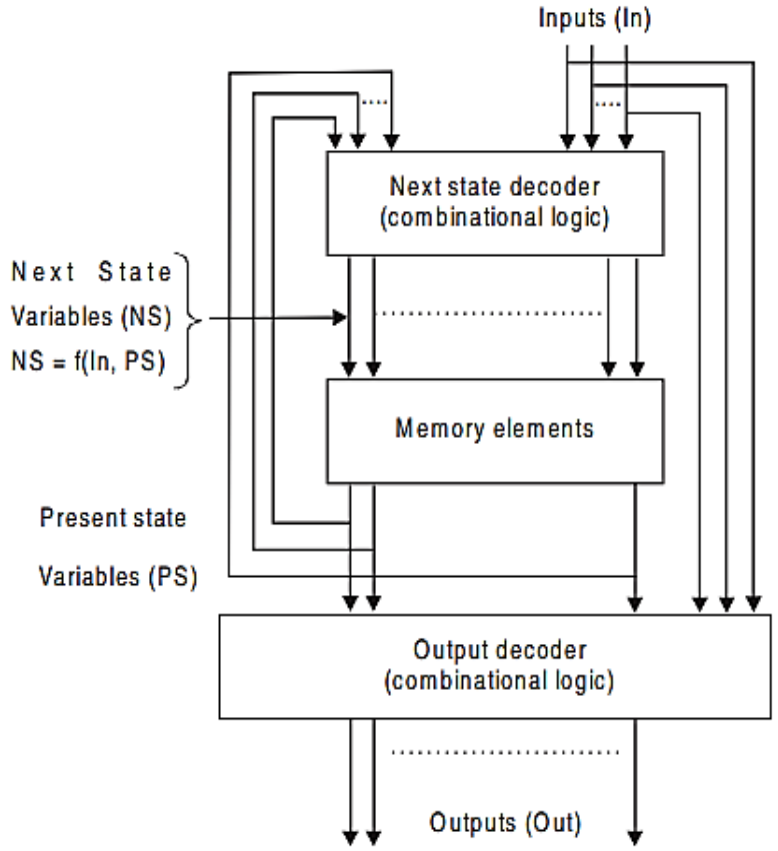


Classification of SC

- Generally **5** different **classes**:
  - Class A circuits
  - Class B circuits
  - Class C circuits
  - Class D circuits
  - Class E circuits

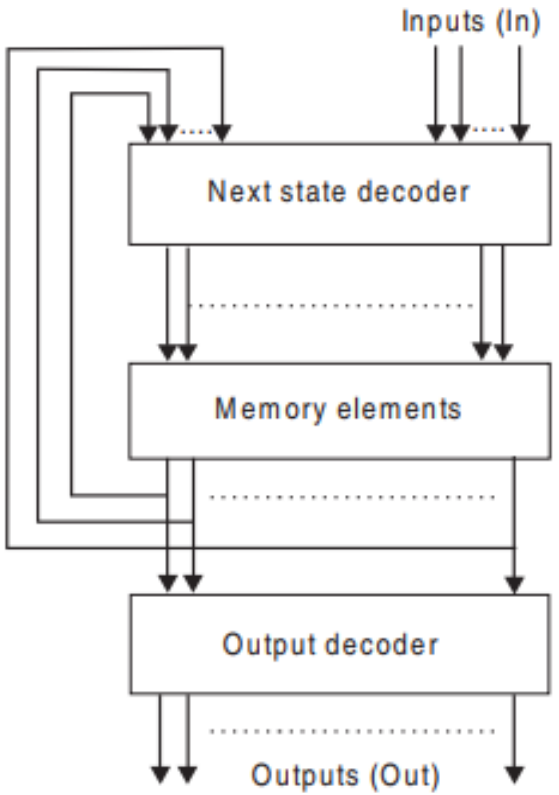
Classification of SC: Class A circuits

- Defined as ***Mealy Machine***
  - Named after ***G.H. Mealy***
- **Basic property:**
  - The output decoder is a function of the present input(s) and the present state(s) of the circuit
  - The next state decoder is a function of present state(s) and present input(s)



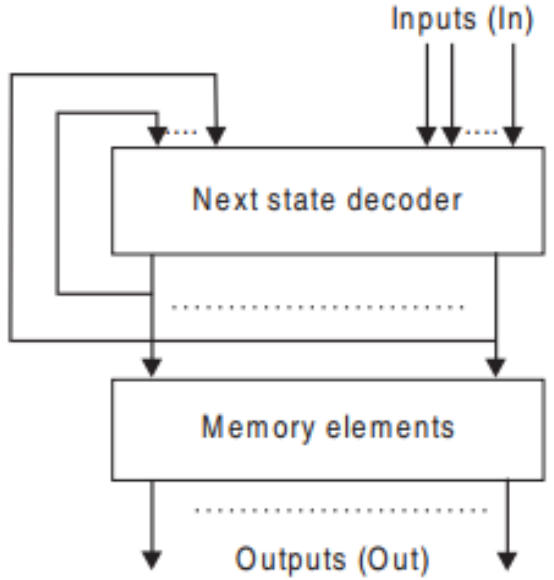
Classification of SC: Class B circuits

- Defined as ***Moore Machine***
  - Named after ***E.F. Moore***
- **Basic property:**
  - The output decoder is a function of the present state(s) only
    - It's a MOORE machine with an output decoder
  - The next state decoder is a function of present state(s) and present input(s)



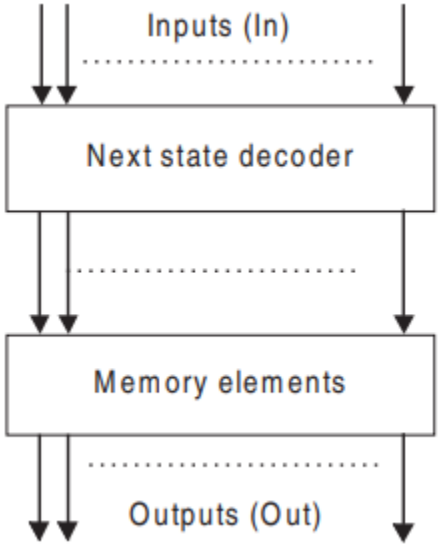
Classification of SC: Class C circuits

- Also, defined as ***Moore Machine***
- **Basic property:**
  - The output decoder is not included in this circuit
    - It's a MOORE machine without an output decoder
    - Contents of memory elements are considered as the output directly
  - The next state decoder is a function of present state(s) and present input(s)



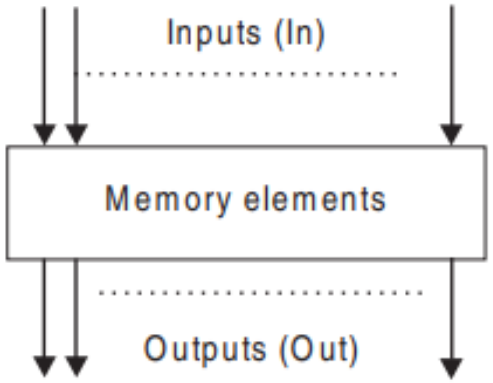
Classification of SC: Class D circuits

- **Basic property:**
  - The next state decoder is a function of present input(s) only
  - No output decoder
    - Contents of memory elements are considered as the output directly



Classification of SC: Class E circuits

- **Basic property:**
  - The next state decoder is not included in this circuit
    - Inputs are directly inserted into the memory element
  - No output decoder
    - Contents of memory elements are considered as the output directly



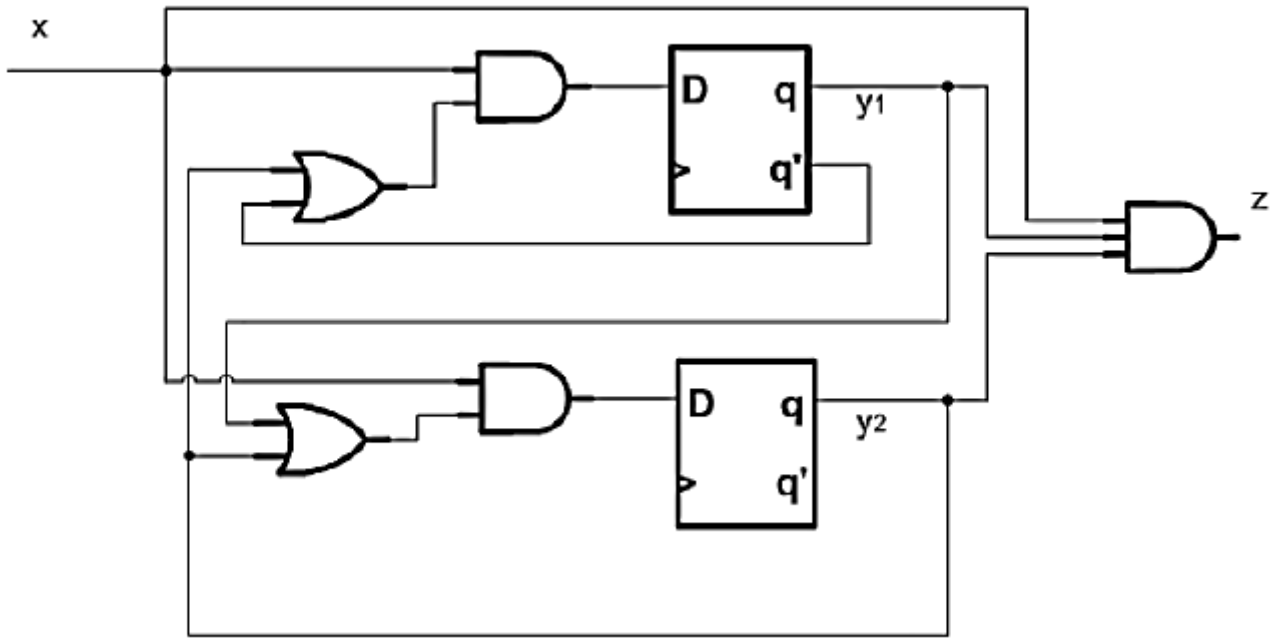
State Diagram or State Machine to Represent a Sequential Circuit

State Diagram or State Machine

- A **transitional diagram** from one state to another
  - Also called as finite automata (or machine)
  - ***Alternatively,*** Finite state machine (FSM)

Mealy State Machine: Circuit Diagram

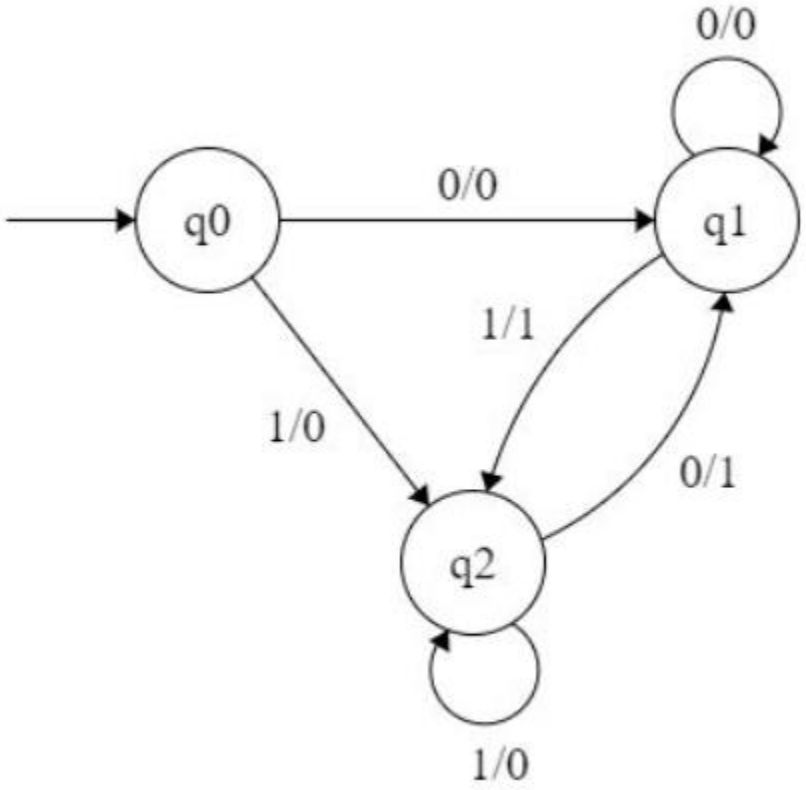
Random example



Mealy State Machine: State Diagram

Random example

- **Outputs are represented along with *current input(s)* separated by '/'**

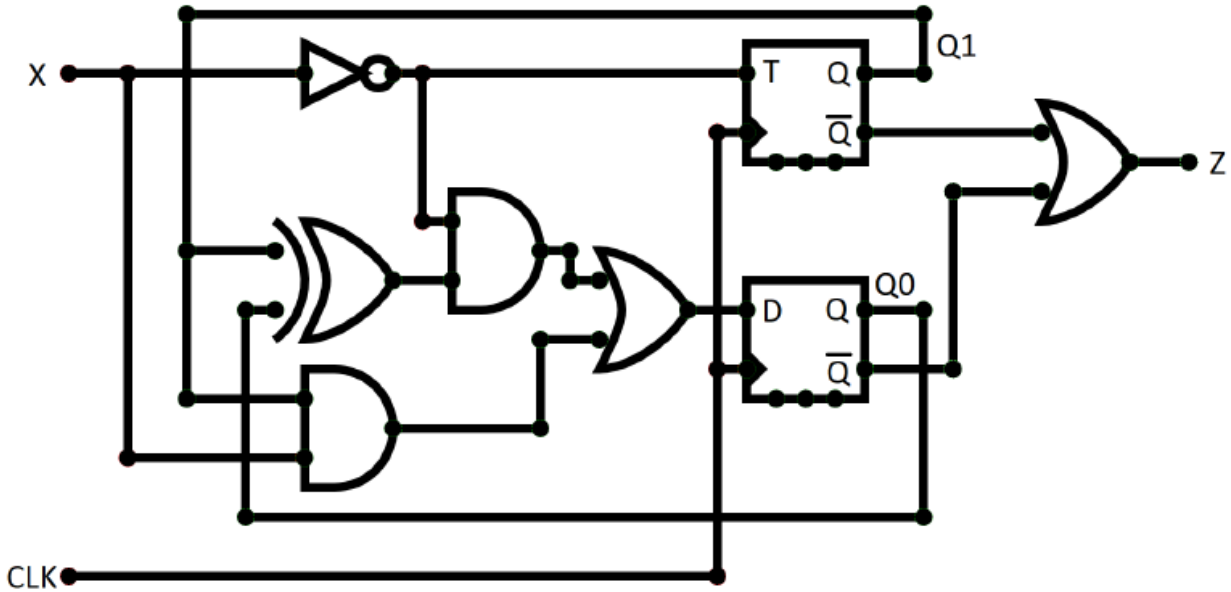


Moore State Machine

- Where output depends on only current state, ***i.e.*** output is solely related (**paired**) to current state.
  - But, **states** change based on *current input(s)* and *current state(s)*
  - ***Output*** is a function of current states
  - Basically, ***synchronous*** to state changes
  - **Application:** Edge detector of clock pulses, Elevator (Loop), Binary adder in FSM, Clocked sequential circuit

Moore State Machine: Circuit Diagram

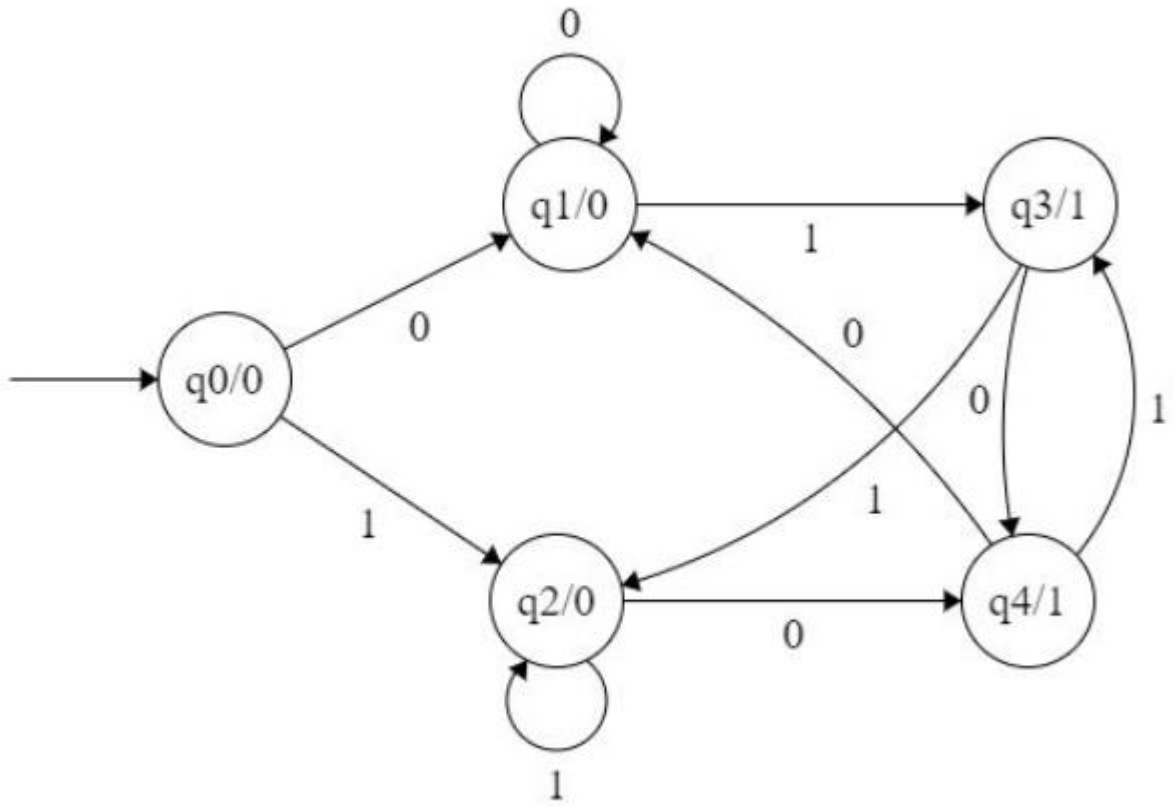
Random example



Moore State Machine: State Diagram

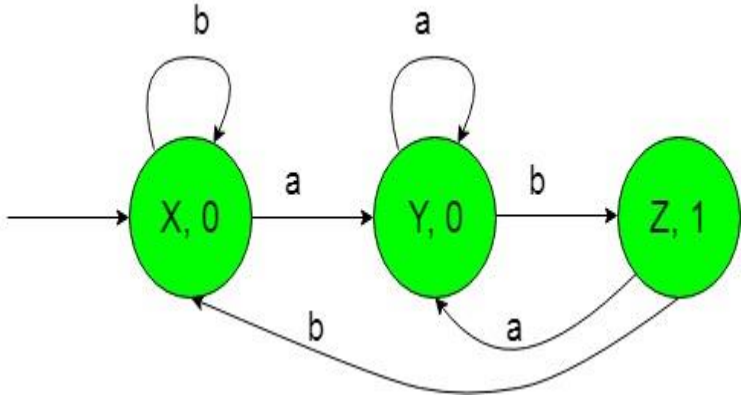
Random example

- **Outputs are represented along with *current state* separated by '/'**



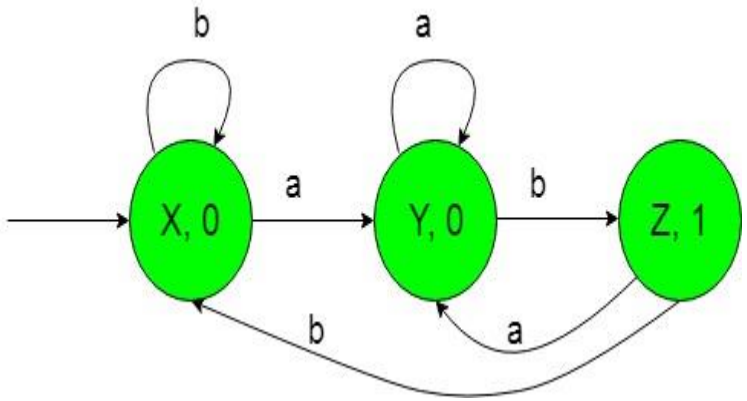
Moore to Mealy FSM Conversion

- **Steps for Moore to Mealy conversion:**
  1. Create the state table for given Moore FSM
  2. Derive the state table of Mealy FSM from the above state table of Moore FSM
  3. Draw the Mealy FSM from the derived state table of Mealy machine



Example of a Moore FSM

Moore to Mealy FSM Conversion



	a	b	Δ
→ X	Y	X	0
Y	Y	Z	0
Z	Y	X	1

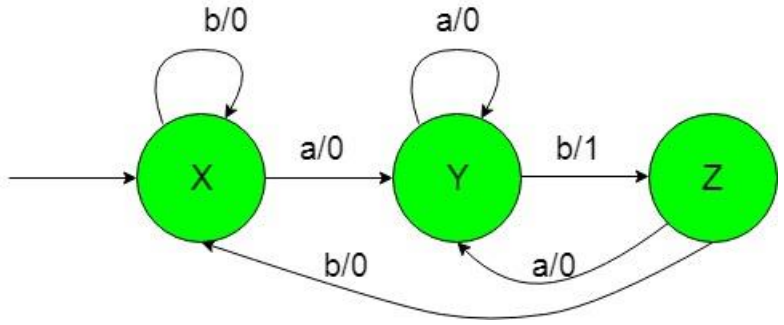
$\Delta = \boldsymbol{Output}$

Step 1

$Input = I_i$
(Next state, Output)

	a	b
→ X	(Y, 0)	(X, 0)
Y	(Y, 0)	(Z, 1)
Z	(Y, 0)	(X, 0)

Step 2

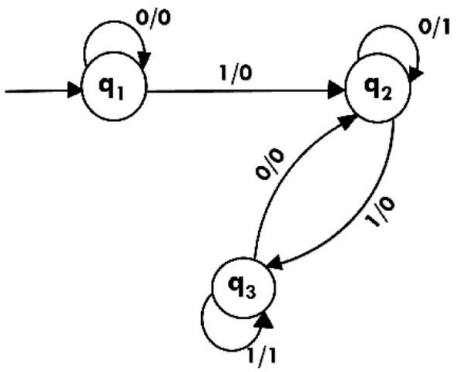


Step 3

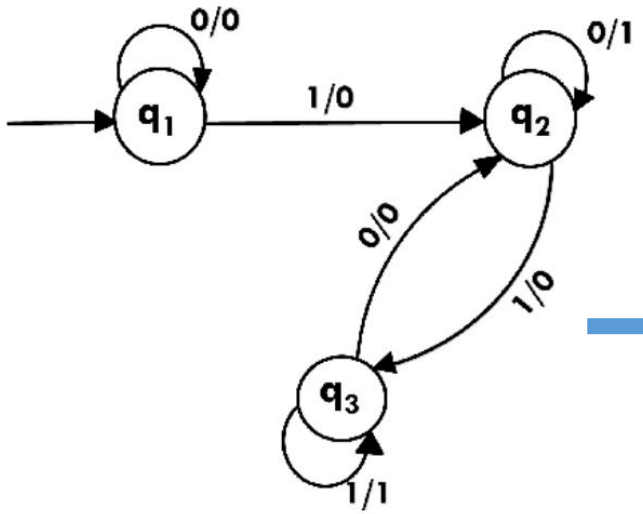
Note: If any two rows are identical, one of the nodes is redundant

Mealy to Moore FSM Conversion

- **Steps for Mealy to Moore conversion (more nodes are produced):**
  1. For each state(Qi), calculate the number of different outputs based on the input flow (also available in the transition table) of the Mealy machine.
     - **Why input flow**: Because it indicates when that state is produced which output is associated with that state.
  2. One of two:
     - Keep the same state ,Qi, if all the input flows produce the same output with that state.
     - Or, break the state into n states (as Qi1…Qin) if the input flows produce ***n*** different outputs
  3. Associate appropriate output with each state separated by a slash (/) – (q01/1)
  4. Check all possible inputs for each state from the diagram



Mealy to Moore FSM Conversion



PS	Input = 0		Input = 1	
	NS	Output	NS	Output
q1	q1	0	q2	0
q2	q2	1	q3	0
q3	q2	0	q3	1

When q1, output is same = 0, no split.
Two outputs when q2 – q20 and q21
Two outputs when q3 – q30 and q31