Digital System Design EC 503

Mealy and Moore Circuit

Subject Coordinator
Dr. GAURAV KUMAR BHARTI
Department of ECE

Finite State Machine

- A Finite State Machine (FSM) is a mathematical model used to represent the behavior of a system that can exist in a finite number of states, with transitions between these states governed by certain conditions or inputs.
- FSMs are widely used in various fields such as computer science, engineering, linguistics, and more, for modeling and controlling the behavior of systems.
- Finite State Machines can be classified into two main types based on how they handle transitions:

- **Deterministic Finite Automaton (DFA)**: In a DFA, for every state and input, there is exactly one possible transition. The transition function is deterministic, meaning that it is uniquely determined by the current state and input. DFAs are often represented using state transition diagrams or tables.
- **Nondeterministic Finite Automaton (NFA)**: In an NFA, there can be multiple possible transitions for a given state and input. The transition function is non-deterministic, allowing for more flexibility in modeling certain types of systems. NFAs are often represented using state transition diagrams.

Applications of FSMs

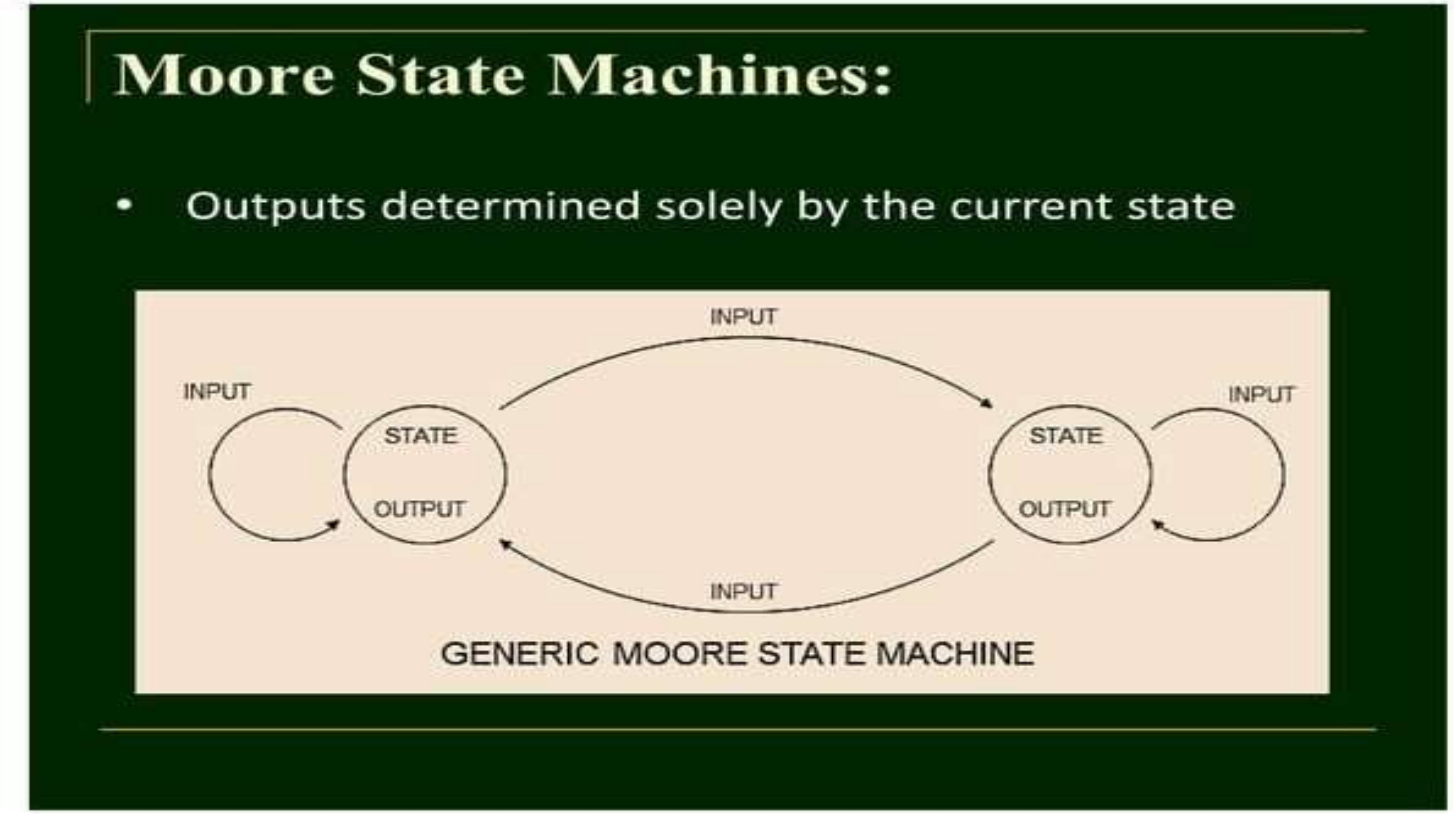
- Finite State Machines are used in various applications, including digital circuit design, parsing and lexical analysis in compiler construction.
- And Exclusively in protocol specification and verification, modeling software behavior, and more.
- They provide a structured way to analyze and design systems with discrete states and transitions.

Mealy and Moore Machine

- Mealy and Moore machines are two common variations of finite state machines that differ in how they handle output generation. Both are used for modeling and controlling systems with discrete states and transitions, but they have distinct characteristics regarding when they produce output.
- **Mealy Machine**:
  - In a Mealy machine, outputs are associated with transitions between states.
  - Output is produced when a transition occurs, and it is based on both the current state and the input that triggers the transition.
  - The output of a Mealy machine is a function of both the current state and the input.
  - Mealy machines are often more compact than Moore machines because output information is tied directly to transitions.
- **Moore Machine**:
  - In a Moore machine, outputs are associated with states.
  - Output is produced when the system enters a state, and it is based solely on the current state.
  - The output of a Moore machine is a function of only the current state.
  - Moore machines are often easier to understand and design than Mealy machines because output is directly tied to states.

Moore Machine

States in Moore machines represent distinct conditions or modes of operation, and each state is associated with a fixed output value. The behavior of the system is determined solely by the current state, making Moore machines easier to analyze and design for certain applications.



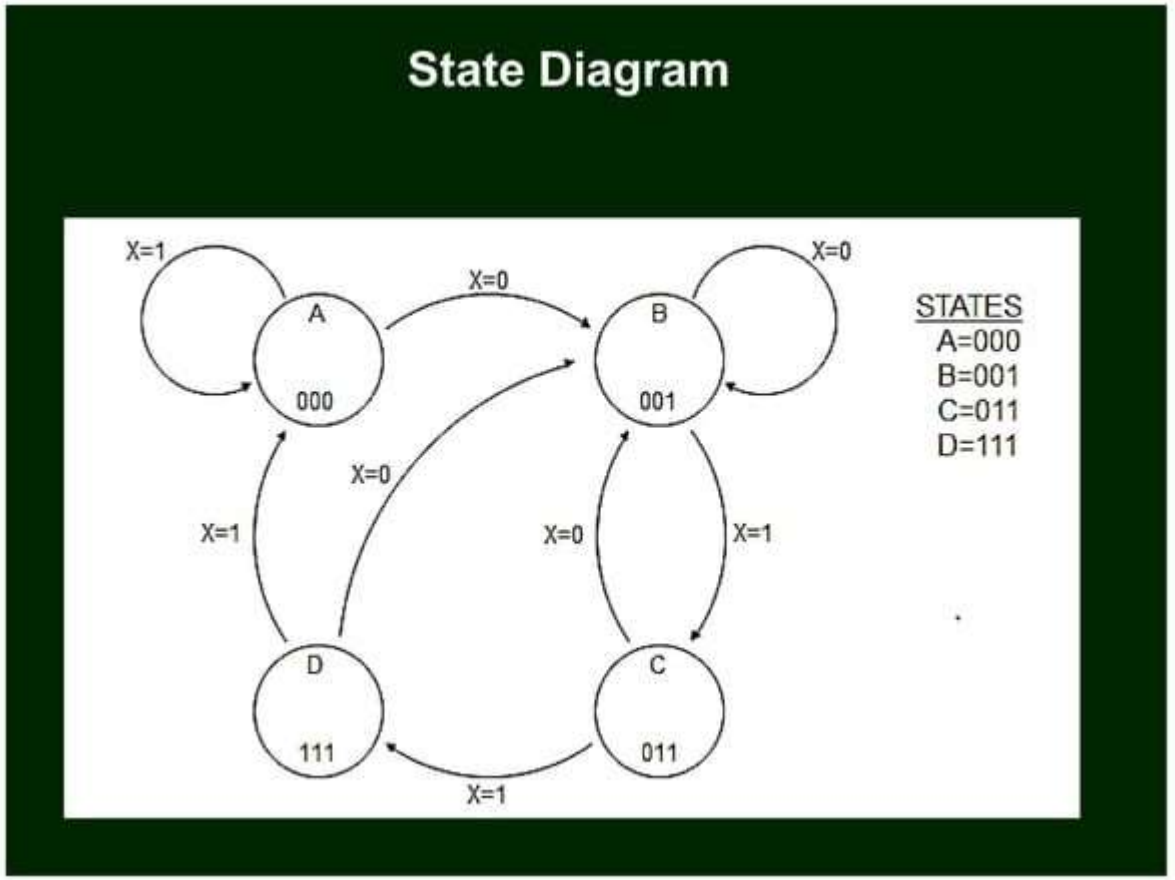
Moore State Table

State Table

Prev. State	X	O_2	O_1	O_0	Next State$^+$
A	0	0	0	0	B
A	1	0	0	0	A
B	0	0	0	1	B
B	1	0	0	1	C
D	0	1	1	1	B
D	1	1	1	1	A
C	0	0	1	1	B
C	1	0	1	1	D

Moore State Diagram

A Moore state diagram represents a type of finite state machine where outputs are associated with states rather than transitions.



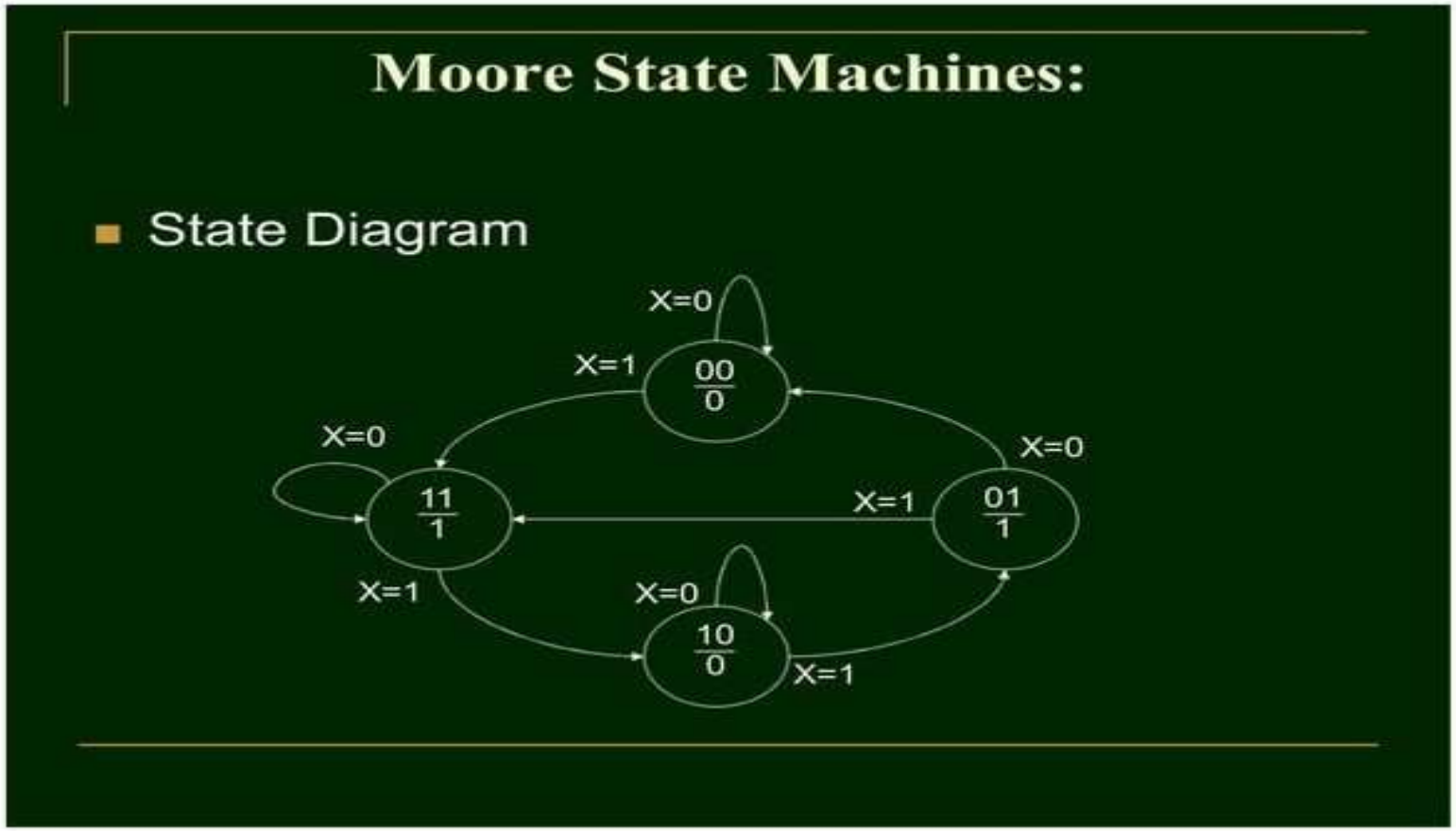
Moore State Machine

Moore State Machines:

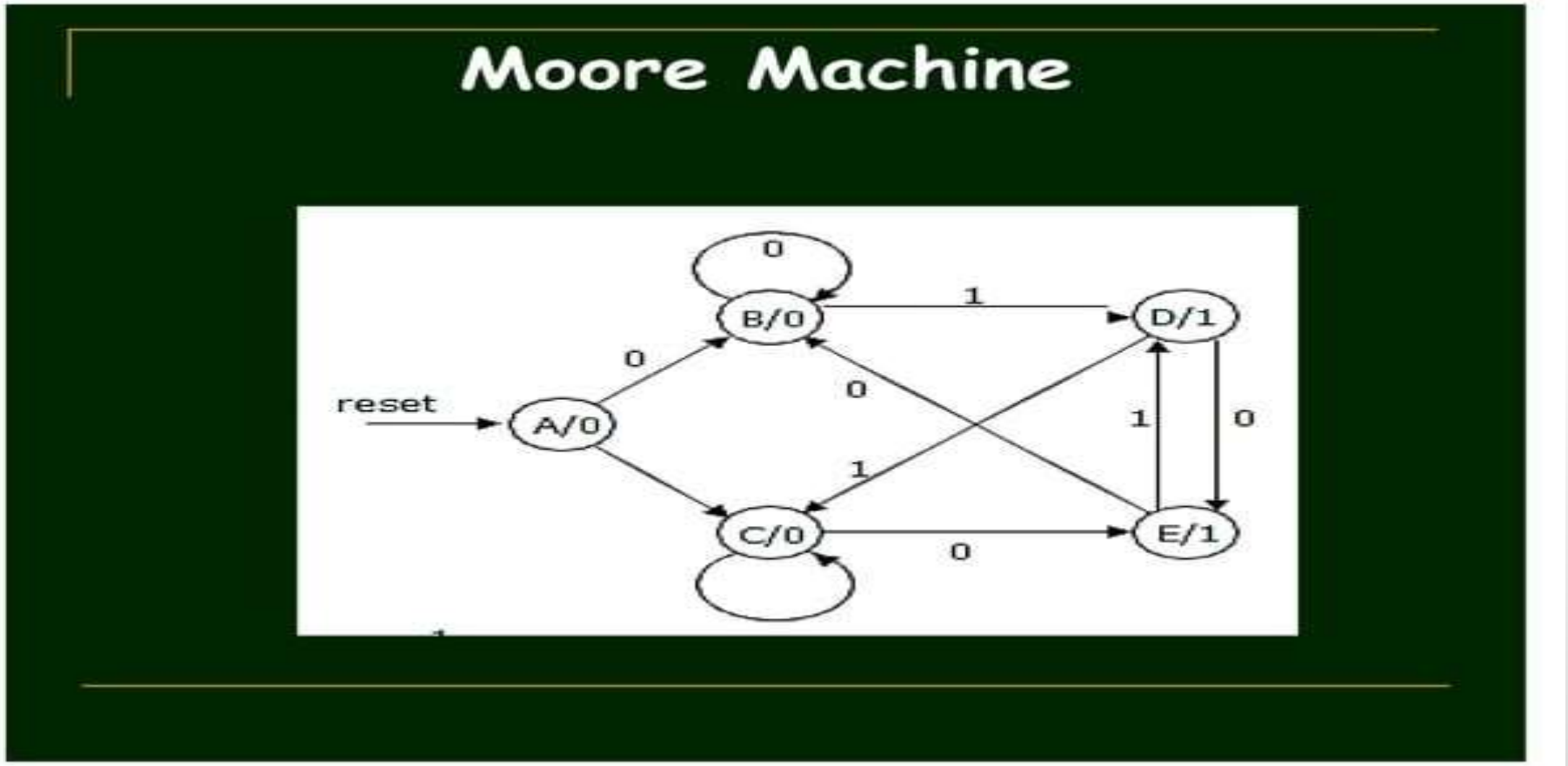
- State Table

	NS		
PS	X=0	X=1	
AB	AB	AB	z (=B)
00	00	11	0
01	00	11	1
10	10	01	0
11	11	10	1

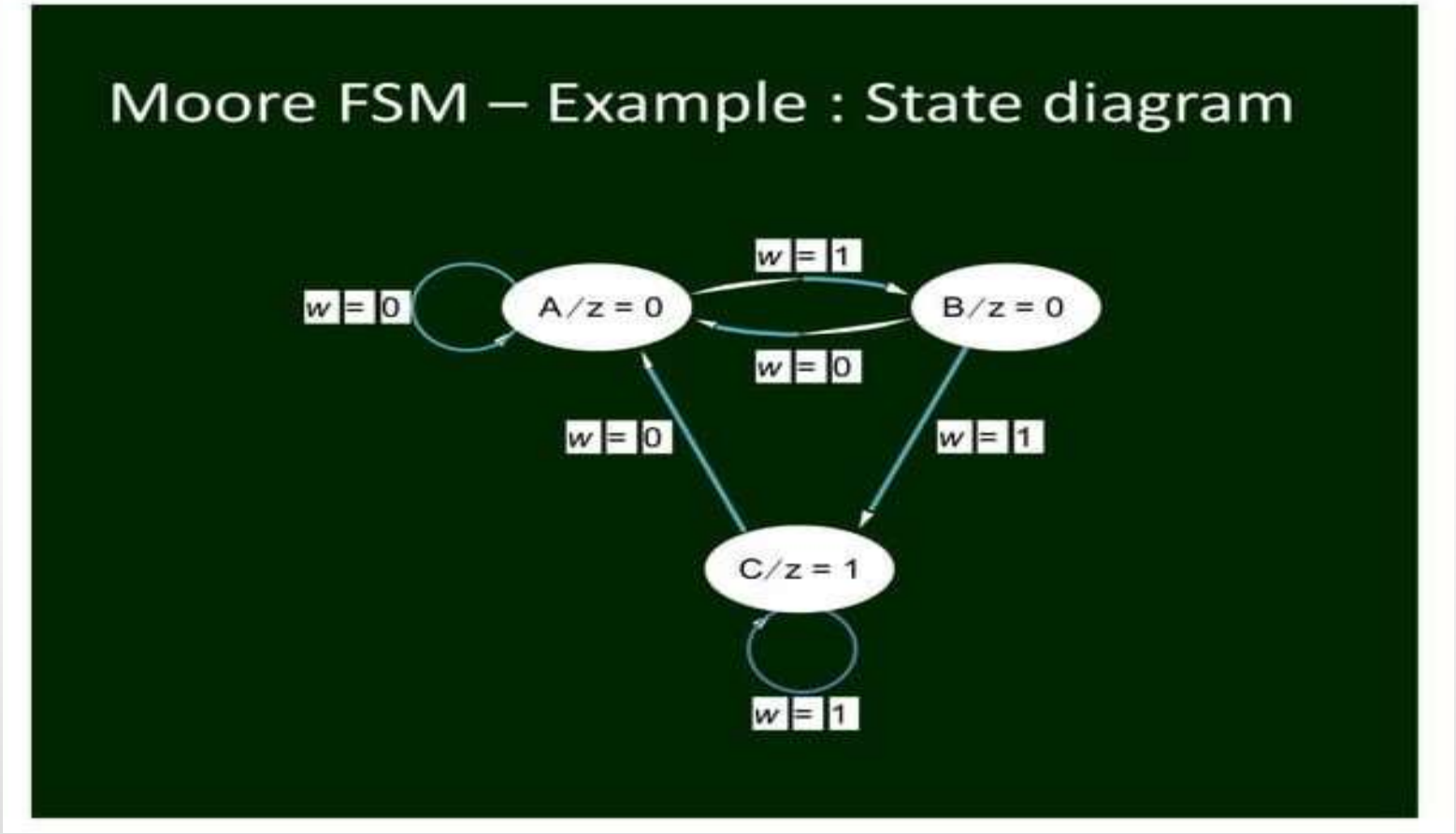
Examples


Moore State Machines:
State Diagram
X=0
X=1
00
0
X=0
X=0
11
1
X=1
01
1
X=1
X=0
10
0
X=1

Examples



Moore FSM



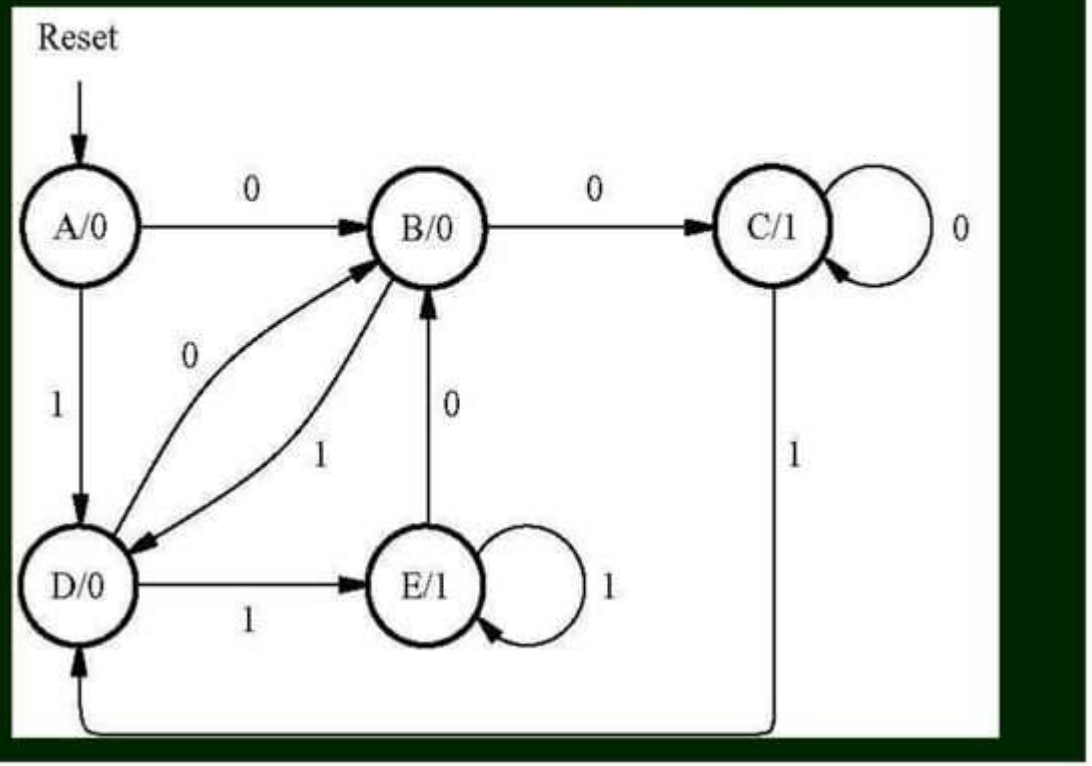
Moore State Table

Moore Machine

reset	input	current state	next state	output
1	—	—	A	
0	0	A	B	0
0	1	A	C	0
0	0	B	B	0
0	1	B	D	0
0	0	C	E	0
0	1	C	C	0
0	0	D	E	1
0	1	D	C	1
0	0	E	B	1
0	1	E	D	1

Moore State Diagram and Table

State Diagram

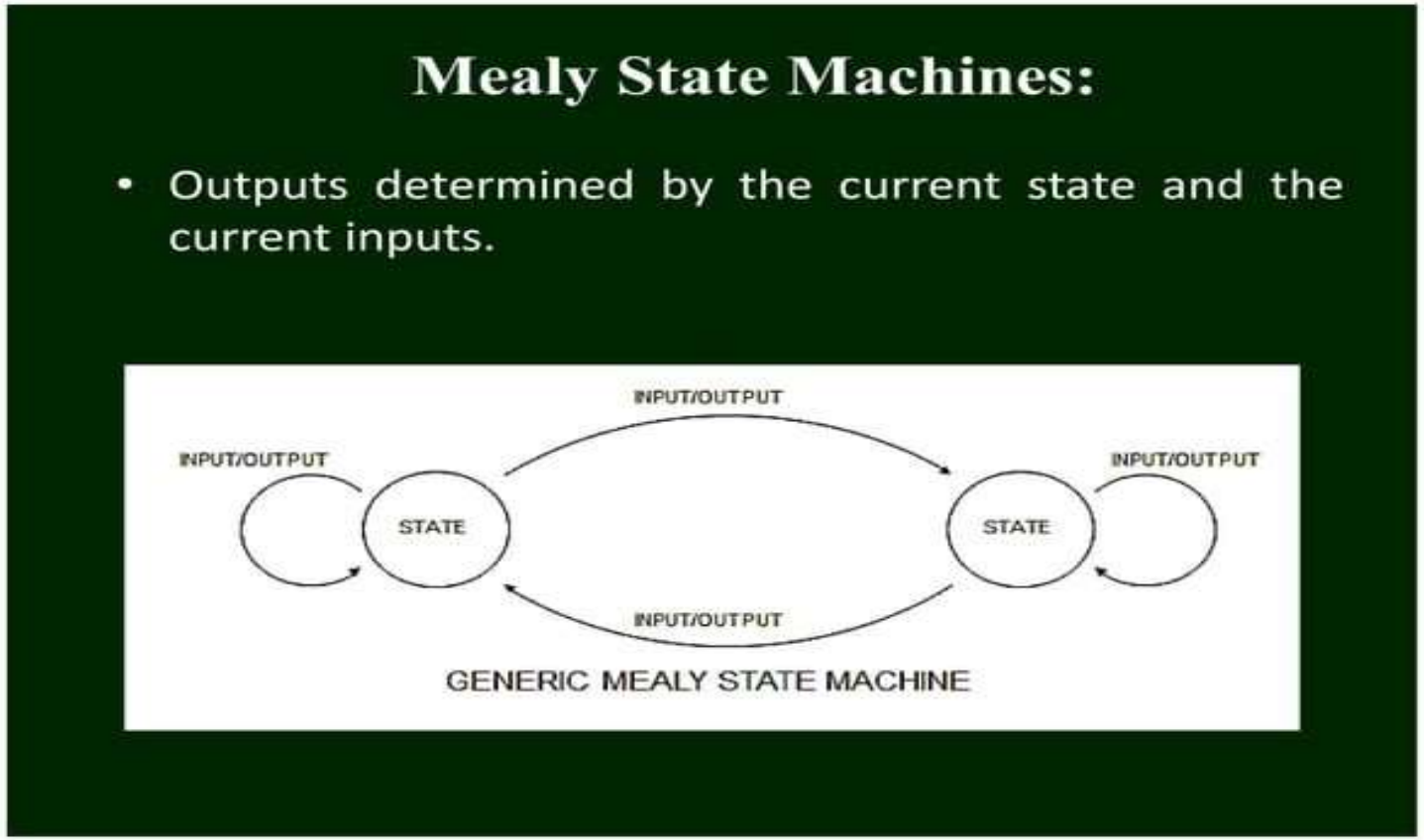


State Table

Present state	Next state $w=0$	Next state $w=1$	Output z
A	B	D	0
B	C	D	0
C	C	D	1
D	B	E	0
E	B	E	1

Mealy Machines

- A Mealy Finite State Machine (FSM) is a type of finite state machine where outputs are associated with transitions between states. In a Mealy machine, both the state transitions and the outputs are determined by the inputs received by the system..



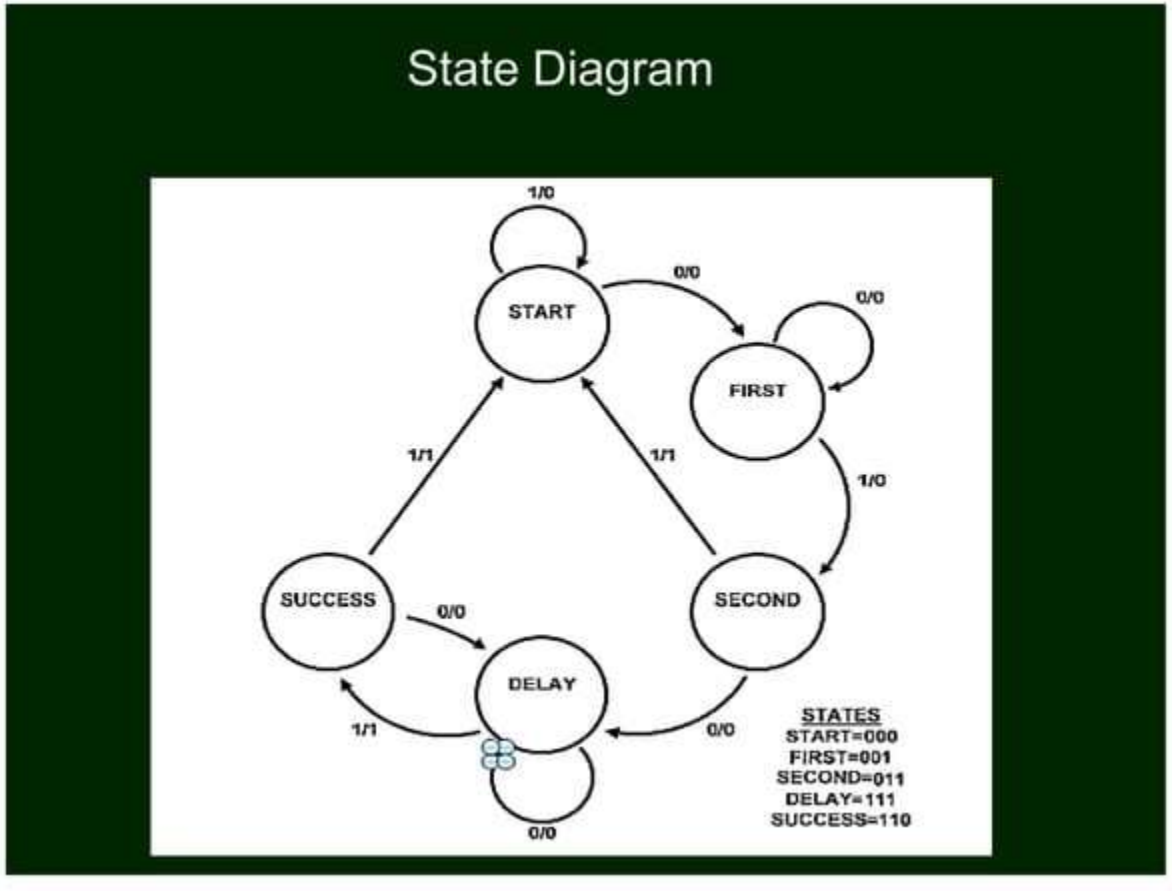
Mealy FSM

- **States**: Similar to other finite state machines, a Mealy FSM consists of a finite set of states that represent different conditions or configurations of the system.
- **Transitions**: The transitions between states are triggered by inputs. Each transition is labeled with the input condition that causes the transition. These transitions indicate how the system moves from one state to another based on the input it receives.
- **Output Functions**: Unlike Moore machines where outputs are associated with states, in Mealy machines, outputs are associated with transitions. Each transition is associated with an output value, and this output value is produced when the system makes that particular transition. The output can depend on both the current state and the input that triggers the transition.
- **Input Labels**: Alongside each transition arrow, the input condition that triggers the transition is specified, indicating what input causes the system to move from one state to another.

Mealy Table

State	Q_2	Q_1	Q_0	X	Z	State$^+$	$Q2^+$	$Q1^+$	$Q0^+$
Start	0	0	0	0	0	First	0	0	1
Start	0	0	0	1	0	Start	0	0	0
First	0	0	1	0	0	First	0	0	1
First	0	0	1	1	0	Second	0	1	1
Success	0	1	0	0	0	First	0	0	1
Success	0	1	0	1	0	Start	0	0	0
Second	0	1	1	0	0	Delay	1	1	1
Second	0	1	1	1	1	Success	0	1	0
unused	1	0	*	*	X	X	X	X	X
SuccessD	1	1	0	0	0	Delay	1	1	1
SuccessD	1	1	0	1	1	Success	0	1	0
Delay	1	1	1	0	0	Delay	1	1	1
Delay	1	1	1	1	1	SuccessD	1	1	0

Mealy State Diagram



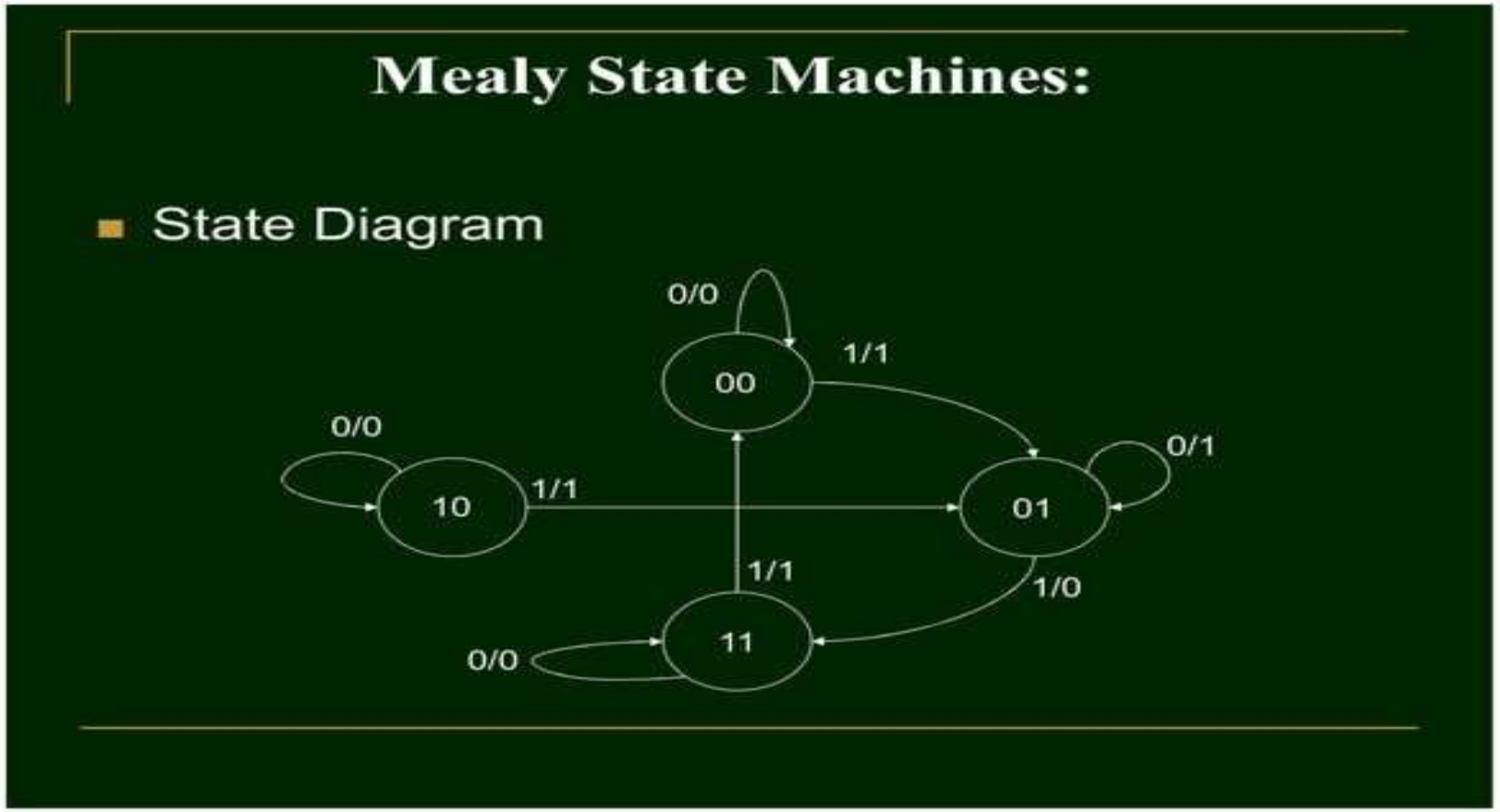
Mealy Machine

Mealy State Machines:

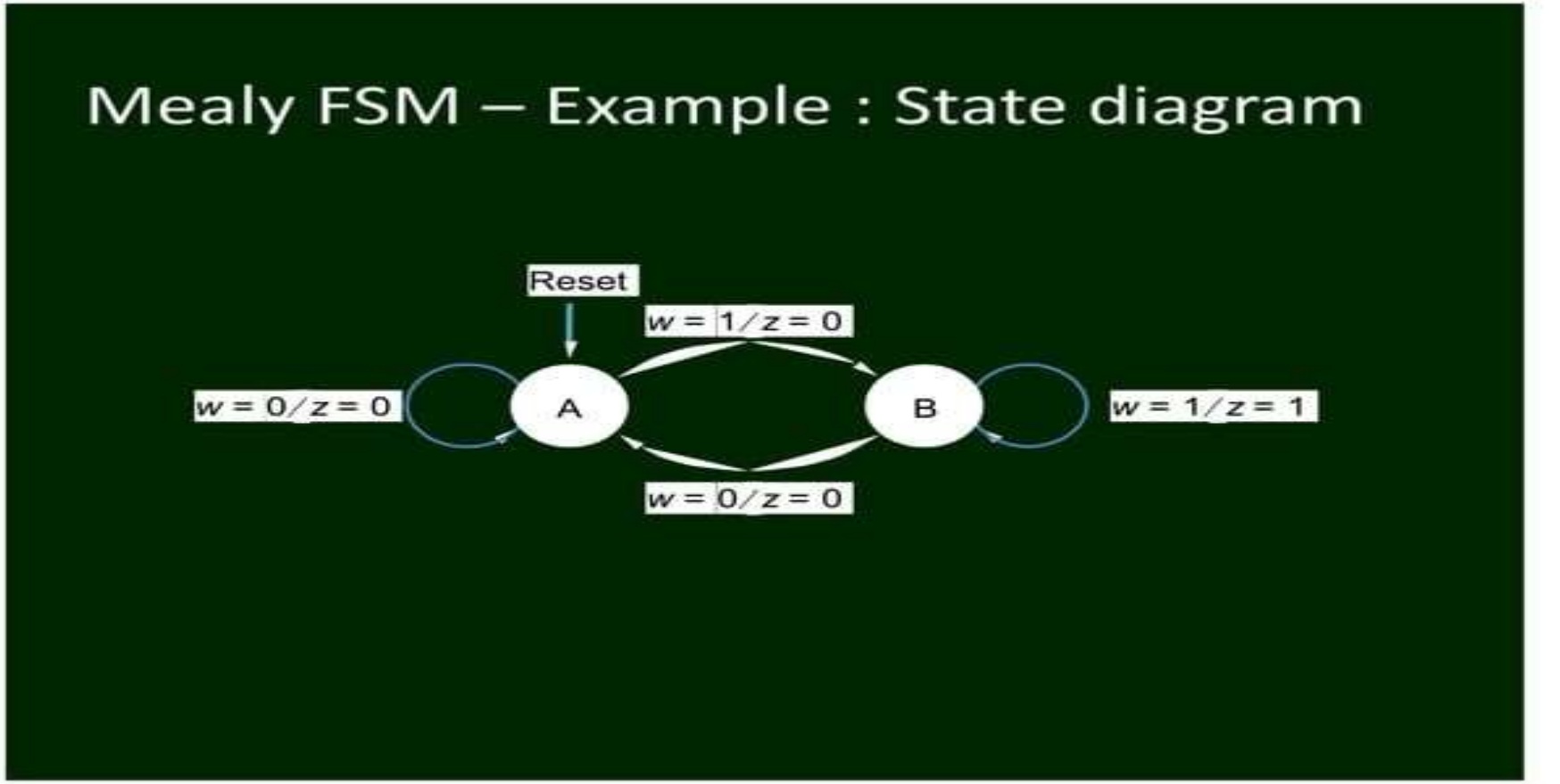
- State Table

	NS	
PS	x=0	x=1
AB	AB,z	AB,z
00	00,0	01,1
01	01,1	11,0
10	10,0	01,1
11	11,0	00,1

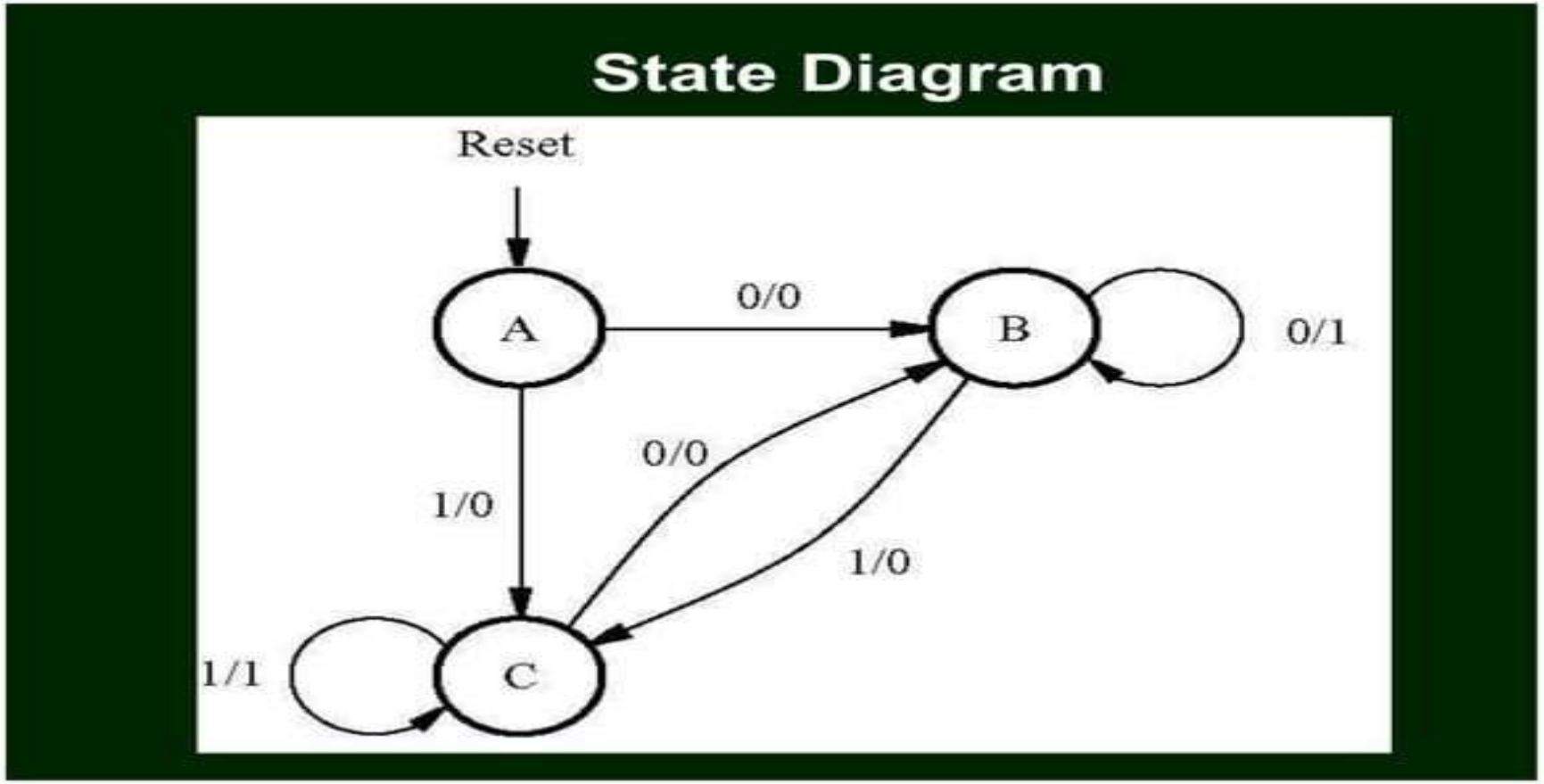
Mealy Machine



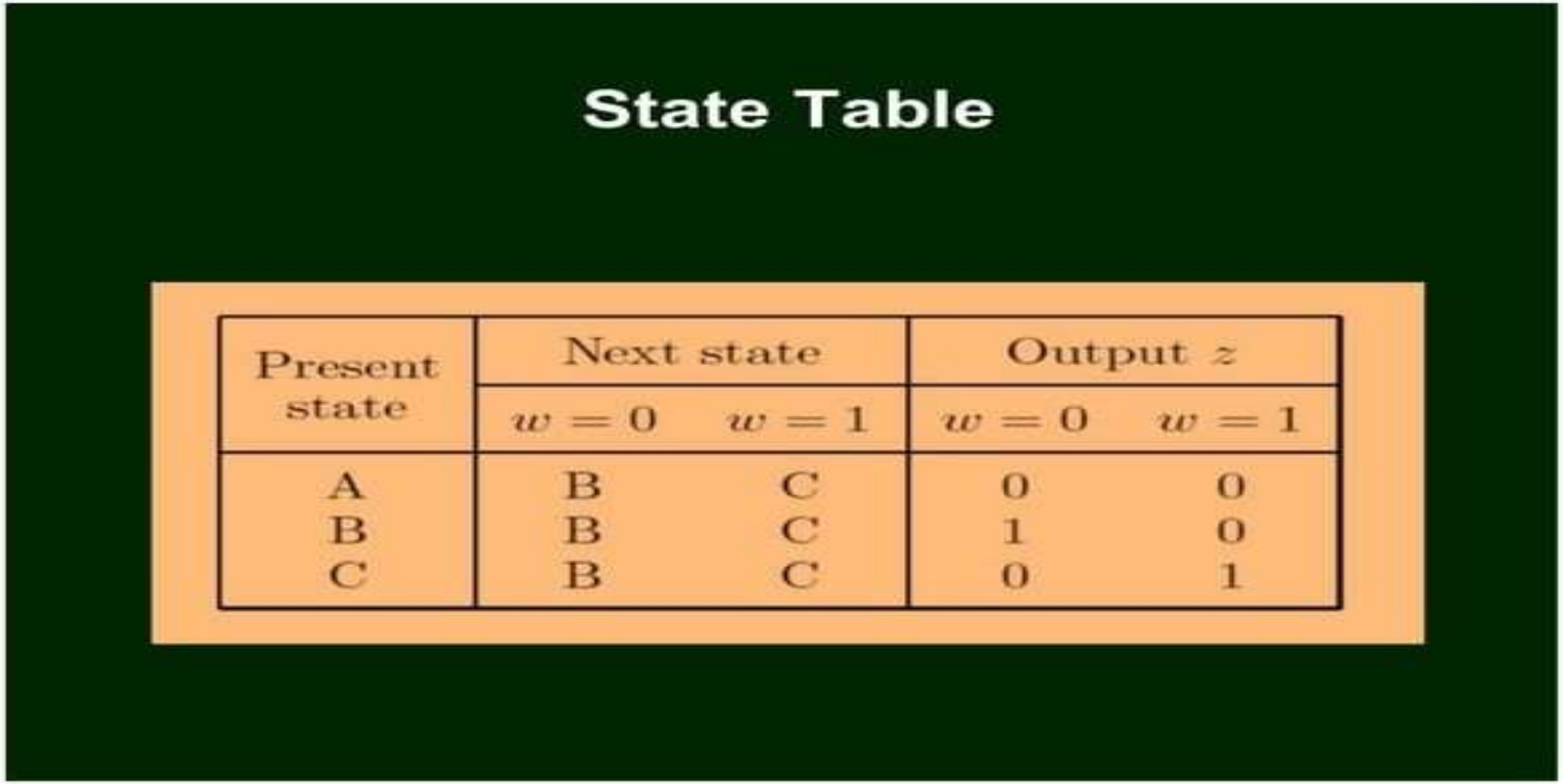
Mealy Machine Examples



State Diagram



State Table



Present state	Next state		Output z	
	$w = 0$	$w = 1$	$w = 0$	$w = 1$
A	B	C	0	0
B	B	C	1	0
C	B	C	0	1

THANK YOU