Lecture 8
EE 421 / CS 425
Digital System Design

Spring 2023

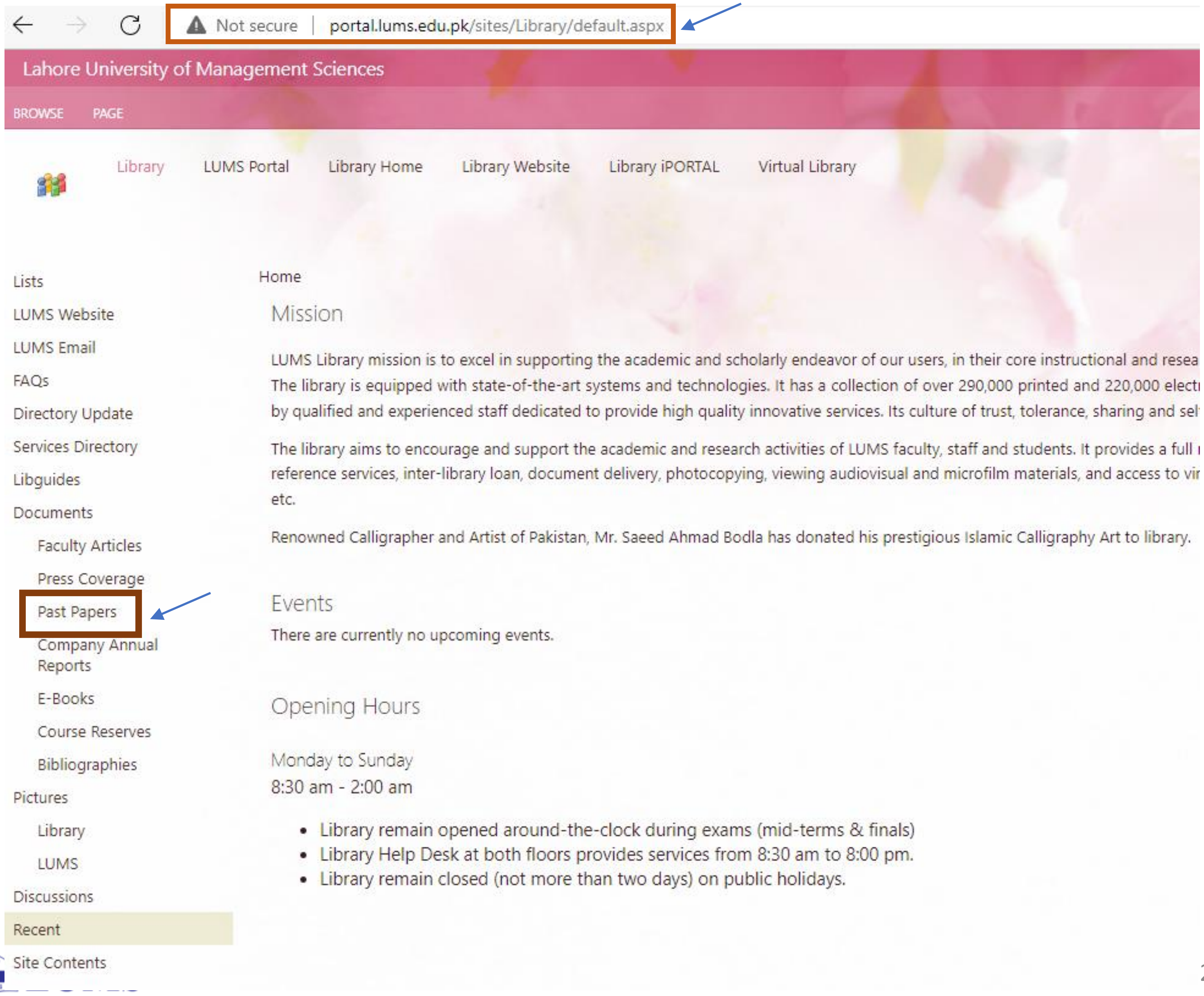
Shahid Masud

Topics

- State Diagrams
- Introduction to State Machines
- Moore State Machine and Mealy State Machine
- State Tables Description of State Machines
- Design Example of a State Machine (Vending Machine)
- Sequential State Machine Circuit Design
- One Hot Encoding and ASMD

ANNOUNCEMENT
QUIZ 2
NEXT WEEK

Practice Questions from Past Papers



Check folders:

SSE → EE 421

and

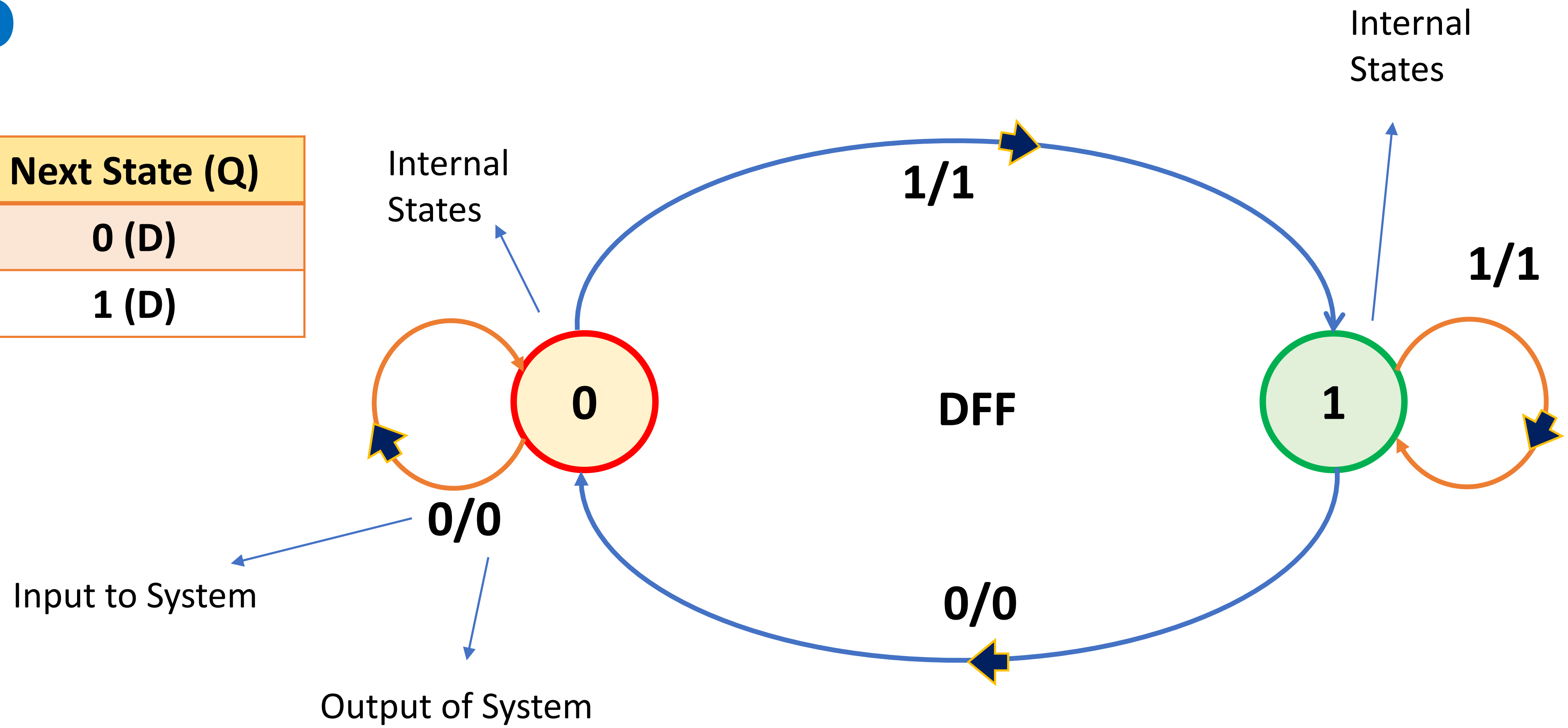
CS → CMPE 424

and

CS → CS 424

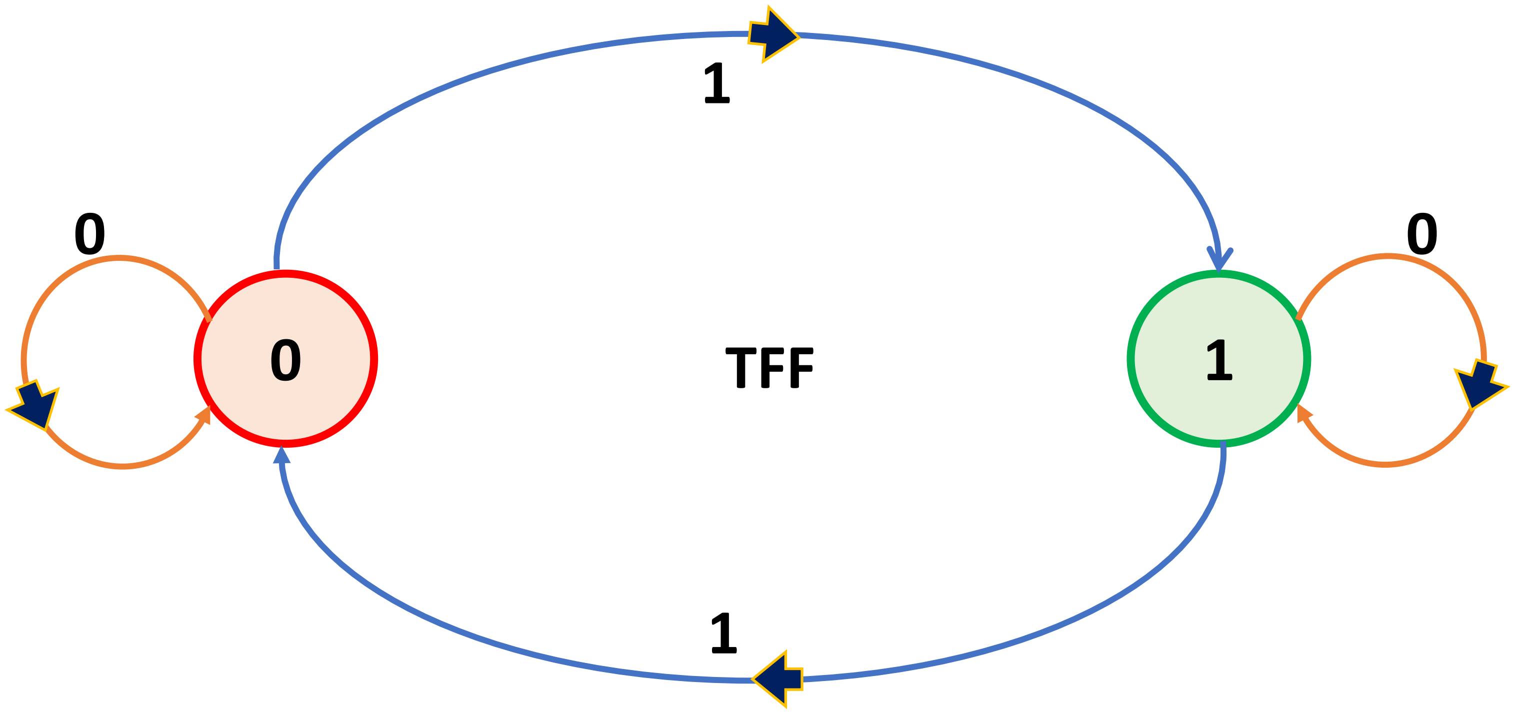
State Diagram of D Flipflop

Input (D)	Next State (Q)
0	0 (D)
1	1 (D)



State Diagram of T Flipflop

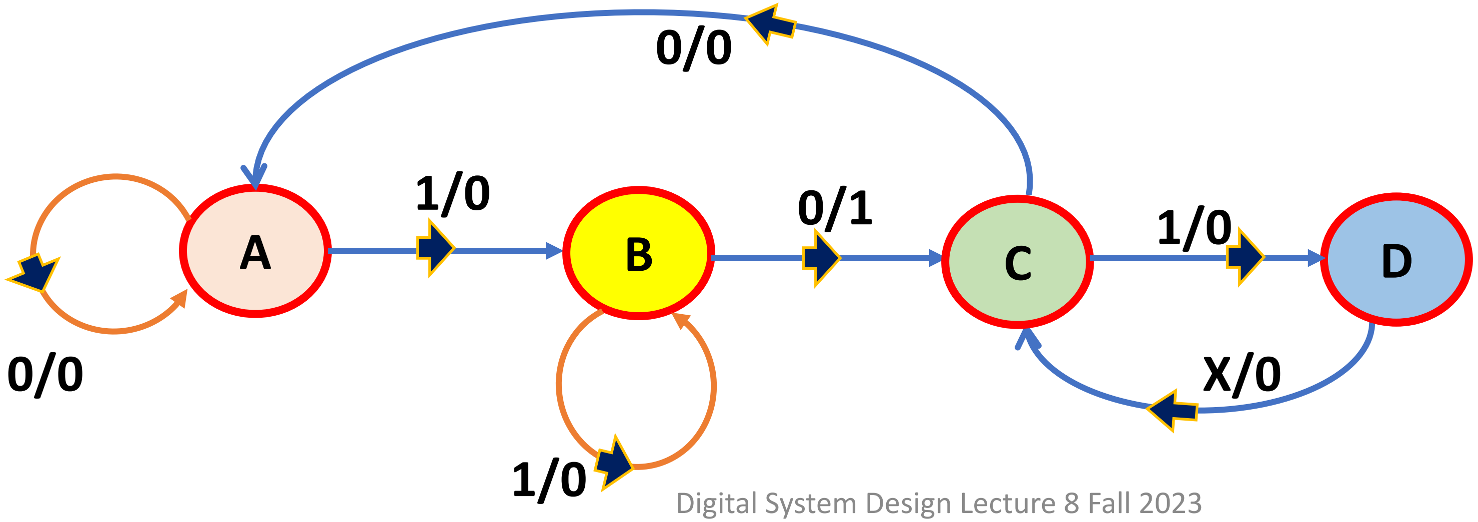
Input (T)	Next State (Q)
0	Q
1	Q’



Linking State Table with State Diagram

Complete State Description Including Inputs, Present State, Outputs and Next State

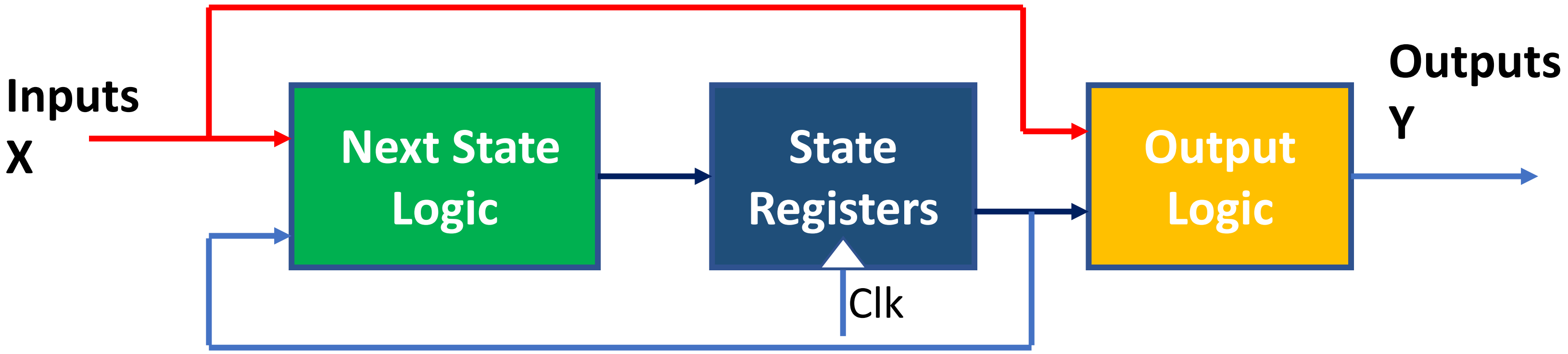
Q (t)	Q (t+1)		Z (Output)	
	Input X=0	Input X=1	Input X=0	Input X=1
A	A	B	0	0
B	C	B	1	0
C	A	D	0	0
D	C	C	0	0



Types of State Machines

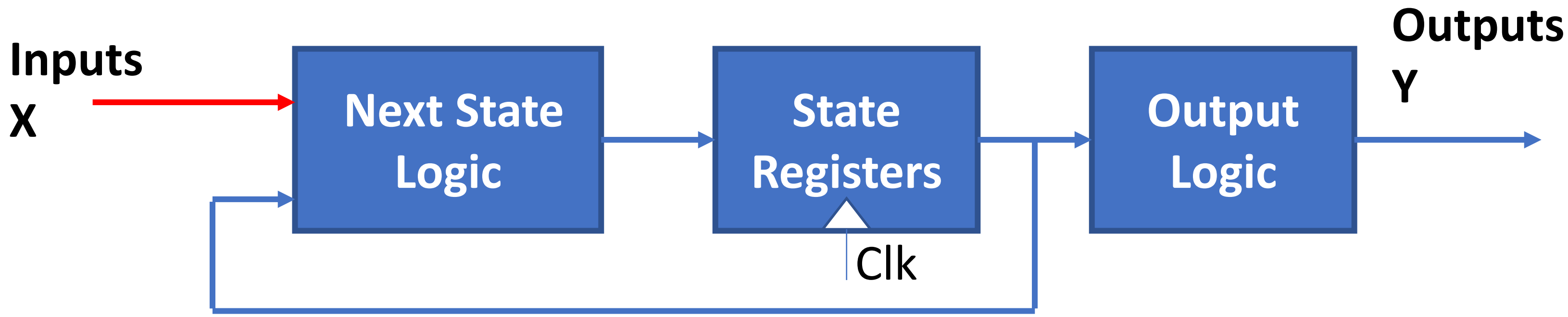
- **Mealy Machine**
  - Output depends upon Internal State plus External Inputs
  - Output can change at any time and not necessarily after a Clocked event
- **Moore Machine**
  - Output depends upon External Inputs and Current Internal State
  - Output is Synchronized with the Change in Internal States

Mealy State Machine Block Diagram



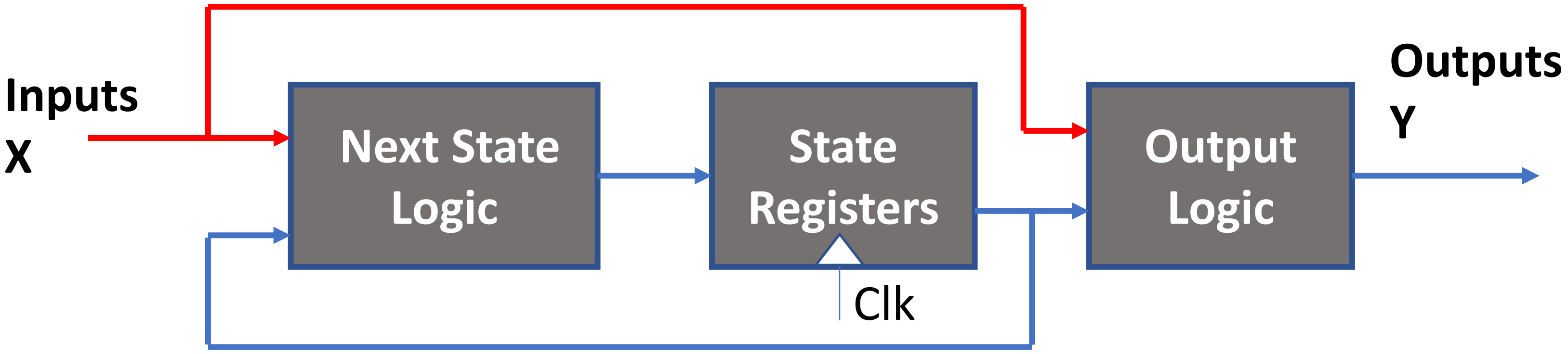
State Machine Block Diagrams - Comparison

Moore



Side by side

Mealy



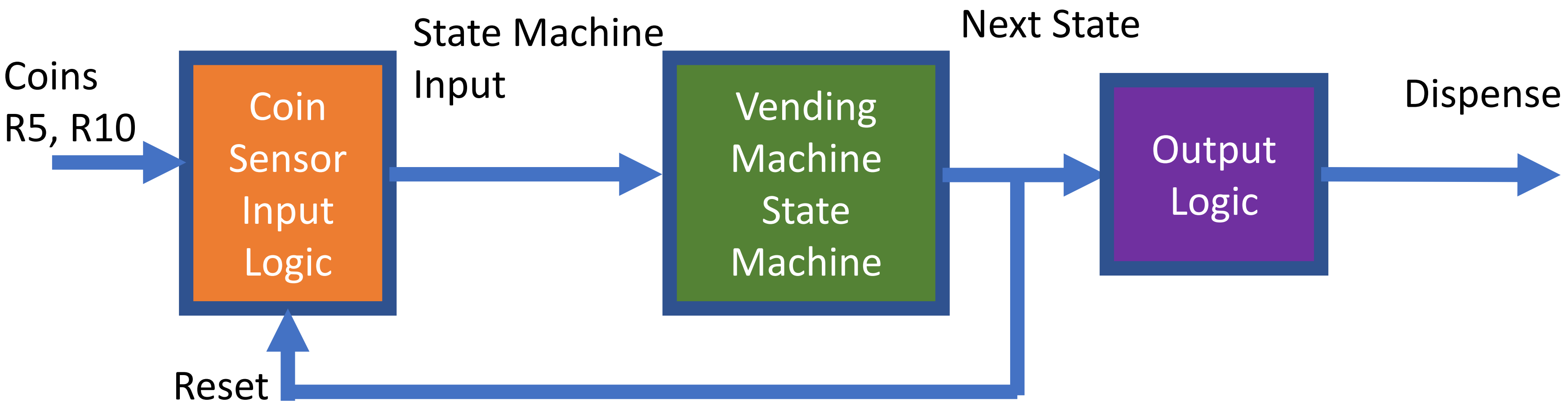
State Machine Based Synchronous Design

- Understand the Problem – desired sequence based on inputs and present state
- Develop abstract representation of FSM – A state diagram or a state table that shows all possible states and transitions
- Perform state minimization – to achieve efficient implementation
- Perform state assignment
- Choose appropriate flipflop for storage elements (eg. D flipflops)
- Use K-maps to determine characteristic equations for Next State
- After K-maps based minimization; draw complete logic circuit using combinational and sequential elements

Example of a State Machine Based Design – Coke Vending Machine

- Machine dispenses a can of coke for Rs. 15/-
- You can provide coins of either Rs. 5/- or Rs. 10/-
- The Machine does not provide any change back
- The Machine is 'Reset' after the can has been dispensed

Block Diagram of Vending Machine Example



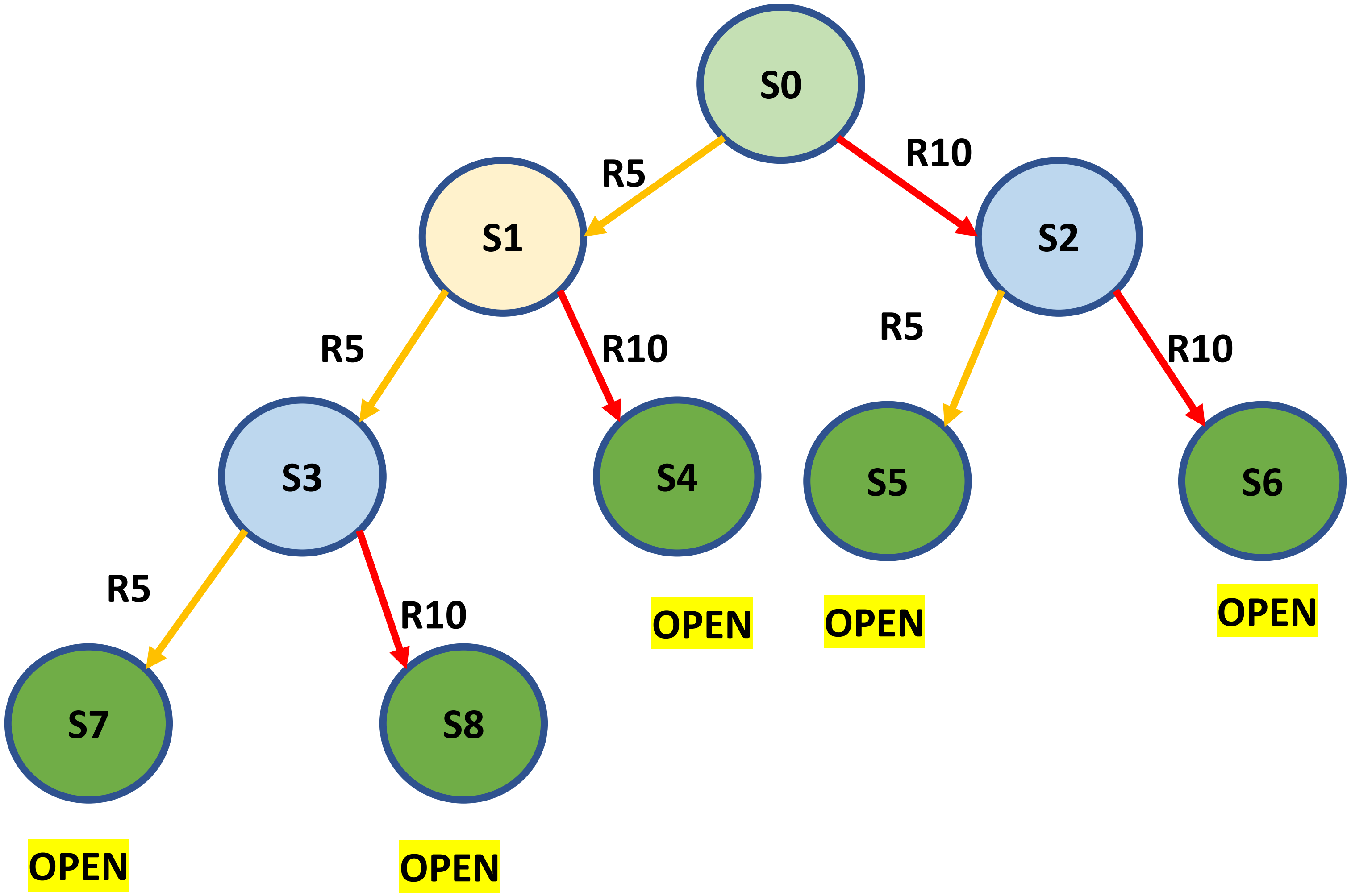
Is Input Directly Connected to Output Logic?

Question: Which type is this? Moore or Mealy??

Elaborate State Machine in text description

- Enumerate all possible inputs and outputs
- Objective: Insert sufficient coins to release a can of Coke
- Either Insert R5 + R5 + R5 in sequence
- OR Insert R5 + R10 in sequence
- OR Insert R10 + R5 in sequence
- OR Insert R10 + R10 in sequence
- OR Insert R5 + R5 + R10 in sequence

Make a State Diagram Representation of State Machine (Moore Machine)

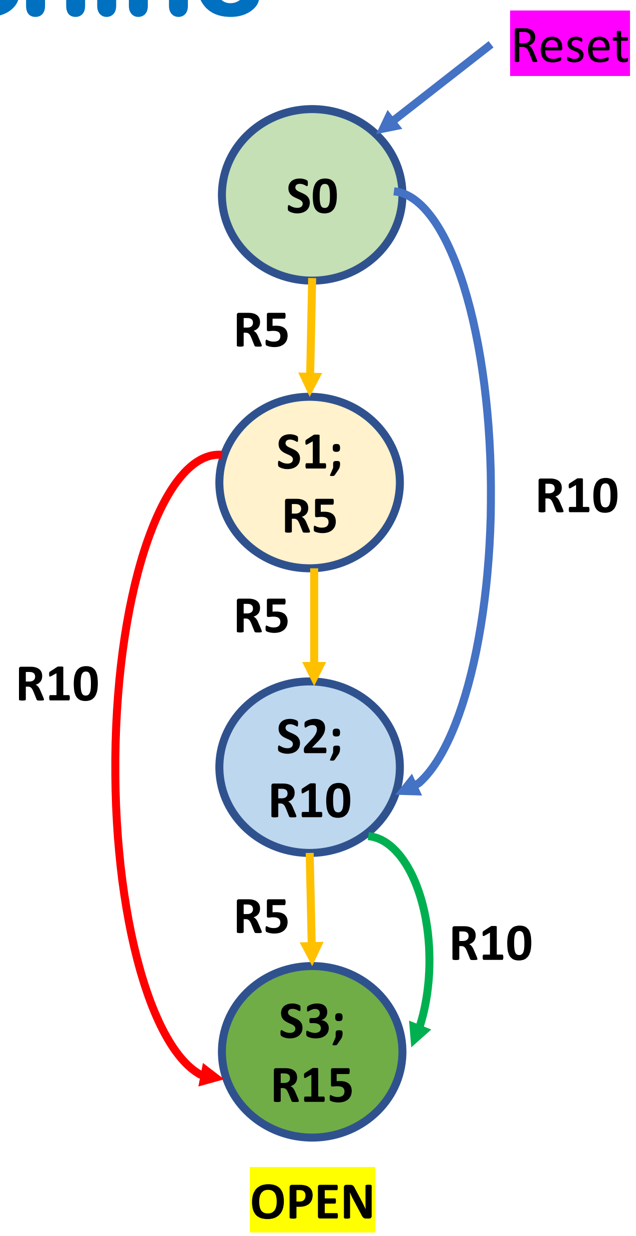


Simplifying the State Machine

Identify Similar States (same present state and next state)

State Minimization by Observation

- **Reset brings to State S0**
- **State S1 represents R5 received so far, one possible path**
- **State S2 represents R10 received so far, two possible paths;**
- **State S3 represents R15 received so far, three possible paths**
- **S4, S5, S6, S7 have identical behaviour so they can be Combined into one state**



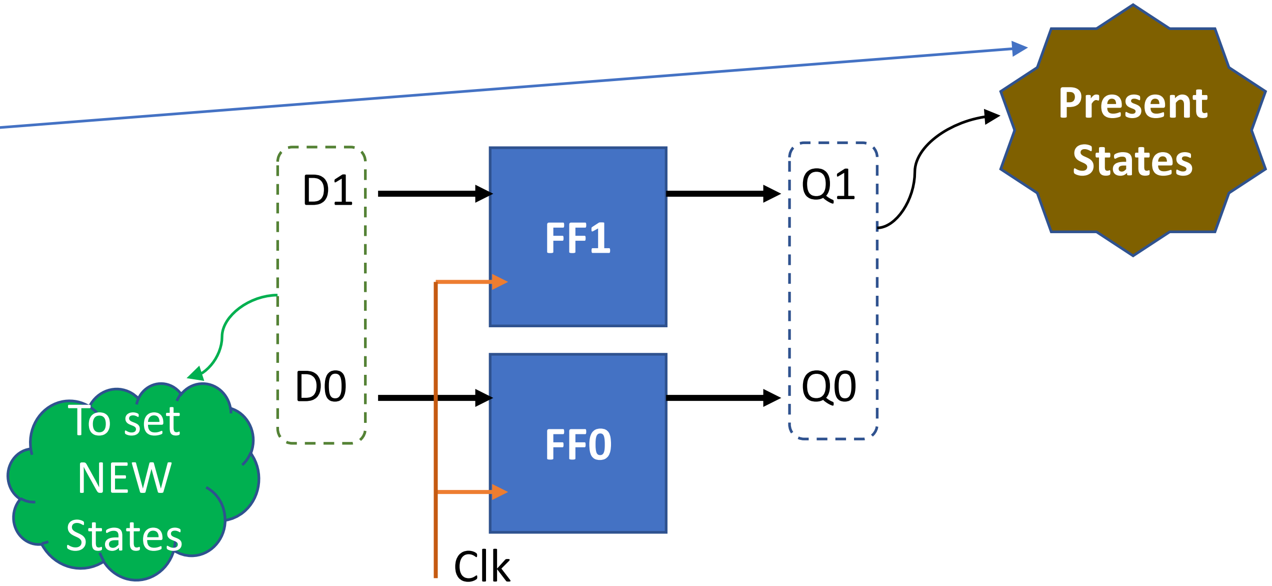
State Transition Table Describing Vending Machine

Q1	Q0	Present State	Inputs		Next State	Output OPEN
			R10	R5		
0	0	S0; R0	0	0	S0; R0	0
			0	1	S1; R5	0
			1	0	S2; R10	0
			1	1	Not Allowed	X
0	1	S1; R5	0	0	S1; R5	0
			0	1	S2; R10	0
			1	0	S3; R15	0
			1	1	Not Allowed	X
1	0	S2; R10	0	0	S2; R10	0
			0	1	S3; R15	0
			1	0	S3; R20	0
			1	1	Not Allowed	X
1	1	S3; R15	0	0	S3; R15	1
			0	1	S3; R15	1
			1	0	S3; R15	1
			1	1	Not Allowed	X

State Mapping to Flipflops

- In the reduced State Diagram, there are 4 states
- We can distinctly represent these states using two Flipflops
- Assign states as follows:

States	Code	Q1	Q0
S0	00	0	0
S1	01	0	1
S2	10	1	0
S3	11	1	1



Present State to Next State Table using DFF

Present State		Inputs		Next State		Output OPEN
Q1	Q0	R10	R5	D1	D0	
0	0 (S0)	0	0	0	0 (S0)	0
		0	1	0	1 (S1)	0
		1	0	1	0 (S2)	0
		1	1	X	X	X
0	1 (S1)	0	0	0	1 (S1)	0
		0	1	1	0 (S2)	0
		1	0	1	1 (S3)	0
		1	1	X	X	X
1	0 (S2)	0	0	1	0 (S2)	0
		0	1	1	1 (S3)	0
		1	0	1	1 (S3)	0
		1	1	X	X	X
1	1 (S3)	0	0	1	1 (S3)	1
		0	1	1	1 (S3)	1
		1	0	1	1 (S3)	1
		1	1	X	X	X

Characteristic Equation for D0 using K-Map

R10,R5\Q1Q0	00	01	11	10
00	0	1	1	0
01	1	0	1	1
11	X	X	X	X
10	0	1	1	1

R5'.Q0

R5.Q0'

R10.Q1

Q1Q0

$$\boxed{D0 = (R5'.Q0) + (R5.Q0') + (R10.Q1) + (Q1.Q0)}$$

LUMS

Characteristic Equation for D1 using K-Map

Q1

R10,R5\Q1Q0	00	01	11	10
00	0	0	1	1
01	0	1	1	1
11	X	X	X	X
10	1	1	1	1

R5.Q0

R10

$\boxed{D1 = Q1 + R10 + (R5.Q0)}$

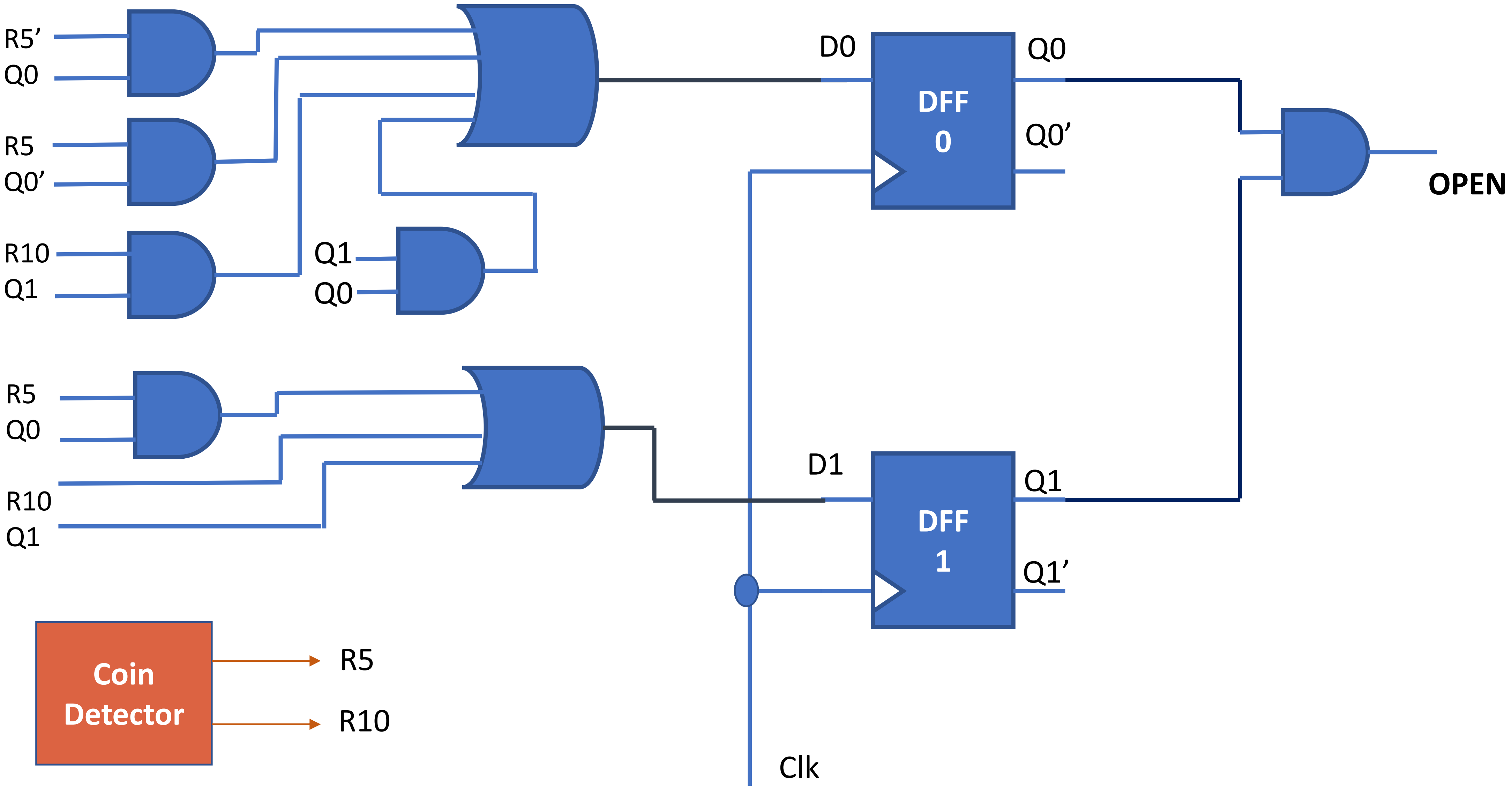
Characteristic Equation for OPEN using K-Map

R10,R5\Q1Q0	00	01	11	10
00	0	0	1	0
01	0	0	1	0
11	X	X	X	X
10	0	0	1	0

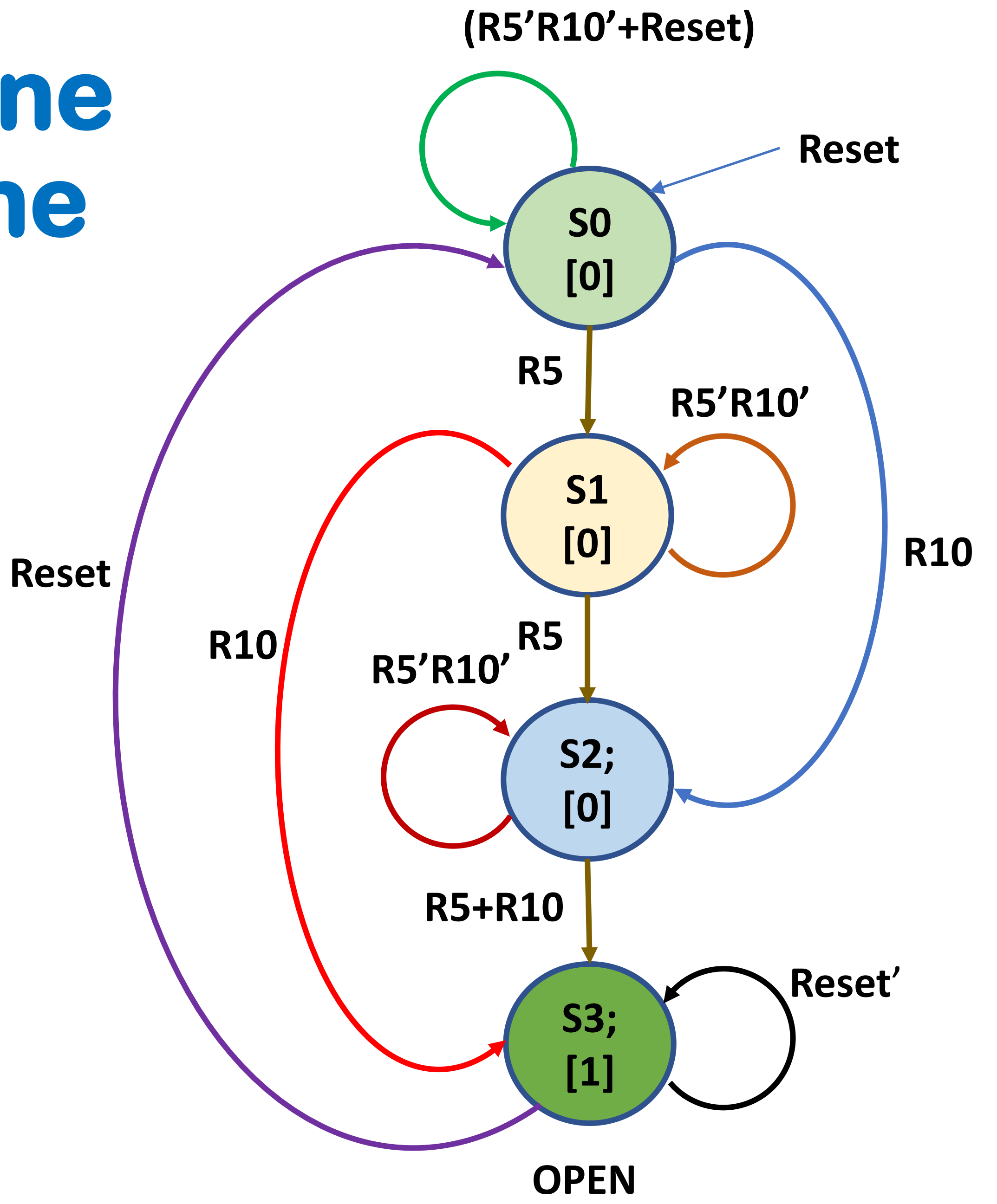
Q1Q0

OPEN = Q1Q0

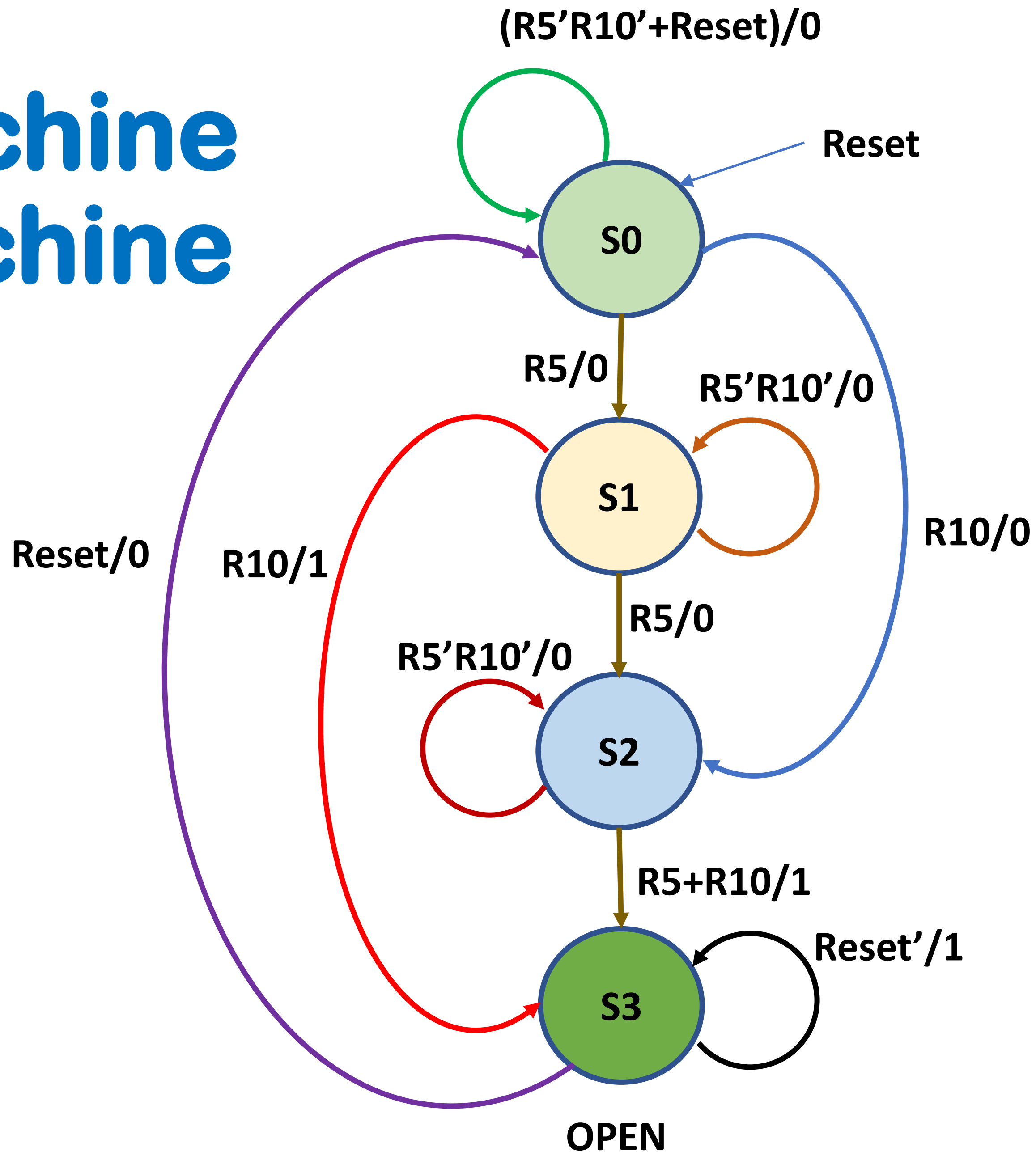
Vending Machine Circuit using DFF



Moore State Machine for Vending Machine



Mealy State Machine for Vending Machine



Some Comparison Moore vs Mealy

- Mealy machine requires fewer states to reach output in comparison with Moore machine
- Mealy machine is more susceptible to glitches
- Explicit output values are shown in Mealy machine associated with each transition
- Output changes after state is changed in Moore machine
- Output in Moore machine depends upon state only; inputs can steer the output towards a particular state that affects output
- Output depends upon present state and the present value at the input; thus, output can change immediately with the change in input, independent of synchronous clock.

One Hot Encoding – one FF for each state

Present State				Inputs		Next State				Output OPEN
Q3	Q2	Q1	Q0	R10	R5	D3	D2	D1	D0	Y
0	0	0	1	0	0	0	0	0	1	
				0	1	0	0	1	0	
				1	0	0	1	0	0	
				1	1	X	X	X	X	
0	0	1	0	0	0	0	0	1	0	
				0	1	0	1	0	0	
				1	0	1	0	0	0	1
				1	1	X	X	X	X	
0	1	0	0	0	0	0	1	0	0	
				0	1	1	0	0	0	1
				1	0	1	0	0	0	1
				1	1	X	X	X	X	
1	0	0	0	X	X	1	0	0	0	1

D3 is directly the Output and Its State

The Design Becomes Simpler
Less Combinational Logic – At the Expense of Extra DFF

Algorithmic State Machine Description - ASMD