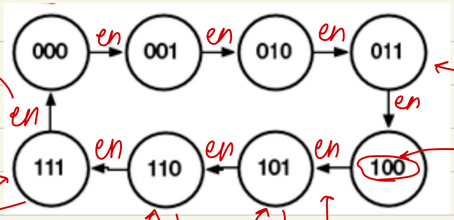
. has a certain number of states
. a plan to transition from one state to another

Finite State Machines

- very efficient model to describe the behavior of our circuits.

Sequence and State of Counters

- at a pt in time it has state A
- at the transition point, it changes to the next state corresponding with the next largest output



⇒ transitions only when en is asserted

en'

self transition when en is LOW

node : shows all possible states

states are labelled by o/p values

en'

en'

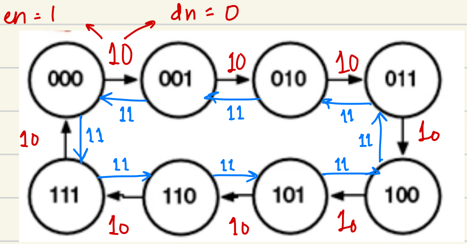
clear movement from one state to another is shown

not always necessary to draw

State transition diagram

- still a sequential circuit ⇒ change of state can only occur at the rising edge of a clock

Up-Down Counters

- count up : dn = LOW
  down : dn = HIGH
- all while : en = HIGH
- 

inputs : en, dn

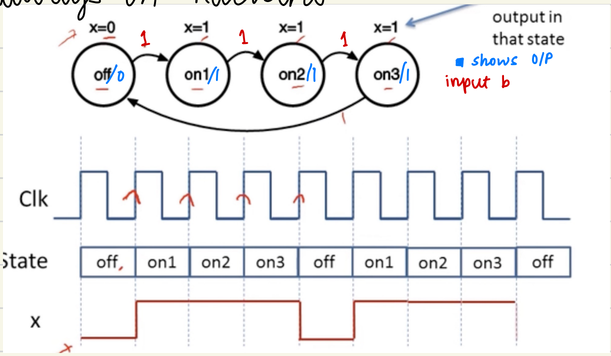
self transitions (00, 01) can be excluded from the diagram

Back to FSM → general object that can execute an abstract language

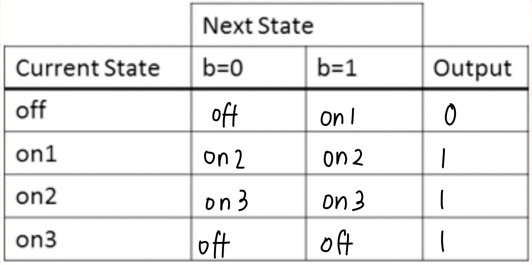
- it's a system using a finite number of states and associated transitions
- in synchronous design, an FSM is in one state for the duration of each clock cycle
- at every rising clock edge, the FSM may transition to another state

  may → depends on input values at the rising edge

- always on machine

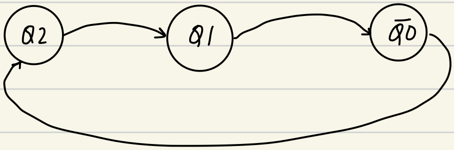


- state transition table



	Next State		
Current State	b=0	b=1	Output
off	off	on1	0
on1	on2	on2	1
on2	on3	on3	1
on3	off	off	1

task 1
Q1



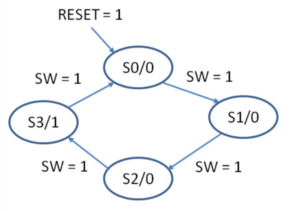
← x

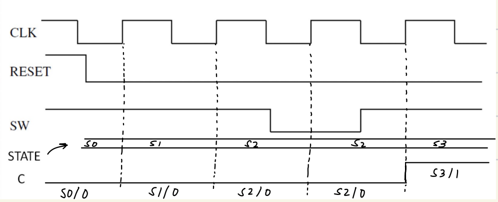
Q2Q1Q0 is the O/P

at power up/ reset Q2Q1Q0 → 000 ⇒ ($\bar{Q}0$ = 1) → 100 → 110 → 111

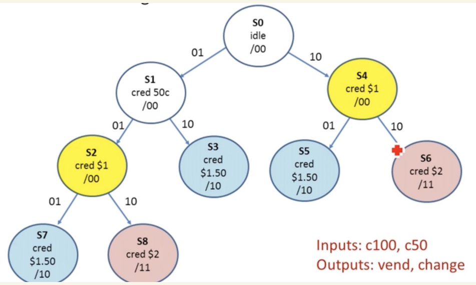
000 ← 001 ← 011

Q2



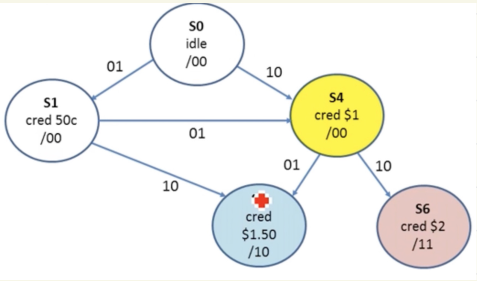


FSM for a vending machine



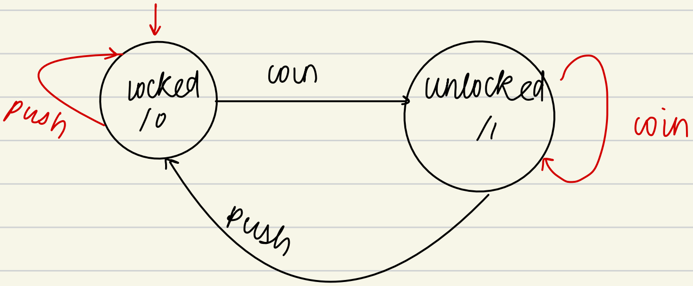
Inputs: c100, c50
Outputs: vend, change

simp-lified →



task 2

→ coin , → push   output : light



Moore vs Mealy

- Moore machines : outputs depend only on state
  ie. in state x, output is always f

- Mealy machines : output depends on the state and current values of inputs
  ie. if inputs change mid-state, then outputs also change