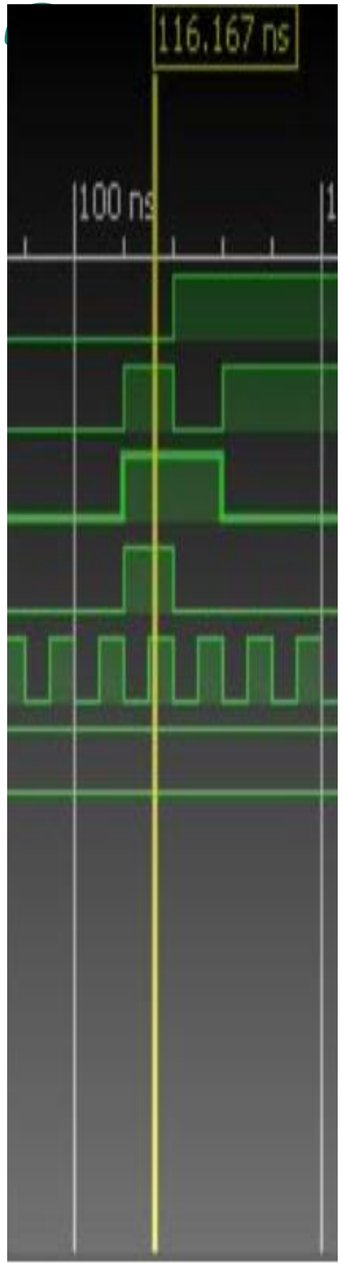
Sequential logic II

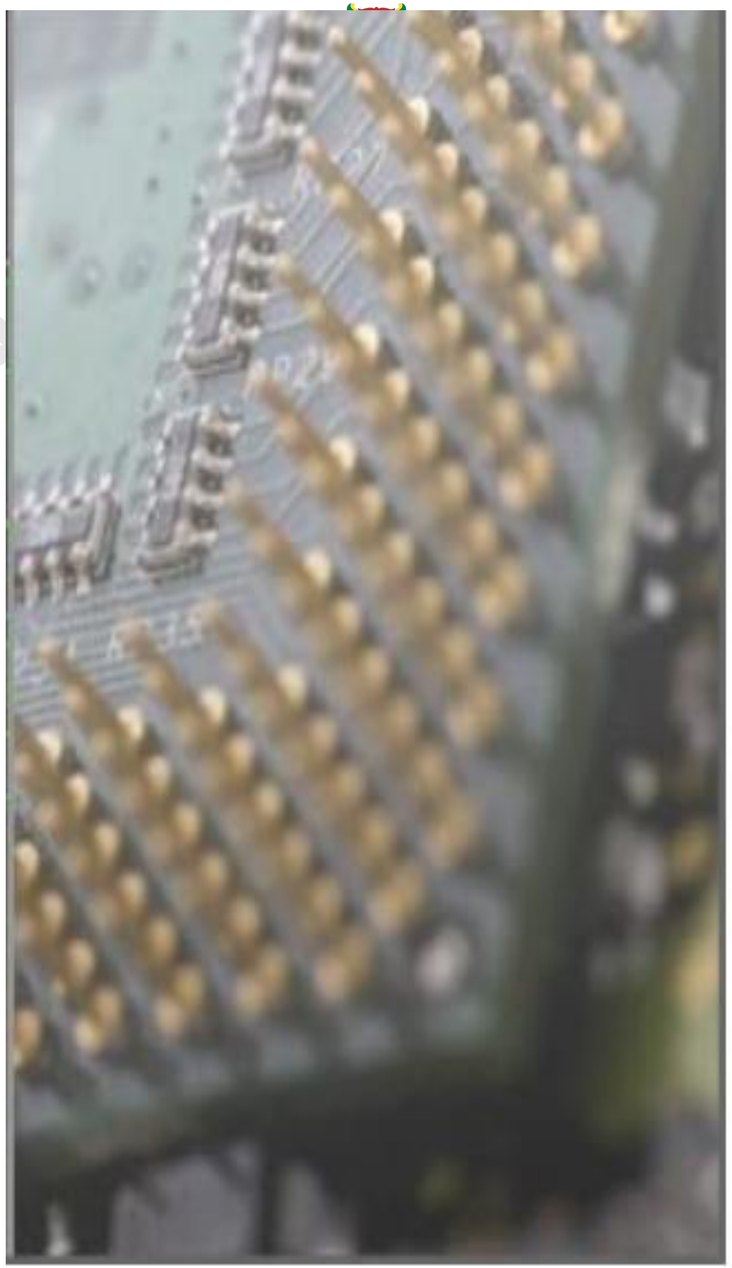
MC. Martin González Pérez

Agenda

- Finite State Machines
- Sequential building blocks

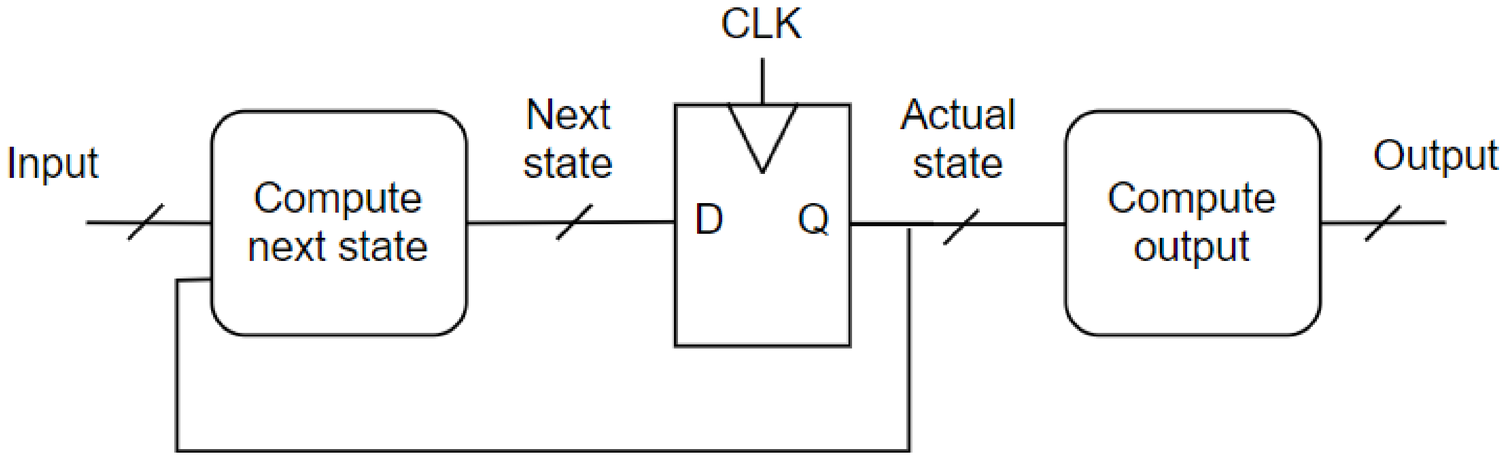


Finite State Machines



Finite State Machines

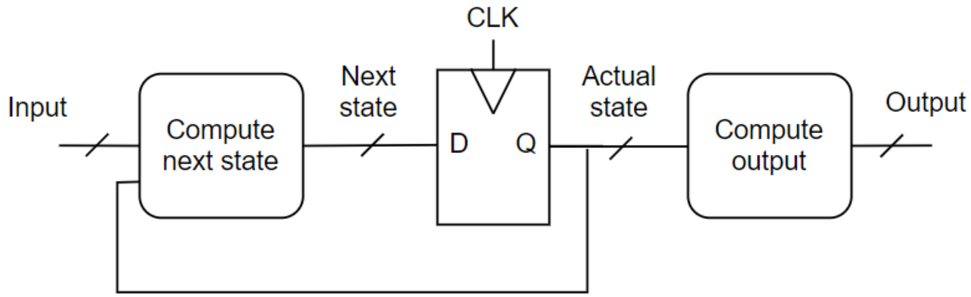
A sequential logic circuit is composed of combinational circuits and registers. The outputs of the registers represent the current state **(Q)**, while the inputs of the registers **(D)** represent the next state. A system that includes **n registers** can be in one of a finite number of **states (2^n)**, which is why these sequential systems are called Finite State Machines (**FSMs**).



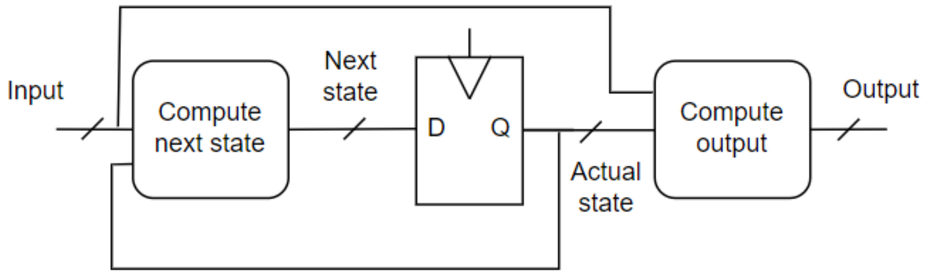
FSM Classification

FSM provides a systematic method to design synchronous sequential circuits. There are 2 types of FSM.

- Moore FSM: Output depend only on current state.



- Mealy FSM: output depend on current state and input.



FSM design

A procedure for designing circuits using state machines consists of four steps.

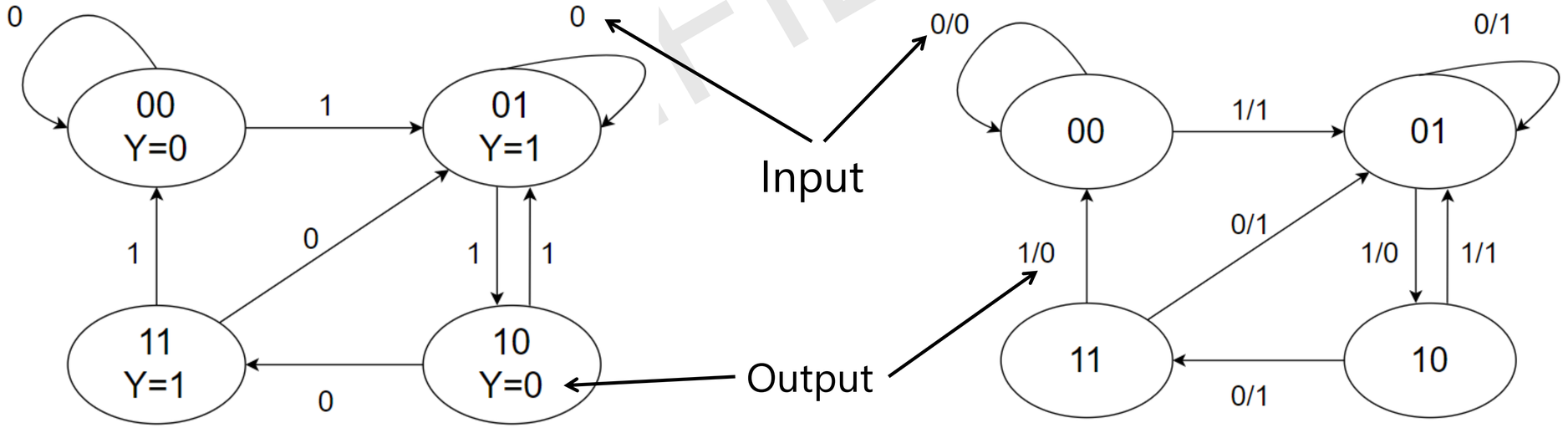
- Draw a state transition diagram.
- Create a state table.
- Obtain state equation.
- Create sequential circuit.

State Diagram

In a state diagram, each possible state is represented as a circle and transitions from one state to another is represented as an arrow.

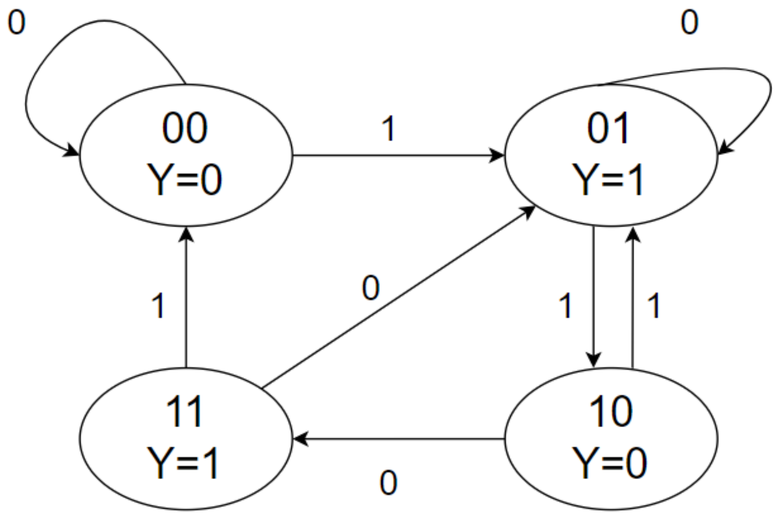
Moore FSM

Mealy FSM



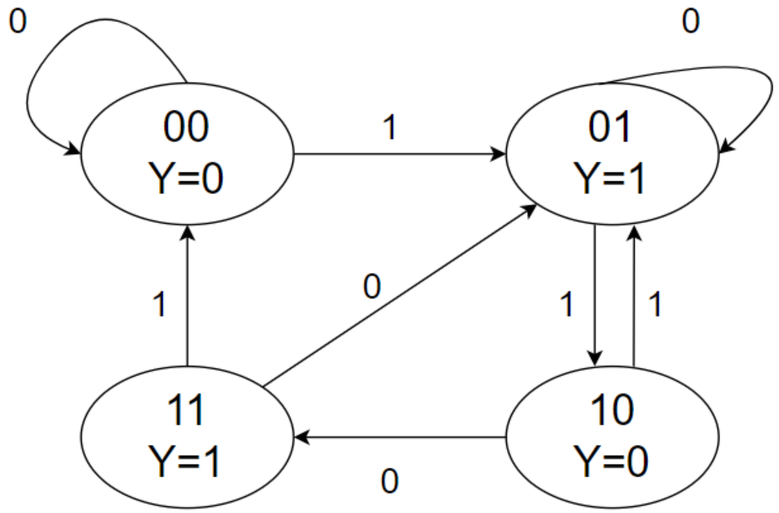
Exercise

Design a Moore machine based in the state diagram using D flip-flop.



State table

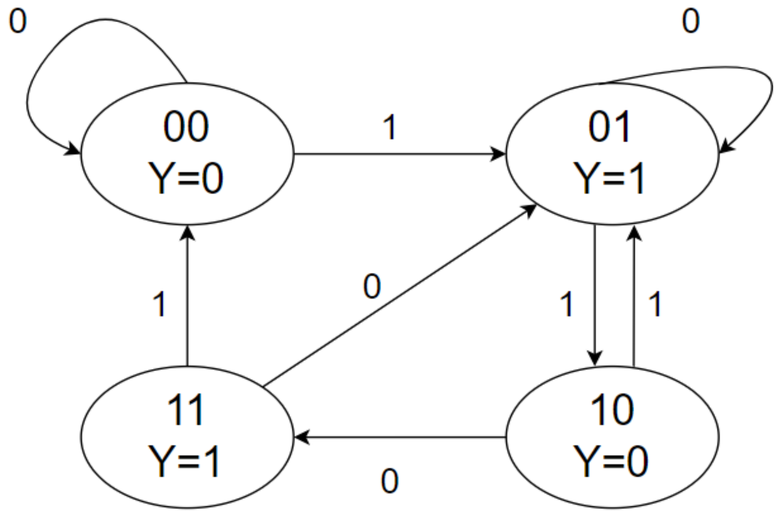
State diagram can be represented as state table.



State		Input	Next State	
Q1	Q0	A	Q1′	Q0′
0	0	0		
0	0	1		
0	1	0		
0	1	1		
1	0	0		
1	0	1		
1	1	0		
1	1	1		

State table

State diagram can be represented as state table.



State		Input	Next State	
Q1	Q0	A	Q1′	Q0′
0	0	0	0	0
0	0	1	0	1
0	1	0	0	1
0	1	1	1	0
1	0	0	1	1
1	0	1	0	1
1	1	0	0	1
1	1	1	0	0

State equation

By using state tables, we can deduce Boolean equations for flip flop inputs.

State		Input	Next State		FF input	
Q1	Q0	A	Q1′	Q0′	D1	D0
0	0	0	0	0	0	0
0	0	1	0	1	0	1
0	1	0	0	1	0	1
0	1	1	1	0	1	0
1	0	0	1	1	1	1
1	0	1	0	1	0	1
1	1	0	0	1	0	1
1	1	1	0	0	0	0

For D Flip Flop
D=Q′

State equation

By using state tables, we can deduce Boolean equations for flip flop inputs.

State		Input	Next State		FF input	
Q1	Q0	A	Q1′	Q0′	D1	D0
0	0	0	0	0	0	0
0	0	1	0	1	0	1
0	1	0	0	1	0	1
0	1	1	1	0	1	0
1	0	0	1	1	1	1
1	0	1	0	1	0	1
1	1	0	0	1	0	1
1	1	1	0	0	0	0

For D Flip Flop
D=Q′

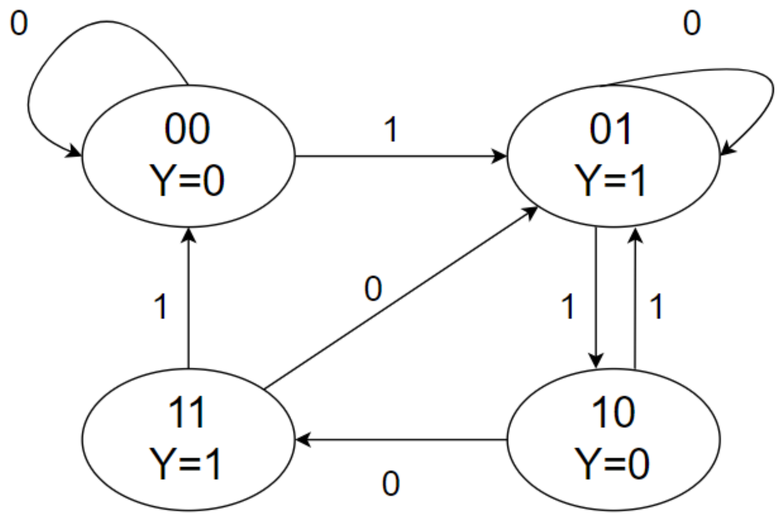
A \ Q1Q0	00	01	11	10
0	0	1	1	1
1	1	0	0	1

$$D0 = Q0\bar{A} + Q1\overline{Q0} + \overline{Q0}A$$

$$D1 = \overline{Q1}Q0A + Q1\overline{Q0}\,\bar{A}$$

State equation

Since Moore state machines only depend on the current state, we can obtain the output through a truth table that only involves the states.



Q1	Q0	Y
0	0	0
0	1	1
1	0	0
1	1	1

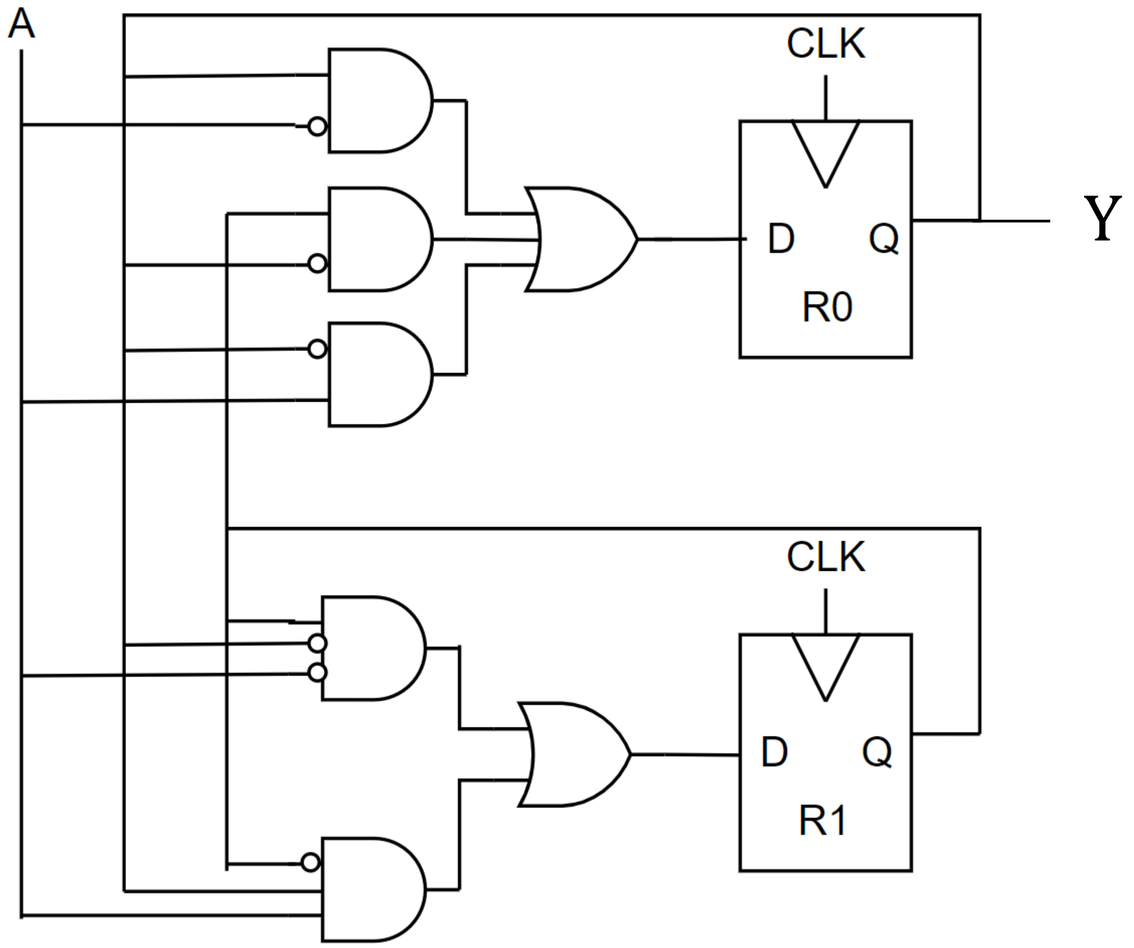
$$Y = Q0\overline{Q1} + Q0Q1 = Q0$$

Creating schematic

$$\mathrm{D1} = \overline{Q1}Q0A + Q1\overline{Q0}\,\bar{A}$$

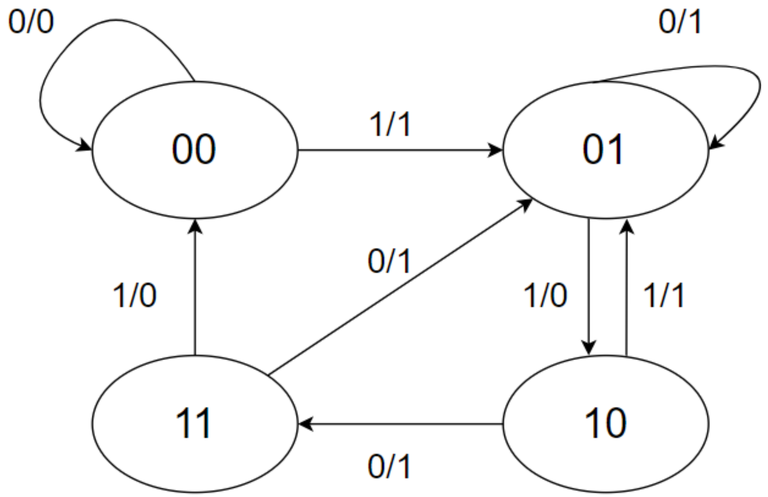
$$\mathrm{D0} = Q0\bar{A} + Q1\overline{Q0} + \overline{Q0}A$$

$$\mathrm{Y} = Q0$$



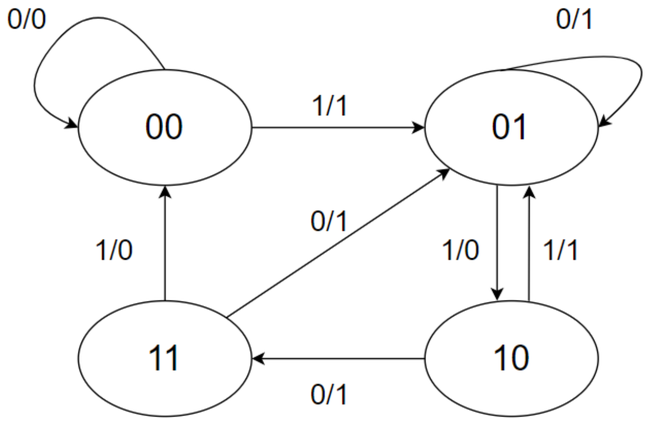
Exercise

Design a Mealy machine based in the state diagram using JK flip-flop.



State equation

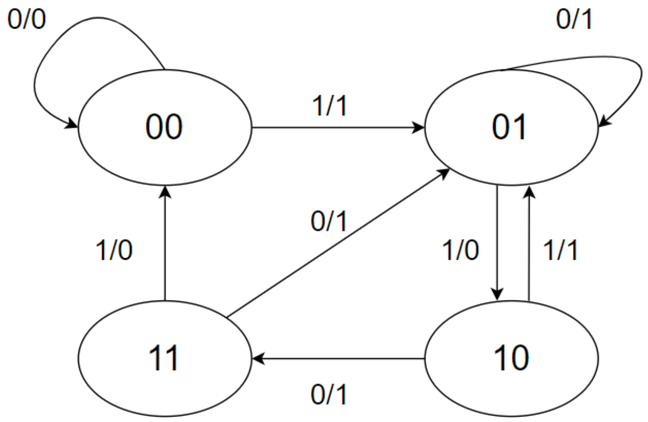
By using state tables, we can deduce Boolean equations for flip flop inputs.



State		Input	Next State		Output	FF input	
Q1	Q0	A	Q1′	Q0′	Y	D1	D0
0	0	0					
0	0	1					
0	1	0					
0	1	1					
1	0	0					
1	0	1					
1	1	0					
1	1	1					

State equation

By using state tables, we can deduce Boolean equations for flip flop inputs.



State		Input	Next State		Output	FF input	
Q1	Q0	A	Q1′	Q0′	Y	D1	D0
0	0	0	0	0	0	0	0
0	0	1	0	1	1	0	1
0	1	0	0	1	1	0	1
0	1	1	1	0	0	1	0
1	0	0	1	1	1	1	1
1	0	1	0	1	1	0	1
1	1	0	0	1	1	0	1
1	1	1	0	0	0	0	0

State equation

By using state tables, we can deduce Boolean equations for flip flop inputs.

State		Input	Next State		Output	FF input	
Q1	Q0	A	Q1′	Q0′	Y	D1	D0
0	0	0	0	0	0	0	0
0	0	1	0	1	1	0	1
0	1	0	0	1	1	0	1
0	1	1	1	0	0	1	0
1	0	0	1	1	1	1	1
1	0	1	0	1	1	0	1
1	1	0	0	1	1	0	1
1	1	1	0	0	0	0	0

$$Y = D0 = Q0\bar{A} + Q1\overline{Q0} + \overline{Q0}A$$

$$D1 = \overline{Q1}Q0A + Q1\overline{Q0}\bar{A}$$

Y

A \ Q1Q0	00	01	11	10
0	0	1	1	1
1	1	0	0	1

D1

A \ Q1Q0	00	01	11	10
0	0	0	0	1
1	0	1	0	0

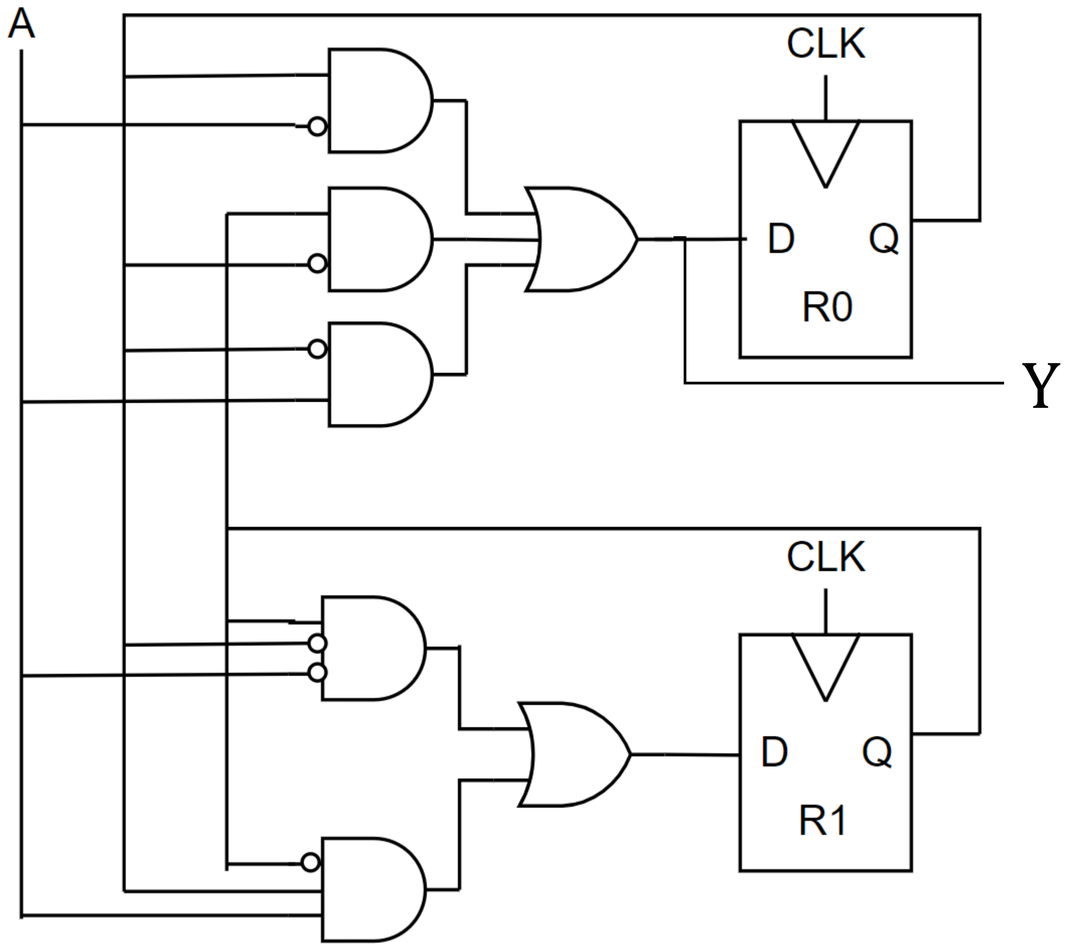
D0

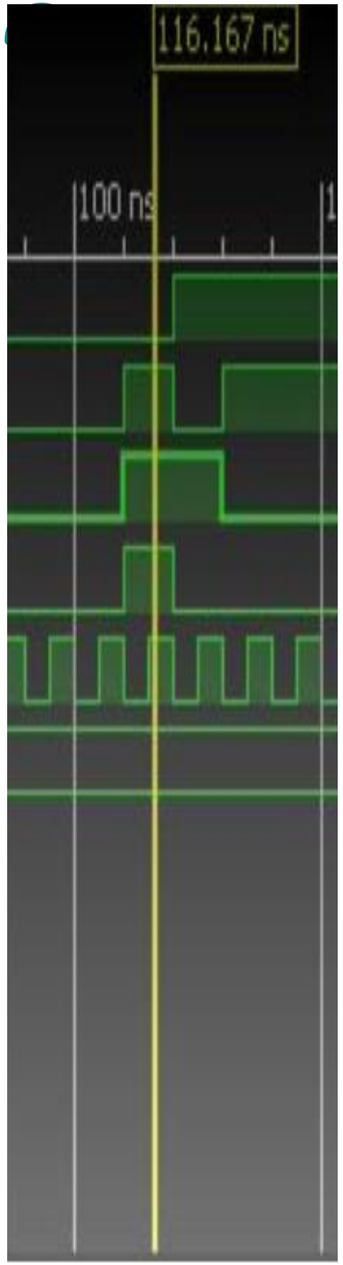
A \ Q1Q0	00	01	11	10
0	0	1	1	1
1	1	0	0	1

Creating schematic

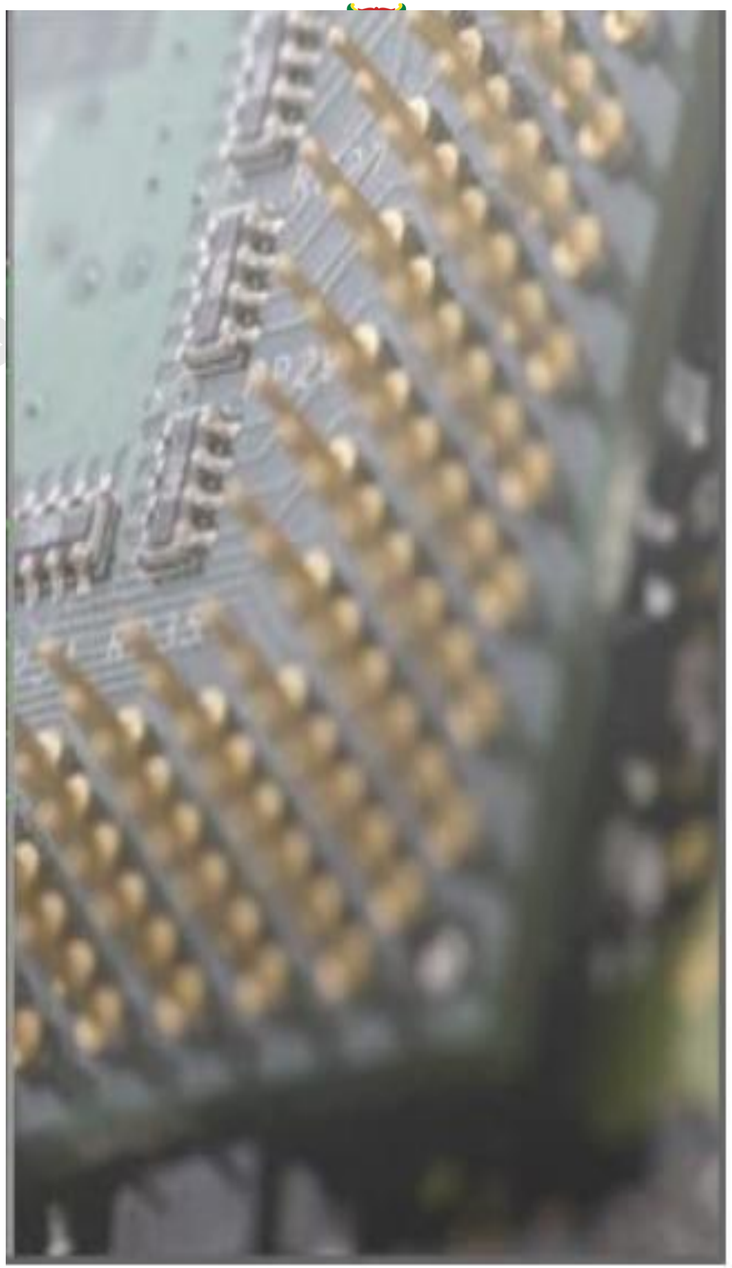
$$\mathrm{D1} = \overline{Q1}Q0A + Q1\overline{Q0}\,\bar{A}$$

$$\mathrm{Y} = \mathrm{D0} = Q0\bar{A} + Q1\overline{Q0} + \overline{Q0}A$$

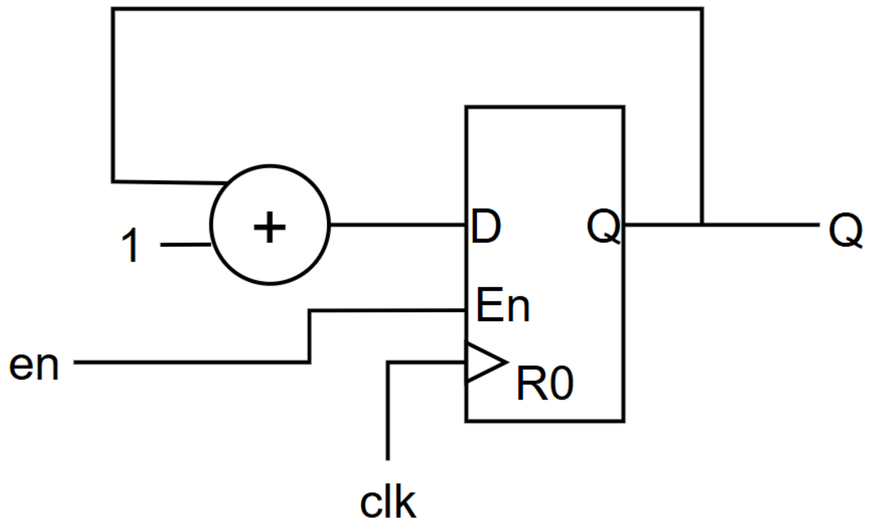




Sequential building blocks

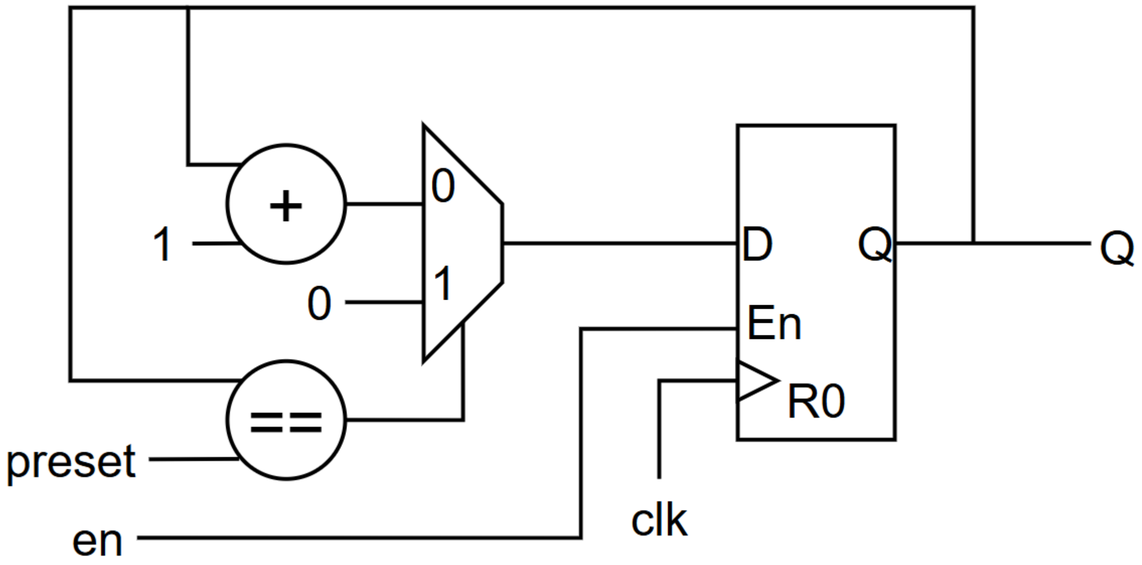


Counter



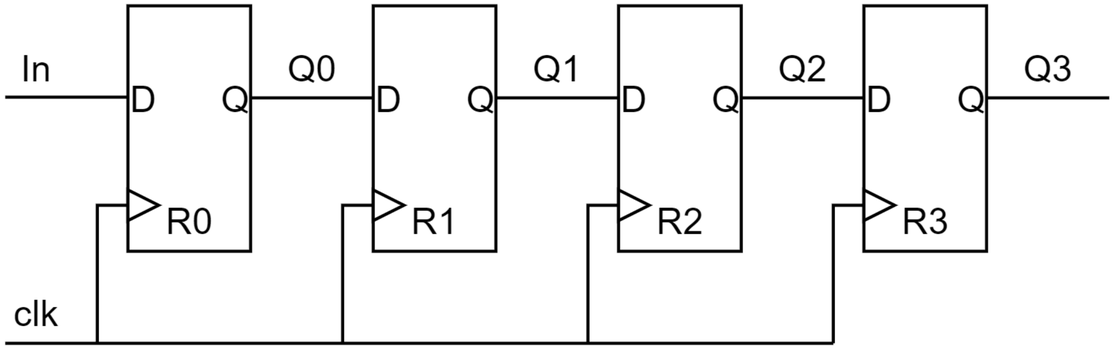

```
if (en)
    Q = Q + 1;
else
    Q = Q;
```

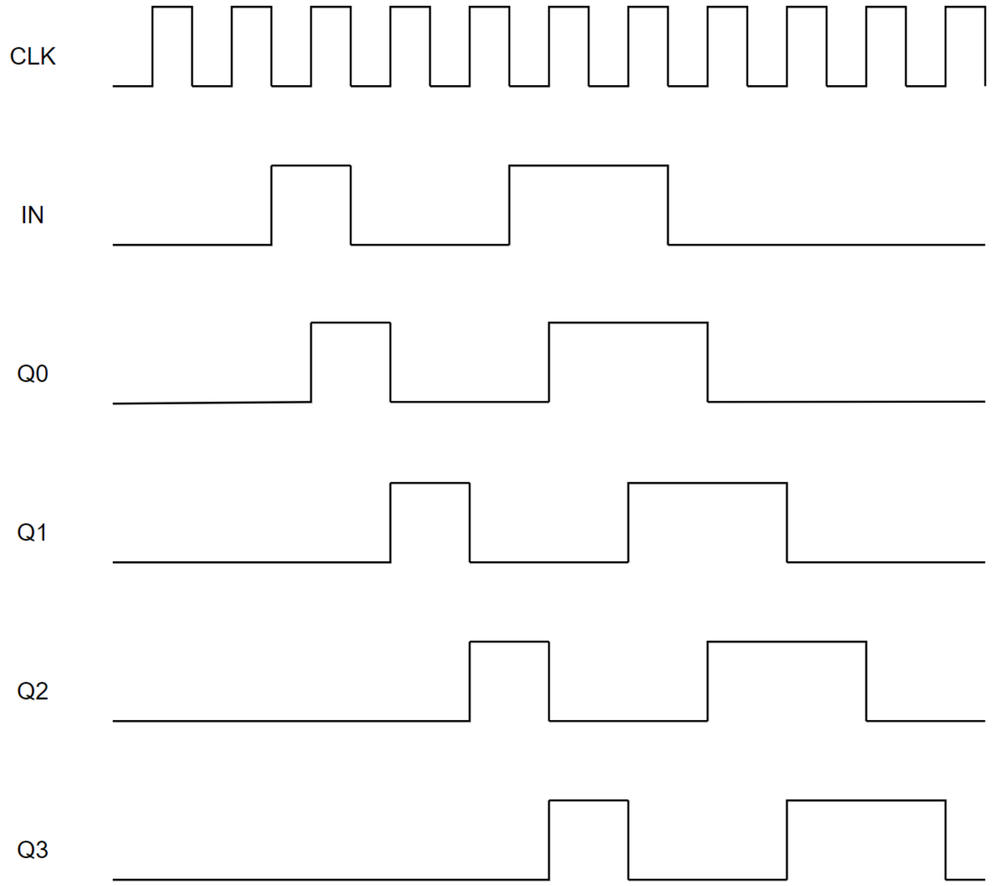
Counter with preset




```
if (en){
    if (Q==preset){
        Q = 0;
    }
    else
        Q = Q + 1;
    }
else
    Q = Q;
```

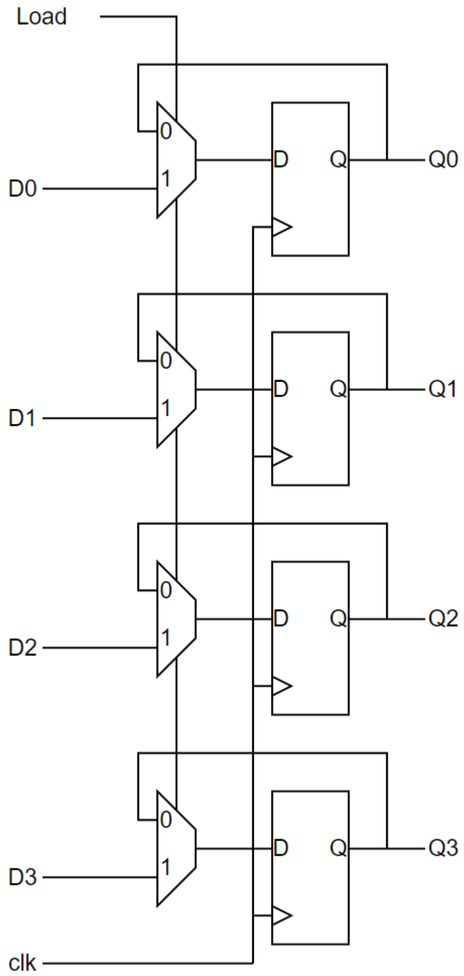
Shift register





Registers with load and hold

$$Q = D \circ Load + Q \circ \overline{Load}$$



Shift with load