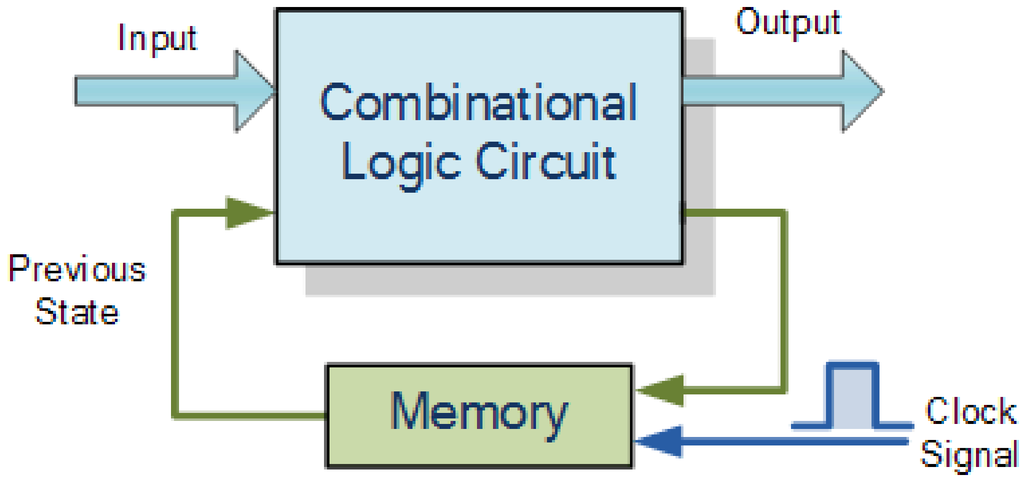
ESC201: Lecture 18

Dr. Imon Mondal

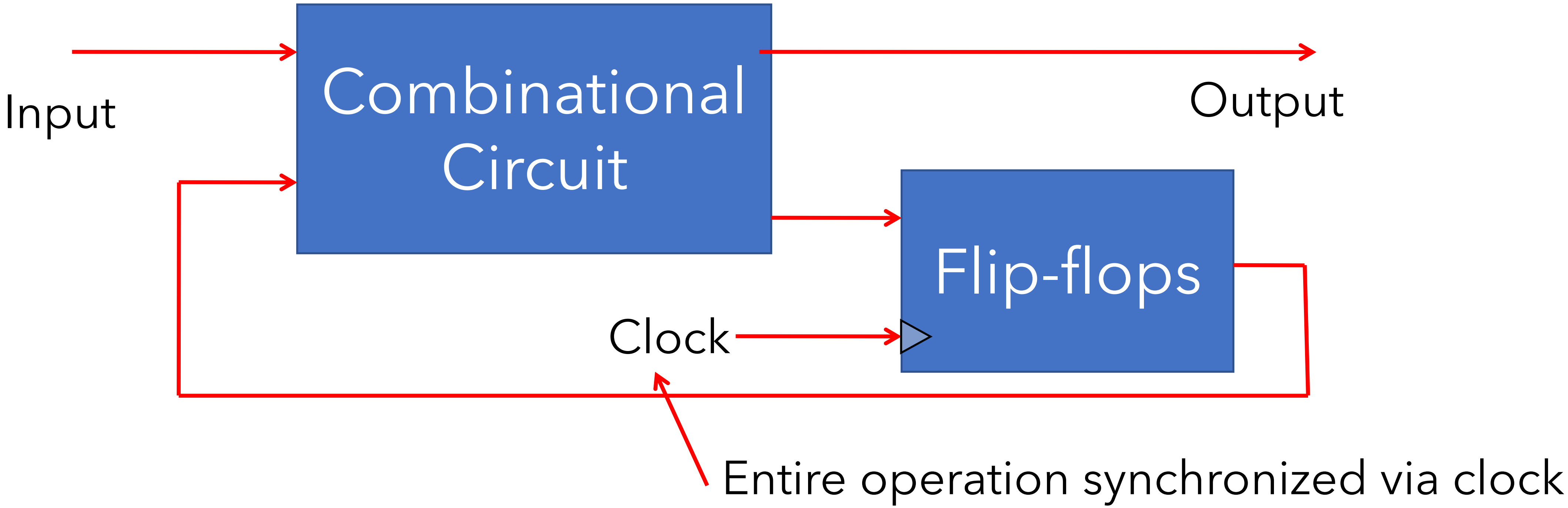
ASSISTANT PROFESSOR,
ELECTRICAL ENGINEERING, IIT KANPUR

Sequential Circuits



- Calculation divided into steps
- Each step is triggered by a clock
- At each step,
  - output is based on the current values of inputs and past values of inputs/outputs.
- Requires memory

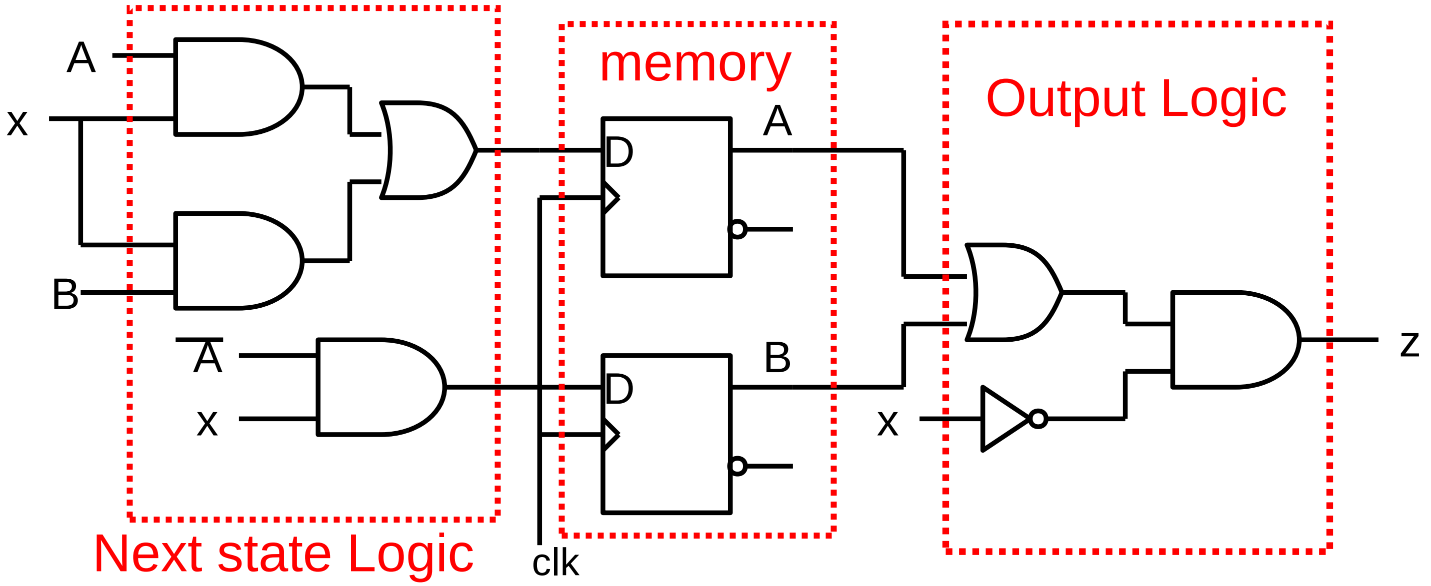
Synchronous Clocked Sequential Circuits



Employs signals that affect the stored value only at discrete instants of time.

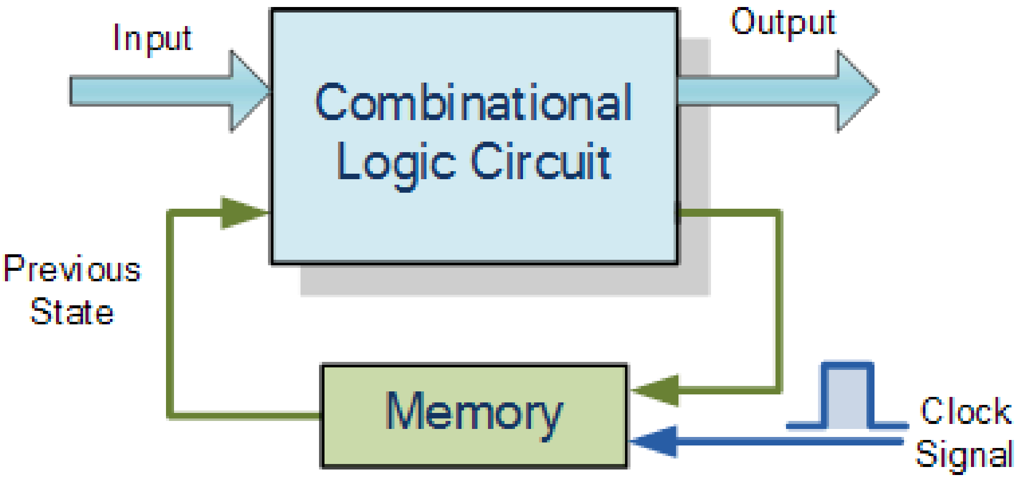
Synchronization is achieved via the ***clock pulses***.

Analyzing sequential circuits



- Output z depends on the input x and on the state of the memory (A,B)
- The memory has 2 FFs and each FF can be in state 0 or 1.
- Thus there are four possible states: AB: 00,01,10,11
- To describe the behavior of a sequential circuit, we need to show
  - How the system goes from one memory state to the next as the input changes
  - How the output responds to input in each state

Sequential Circuits

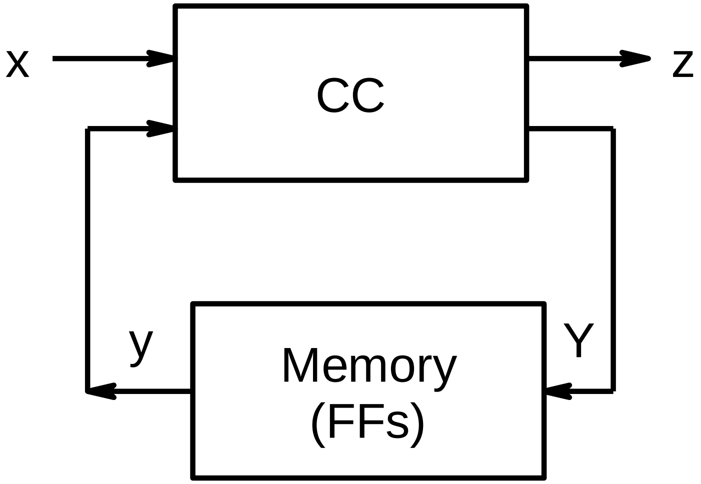


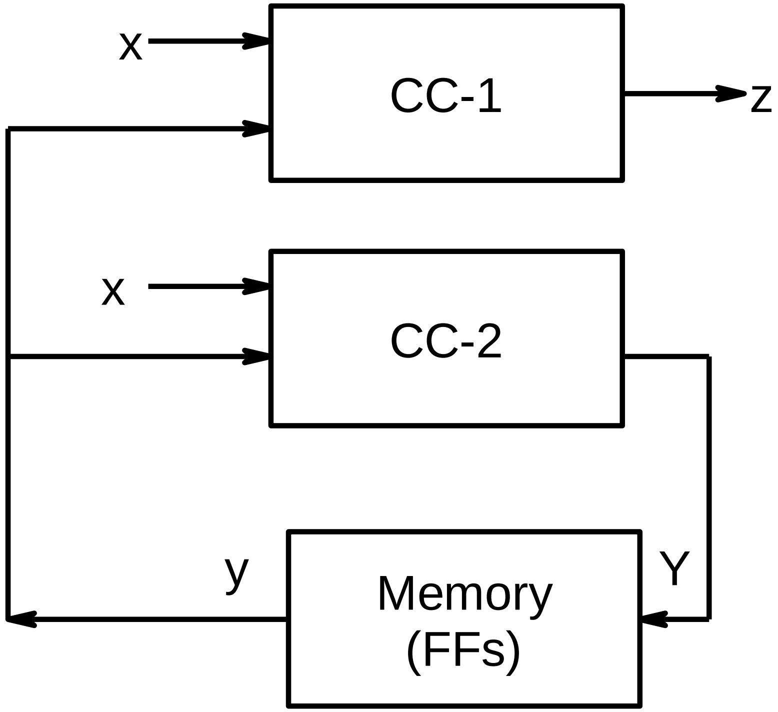
The binary information stored in the storage elements at any given time defines the ***state*** of the sequential circuit at that time

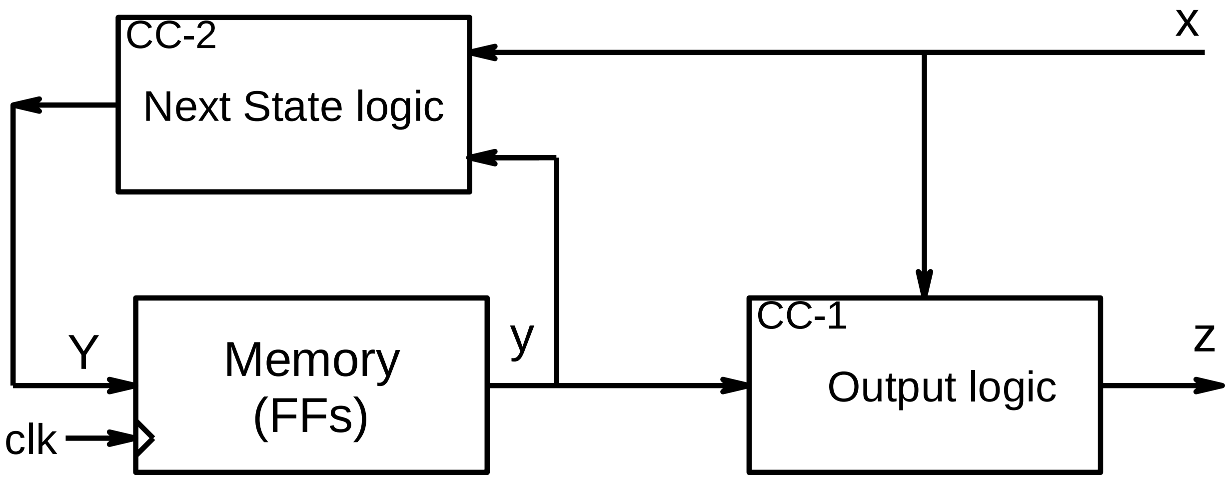
Output is a function of input as well as the present state (the stored value).

Next state is also a function of the present state and inputs.

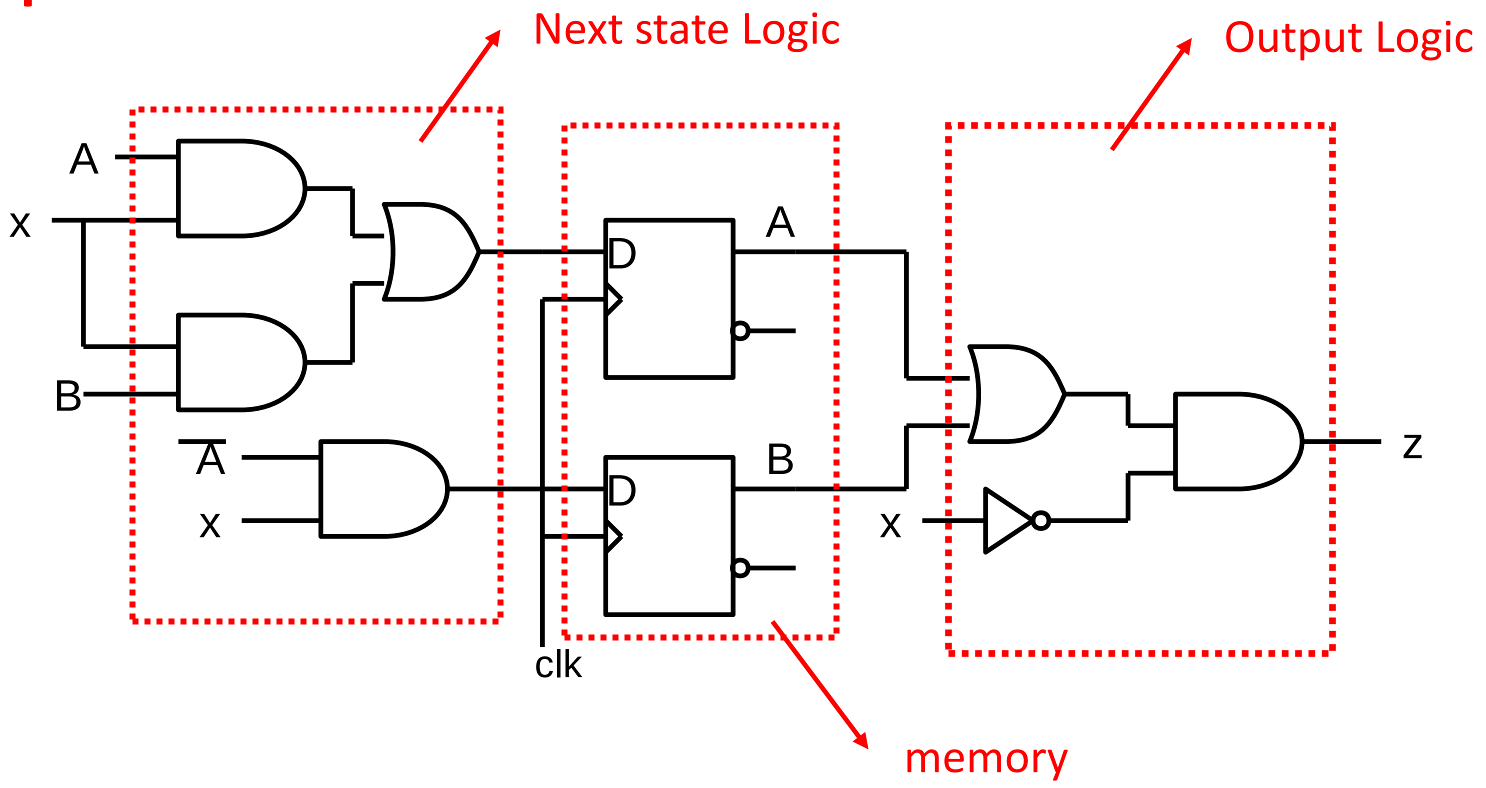
Sequential Circuits



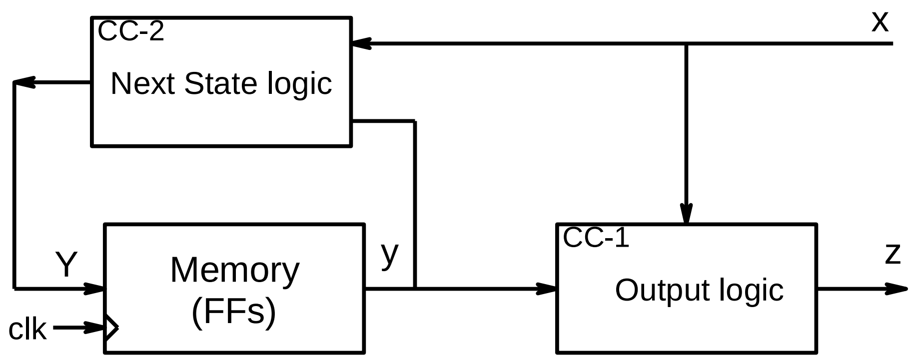




Example

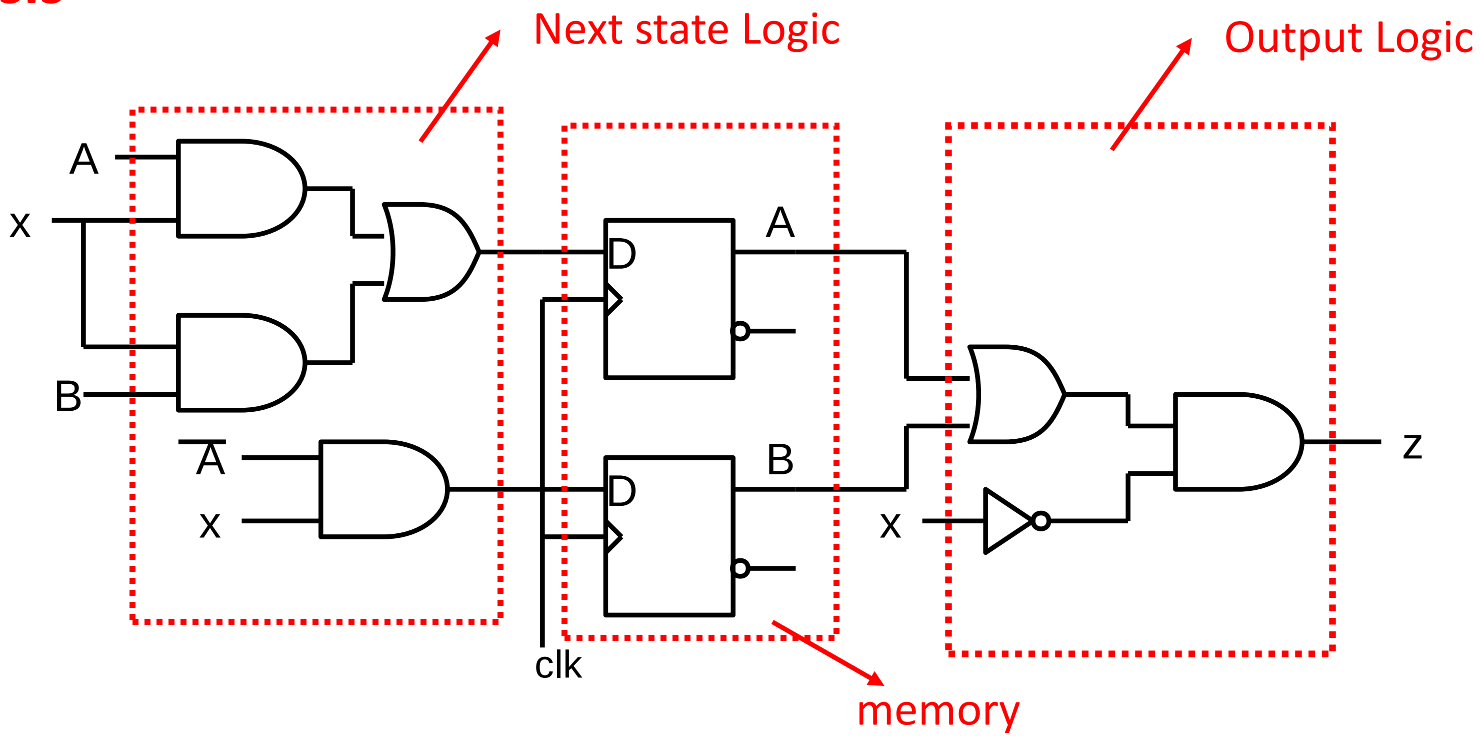


Next state Logic
Output Logic
A
x
B
A
x
D
A
D
B
clk
x
z
memory



CC-2
Next State logic
x
Y
Memory
(FFs)
y
CC-1
Output logic
z
clk

Analysis



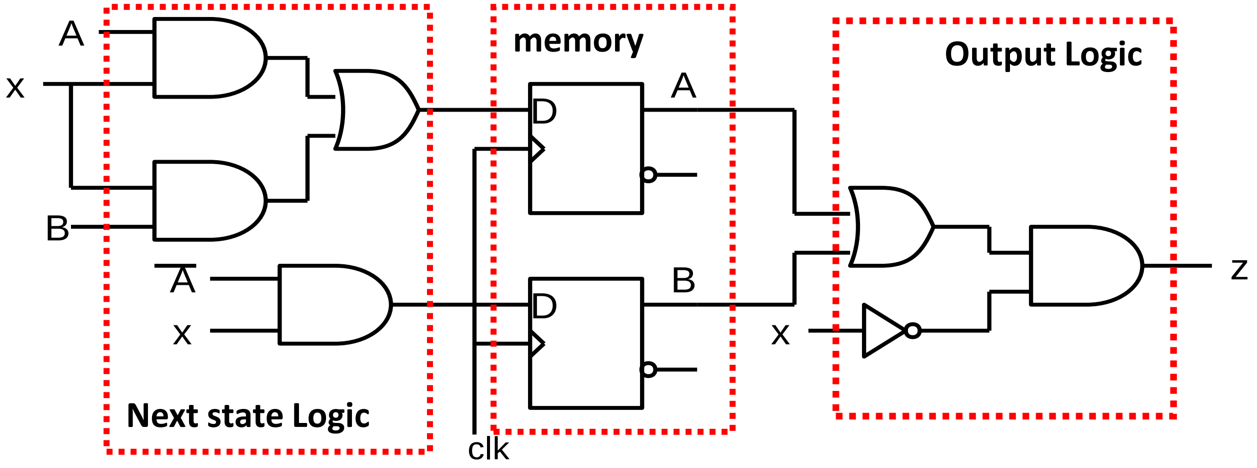
The dependence of output z on input x depends on the state of the memory (A,B)

The memory has 2 FFs and each FF can be in state 0 or 1. Thus there are four possible states: AB: 00,01,10,11.

To describe the behavior of a sequential circuit, we need to show

1. how the system goes from one memory state to the next as the input changes
2. How the output responds to input in each state

Analysis of Sequential Circuits



$$D_A = A.x + B.x \;\; ; D_B = \overline{A}.x ; z = (A+B).\overline{x}$$

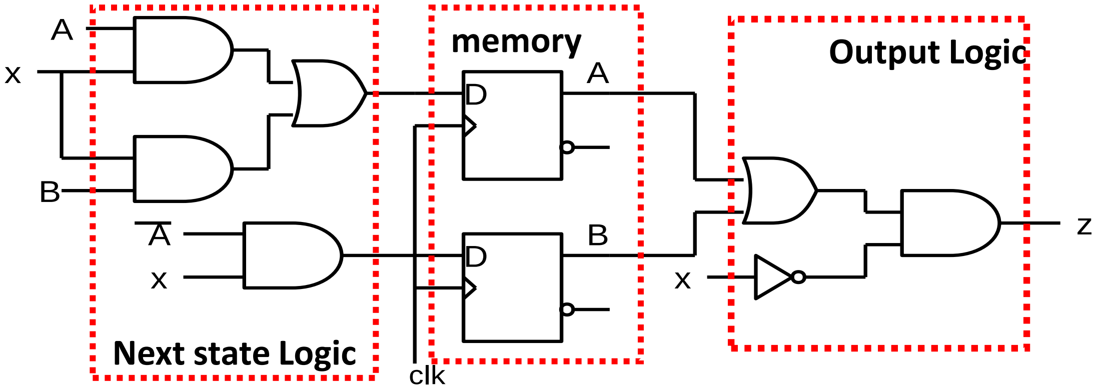
$$A(t+1) = A(t).x + B(t).x$$

$$B(t+1) = \overline{A(t)}.x$$

$$z = (A+B).\overline{x}$$

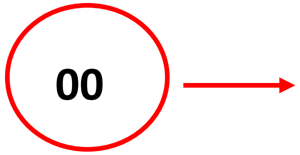
State Transition Table

Present State		Input	Next State		Output
A(t)	B(t)	x(t)	A(t+1)	B(t+1)	z(t)
0	0	0	**0**	**0**	**0**
0	0	1	**0**	**1**	**0**
0	1	0	**0**	**0**	**1**
0	1	1	**1**	**1**	**0**
1	0	0	**0**	**0**	**1**
1	0	1	**1**	**0**	**0**
1	1	0	**0**	**0**	**1**
1	1	1	**1**	**0**	**0**

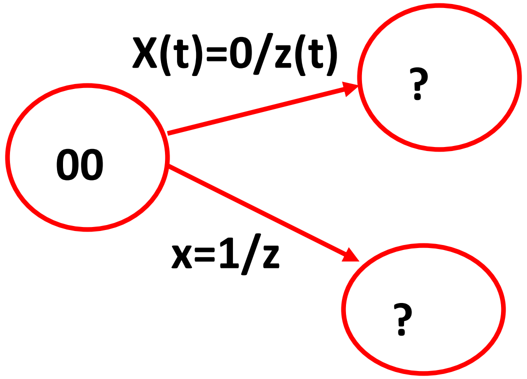


State Transition Table

Present State		Input	Next State		Output
A(t)	B(t)	x(t)	A(t+1	B(t+1)	z(t)
0 0	0 0	0 1	0 0	0 1	0 0
0 0	1 1	0 1	0 1	0 1	1 0
1 1	0 0	0 1	0 1	0 0	1 0
1 1	1 1	0 1	0 1	0 0	1 0

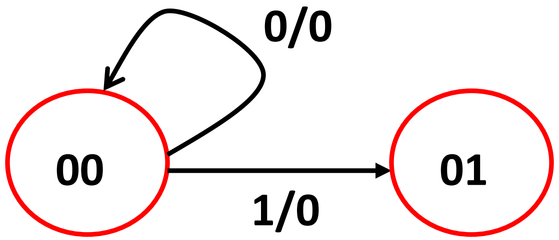


Memory state in which FF A& B have output values 00

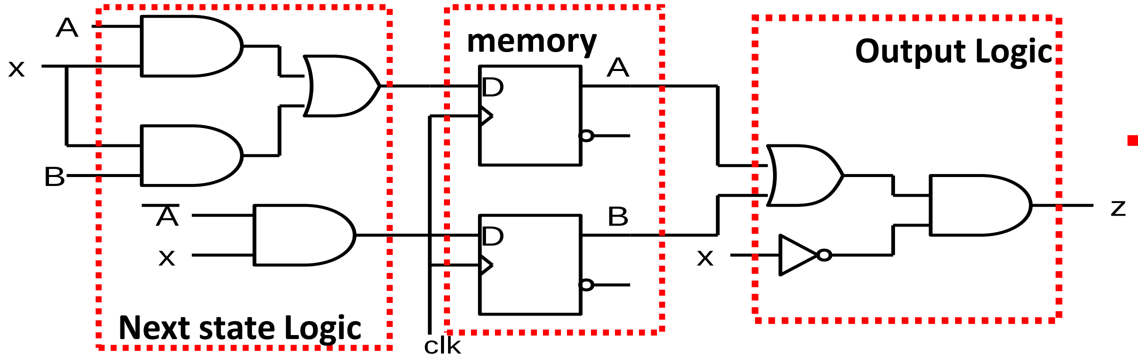


If x = 0 then z = 0, When the clock edge comes the system would stay in 00 state.

If x = 1 then z = 0. When the clock edge comes the system would go to 01 state.



Analysis of Sequential Circuits



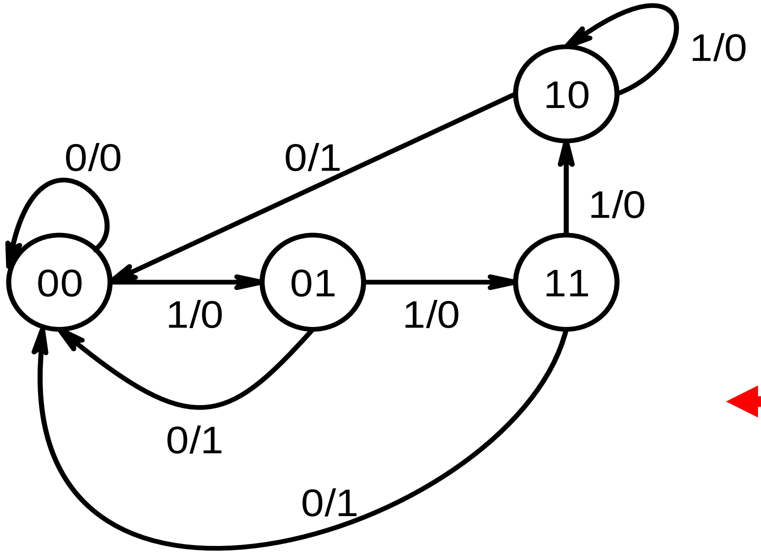
$$A(t+1) = A(t).x + B(t).x$$

$$B(t+1) = \overline{A(t)}.x$$

$$z = (A+B).\overline{x}$$

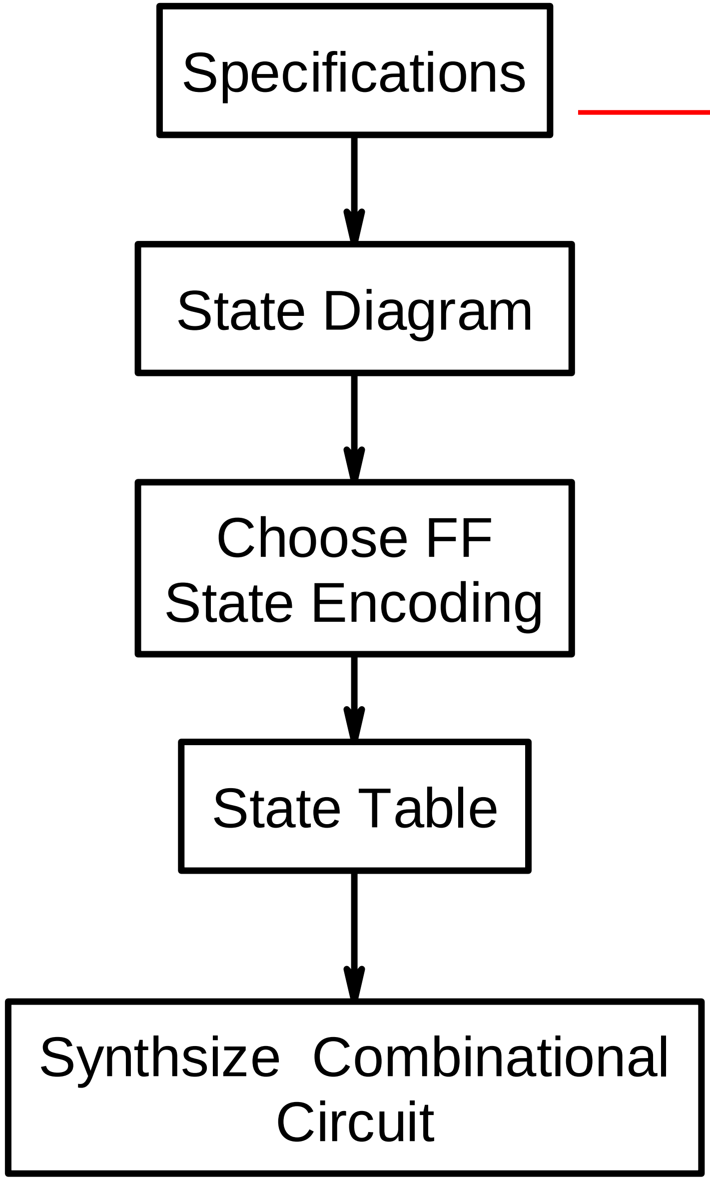
State Transition Table

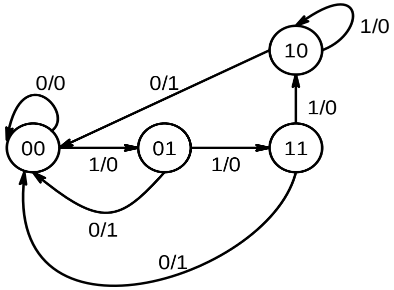
Present State A(t)	Present State B(t)	Input x(t)	Next State A(t+1)	Next State B(t+1)	Output z(t)
0	0	0	0	0	0
0	0	1	0	1	0
0	1	0	0	0	1
0	1	1	1	1	0
1	0	0	0	0	1
1	0	1	1	0	0
1	1	0	0	0	1
1	1	1	1	0	0



State transition Graph

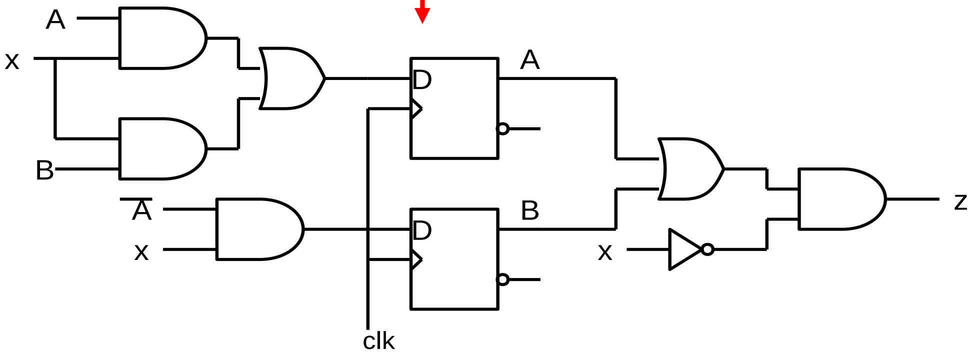
Design of Sequential Circuits



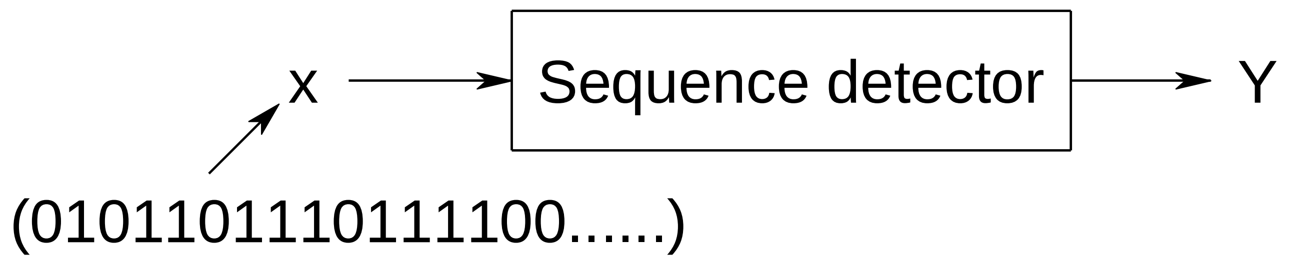


State Transition Table

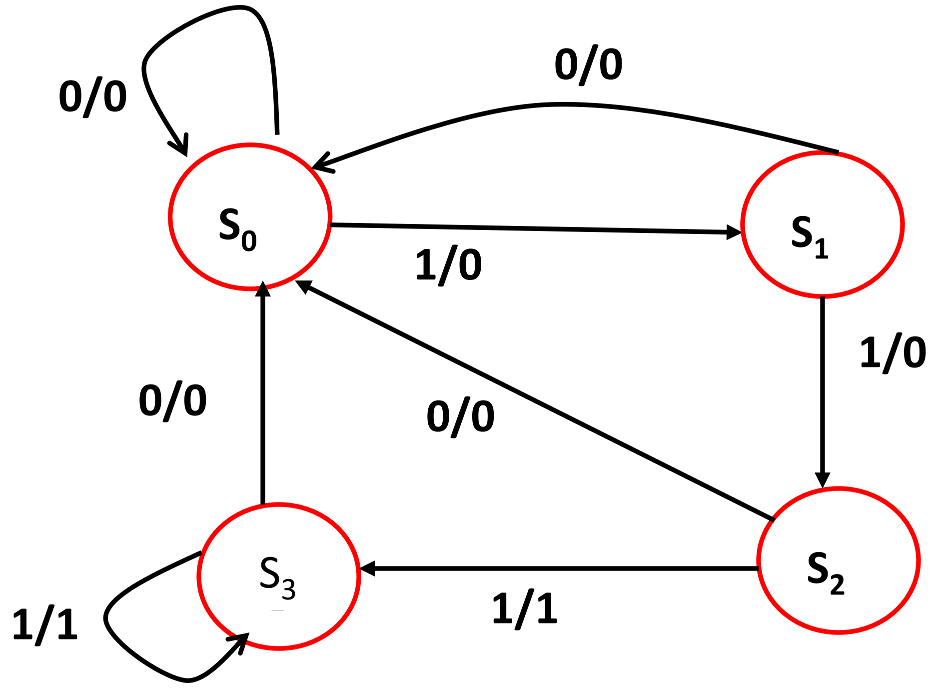
Present State A	Present State B	Input x	Next State A	Next State B	Output z
0	0	0	0	0	0
0	0	1	0	1	0
0	1	0	0	0	1
0	1	1	1	1	0
1	0	0	0	0	1
1	0	1	1	0	0
1	1	0	0	0	1
1	1	1	1	0	0



System specification to State diagram

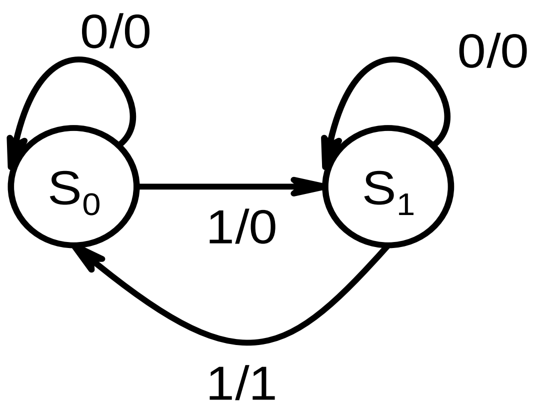


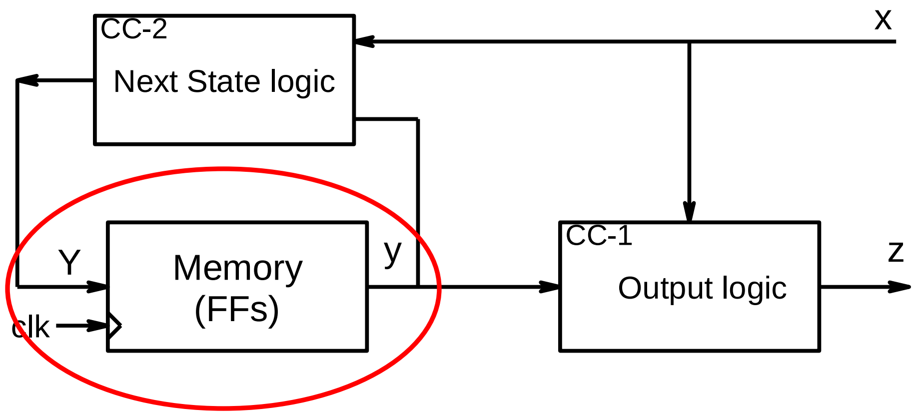
Detect 3 or more consecutive 1’s in the input stream



Conversion of State transition graph to a circuit

Example-1



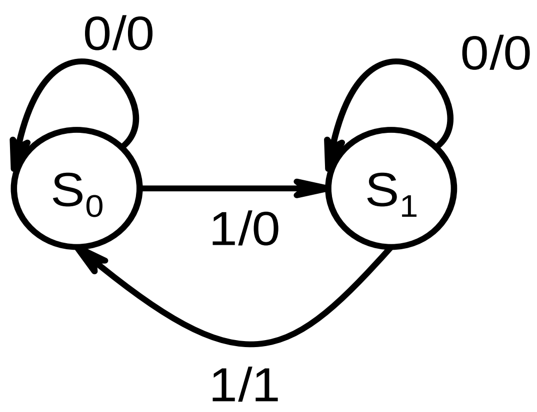


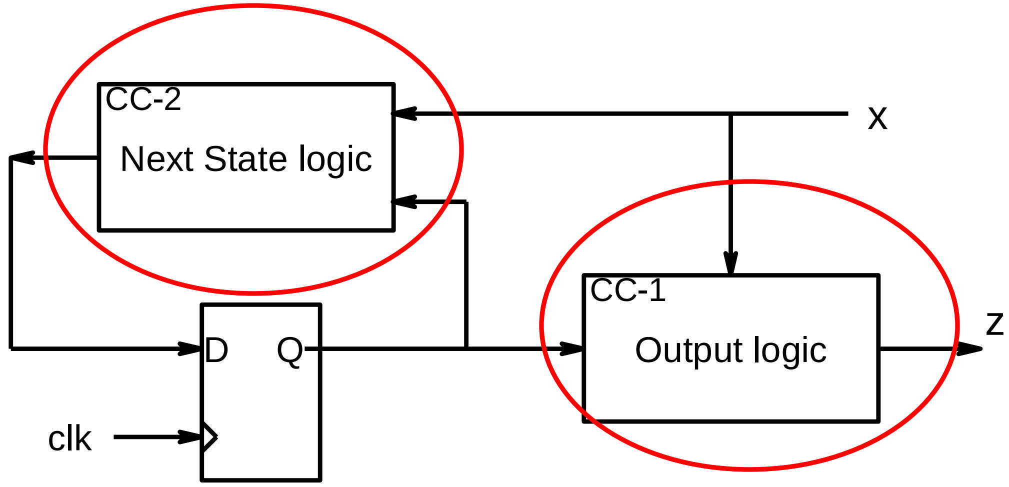
3 blocks need to be designed

1. How many FFs do we need? **N FFS can represent 2^N states so Minimum is 1**

2. Which FF do we choose? **Say D FF**

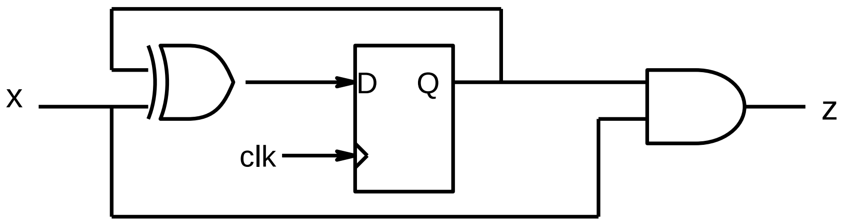
3. How are the states encoded? **Say FF output Q=0 represents S_0 and Q=1 represents S_1 state**





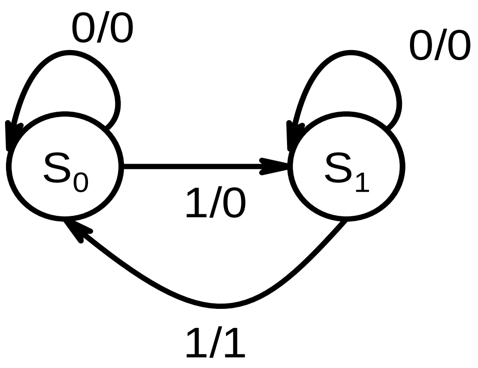
State Transition Table

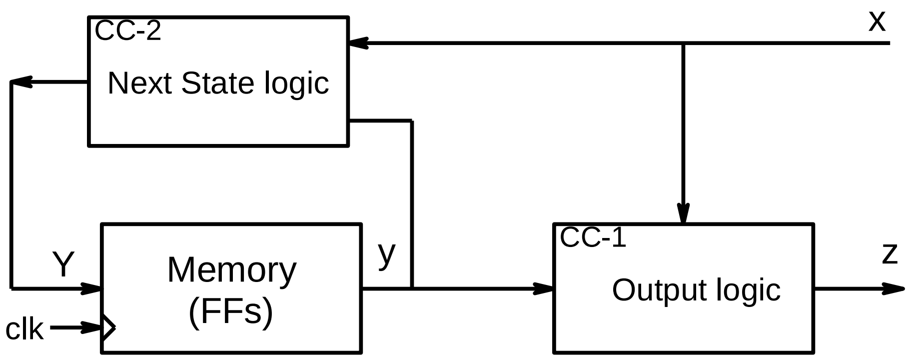
Present State Q(t)	Input x	Next State Q(t+1)	D	Output z
0	0	0	**0**	0
0	1	1	**1**	0
1	0	1	**1**	0
1	1	0	**0**	1



$$D=\overline{Q}.x+Q.\overline{x} \quad ; \quad z=Q.x$$

Example-2

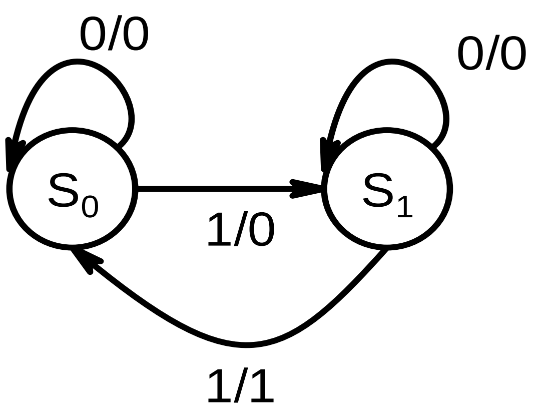


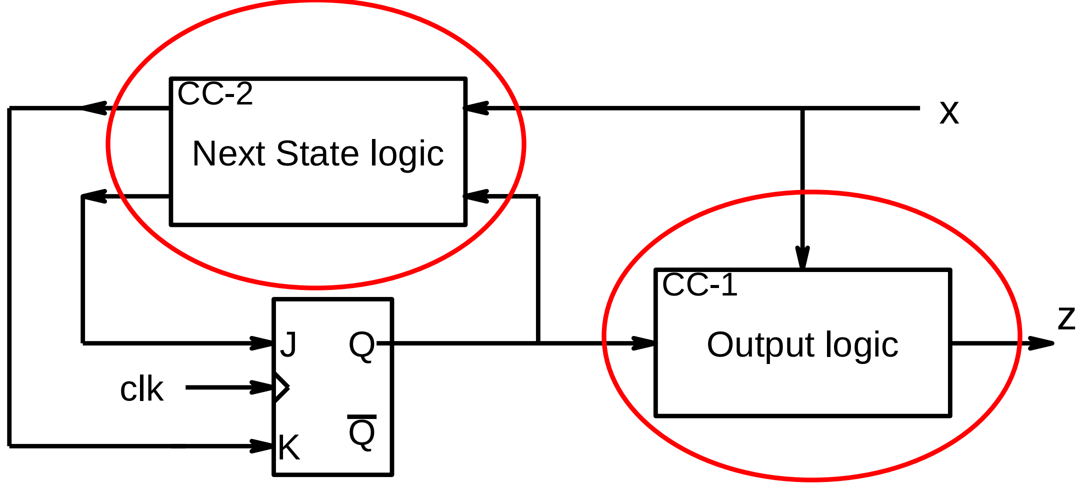


1. How many FFs do we need? **1**

2. Which FF do we choose? **Say JK FF**

3. How are the states encoded? **Say FF output Q=0 represents S_0 and Q=1 represents S_1 state**

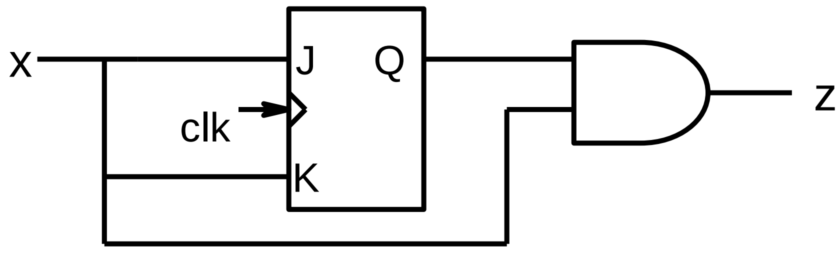




Q(t)	Q(t+1)	J K
0	0	0 X
0	1	1 X
1	0	X 1
1	1	X 0

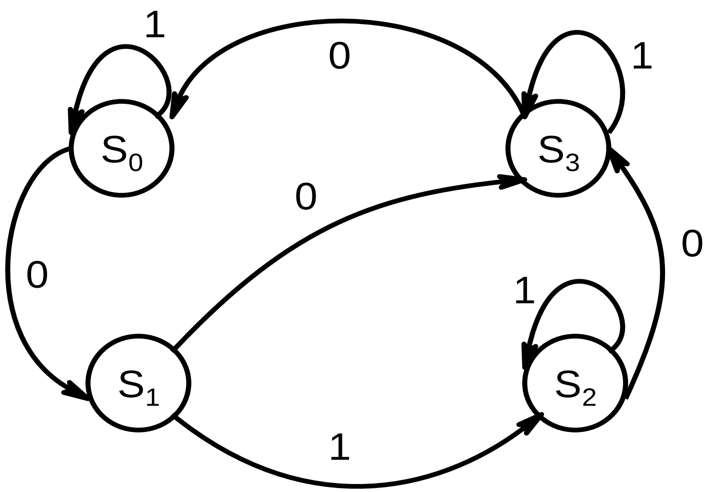
State Transition Table

Present State Q(t)	Input x	Next State Q(t+1)	J K	Output z
0	0	0	**0 X**	0
0	1	1	**1 X**	0
1	0	1	**X 0**	0
1	1	0	**X 1**	1

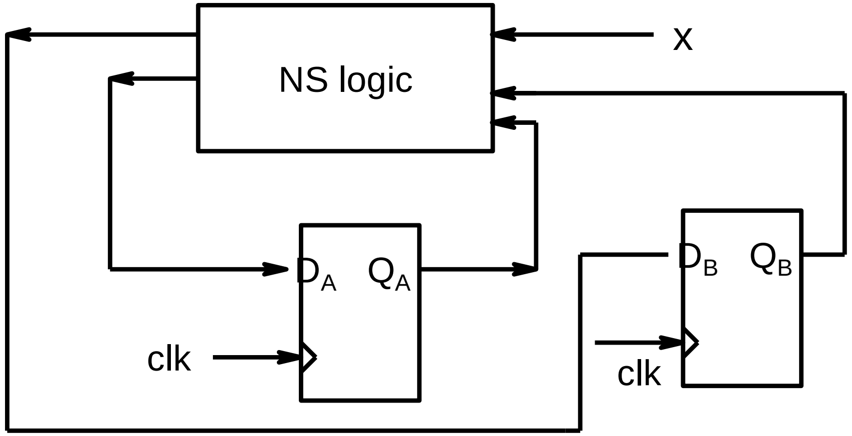


$$J = x \ ; K = x; \ \ z = Q.x$$

Example-3



For 4 states a minimum of two FFs will be required. Let us choose 2 D FFs A &B



State	FF O/P A	B
S_0	0	0
S_1	0	1
S_2	1	0
S_3	1	1

Present State A	B	Input x	Next State A	B	D_A	D_B
0	0	0	0	1	0	1
0	0	1	0	0	0	0
0	1	0	1	1	1	1
0	1	1	1	0	1	0
1	0	0	1	1	1	1
1	0	1	1	0	1	0
1	1	0	0	0	0	0
1	1	1	1	1	1	1

Present State		Input	Next State			
A	B	x	A	B	D_A	D_B
0	0	0	0	1	0	1
0	0	1	0	0	0	0
0	1	0	1	1	1	1
0	1	1	1	0	1	0
1	0	0	1	1	1	1
1	0	1	1	0	1	0
1	1	0	0	0	0	0
1	1	1	1	1	1	1

D_A

x \ AB	00	01	11	10
0	0	1	0	1
1	0	1	1	1

$$D_A = \overline{A}B + xB + A\overline{B}$$
$$= A \oplus B + x.B$$

D_B

x \ AB	00	01	11	10
0	1	1	0	1
1	0	0	1	0

$$D_B = \overline{x}.\overline{A} + \overline{x}.\overline{B} + x.A.B$$
$$= \overline{x}.(\overline{A} + \overline{B}) + x.A.B$$
$$= \overline{x}.\overline{AB} + x.AB = \overline{x \oplus AB}$$

$$D_A = A \oplus B + x.B \qquad D_B = \overline{x \oplus AB}$$