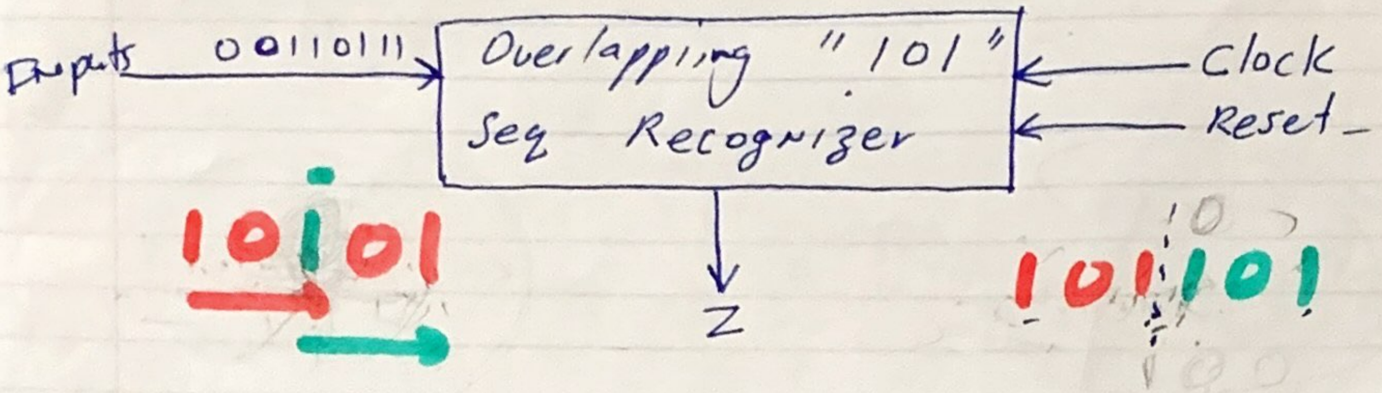
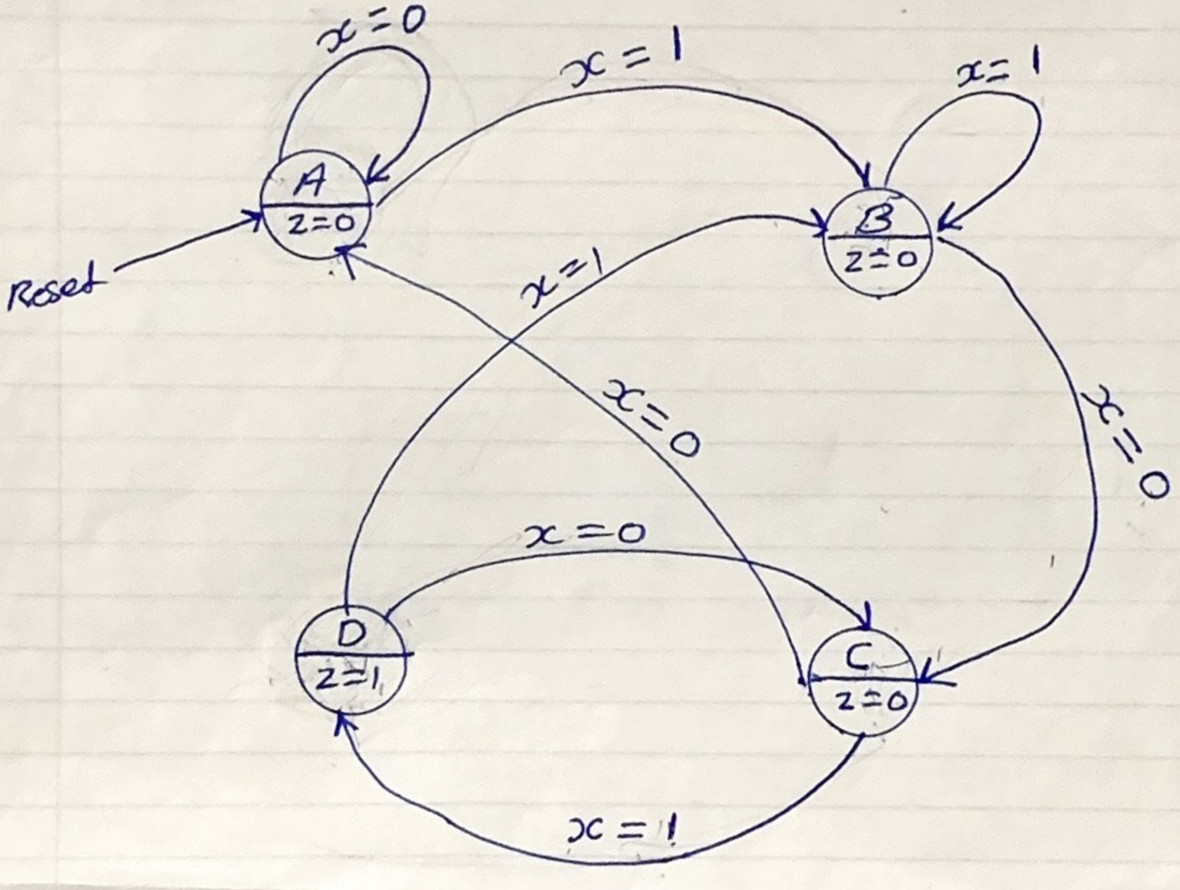
Example : Design of a **Moore** FSM that detects overlapping Seq "101"



Step 1: Create a Moore FSD - Finite State Diagram



Design of Moore FSM that detects Overlapping Sequence "101" - Cont.

Step 2: Determine the minimum number of bits required to store the states

Number of bits $= \text{Log}_2 [K] = \text{Log}_2 [4] = 2$

$K =$ # of states

Step 3: From the FSD, create the Truth table for NSG + OG

A = 00, B = 01
C = 10 D = 11

NSG

		Current state		Input X	Next state		
		q_1	q_0	x	d_1	d_0	
1	A	0	0	0	0	0	A
2		0	0	1	0	1	B
3	B	0	1	0	1	0	C
4		0	1	1	0	1	B
5	C	1	0	0	0	0	A
6		1	0	1	1	1	D
7	D	1	1	0	1	0	C
8		1	1	1	0	1	B

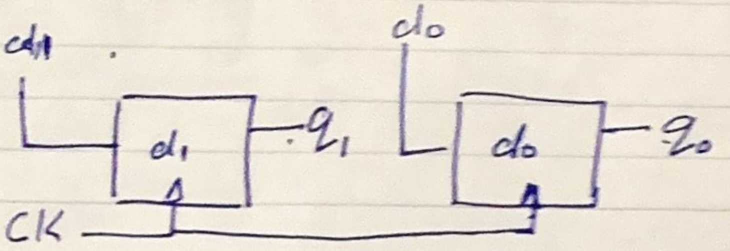
step 3 - Cont

Design of a Moore FSM that detects Overlapping Sequence "101"

Step 3: Create Output Generator (OG)

OG

	Current state			output
	q_1	q_0		Z
1	0	0	A	0
2	0	1	B	0
3	1	0	C	0
4	1	1	D	1



clk= Clock signal

Design of a Moore FSM that detects Overlapping Sequence "101"

Step 4: From the truth table, Determine Min SOP for each of the states-var $d_1\ d_0$, output Z

$$d_1 = \boxed{\bar{x}\,\bar{q}_1 q_0 + x\, q_1 \bar{q}_0 + \bar{x}\, q_1 q_0}$$

$$= \bar{x}\, q_1 q_0 + \bar{x}\, q_1 q_0 + x\, q_1 \bar{q}_0$$

$$= \bar{x}\,\underbrace{(\bar{q}_1 q_0 + q_1 q_0)}_{q_0} + x\, q_1 \bar{q}_0$$

$q_0 = 0$

$q_0 = 1$

$\downarrow$

q_0

Complement Law

$q_0(\bar{q}_1 + q_1)$

1

$q_0 \cdot 1$

q_0

$$= \bar{x}\, q_0 + x\, q_1 \bar{q}_0$$

$$d_1 = \boxed{\bar{x}\, q_0 + x\, q_0 \bar{q}_0}$$

$d_0 = \boxed{x}$; $\boxed{Z = q_1 q_0}$ ✓

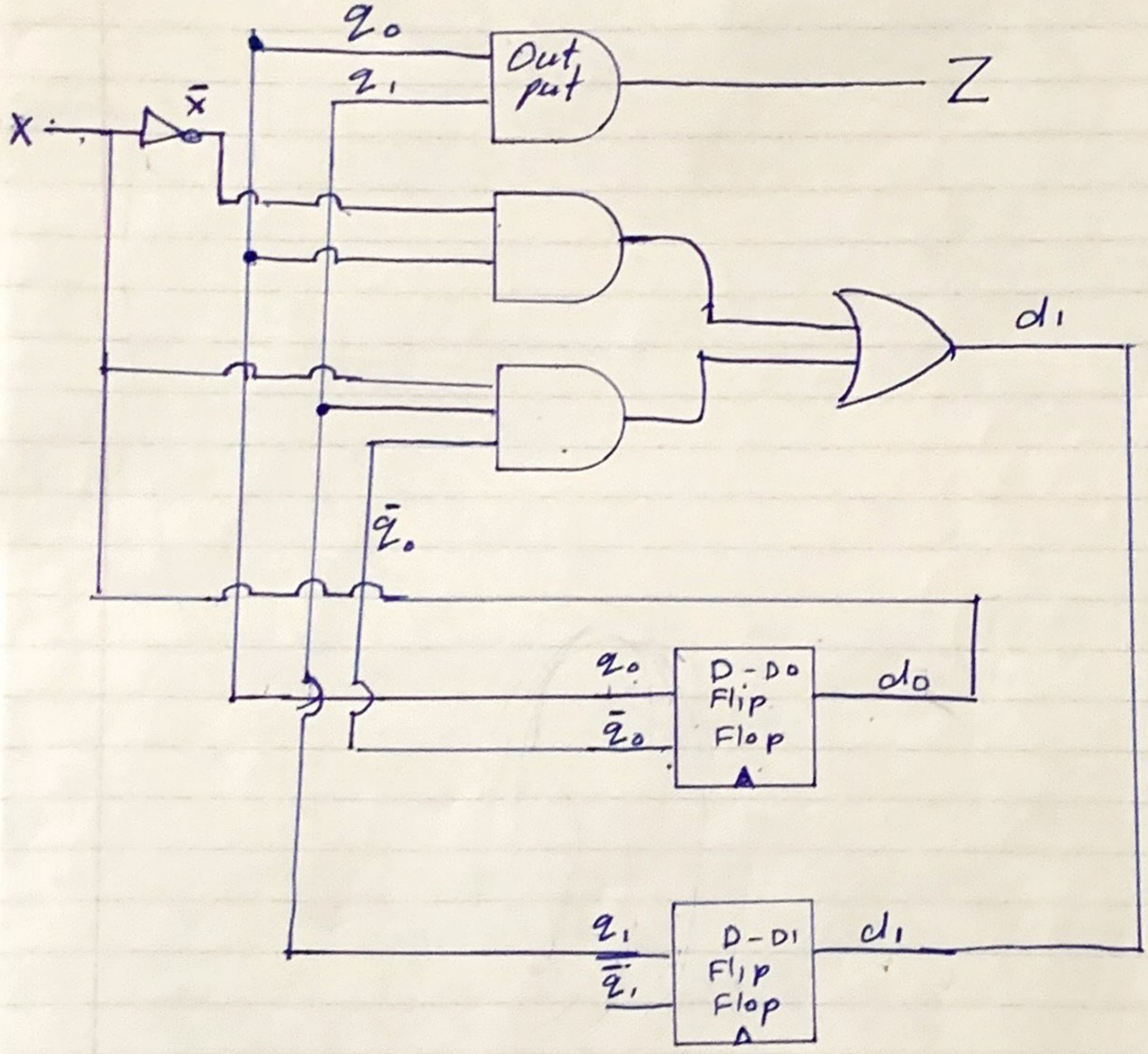
$$\underbrace{\bar{q}_1 q_0 + q_1 q_0} \Leftrightarrow q_0$$

$q_0 = 0 \Rightarrow$ whole expression $= 0$

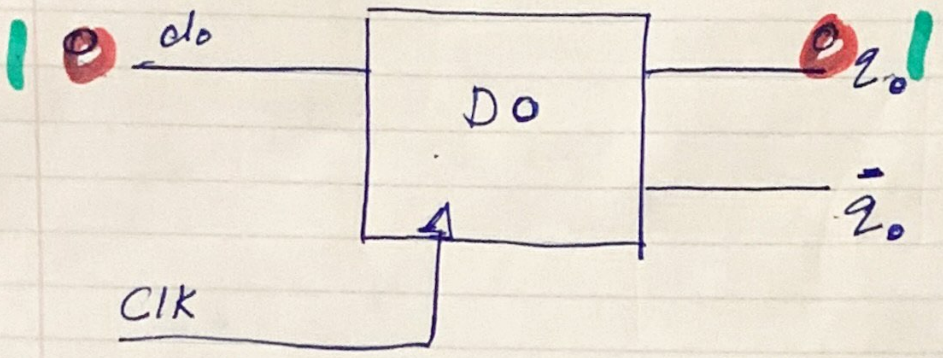
$\bar{q}_1 \cdot 0 + q_1 \cdot 0 =$

Design od a Moore FSM that Detects "101" Sequence

Step 5: Built or design the circuit that detects sequence "101"



D flip Flops



Clk = Clock

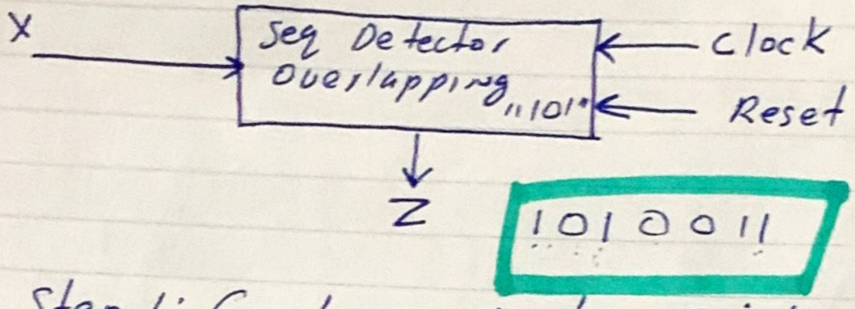
d_0 = Input (1 bit)

q_0 = Output

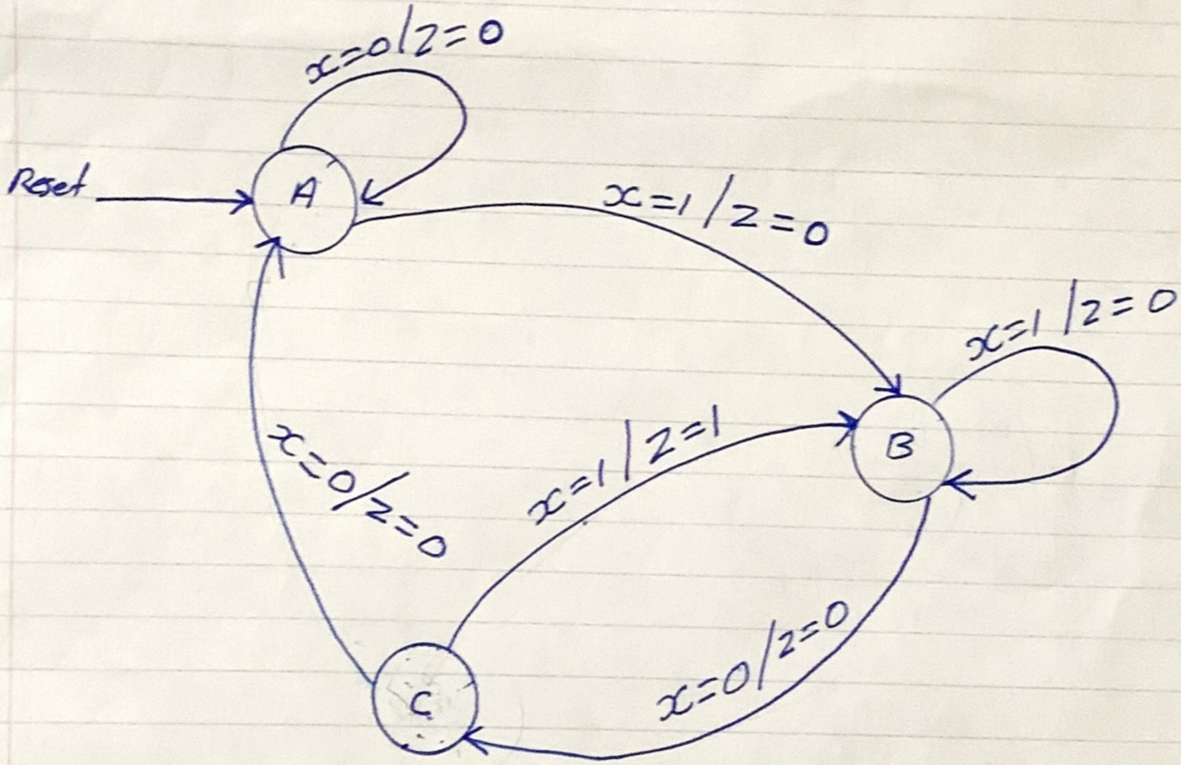
$\bar{q}_0$ = Inverted Output

Design od Mealy FSM - Overlapping Sequence " 101 "

* Mealy output is assigned to the arcs and Not to the states



Step 1: Create a Mealy Finite state Diagram



Design of Mealy FSM - Overlapping Sequence "101"

step 2: Determine the Min number of states/bits required to store the states

Number of bits = $\log_2[K] = \log_2[3] \approx 2$

K = Total # of states

step 3: From FSD, Create the truth table.
Let 00 = A, 01 = B, 10 = C, 11 = D.

NSG/OG

		Current States		Input	Next state			
		q_1	q_0	X	d_1	d_0		Z
1	A	0	0	0	0	0	A	0
2		0	0	1	0	1	B	0
3	B	0	1	0	1	0	C	0
4		0	1	1	0	1	B	0
5	C	1	0	0	0	0	A	0
6		1	0	1	0	1	B	1
7	D	1	1	0	d	d		d
8		1	1	1	d	d		d

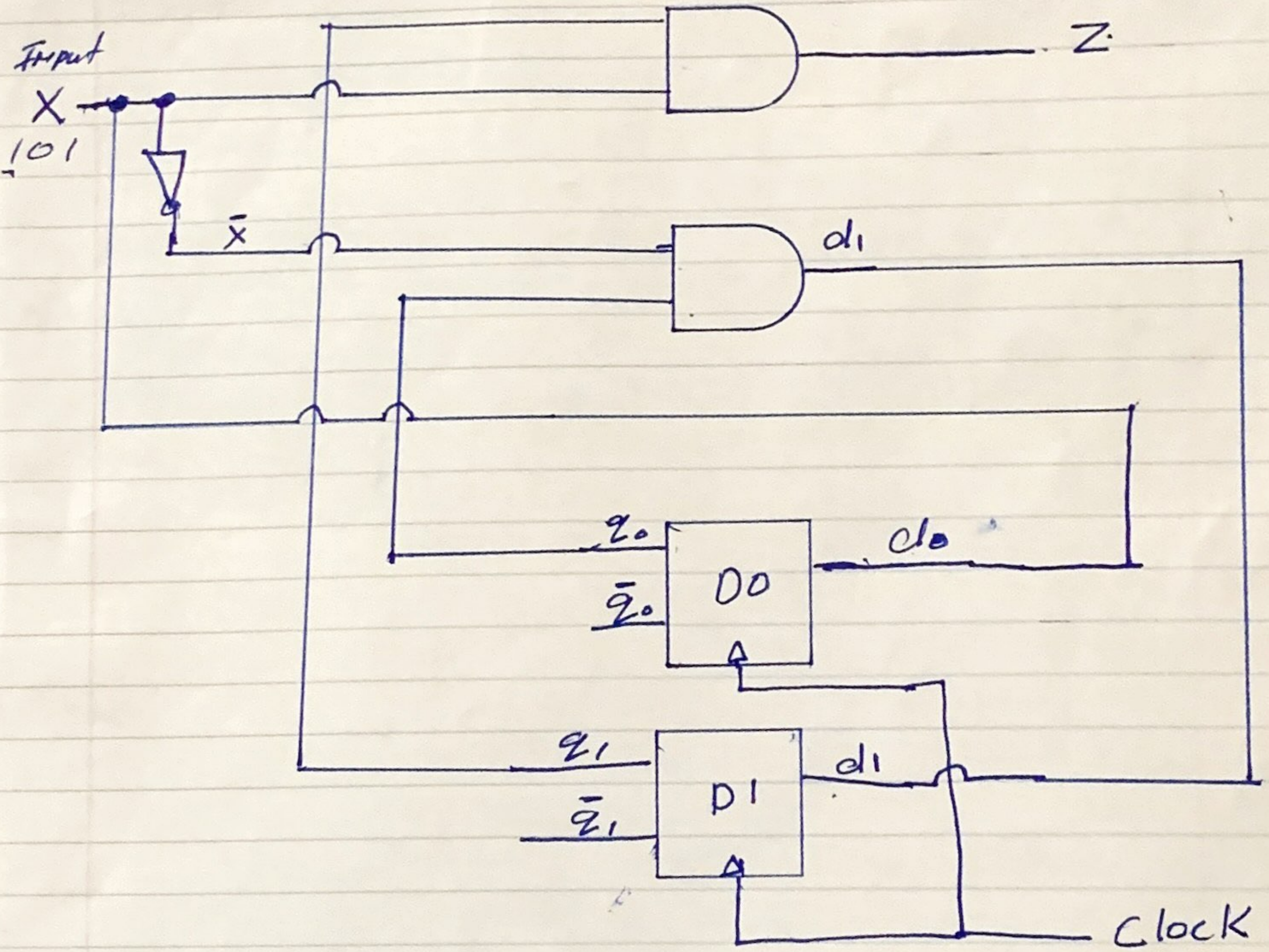
Design of Mealy FSM - Overlapping
Sequence "101"

step 4: Determine the logical Expression

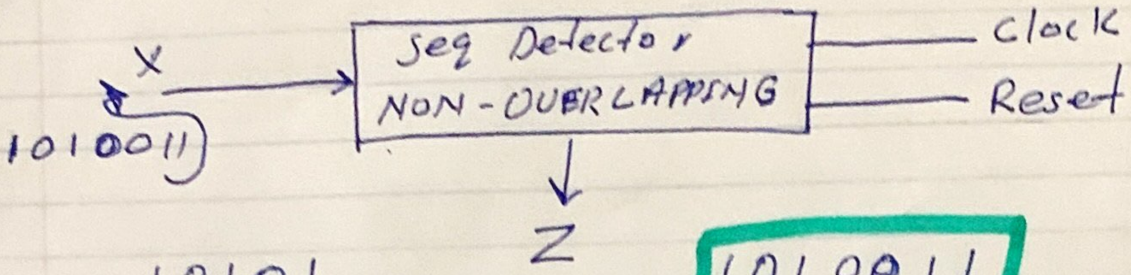
$d_1 = q_0 \bar{x}$; $d_1 = q_0 \bar{q}_1 \bar{x}$

$d_0 = x$ $z = q_1 x$

step 5: Draw the Circuit Diagram



Example: Design of Mealy Finite State machine that detects NON-overlapping Sequence "101"

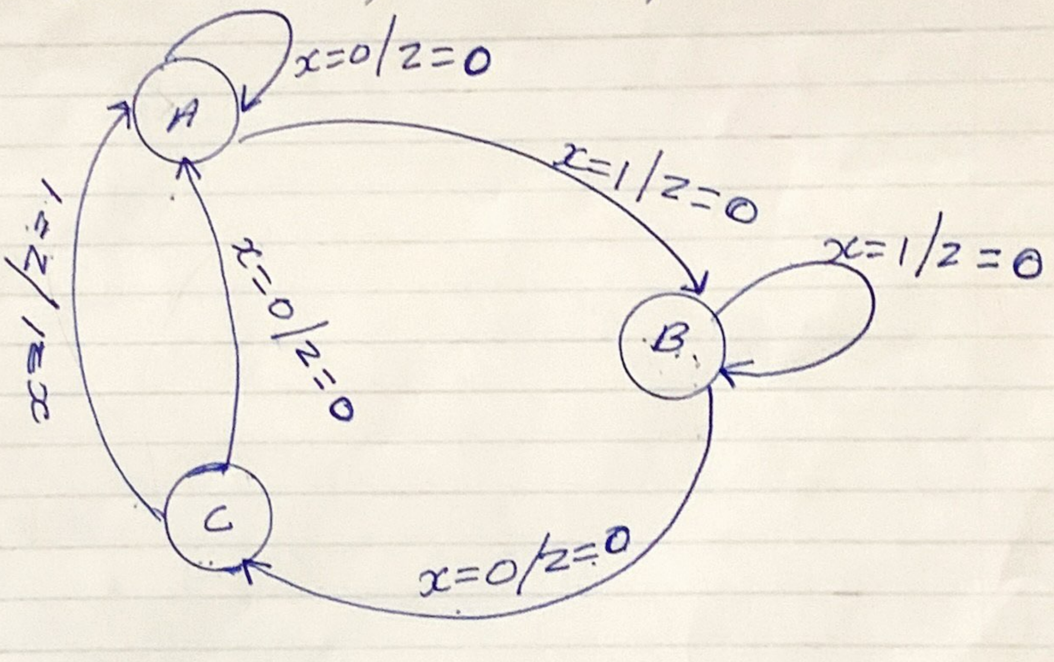


10101

1010011

Step 1: Create a mealy FSD - Finite state Diagram.

Let A = 00, B = 01, C = 10, D = 11



A
B
C
x=0/z=0
x=1/z=0
x=1/z=0
x=0/z=0
x=0/z=0
x=1/z=1