SEQUENTIAL MACHINE DESIGN

INTRODUCTION

- Design problem normally starts with a word description of input output relation and ends with a circuit diagram having sequential and combinatorial logic elements.
- Sequential logic circuit design refers to synchronous clock-triggered circuit because of its design and implementation advantages.
- Design a digital circuit whose outputs are to take on specific values after a specific sequence of inputs has taken place.

SYNCHRONOUS VS ASYNCHRONOUS DESIGN

ASYNCHRONOUS	SYNCHRONOUS
Internal states can change at any instant of time when there is a change in the input variables	Synchronous sequential circuits Synchronized by a periodic train of clock pulses
No clock signal is required	Required
Have better performance but hard to design due to timing problems	Much easier to design (preferred design style)

WHY ASYNCHRONOUS DESIGN

- Used when speed of operation is important
  - Response quickly without waiting for a clock pulse
- Used in small independent systems
  - Only a few components are required
- Used when the input signals may change independently of internal clock
  - Asynchronous in nature
- Used in the communication between two units that have their own independent clocks
  - Must be done in an asynchronous fashion

PROBLEMS WITH ASYNCHRONOUS DESIGN

Problems are:-

- Oscillation of output
- Critical race conditions
- Logical Hazards
- Larger propagation delay
- Glitch problem

- Can cause major problem unless they are addressed at design stage.
- These problems are non-issues in synchronous circuit where external clock trigger arrives after all the inputs are stabilized.
  - Hence, complicated design procedure and complex circuitry

STEPS FOR SYNCHRONOUS DESIGN

- State diagram (Input output relation)
  - Used to illustrate the progression of states through which machine advances when it is clocked
- Next State table
  - List next states that machine advances from its present state when clock is triggered
- State synthesis table (Transition table)
  - Output transitions are listed showing output from present and next and for each corresponding output transition, flipflop inputs that will cause transition to occur are listed

Flip Flop based implementation- excitation tables are used to generate design equations through Karnaugh Map

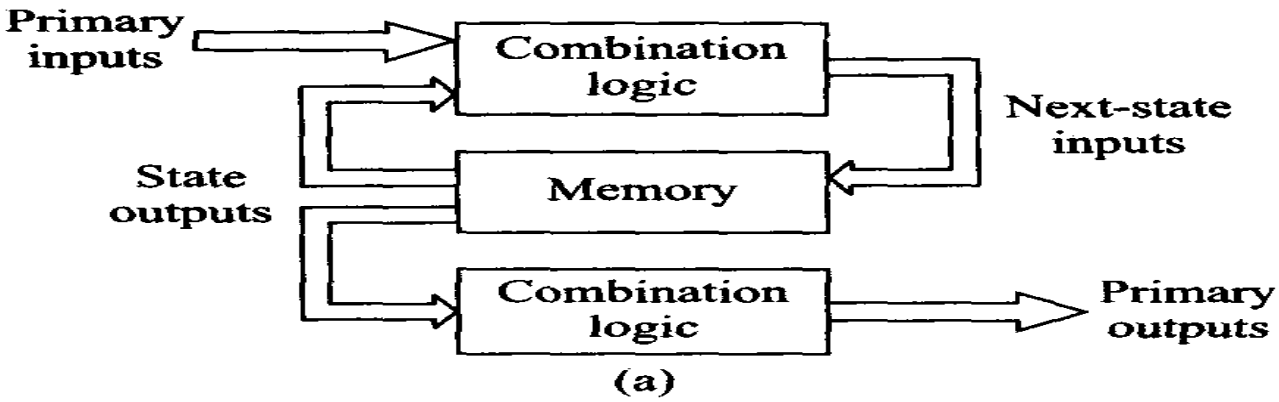
- Circuit diagram is developed from these design equations.

REDUNDANT STATES

- Two states are equivalent if they have the same output and go to the same (equivalent) next states for each possible input
- Removal of redundant states will reduce the amount of logic required and hence reduce the no. of flip-flops required

State machine design

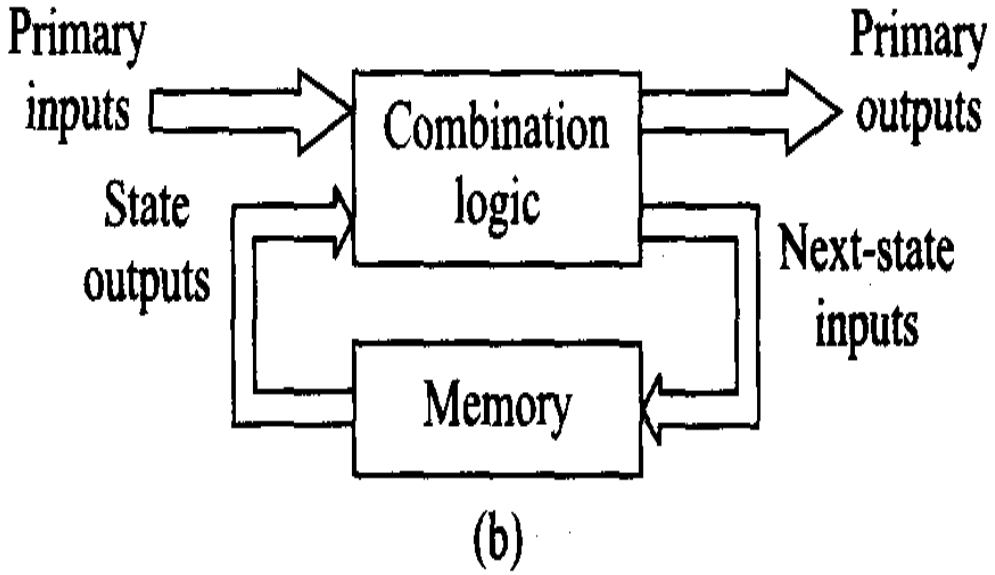
- Two different approaches of state machine design
  - Moore model
  - Mealy model.
- In Moore model circuit outputs, also called primary outputs are generated solely from secondary outputs or memory values.



(a)

State machine design

- In Mealy model circuit inputs, also known as primary inputs combine with memory elements to generate circuit output.



(b)

State machine design

MOORE MODEL	MEALY MODEL
The output depends only on present state and not on input	The output is derived from present state as well as input
It requires more number of states and thereby more hardware to solve any problem	It requires less number of states and thereby less hardware to solve any problem
The output is generated one clock cycle after.	The output is generated one clock cycle earlier, so faster
The output remains stable over entire clock period and changes only when there occurs a state change at clock trigger based on input available at that time.	The glitches occurs

SEQUENCE RECOGNIZER

- A Sequence Recognizer is a special kind of sequential circuit that looks for a special bit pattern in some input
- The recognizer circuit has only one input, X
  - One bit of input is supplied on every clock cycle
- There is one output Z, which is 1 when the desired pattern is found

Our example will detect the bit pattern "1001":

Inputs: X 1 1 0 0 1 1 0 1 0 0 1 0 0 1 1 0 ...

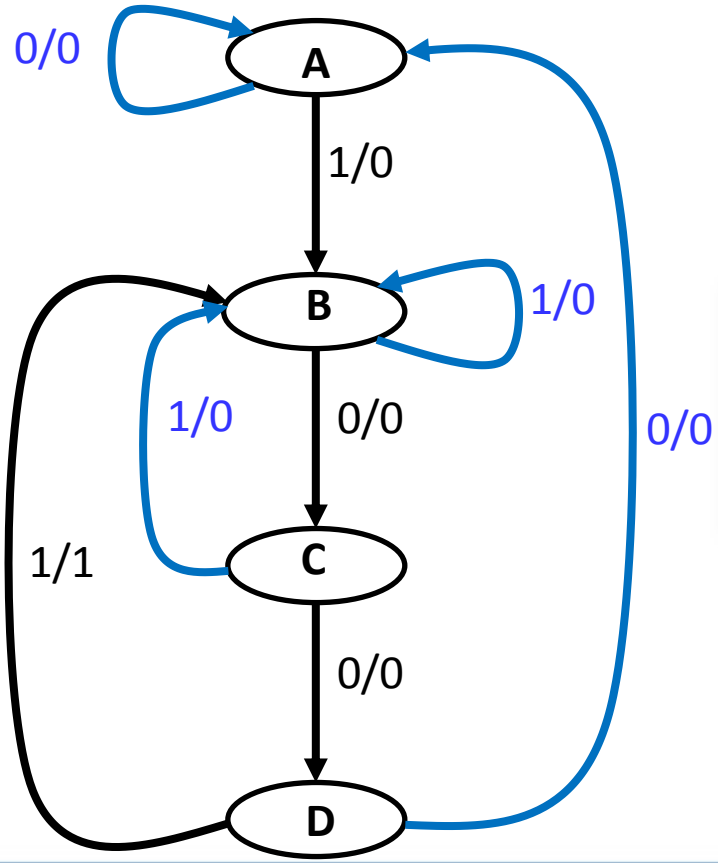
Outputs: Z 0 0 0 0 1 0 0 0 0 0 1 0 0 1 0 0 ...

- A sequential circuit is required because the circuit has to "remember" the inputs from previous clock cycles, in order to determine whether or not a match was found

SEQUENCE RECOGNIZER
MEALY MODEL

Step: 1 State diagram

X= 1001
Z= 0001



A ← 0/0
B ← 1/0
C ← 0/0
D ← 0/0

*Note

present state

input/ output

next state

SEQUENCE RECOGNIZER

Step: 2 Assigning binary codes to States

- We have four states ABCD, so we need at least two flip-flops Q1Q0.
- The easiest thing to do is represent state A with Q1Q0 = 00, B with 01, C with 10, and D with 11.

Present state	Next state		Output	
	X=0	X=1	X=0	X=1
A	A	B	0	0
B	C	B	0	0
C	D	B	0	0
D	A	B	0	1

Present state		Next state			
Q_1	Q_0	X=0		X=1	
		Q_1	Q_0	Q_1	Q_0
0	0	0	0	0	1
0	1	1	0	0	1
1	0	1	1	0	1
1	1	0	0	0	1

SEQUENCE RECOGNIZER

Step: 3 Finding flip-flop input values

Transition table of JK flip flop

Q_P	Q_N	J	K
0	0	0	X
0	1	1	X
1	0	X	1
1	1	X	0

SEQUENCE RECOGNIZER

Step: 4 Find equations for the FF inputs and output

For J_0 and K_0
When X=0

Q_{P0}	Q_{N0}	J_0	K_0
0	0	0	X
1	0	X	1
0	1	1	X
1	0	X	1

When X=1

Q_{P0}	Q_{N0}	J_0	K_0
0	1	1	X
1	1	X	0
0	1	1	X
1	1	X	0

X \ Q_1Q_0	00	01	11	10
0	0	X	X	1
1	1	X	X	1

$J_0 = X+Q_1$

X \ Q_1Q_0	00	01	11	10
0	X	1	1	X
1	X	0	0	X

$K_0 = X'$

SEQUENCE RECOGNIZER

Step: 4 Find equations for the FF inputs and output

For J_1 and K_1
When X=0

Q_{P1}	Q_{N1}	J_1	K_1
0	0	0	X
0	1	1	X
1	1	X	0
1	0	X	1

When X=1

Q_{P1}	Q_{N1}	J_1	K_1
0	0	0	X
0	0	0	X
1	0	X	1
1	0	X	1

X \ Q_1Q_0	00	01	11	10
0	**0**	**1**	**X**	**X**
1	0	0	X	X

$J_1 = X'Q_0$

X \ Q_1Q_0	00	01	11	10
0	**X**	**X**	**1**	**0**
1	X	X	1	1

$K_1 = X+Q_0$

SEQUENCE RECOGNIZER

Step: 5 Build the circuit

For Output Z

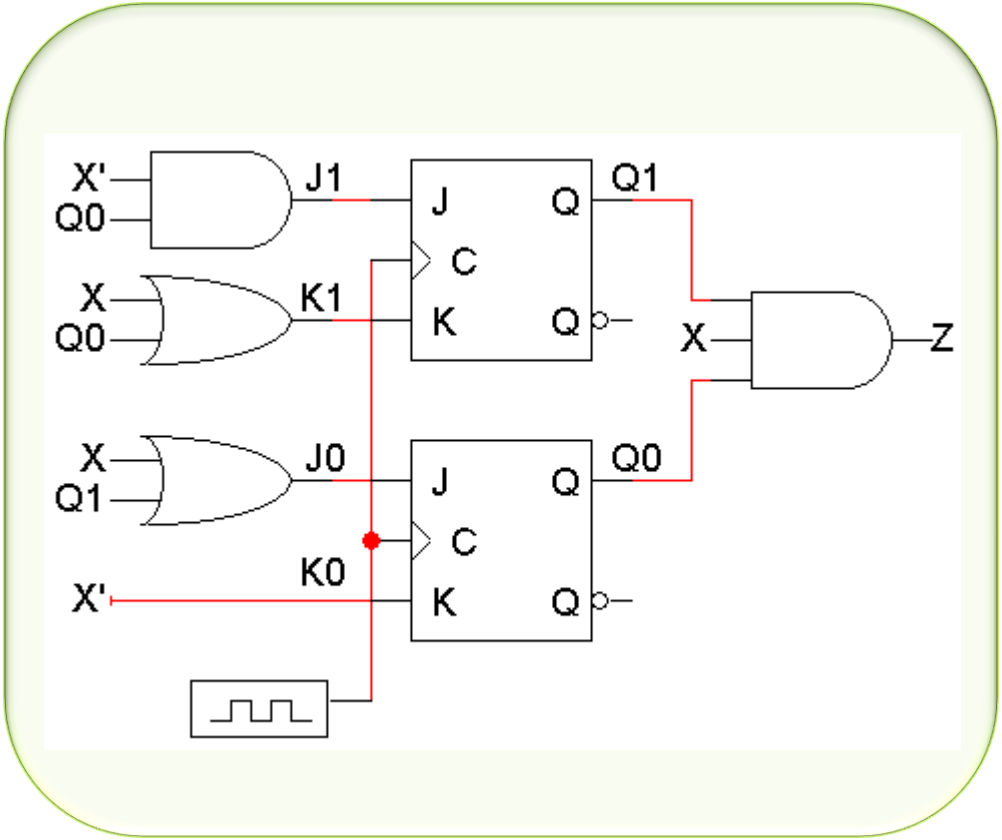
K-map

X \ Q_1Q_0	00	01	11	10
0	0	0	0	0
1	0	0	1	0

$Z = XQ_1Q_0$

$J_1 = X' Q_0$
$K_1 = X + Q_0$

$J_0 = X + Q_1$
$K_0 = X'$



SEQUENCE RECOGNIZER

Building the same circuit with D flip-flops

- We already have the state table and state assignments, so we can just start from Step 3, finding the flip-flop input values

Step: 3 Finding flip-flop input values

Transition table of D flip flop

Q_P	Q_N	D
0	0	0
0	1	1
1	0	0
1	1	1

SEQUENCE RECOGNIZER

Step: 4 Find equations for the FF inputs and output

For D_0
When X=0

Q_{P0}	Q_{N0}	D_0
0	0	0
1	0	0
0	1	1
1	0	0

When X=1

Q_{P0}	Q_{N0}	D_0
0	1	1
1	1	1
0	1	1
1	1	1

X \ Q_1Q_0	00	01	11	10
0	0	0	0	1
1	1	1	1	1

$D_0 = X+Q_1Q_0'$

SEQUENCE RECOGNIZER

Step: 4 Find equations for the FF inputs and output

For D_1
When X=0

Q_{P0}	Q_{N0}	D_0
0	0	0
0	1	1
1	1	1
1	0	0

When X=1

Q_{P0}	Q_{N0}	D_0
0	0	0
0	0	0
1	0	0
1	0	0

X \ Q_1Q_0	00	01	11	10
0	**0**	**1**	**0**	**1**
1	0	0	0	0

$D_1 = X'Q_1'Q_0 + X'Q_1Q_0'$

SEQUENCE RECOGNIZER

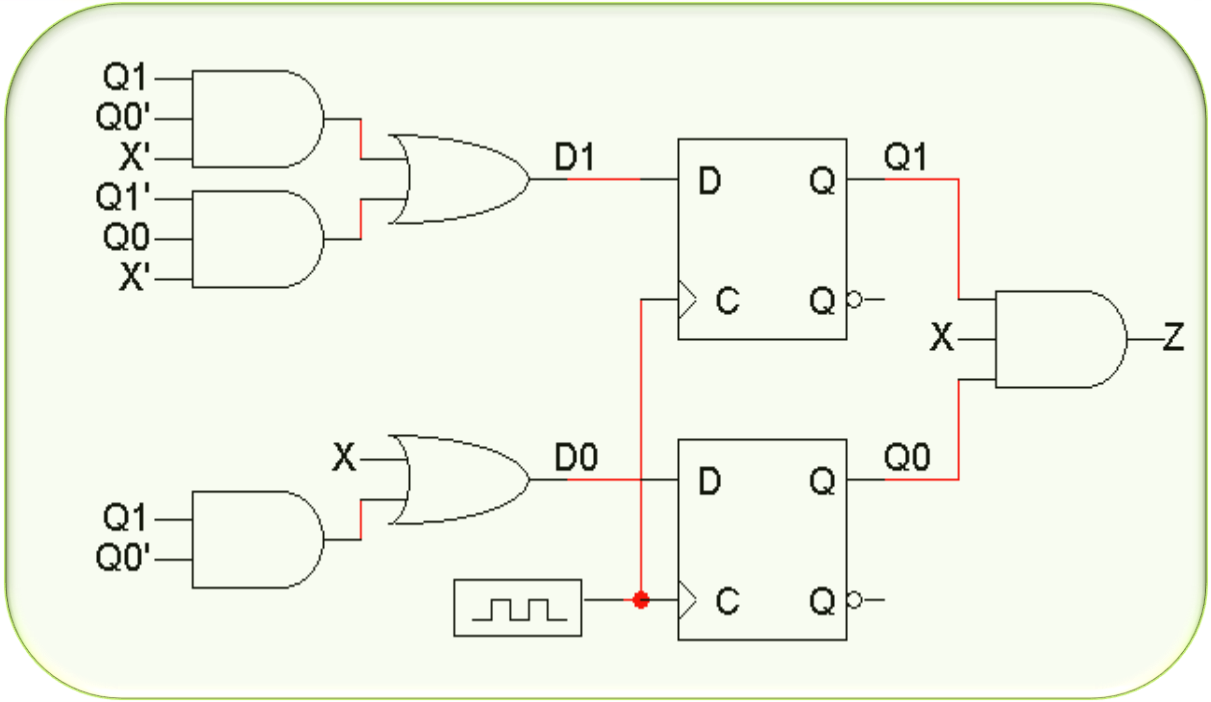
Step: 5 Build the circuit

For Output Z

K-map

X \ Q_1Q_0	00	01	11	10
0	0	0	0	0
1	0	0	1	0

$Z = XQ_1Q_0$



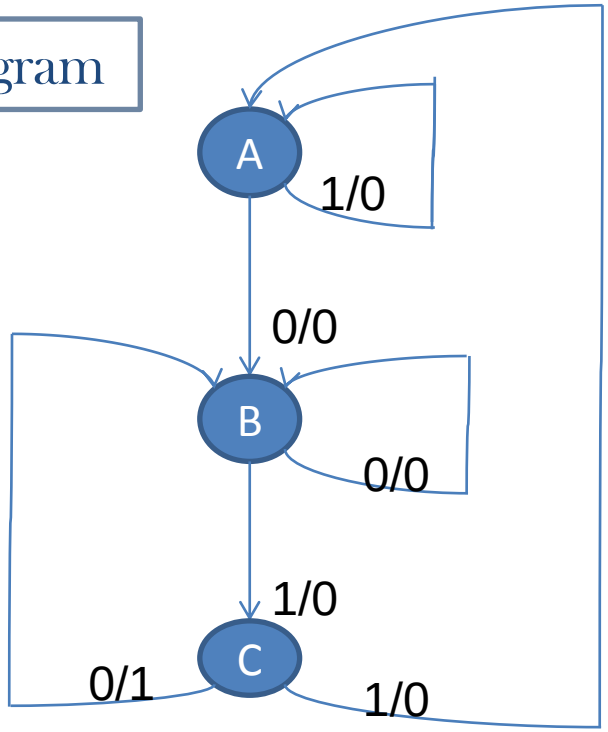
$D_1 = Q_1\ Q_0'\ X' + Q_1'\ Q_0\ X'$

$D_0 = X + Q_1\ Q_0'$

SEQUENCE RECOGNIZER

Design a circuit of sequential machine using T-flipflop, whose output is 1 whenever pattern is 010

Step: 1 State diagram



A ⟵ 1/0
B ⟵ 0/0
C ⟵ 1/0

SEQUENCE RECOGNIZER

Step: 2 Assigning binary codes to States

Present	Next		Output	
	X=0	X=1	X=0	X=1
A	B	A	0	0
B	B	C	0	0
C	B	A	1	0

Next State Table

Put A=00
B=01
C=10

Present		Next			
		X=0		X=1	
Q_1	Q_0	Q_1	Q_0	Q_1	Q_0
0	0	0	1	0	0
0	1	0	1	1	0
1	0	0	1	0	0

SEQUENCE RECOGNIZER

Step: 3 Finding flip-flop input values

Transition table of T flip flop

Q_P	Q_N	T
0	0	0
0	1	1
1	0	1
1	1	0

SEQUENCE RECOGNIZER

Step: 4 Find equations for the FF inputs and output

X=0			X=1		
QP_0	QN_0	T_0	QP_0	QN_0	T_0
0	1	1	0	0	0
1	1	0	1	0	1
0	1	1	0	0	0

K-map for T_0

Q_1Q_0 / x	00	01	11	10
0	1	0	x	1
1	0	1	X	0

$T_0=X'Q_0'+XQ_0$

SEQUENCE RECOGNIZER

Step: 4 Find equations for the FF inputs and output

X=0			X=1		
QP_1	QN_1	T_1	QP_1	QN_1	T_1
0	0	0	0	0	0
0	0	0	0	1	1
1	0	1	1	0	1

K-map for T_1

x \ Q_1Q_0	00	01	11	10
0	0	0	x	1
1	0	1	X	1

$T_1=Q_1+XQ_0$

SEQUENCE RECOGNIZER

Step: 5 Build the circuit

For Output Z

K-map

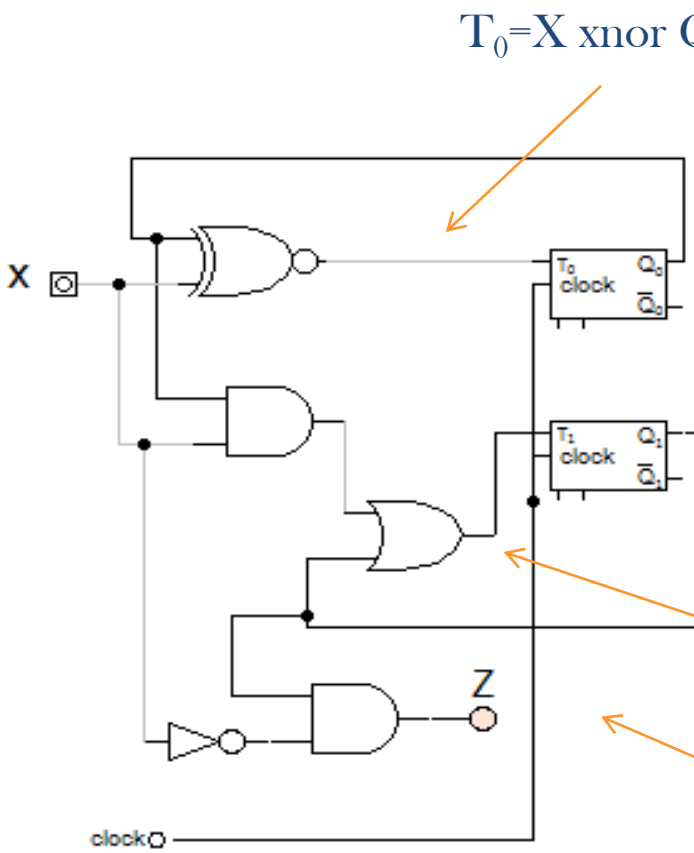
X \ Q_1Q_0	00	01	11	10
0	0	0	x	1
1	0	0	x	0

$Z = X'Q_1$

$T_0 = X'Q_0' + XQ_0$

$T_0 = X \text{ xnor } Q_0$

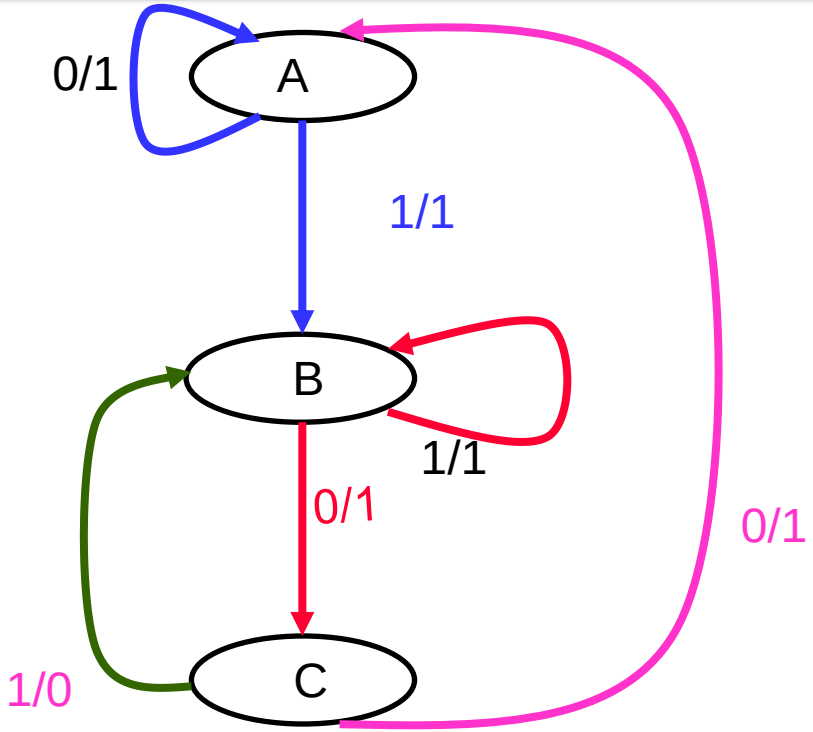
$T_1 = Q_1 + XQ_0$



SEQUENCE RECOGNIZER

Design a circuit of sequential machine using T-flipflop, whose output is 0 whenever pattern is 101

Step: 1 State diagram



SEQUENCE RECOGNIZER

Step: 2 Assigning binary codes to States

Present	Next		Output	
	X=0	X=1	X=0	X=1
A	A	B	1	1
B	C	B	1	1
C	A	B	1	0

Next State Table

Put A=00
B=01
C=10

Present		Next			
		X=0		X=1	
Q_1	Q_0	Q_1	Q_0	Q_1	Q_0
0	0	0	0	0	1
0	1	1	0	0	1
1	0	0	0	0	1

SEQUENCE RECOGNIZER

Step: 3 Finding flip-flop input values

Transition table of T flip flop

Q_P	Q_N	T
0	0	0
0	1	1
1	0	1
1	1	0

SEQUENCE RECOGNIZER

Step: 4 Find equations for the FF inputs and output

X=0			X=1		
QP_0	QN_0	T_0	QP_0	QN_0	T_0
0	0	0	0	1	1
1	0	1	1	1	0
0	0	0	0	1	1

K-map for T_0

x \ Q_1Q_0	00	01	11	10
0	0	1	x	0
1	1	0	X	1

$T_0=X'Q_0+XQ'_0$

SEQUENCE RECOGNIZER

Step: 4 Find equations for the FF inputs and output

X=0			X=1		
QP_1	QN_1	T_1	QP_1	QN_1	T_1
0	0	0	0	0	0
0	1	1	0	0	0
1	0	1	1	0	1

K-map for T_1

x \ Q_1Q_0	00	01	11	10
0	0	1	x	1
1	0	0	X	1

$T_1=Q_1+X'Q_0$

SEQUENCE RECOGNIZER

Step: 5 Build the circuit

For Output Z

K-map

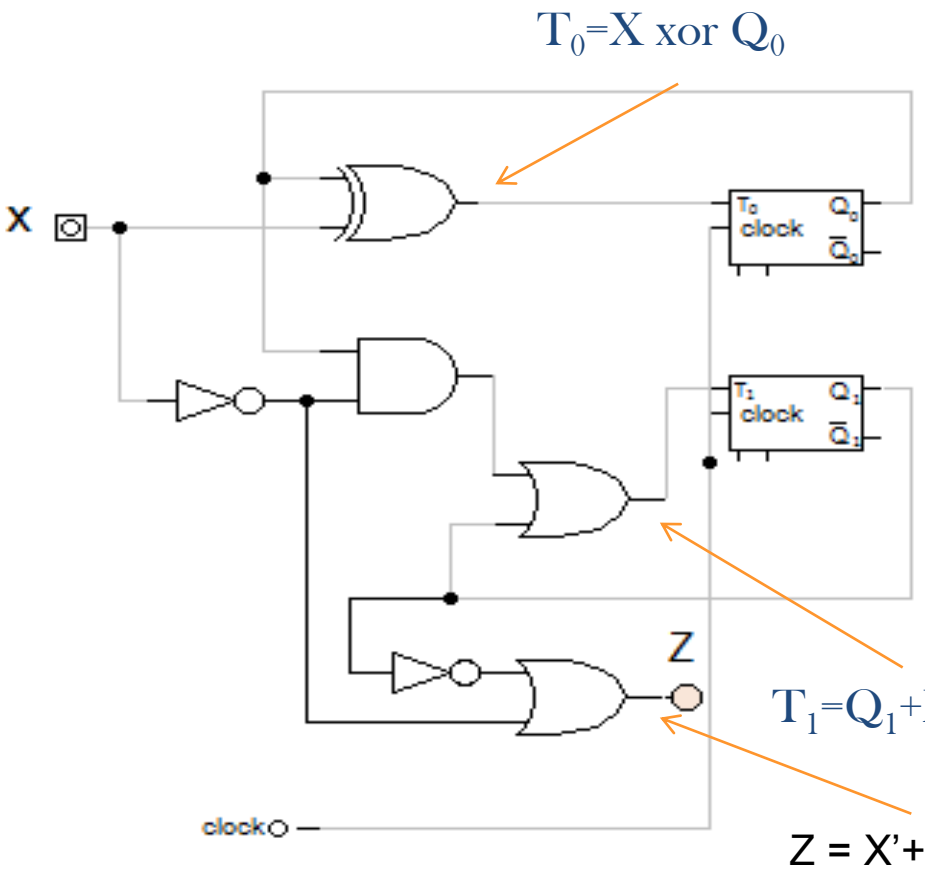
X \ Q_1Q_0	00	01	11	10
0	1	1	x	1
1	1	1	x	0

$Z = X' + Q'_1$

$T_0 = X'Q_0 + XQ'_0$

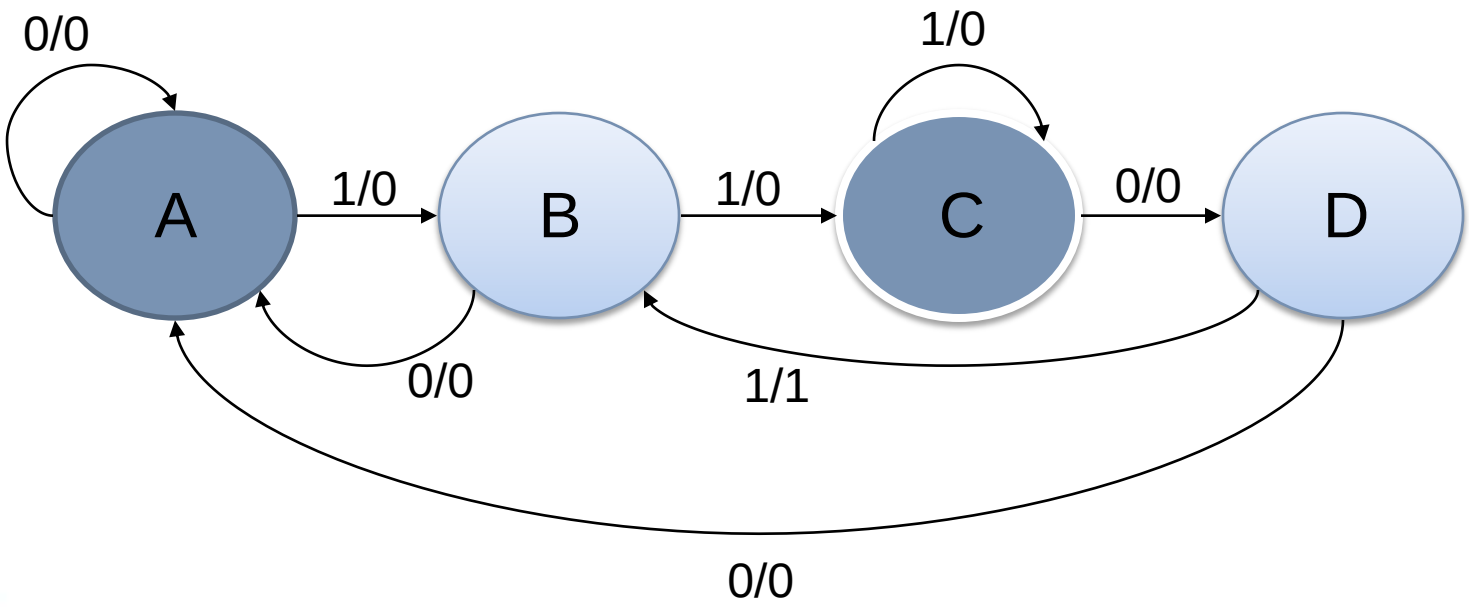
$T_0 = X \text{ xor } Q_0$

$T_1 = Q_1 + X'Q_0$



SEQUENCE RECOGNIZER

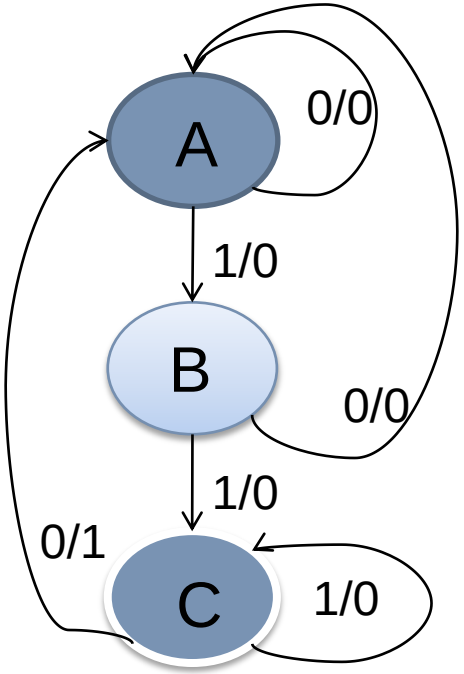
State diagram for 1101: Output 1 if the sequence 1101 has been read, output 0 otherwise.



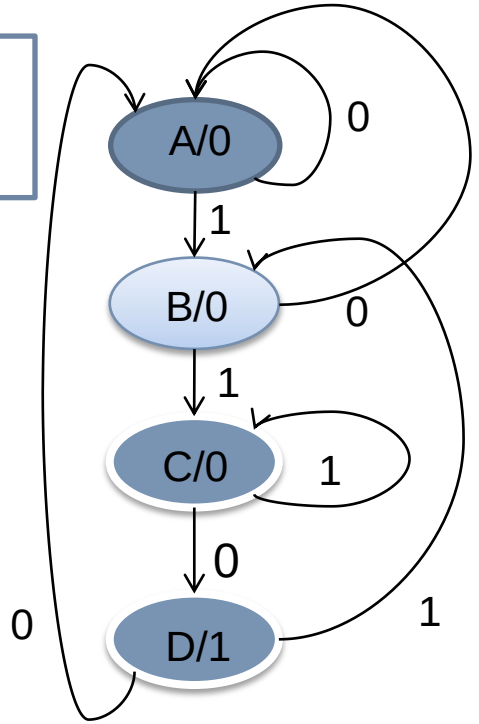
Sequence Recognizer
Moore vs Mealy Model

State diagram for 110: Output 1 if the sequence 110 has been read, output 0 otherwise.

Mealy Model



Moore Model



ANY QUERIES???

Hope it helped you!!!