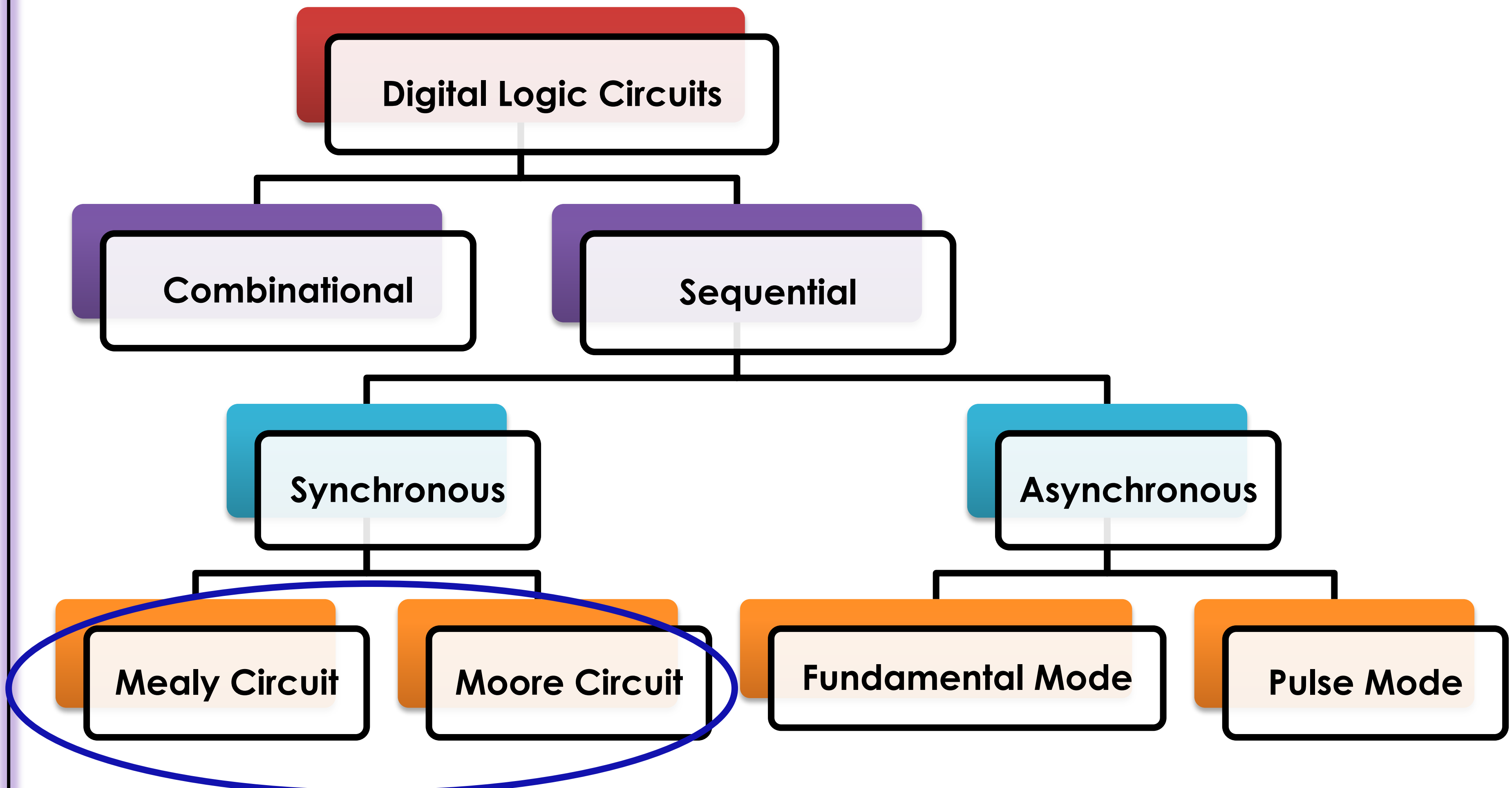
MEALY & MOORE MODEL

CLASSIFICATION OF DIGITAL LOGIC CIRCUITS



CLASSIFICATION OF DIGITAL LOGIC CIRCUITS

- The behavior of synchronous sequential circuits can be represented in the graphical form and it is known as state diagram.

- A synchronous sequential circuit is also called as Finite State Machine (FSM), if it has finite number of states. There are two types of FSMs.

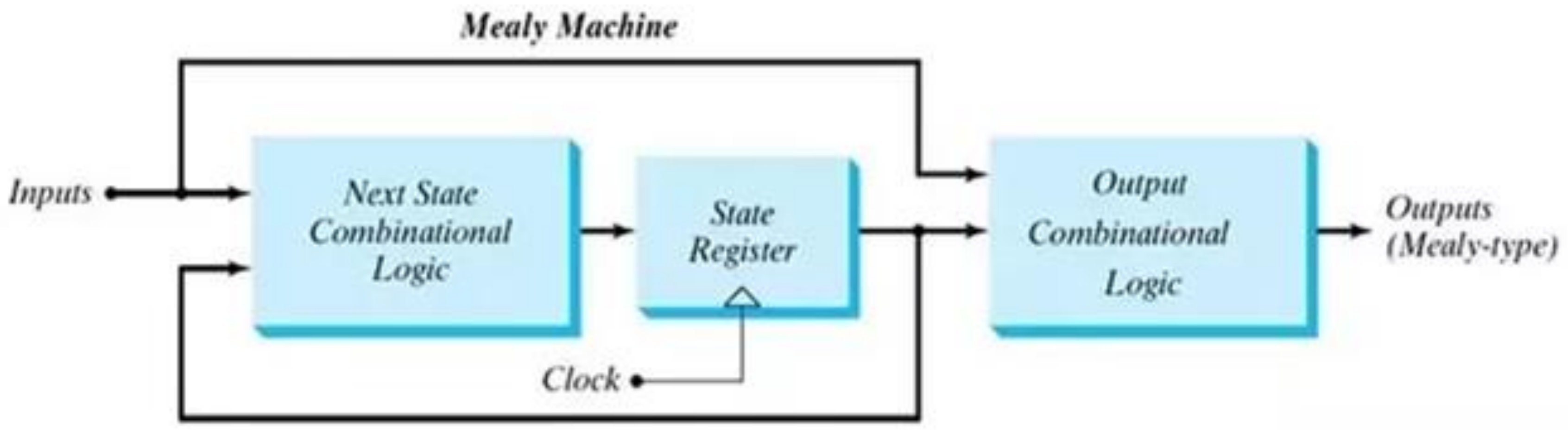
  - Mealy State Machine
  - Moore State Machine

MEALY MODEL

MEALY MODEL

- There are two types of finite state machines that generate output Mealy Machine & Moore machine

- A Mealy Machine is an FSM whose output depends on the present state as well as the present input

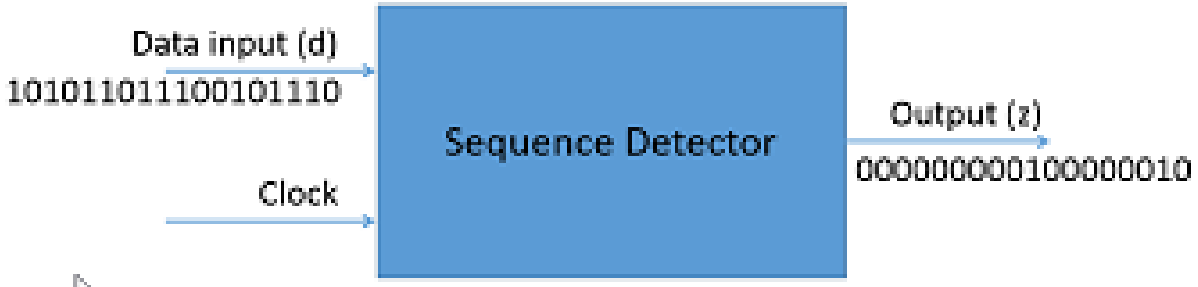


SEQUENCE DETECTOR

- A sequence detector is a sequential circuit that outputs 1 when a particular pattern of bits sequentially arrives at its data input.

- The data input receives the input sequence and the clock is used to synchronize the functionality of the circuit.

- If you analyse the input and output sequences, only when the last 4-bits of the input sequence are 1001 the output turn to 1, then it turns back to 0.



MEALY MODEL

MEALY MODEL

- Types of sequence detector: Overlapping and Non-overlapping.

- In an overlapping sequence detector, the final bits of one sequence becomes the start of next sequence. Example - bit pattern“1001”,

  Inputs: 11100110100100110

  Outputs: 00000100000100100

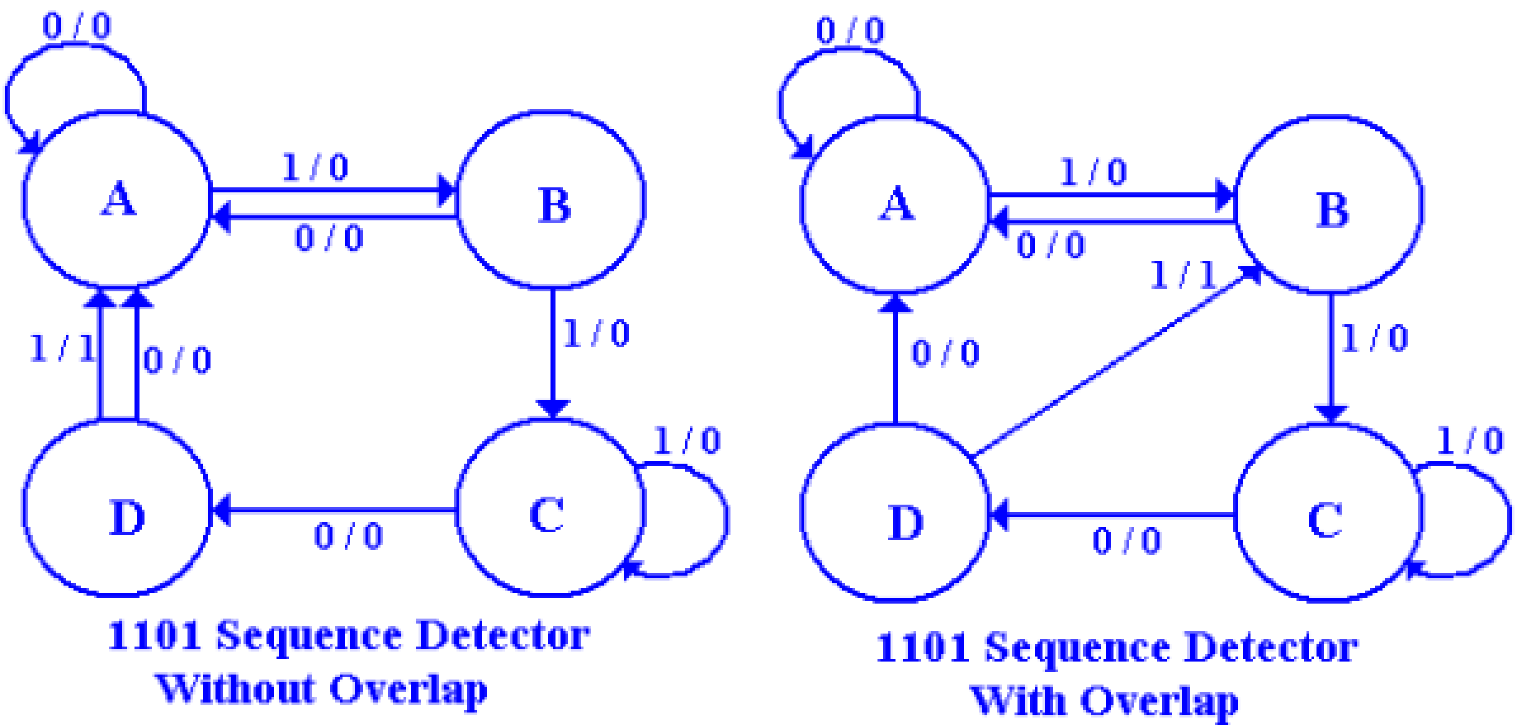
- In Non-overlapping sequence detector, the last bit of one sequence can’t considered as first bit of the next sequence i.e it resets itself to the start state when the sequence has been detected. Example - bit pattern“1001”:

  Inputs: 11100110100100110

  Outputs: 00000100000100000

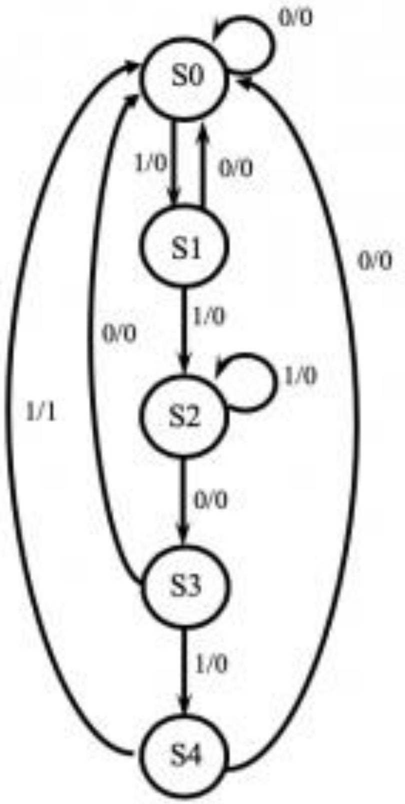
MEALY MODEL

MEALY MODEL – SEQUENCE DETECTOR EXAMPLES

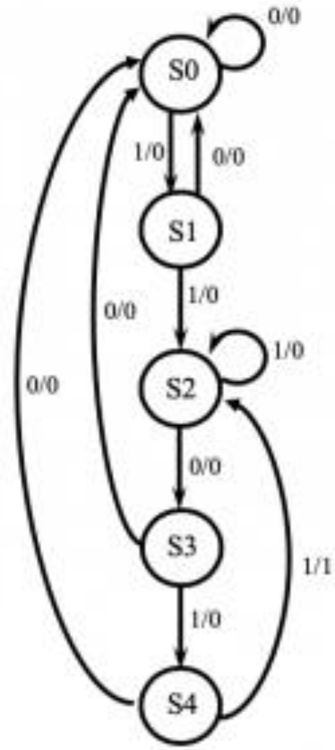


MEALY MODEL

MEALY MODEL – SEQUENCE DETECTOR EXAMPLES



11011 sequence detector
without overlapping



11011 sequence detector
with overlapping

MEALY MODEL

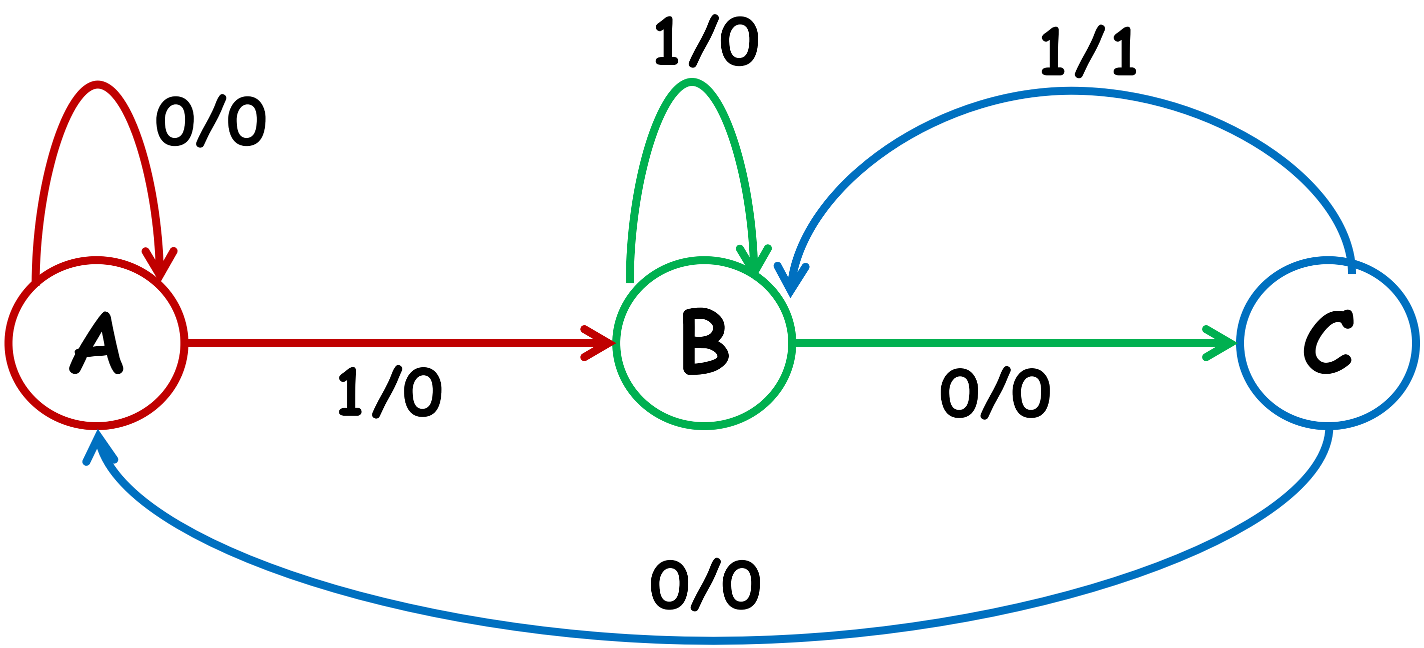
STEPS TO DESIGN MEALY MODEL - SEQUENCE DETECTOR

1. Draw the state diagram
2. Construct state table
3. Construct state table with state values
4. Determine excitation table
5. Construct the transition table
6. K-Map simplification procedures for driving expressions
7. Draw the logic diagram

MEALY MODEL – EXAMPLE-1

STEPS TO DESIGN MEALY MODEL – SEQUENCE DETECTOR "101"

1. Draw the state diagram



MEALY MODEL – EXAMPLE-1

STEPS TO DESIGN MEALY MODEL – SEQUENCE DETECTOR "101"

2. Construct state table

Present State	Input	Next State	Output
A	0	A	0
A	1	B	0
B	0	C	0
B	1	B	0
C	0	A	0
C	1	B	1

NOTE: For state C when input X=1 then it move to state B and produce the output as 1. For all other cases the output remains 0.

MEALY MODEL – EXAMPLE-1

STEPS TO DESIGN MEALY MODEL – SEQUENCE DETECTOR "101"

3. Construct state table with state values

Present State Q1	Present State Q0	Input X	Next State Q1	Next State Q0	Output Z
0	0	0	0	0	0
0	0	1	0	1	0
0	1	0	1	0	0
0	1	1	0	1	0
1	0	0	0	0	0
1	0	1	0	1	1

MEALY MODEL – EXAMPLE-1

STEPS TO DESIGN MEALY MODEL – SEQUENCE DETECTOR "101"

4. Determine excitation table (D-Flip Flop)

Q(t)	Q(t+1)	D
0	0	0
0	1	1
1	0	0
1	1	1

MEALY MODEL – EXAMPLE-1

STEPS TO DESIGN MEALY MODEL – SEQUENCE DETECTOR "101"

5. Construct the transition table

Present State Q1 Q0	Input X	Next State Q1 Q0	Flip-Flop inputs D1 D0	Output Z
0 0	0	0 0	0 0	0
0 0	1	0 1	0 1	0
0 1	0	1 0	1 0	0
0 1	1	0 1	0 1	0
1 0	0	0 0	0 0	0
1 0	1	0 1	0 1	1

NOTE: Number of flip flops required for the design is calculated based on number of state. In this case number state is 3, so we need 2 flip flop for the design (2^2=4).

MEALY MODEL – EXAMPLE-1

STEPS TO DESIGN MEALY MODEL – SEQUENCE DETECTOR "101"

6. K-Map simplification procedures for driving expressions

Q1 \ Q0X	00	01	11	10
0	0	0	0	1
1	0	0	X	X

D1 = Q0X'

Q1 \ Q0X	00	01	11	10
0	0	1	1	0
1	0	1	X	X

D0 = X

NOTE: In K-Map, we must assume don't care "x" values for the remaining unknown states. In this case "11" state is unknown state and its output is "X" irrespective of input is 0 or 1.

Q1 \ Q0X	00	01	11	10
0	0	0	0	0
1	0	1	X	X

Z = Q1X

MEALY MODEL – EXAMPLE-1

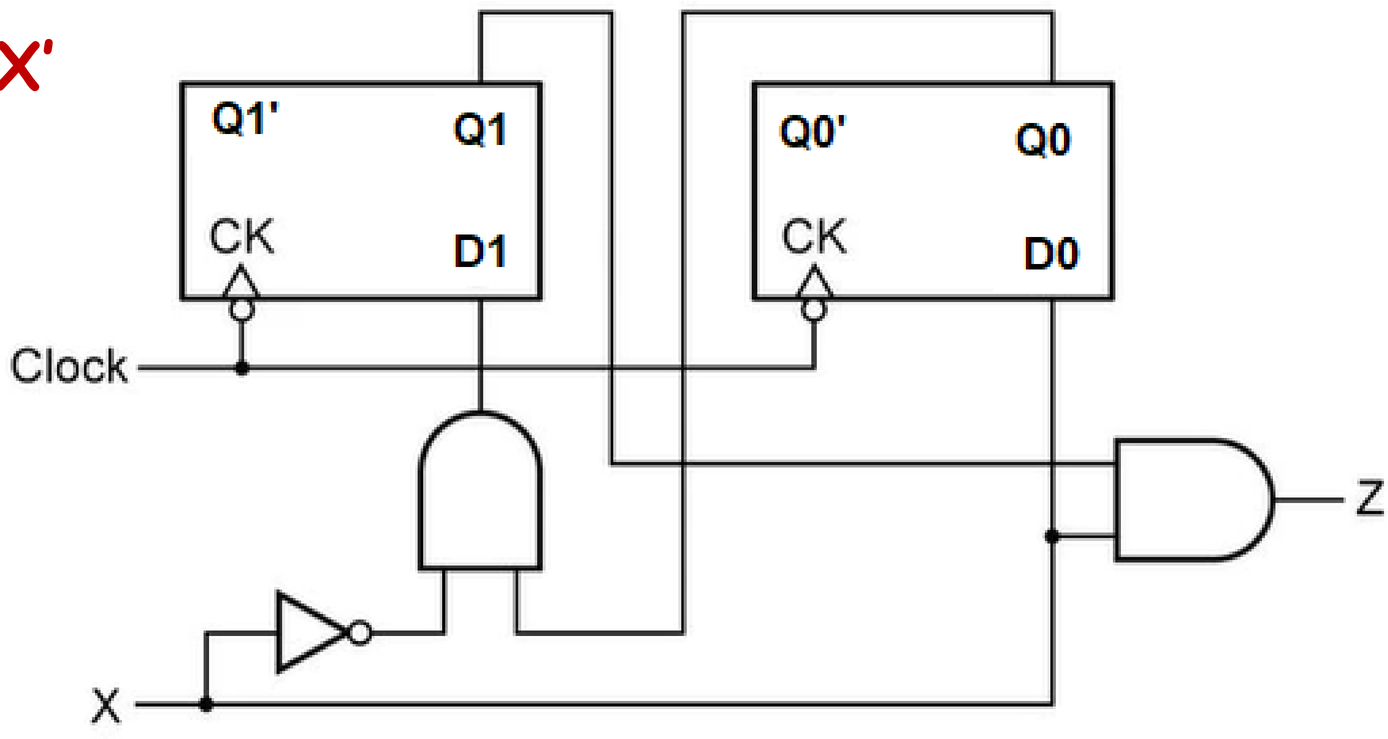
STEPS TO DESIGN MEALY MODEL – SEQUENCE DETECTOR “101”

7. Draw the logic diagram

D1 = Q0X’

D0 = X

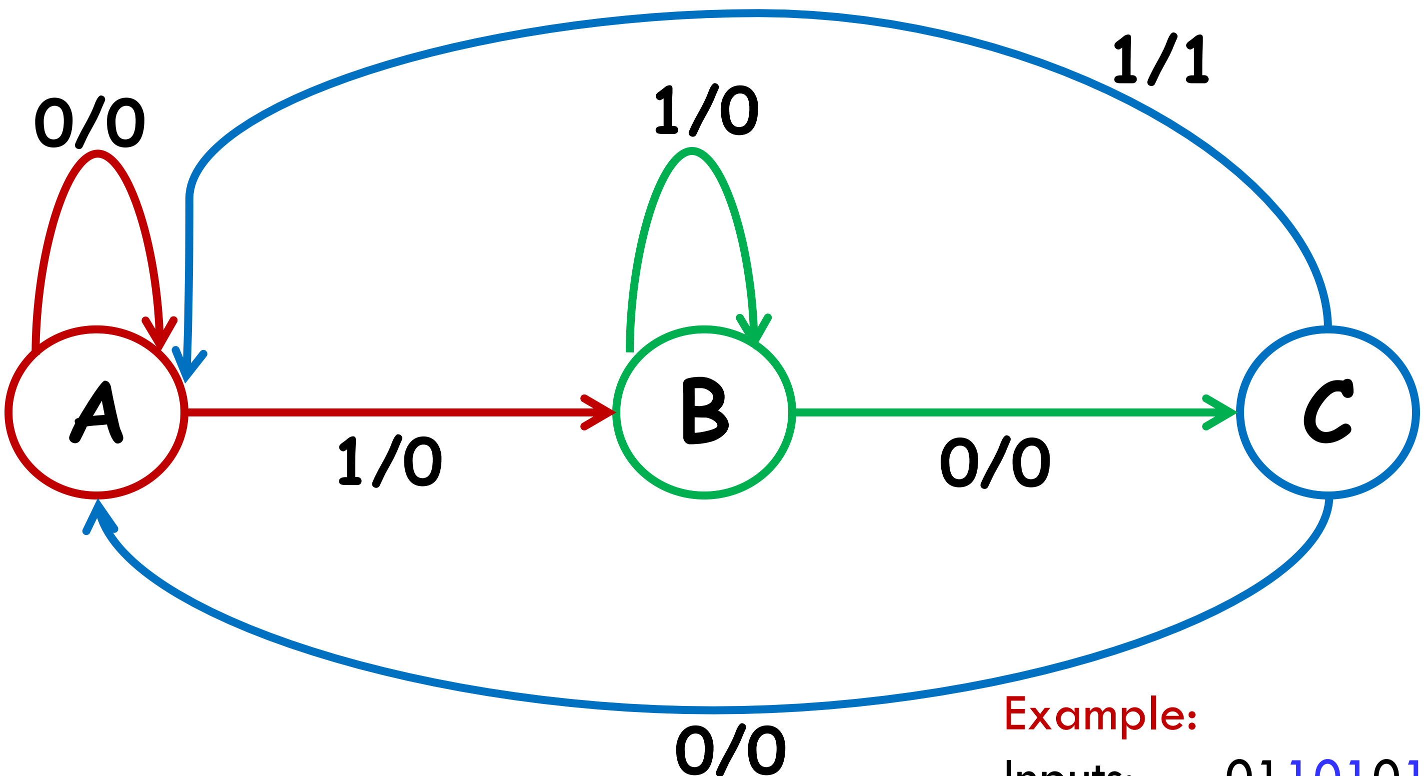
Z = Q1X



MEALY MODEL – EXAMPLE-2

STEPS TO DESIGN MEALY MODEL – SEQUENCE DETECTOR "101" (Non-Overlapping)

1. Draw the state diagram



Example:

Inputs: 01101010111001

Outputs: 00001000010000

MEALY MODEL – EXAMPLE-2

STEPS TO DESIGN MEALY MODEL – SEQUENCE DETECTOR "101" (Non-Overlapping)

2. Construct state table

Present State	Input	Next State	Output
A	0	A	0
A	1	B	0
B	0	C	0
B	1	B	0
C	0	A	0
C	1	A	1

NOTE: For state C when input X=1 then it move to state B and produce the output as 1. For all other cases the output remains 0.

MEALY MODEL – EXAMPLE-2

STEPS TO DESIGN MEALY MODEL – SEQUENCE DETECTOR "101" (Non-Overlapping)

3. Construct state table with state values

Present State Q1 Q0	Input X	Next State Q1 Q0	Output Z
0 0	0	0 0	0
0 0	1	0 1	0
0 1	0	1 0	0
0 1	1	0 1	0
1 0	0	0 0	0
1 0	1	0 0	1

MEALY MODEL – EXAMPLE-2

STEPS TO DESIGN MEALY MODEL – SEQUENCE DETECTOR "101" (Non-Overlapping)

4. Determine excitation table (D-Flip Flop)

Q(t)	Q(t+1)	D
0	0	0
0	1	1
1	0	0
1	1	1

MEALY MODEL – EXAMPLE-2

STEPS TO DESIGN MEALY MODEL – SEQUENCE DETECTOR "101" (Non-Overlapping)

5. Construct the transition table

Present State Q1 Q0	Input X	Next State Q1 Q0	Flip-Flop inputs D1 D0	Output Z
0 0	0	0 0	0 0	0
0 0	1	0 1	0 1	0
0 1	0	1 0	1 0	0
0 1	1	0 1	0 1	0
1 0	0	0 0	0 0	0
1 0	1	0 0	0 0	1

NOTE: Number of flip flops required for the design is calculated based on number of state. In this case number state is 3, so we need 2 flip flop for the design (2^2=4).

MEALY MODEL – EXAMPLE-2

STEPS TO DESIGN MEALY MODEL – SEQUENCE DETECTOR "101"
(Non-Overlapping)

6. K-Map simplification procedures for driving expressions

Q1 \ Q0X	00	01	11	10
0	0	0	0	1
1	0	0	X	X

D1 = Q0X'

Q1 \ Q0X	00	01	11	10
0	0	1	1	0
1	0	0	X	X

D0 = Q1'X

NOTE: In K-Map, we must assume don't care "x" values for the remaining unknown states. In this case "11" state is unknown state and its output is "X" irrespective of input is 0 or 1.

Q1 \ Q0X	00	01	11	10
0	0	0	0	0
1	0	1	X	X

Z = Q1X

MEALY MODEL – EXAMPLE-2

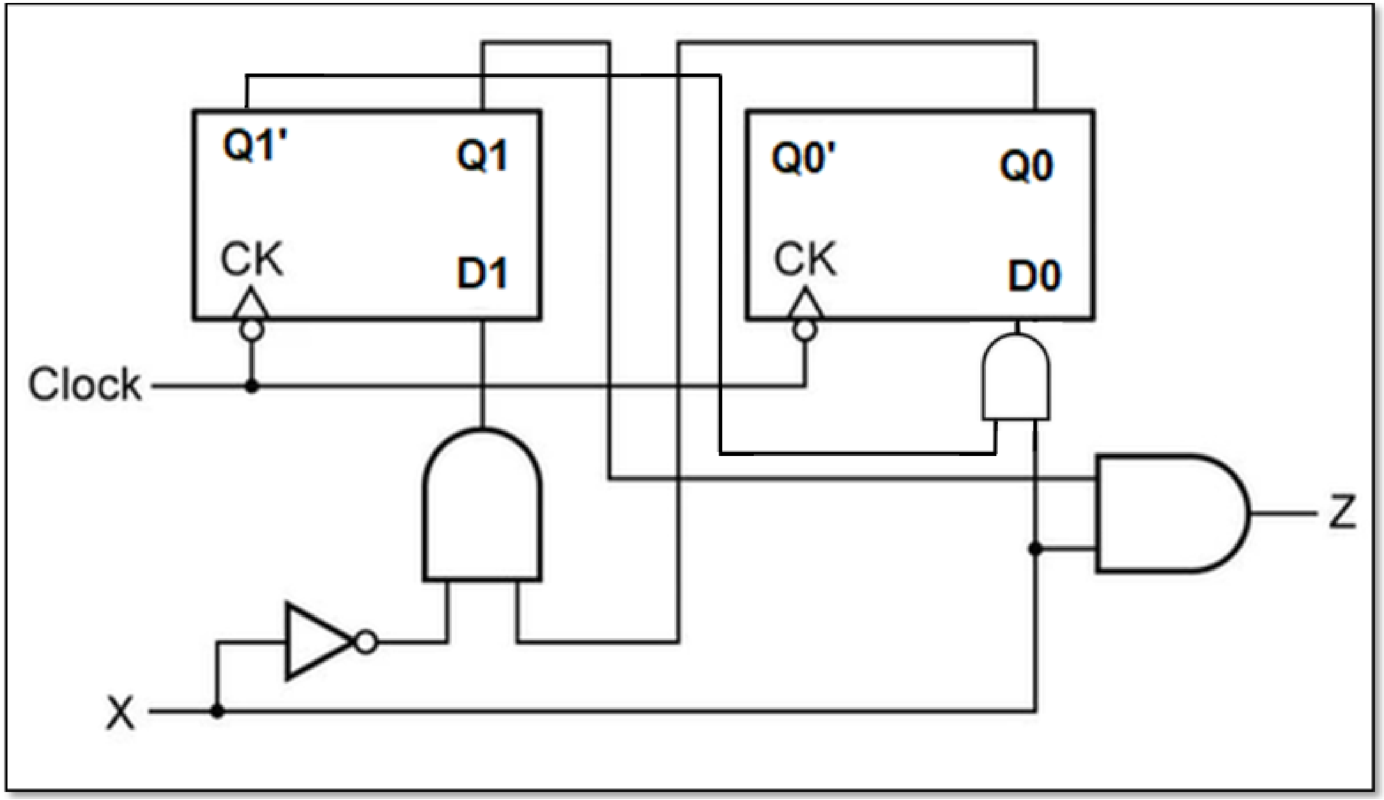
STEPS TO DESIGN MEALY MODEL – SEQUENCE DETECTOR "101" (Non-Overlapping)

7. Draw the logic diagram

$D1 = Q0X'$

$D0 = Q1'X$

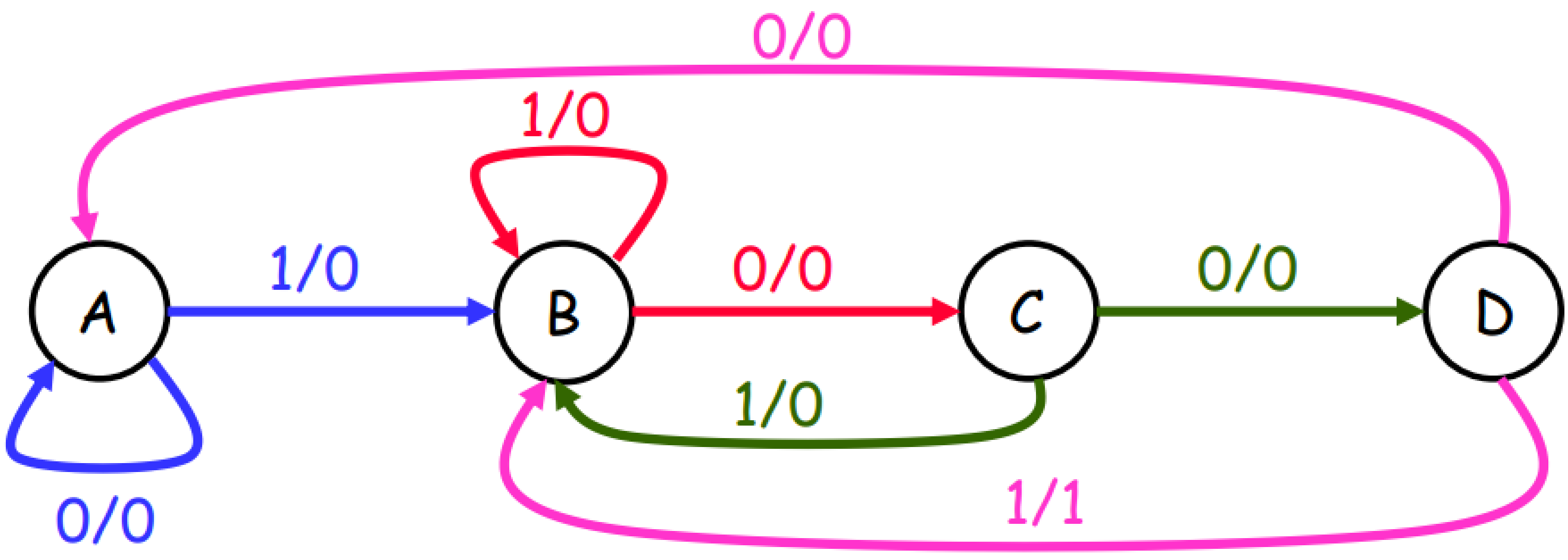
$Z = Q1X$



MEALY MODEL – EXAMPLE-3

STEPS TO DESIGN MEALY MODEL – SEQUENCE DETECTOR "1001"

1. Draw the state diagram



MEALY MODEL – EXAMPLE-3

STEPS TO DESIGN MEALY MODEL – SEQUENCE DETECTOR “1001”

2. Construct state table

Present State	Input	Next State	Output
A	0	A	0
A	1	B	0
B	0	C	0
B	1	B	0
C	0	D	0
C	1	B	0
D	0	A	0
D	1	B	1

NOTE: For state D when input X=1 then it move to state B and produce the output as 1. For all other cases the output remains 0.

MEALY MODEL – EXAMPLE-3

STEPS TO DESIGN MEALY MODEL – SEQUENCE DETECTOR "1001"

3. Construct state table with state values

Present State Q_1	Present State Q_0	Input X	Next State Q_1	Next State Q_0	Output Z
0	0	0	0	0	0
0	0	1	0	1	0
0	1	0	1	0	0
0	1	1	0	1	0
1	0	0	1	1	0
1	0	1	0	1	0
1	1	0	0	0	0
1	1	1	0	1	1

MEALY MODEL – EXAMPLE-3

STEPS TO DESIGN MEALY MODEL – SEQUENCE DETECTOR "1001"

4. Determine excitation table

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

MEALY MODEL – EXAMPLE-3

STEPS TO DESIGN MEALY MODEL – SEQUENCE DETECTOR "1001"

5. Construct the transition table

Present State		Input	Next State		Flip flop inputs				Output
Q_1	Q_0	X	Q_1	Q_0	J_1	K_1	J_0	K_0	Z
0	0	0	0	0	0	x	0	x	0
0	0	1	0	1	0	x	1	x	0
0	1	0	1	0	1	x	x	1	0
0	1	1	0	1	0	x	x	0	0
1	0	0	1	1	x	0	1	x	0
1	0	1	0	1	x	1	1	x	0
1	1	0	0	0	x	1	x	1	0
1	1	1	0	1	x	1	x	0	1

NOTE: Number of flip flops required for the design is calculated based on number of states. In this case number state is 4, so we need 2 flip flop for the design ($2^2=4$).

MEALY MODEL – EXAMPLE-3

STEPS TO DESIGN MEALY MODEL – SEQUENCE DETECTOR "1001"

6. K-Map simplification procedures for driving expressions

Q1 \ Q0X	00	01	11	10
0	0	0	0	1
1	X	X	X	X

J1 = Q0X'

Q1 \ Q0X	00	01	11	10
0	X	X	X	X
1	0	1	1	1

K1 = X + Q0

MEALY MODEL – EXAMPLE-3

STEPS TO DESIGN MEALY MODEL – SEQUENCE DETECTOR "1001"

Q1 \ Q0X	00	01	11	10
0	0	1	X	X
1	1	1	X	X

$J0 = Q1 + X$

Q1 \ Q0X	00	01	11	10
0	X	X	0	1
1	X	X	0	1

$K0 = X'$

Q1 \ Q0X	00	01	11	10
0	0	0	0	0
1	0	0	1	0

$Z = Q1Q0X$

MEALY MODEL – EXAMPLE-3

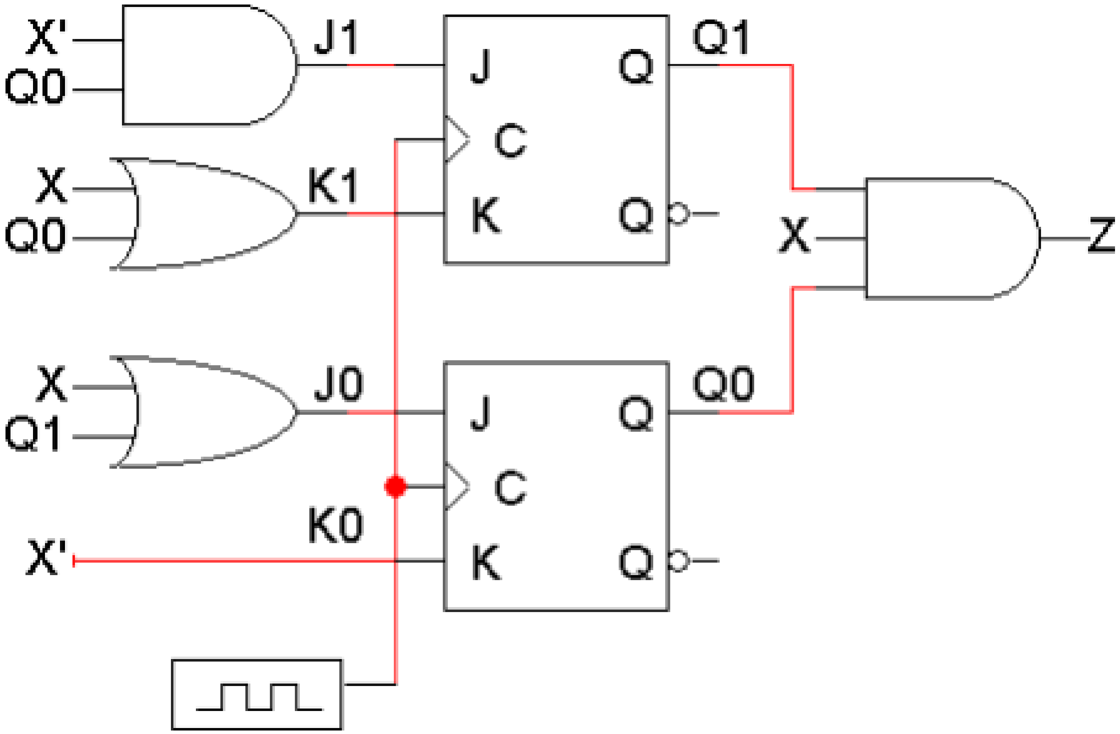
STEPS TO DESIGN MEALY MODEL – SEQUENCE DETECTOR "1001"

7. Draw the logic diagram

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

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

$Z = Q_1 Q_0 X$



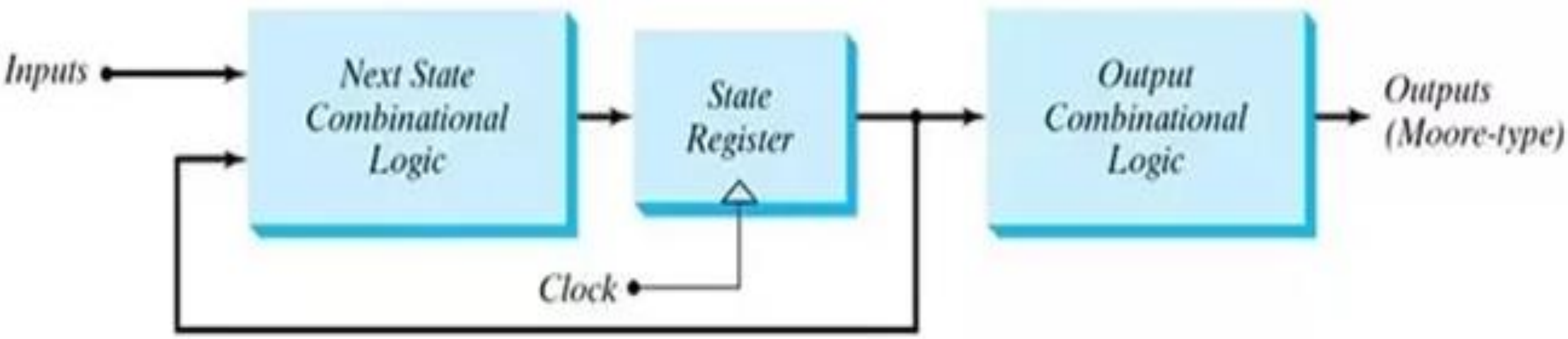
MEALY MODEL – EXERCISE

Design an mealy based sequence detector to detect the bit pattern of "1001" in non-overlapping condition.

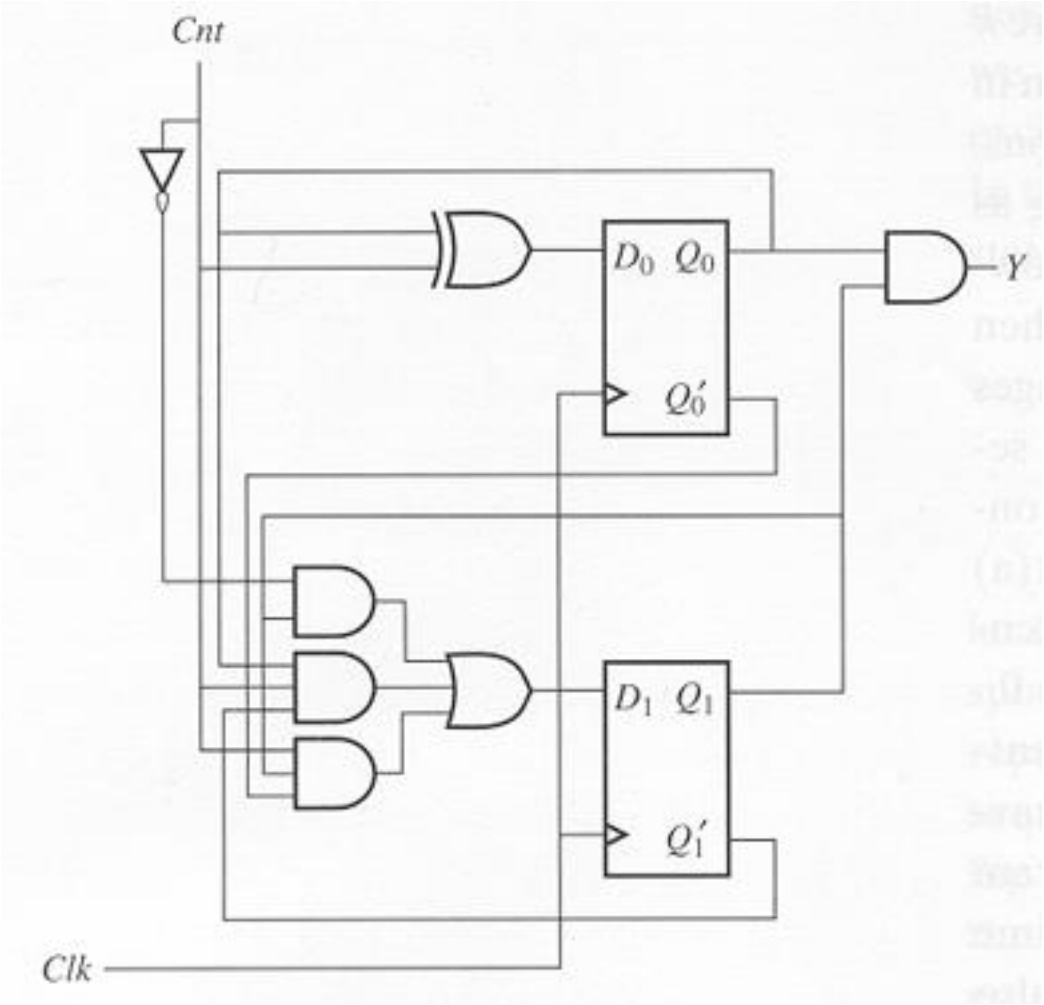
MOORE MODEL

MOORE MODEL

- In Moore state machine, outputs depend on only the present state.
- Next State depends on the Present state and the inputs



MOORE MODEL



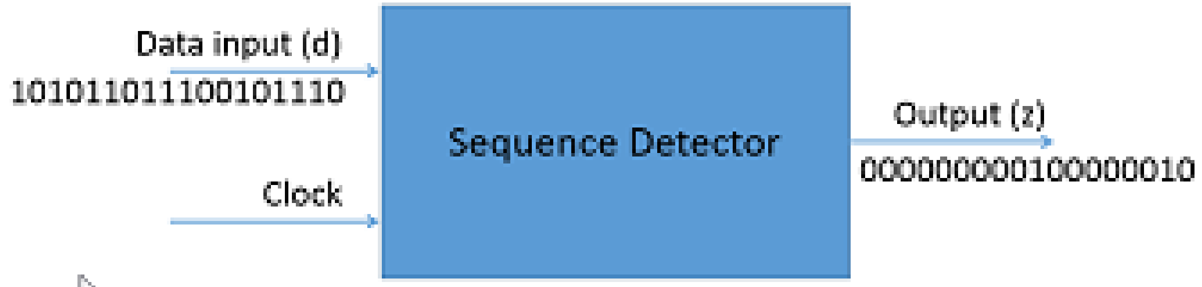
EXAMPLE FOR MOORE MODEL

SEQUENCE DETECTOR

- A sequence detector is a sequential circuit that outputs 1 when a particular pattern of bits sequentially arrives at its data input.

- The data input receives the input sequence and the clock is used to synchronize the functionality of the circuit.

- If you analyse the input and output sequences, only when the last 4-bits of the input sequence are 1001 the output turn to 1, then it turns back to 0.



MOORE MODEL

MOORE MODEL

- Types of sequence detector: Overlapping and Non-overlapping.

- In an overlapping sequence detector, the final bits of one sequence becomes the start of next sequence. Example - bit pattern“1001”,

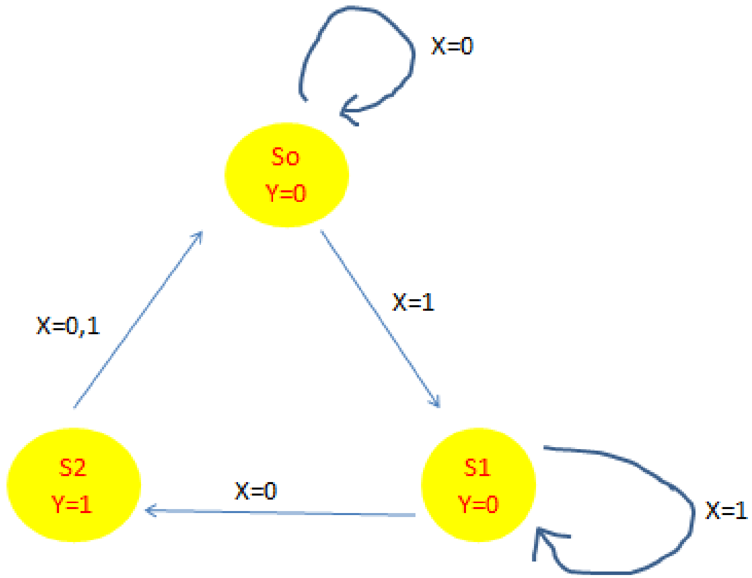
  Inputs: 11100110100100110

  Outputs: 00000100000100100

- In Non-overlapping sequence detector, the last bit of one sequence can’t considered as first bit of the next sequence i.e it resets itself to the start state when the sequence has been detected. Example - bit pattern“1001”:

  Inputs: 11100110100100110

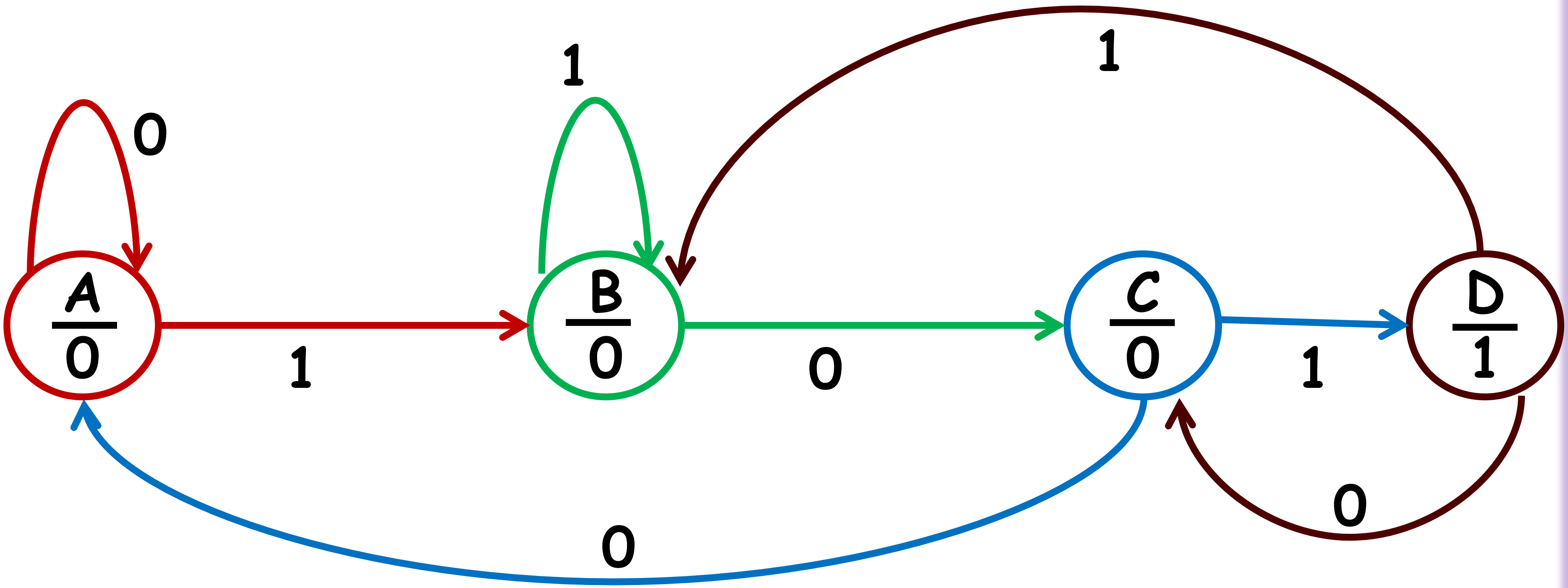
  Outputs: 00000100000100000



Present State		X	Next state		O/p	DA	DB
A	B	X	A+	B+	Y	Da	Db
0	0	0	0	0	0	0	0
0	0	1	0	1	0	0	1
0	1	0	1	0	0	1	0
0	1	1	0	1	0	0	1
1	0	0	0	0	1	0	0
1	0	1	0	0	1	0	0
1	1	0	-	-	-	-	-
1	1	1	-	-	-	-	-

MOORE MODEL

MOORE MODEL – SEQUENCE DETECTOR "101"



MOORE MODEL – EXAMPLE-1

STEPS TO DESIGN MOORE MODEL – SEQUENCE DETECTOR "101"

2. Construct state table

Present State	Input	Next State	Output
A	0	A	0
A	1	B	0
B	0	C	0
B	1	B	0
C	0	A	0
C	1	D	0
D	0	C	1
D	1	B	1

NOTE: In the given state table, the output will be 1 whenever its present state is "D" irrespective of input X(0 or 1). For all other states output remains 0.

MOORE MODEL – EXAMPLE-1

STEPS TO DESIGN MOORE MODEL – SEQUENCE DETECTOR "101"

3. Construct state table with state values

Present State Q1 Q0	Input X	Next State Q1 Q0	Output Z
0 0	0	0 0	0
0 0	1	0 1	0
0 1	0	1 0	0
0 1	1	0 1	0
1 0	0	0 0	0
1 0	1	1 1	0
1 1	0	1 0	1
1 1	1	0 1	1

MOORE MODEL – EXAMPLE-1

STEPS TO DESIGN MOORE MODEL – SEQUENCE DETECTOR "101"

4. Determine excitation table

Q(t)	Q(t+1)	D
0	0	0
0	1	1
1	0	0
1	1	1

MOORE MODEL – EXAMPLE-1

STEPS TO DESIGN MOORE MODEL – SEQUENCE DETECTOR "101"

5. Construct the transition table

Present State Q1 Q0	Input X	Next State Q1 Q0	Flip-Flop inputs D1 D0	Output Z
0 0	0	0 0	0 0	0
0 0	1	0 1	0 1	0
0 1	0	1 0	1 0	0
0 1	1	0 1	0 1	0
1 0	0	0 0	0 0	0
1 0	1	1 1	1 1	0
1 1	0	1 0	1 0	1
1 1	1	0 1	0 1	1

NOTE: Number of flip flops required for the design is calculated based on number of state. In this case number state is 4, so we need 2 flip flop for the design ($2^2=4$).

MOORE MODEL – EXAMPLE-1

STEPS TO DESIGN MOORE MODEL – SEQUENCE DETECTOR "101"

6. K-Map simplification procedures for driving expressions

Q1 \ Q0X	00	01	11	10
0	0	0	0	1
1	0	1	0	1

D1 = Q0X'+Q1Q0'X

Q1 \ Q0X	00	01	11	10
0	0	1	1	0
1	0	1	1	0

D0 = X

Q1 \ Q0X	00	01	11	10
0	0	0	0	0
1	0	0	1	1

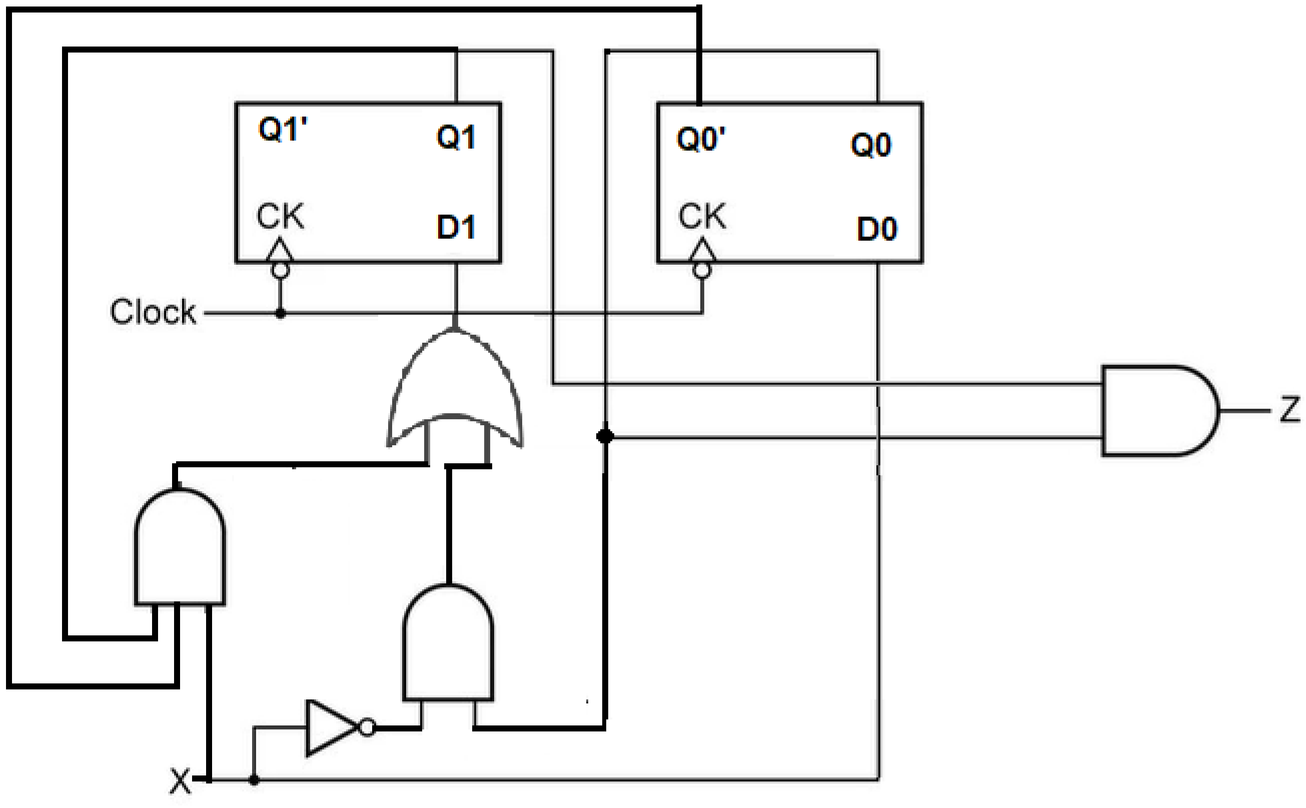
Z = Q1Q0

NOTE: In Moore model, the output expression (Z) depends only on present state values (Q1 & Q0) not on the input (X).

MOORE MODEL – EXAMPLE-1

STEPS TO DESIGN MOORE MODEL – SEQUENCE DETECTOR "101"

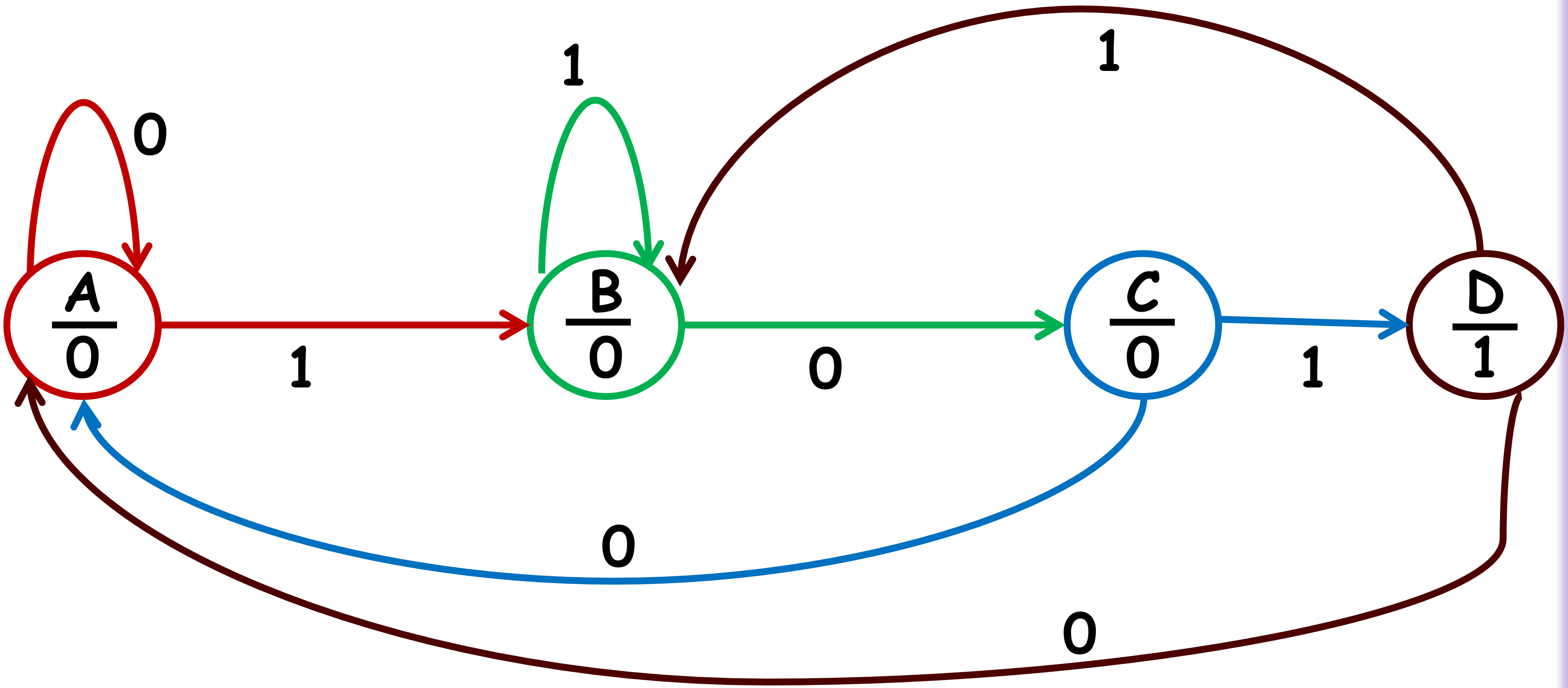
7. Draw the logic diagram



MOORE MODEL – EXAMPLE-2

STEPS TO DESIGN MOORE MODEL – SEQUENCE DETECTOR "101" (Non-overlapping)

1. Draw the state diagram



MOORE MODEL – EXAMPLE-2

STEPS TO DESIGN MOORE MODEL – SEQUENCE DETECTOR "101" (Non-overlapping)

2. Construct state table

Present State	Input	Next State	Output
A	0	A	0
A	1	B	0
B	0	C	0
B	1	B	0
C	0	A	0
C	1	D	0
D	0	A	1
D	1	B	1

NOTE: In the given state table, the output will be 1 whenever its present state is "D" irrespective of input X(0 or 1). For all other states output remains 0.

MOORE MODEL – EXAMPLE-2

STEPS TO DESIGN MOORE MODEL – SEQUENCE DETECTOR "101" (Non-overlapping)

3. Construct state table with state values

Present State Q1 Q0	Input X	Next State Q1 Q0	Output Z
0 0	0	0 0	0
0 0	1	0 1	0
0 1	0	1 0	0
0 1	1	0 1	0
1 0	0	0 0	0
1 0	1	1 1	0
1 1	0	0 0	1
1 1	1	0 1	1

MOORE MODEL – EXAMPLE-2

STEPS TO DESIGN MOORE MODEL – SEQUENCE DETECTOR "101" (Non-overlapping)

4. Determine excitation table

Q(t)	Q(t+1)	D
0	0	0
0	1	1
1	0	0
1	1	1

MOORE MODEL – EXAMPLE-2

STEPS TO DESIGN MOORE MODEL – SEQUENCE DETECTOR "101" (Non-overlapping)

5. Construct the transition table

Present State Q1	Present State Q0	Input X	Next State Q1	Next State Q0	Flip-Flop inputs D1	Flip-Flop inputs D0	Output Z
0	0	0	0	0	0	0	0
0	0	1	0	1	0	1	0
0	1	0	1	0	1	0	0
0	1	1	0	1	0	1	0
1	0	0	0	0	0	0	0
1	0	1	1	1	1	1	0
1	1	0	0	0	0	0	1
1	1	1	0	1	0	1	1

NOTE: Number of flip flops required for the design is calculated based on number of state. In this case number state is 4, so we need 2 flip flop for the design ($2^2=4$).

MOORE MODEL – EXAMPLE-2

STEPS TO DESIGN MOORE MODEL – SEQUENCE DETECTOR "101" (Non-overlapping)

6. K-Map simplification procedures for driving expressions

Q1 \ Q0X	00	01	11	10
0	0	0	0	1
1	0	1	0	0

$D1 = Q1'Q0X' + Q1Q0'X$

Q1 \ Q0X	00	01	11	10
0	0	1	1	0
1	0	1	1	0

$D0 = X$

Q1 \ Q0X	00	01	11	10
0	0	0	0	0
1	0	0	1	1

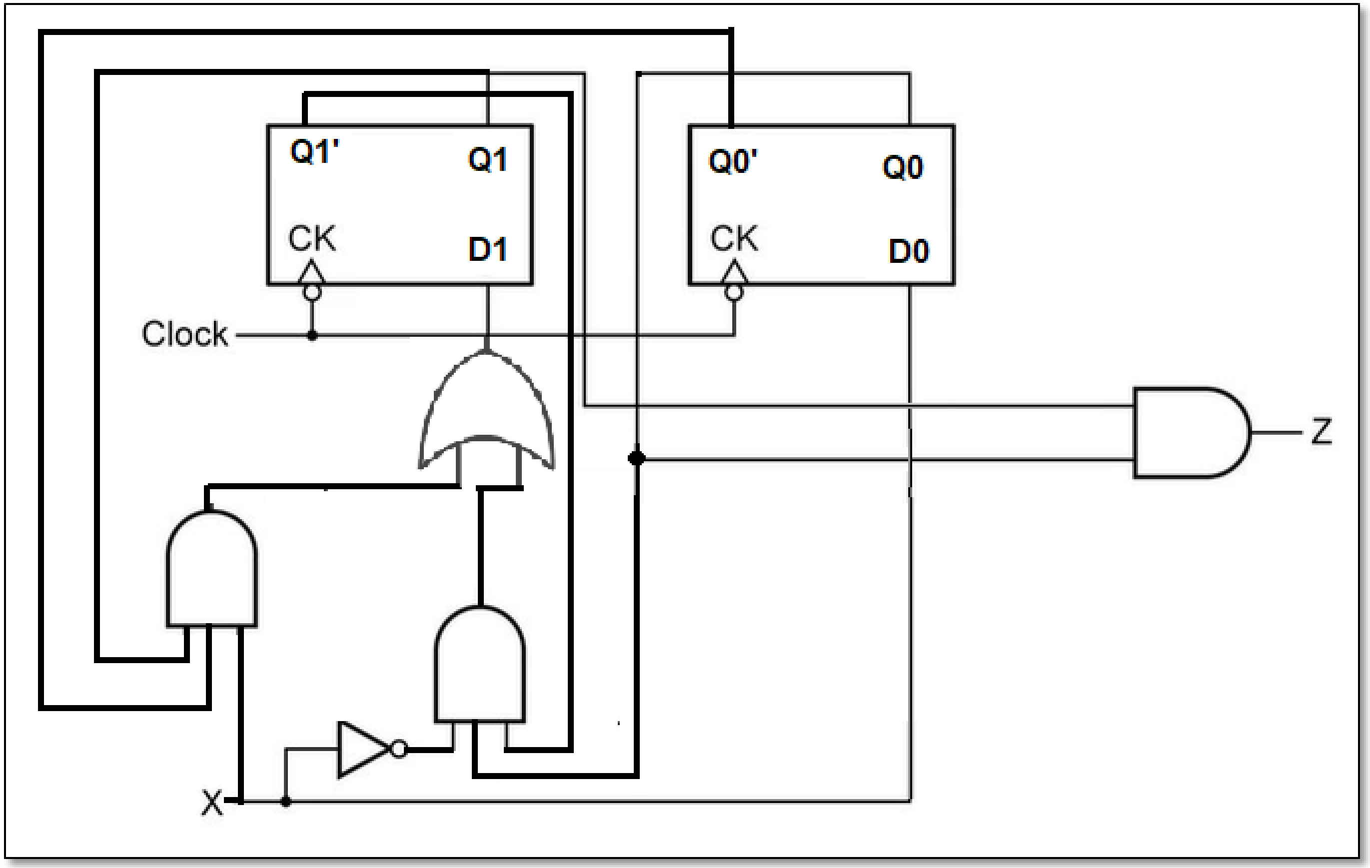
$Z = Q1Q0$

NOTE: In Moore model, the output expression (Z) depends only on present state values (Q1 & Q0) not on the input (X).

MOORE MODEL – EXAMPLE-3

STEPS TO DESIGN MOORE MODEL – SEQUENCE DETECTOR "101" (Non-overlapping)

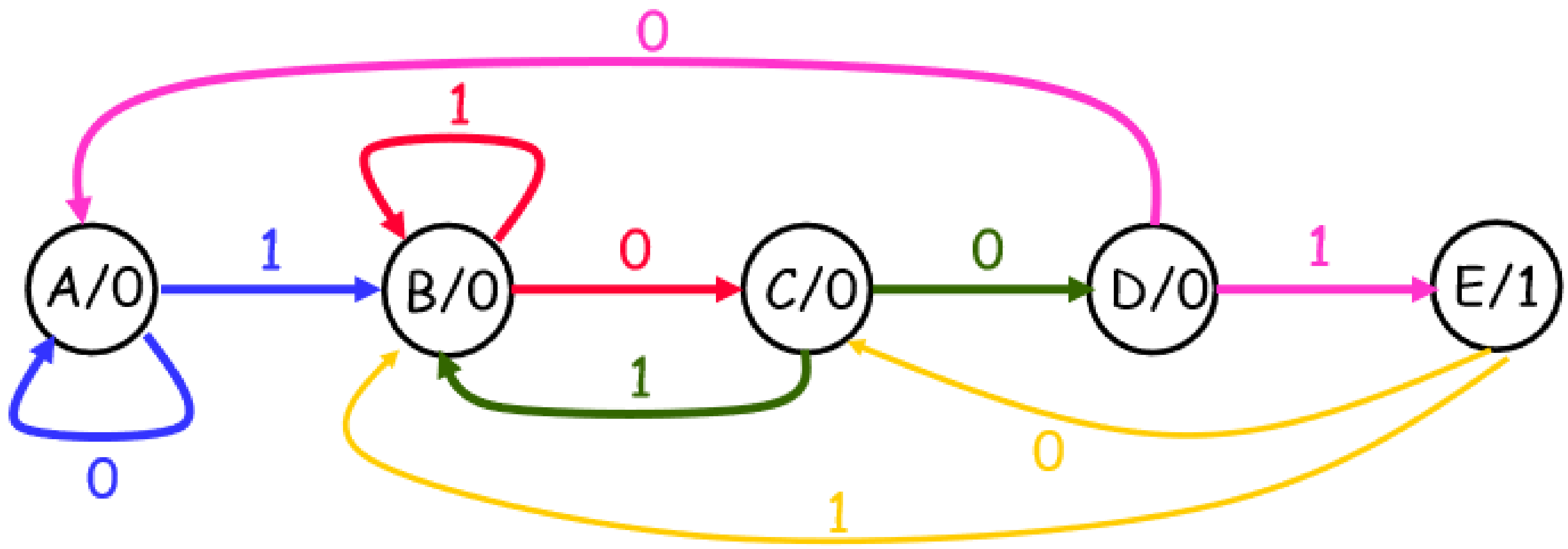
7. Draw the logic diagram



MOORE MODEL – EXAMPLE-3

STEPS TO DESIGN MOORE MODEL – SEQUENCE DETECTOR “1001”

1. Draw the state diagram



MOORE MODEL – EXAMPLE-3

STEPS TO DESIGN MOORE MODEL – SEQUENCE DETECTOR "1001"

2. Construct state table

Present State	Input	Next State	Output
A	0	A	0
A	1	B	0
B	0	C	0
B	1	B	0
C	0	D	0
C	1	B	0
D	0	A	0
D	1	E	0
E	0	C	1
E	1	B	1

NOTE: In the given state table, the output will be 1 whenever its present state is "E" irrespective of input X (0 or 1).

MOORE MODEL – EXAMPLE-3

STEPS TO DESIGN MOORE MODEL – SEQUENCE DETECTOR "1001"

3. Construct state table with state values

Present State Q2	Q1	Q0	Input X	Next State Q2	Q1	Q0	Output Z
0	0	0	0	0	0	0	0
0	0	0	1	0	0	1	0
0	0	1	0	0	1	0	0
0	0	1	1	0	0	1	0
0	1	0	0	0	1	1	0
0	1	0	1	0	0	1	0
0	1	1	0	0	0	0	0
0	1	1	1	1	0	0	0
1	0	0	0	0	1	0	1
1	0	0	1	0	0	1	1

MOORE MODEL – EXAMPLE-3

STEPS TO DESIGN MOORE MODEL – SEQUENCE DETECTOR "1001"

4. Determine excitation table

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

MOORE MODEL – EXAMPLE-3

STEPS TO DESIGN MOORE MODEL – SEQUENCE DETECTOR "1001"

5. Construct the transition table

Present State			Input	Next State			Flip-Flop Inputs						Output
Q2	Q1	Q0	X	Q2	Q1	Q0	J2	K2	J1	K1	J0	K0	Z
0	0	0	0	0	0	0	0	X	0	X	0	X	0
0	0	0	1	0	0	1	0	X	0	X	1	X	0
0	0	1	0	0	1	0	0	X	1	X	X	1	0
0	0	1	1	0	0	1	0	X	0	X	X	0	0
0	1	0	0	0	1	1	0	X	X	0	1	X	0
0	1	0	1	0	0	1	0	X	X	1	1	X	0
0	1	1	0	0	0	0	0	X	X	1	X	1	0
0	1	1	1	1	0	0	1	X	X	1	X	1	0
1	0	0	0	0	1	0	X	1	1	X	0	X	1
1	0	0	1	0	0	1	X	1	0	X	1	X	1

NOTE: Number of flip flops required for the design is calculated based on number of state. In this case number state is 5, so we need 3 flip flop for the design (2^3=8).

MOORE MODEL – EXAMPLE-3

STEPS TO DESIGN MOORE MODEL – SEQUENCE DETECTOR "1001"

6. K-Map simplification procedures for driving expressions

Q2Q1 \ Q0X	00	01	11	10
00	0	0	0	0
01	0	0	1	0
11	X	X	X	X
10	X	X	X	X

J2 = Q1Q0X

Q2Q1 \ Q0X	00	01	11	10
00	X	X	X	X
01	X	X	X	X
11	X	X	X	X
10	1	1	X	X

K2 = 1

MOORE MODEL – EXAMPLE-3

STEPS TO DESIGN MOORE MODEL – SEQUENCE DETECTOR "1001"

6. K-Map simplification procedures for driving expressions

Q2Q1 \ Q0X	00	01	11	10
00	0	0	0	1
01	X	X	X	X
11	X	X	X	X
10	1	0	X	X

J1 = Q2X'+Q0X'

Q2Q1 \ Q0X	00	01	11	10
00	X	X	X	X
01	0	1	1	1
11	X	X	X	X
10	X	X	X	X

K1 = X+Q0

MOORE MODEL – EXAMPLE-3

STEPS TO DESIGN MOORE MODEL – SEQUENCE DETECTOR "1001"

6. K-Map simplification procedures for driving expressions

Q2Q1 \ Q0X	00	01	11	10
00	0	1	X	X
01	1	1	X	X
11	X	X	X	X
10	0	1	X	X

J0 = X+Q1

Q2Q1 \ Q0X	00	01	11	10
00	X	X	0	1
01	X	X	1	1
11	X	X	X	X
10	X	X	X	X

K0 = X'+Q1

MOORE MODEL – EXAMPLE-3

STEPS TO DESIGN MOORE MODEL – SEQUENCE DETECTOR “1001”

6. K-Map simplification procedures for driving expressions

Q2Q1 \ Q0X	00	01	11	10
00	0	0	0	0
01	0	0	0	0
11	X	X	X	X
10	1	1	X	X

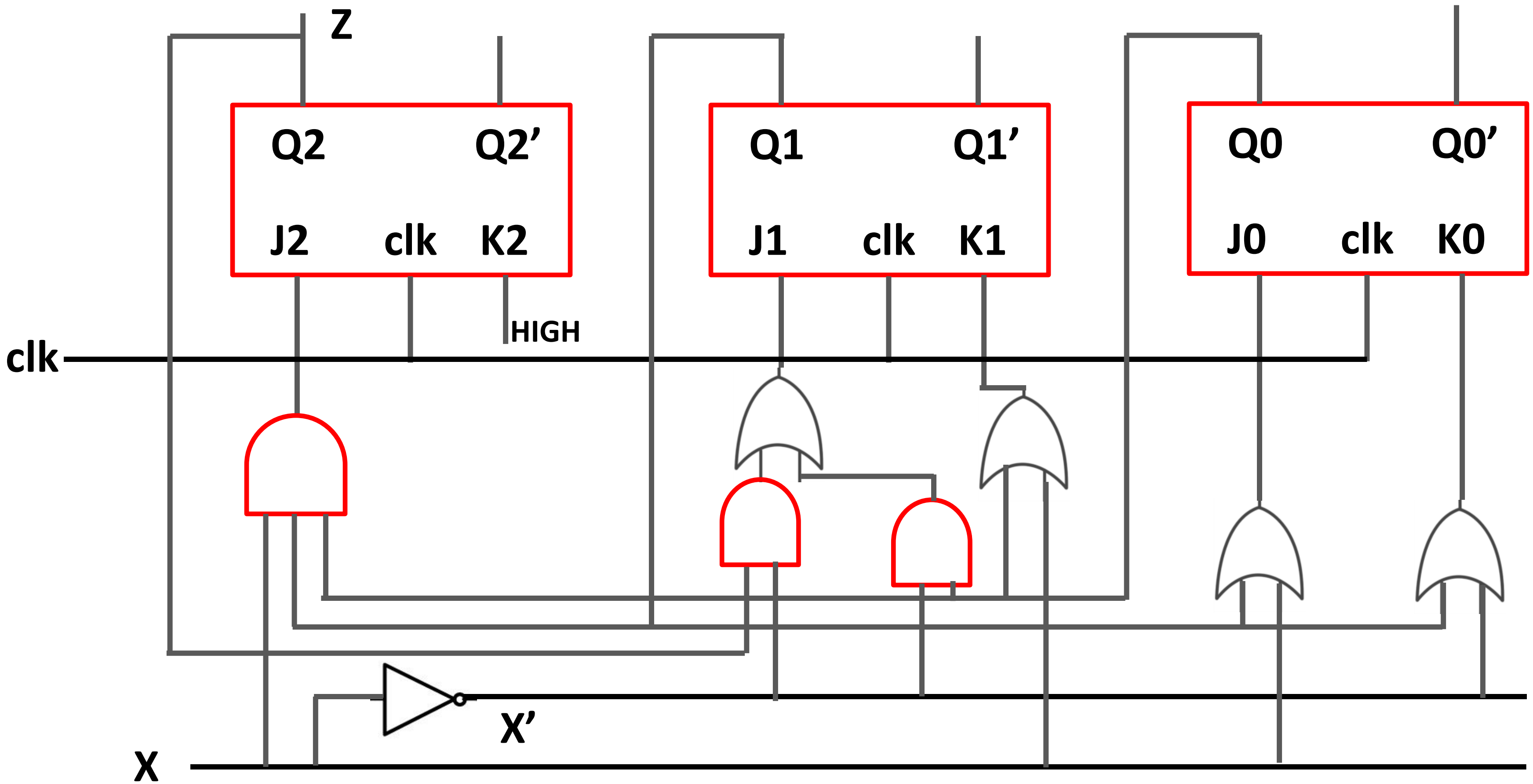
$Z = Q2$

NOTE: In Moore model, the output expression (Z) depends only on present state values (Q2) not on the input (X).

MOORE MODEL – EXAMPLE-3

STEPS TO DESIGN MOORE MODEL – SEQUENCE DETECTOR "1001"

7. Draw the logic diagram



MOORE MODEL – EXERCISE

Design an Moore based sequence detector to detect the bit pattern of “1001” in non-overlapping condition.