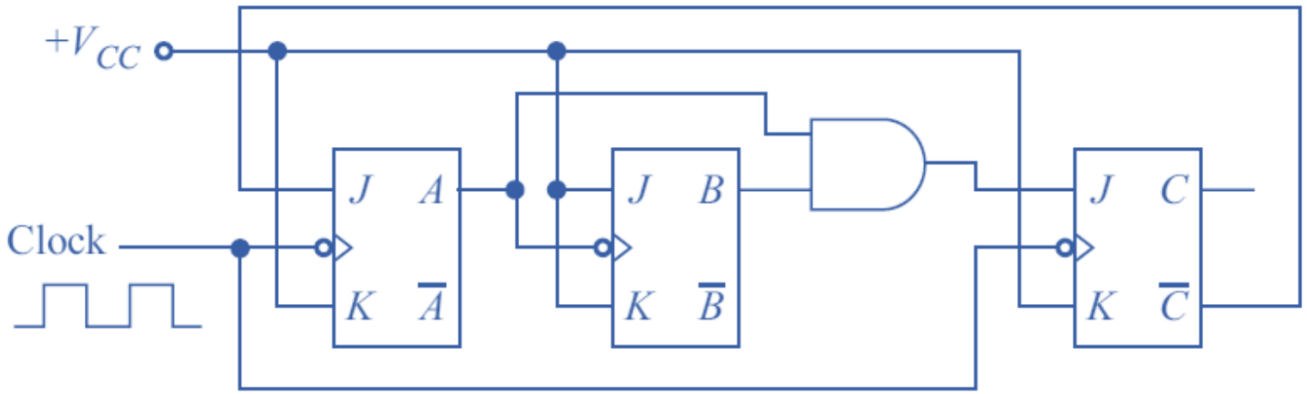
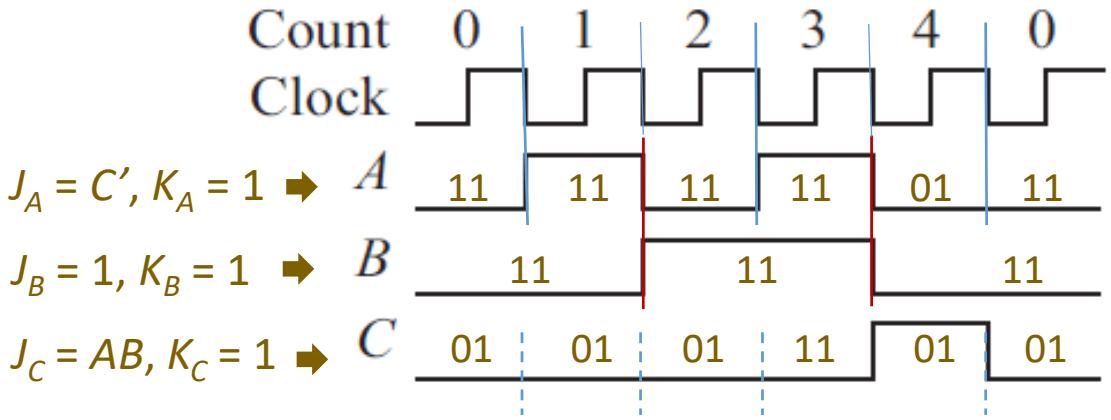
Digital Electronic Circuits
Section 1 (EE, IE)

Lecture 26

Mod 5 Counter

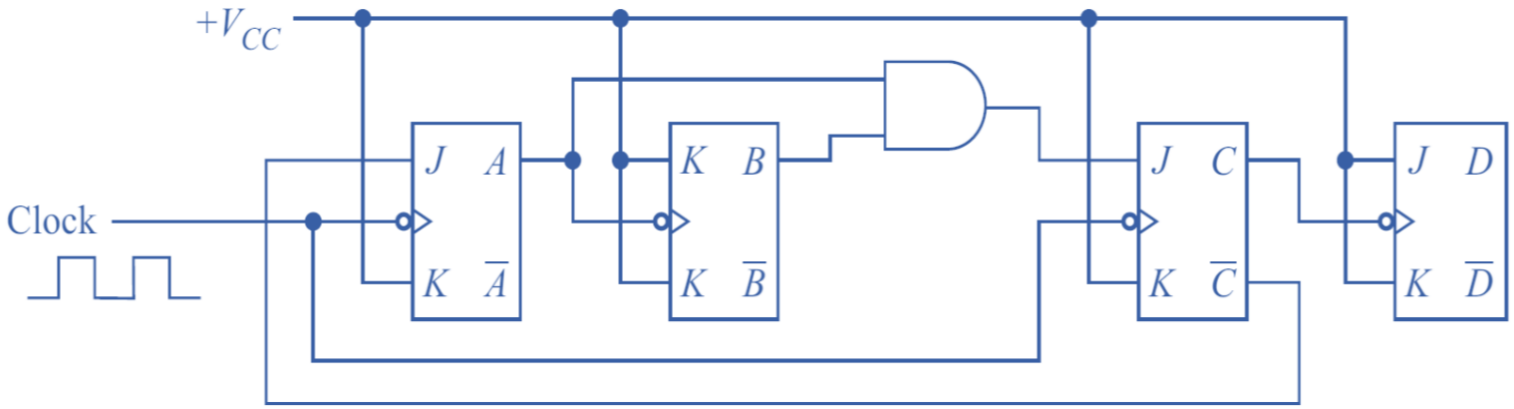


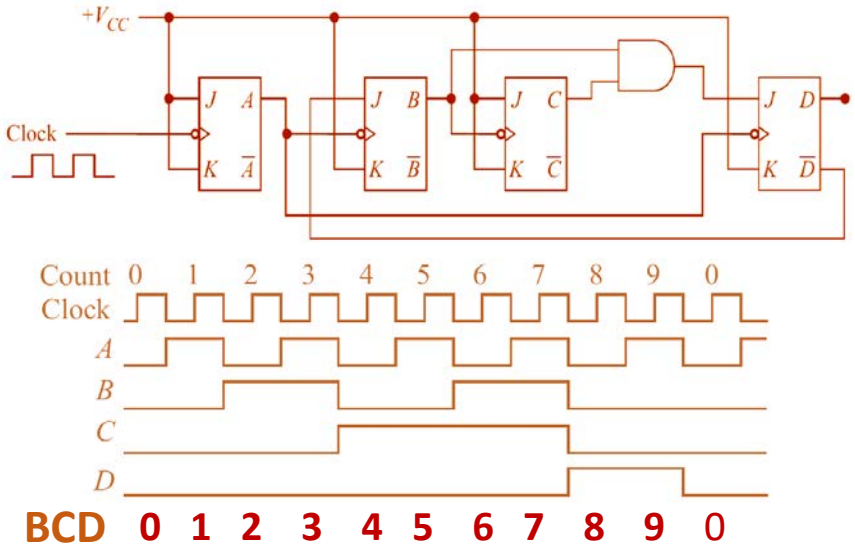
C	B	A	Count
0	0	0	0
0	0	1	1
0	1	0	2
0	1	1	3
1	0	0	4
0	0	0	0

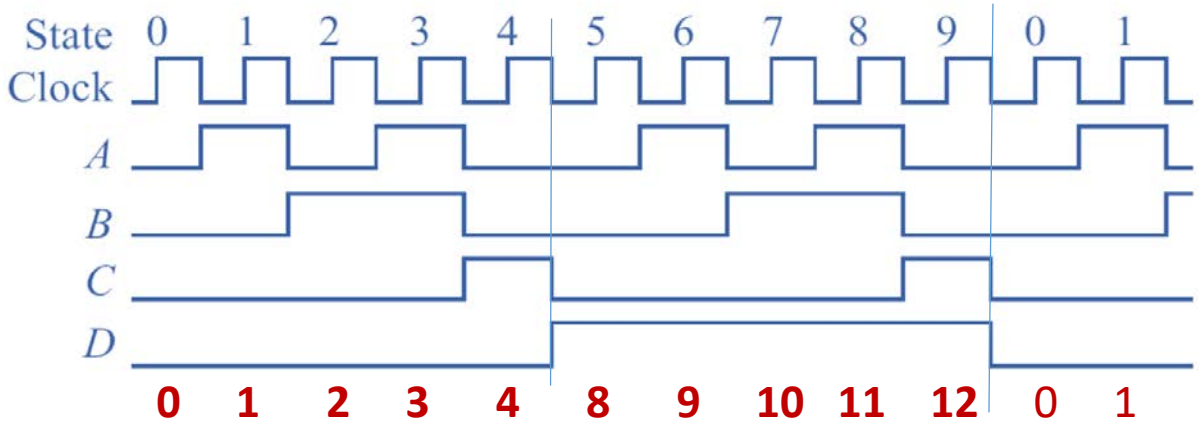


Flip-Flop *JK* inputs shown for each before clocking

Mod 10 Counter by Cascading





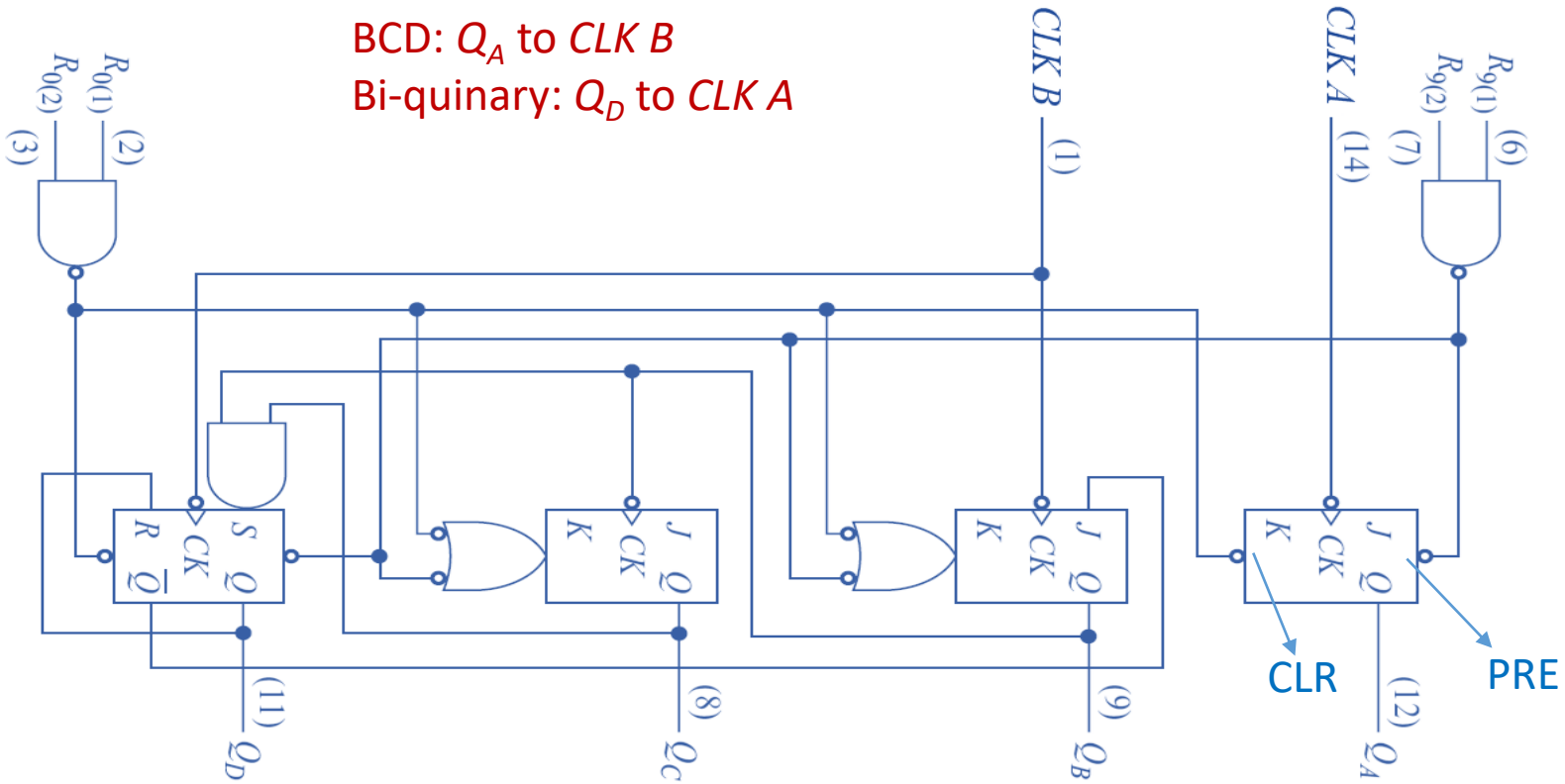


Bi-quinary (5-2)

IC 7490A

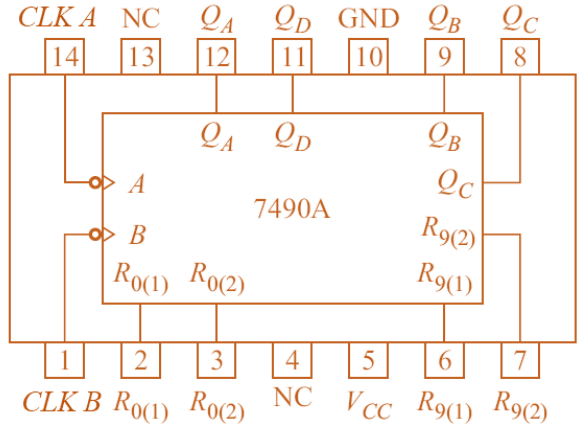
BCD: Q_A to *CLK B*

Bi-quinary: Q_D to *CLK A*

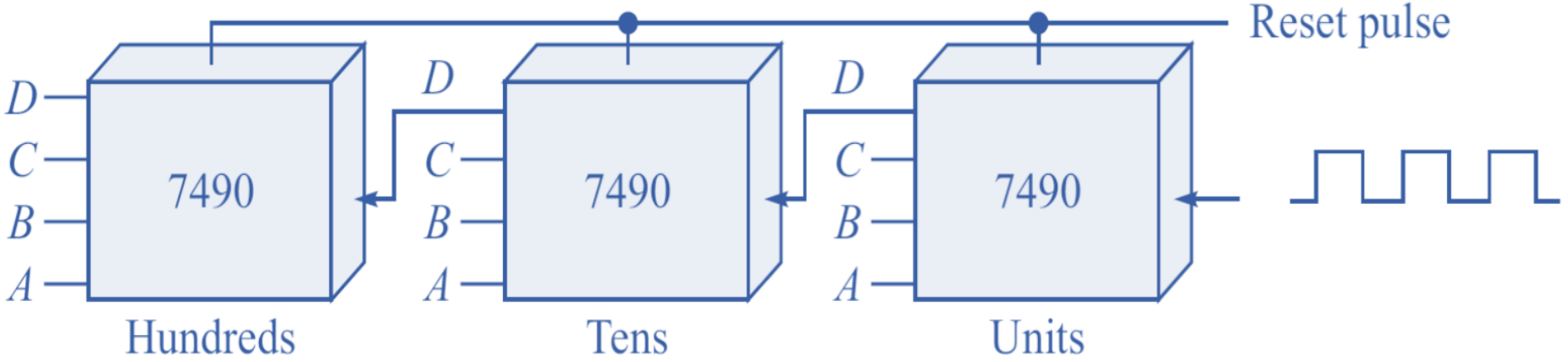


Reset Inputs				Outputs			
R0(1)	R0(2)	R9(1)	R9(2)	Q_D	Q_C	Q_B	Q_A
H	H	L	X	L	L	L	L
H	H	X	L	L	L	L	L
X	X	H	H	H	L	L	H
X	L	X	L	COUNT			
L	X	L	X	COUNT			
L	X	X	L	COUNT			
X	L	L	X	COUNT			

Mod 1000 Counter by Cascading



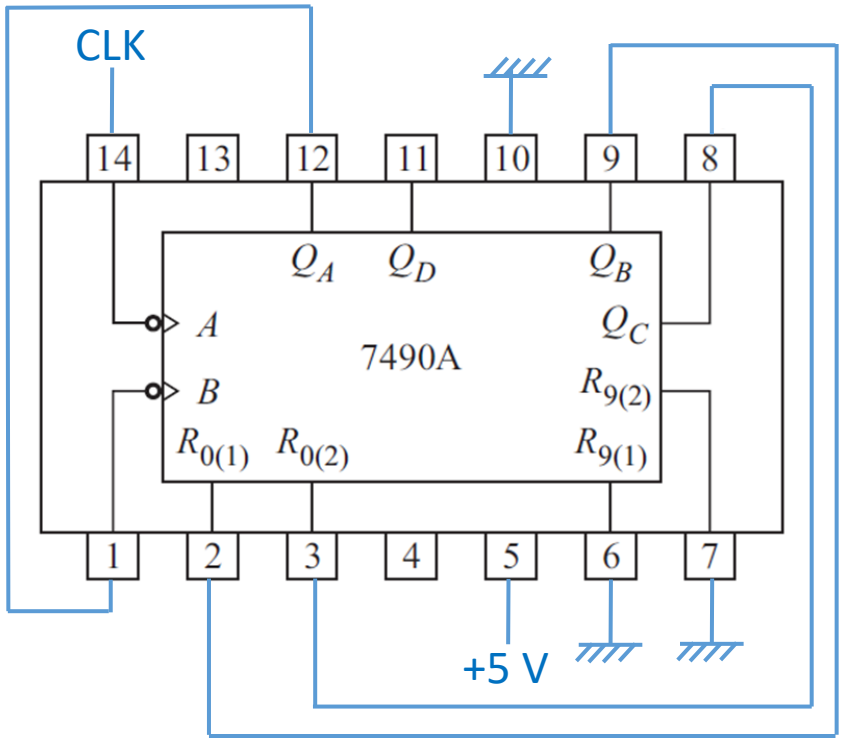
Divided by 10 by cascading ÷5 and ÷2



Counts from 0 to 999

Use of Asynchronous Reset

Clocking for BCD Counter



Clock	$Q_DQ_CQ_BQ_A$	Count
0	0000	0
1	0001	1
2	0010	2
3	0011	3
4	0100	4
5	0101	5
6	(0110) 0000	(6) 0
7	0001	1

Mod 6

Glitch: For the duration of internal AND gate propagation delay << T_{Clock}

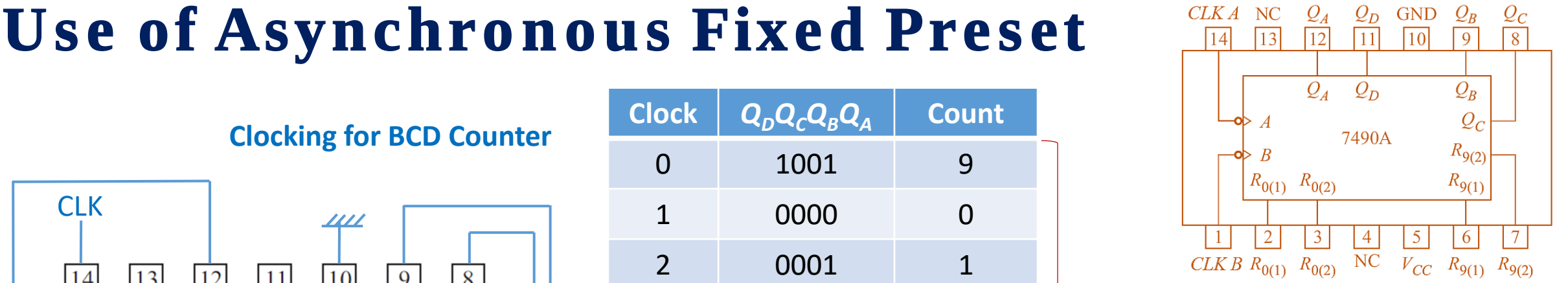
Reset Inputs				Outputs			
R0(1)	R0(2)	R9(1)	R9(2)	Q_D	Q_C	Q_B	Q_A
H	H	L	X	L	L	L	L
H	H	X	L	L	L	L	L
X	X	H	H	H	L	L	H
X	L	X	L	COUNT			
L	X	L	X	COUNT			
L	X	X	L	COUNT			
X	L	L	X	COUNT			

Mod 9: If Q_D and Q_A to $R_{0(1)}$ and $R_{0(2)}$

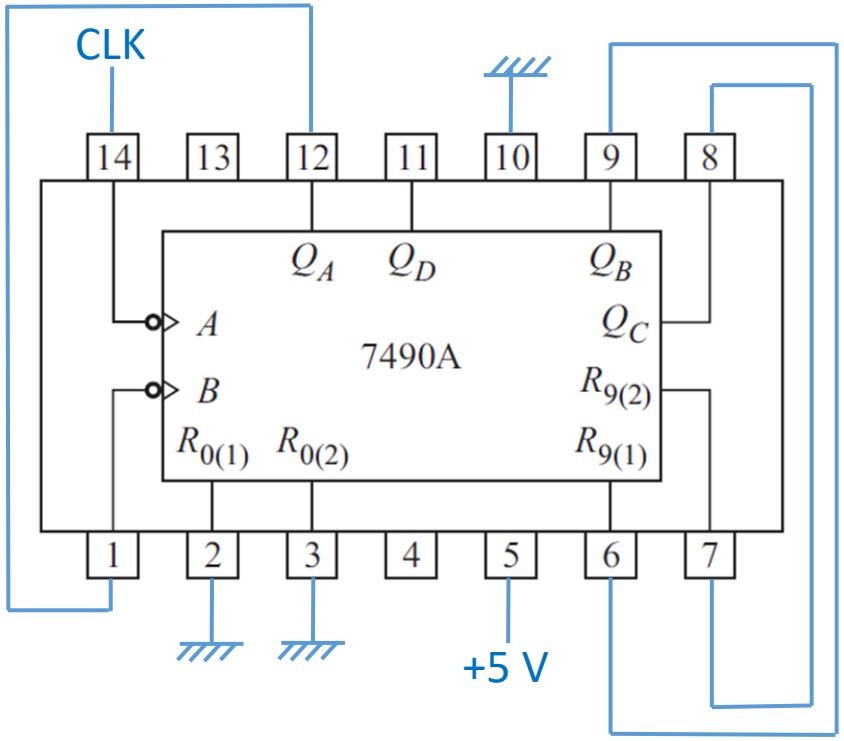
Mod 8: If Q_D to both $R_{0(1)}$ and $R_{0(2)}$

Mod 7: Since $Q_CQ_BQ_A$ = 111 to reset, 3-input gate (external)

Use of Asynchronous Fixed Preset



Clocking for BCD Counter



Clock	$Q_DQ_CQ_BQ_A$	Count
0	1001	9
1	0000	0
2	0001	1
3	0010	2
4	0011	3
5	0100	4
6	0101	5
7	(0110) 1001	(6) 9
8	0000	0
9	0001	1

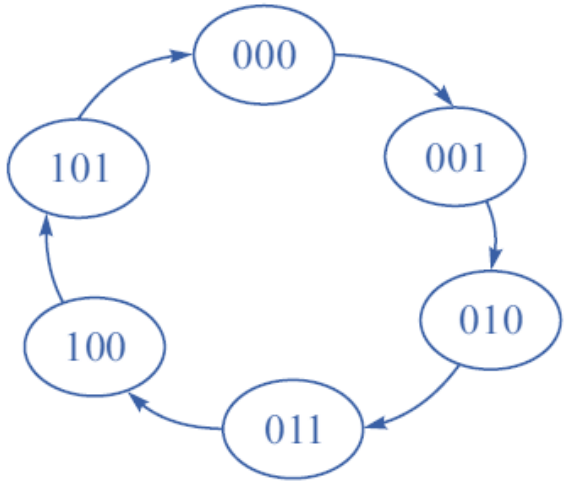
Mod 7

For the duration of internal AND gate propagation delay << T_{Clock}

Synchronous Counter Design

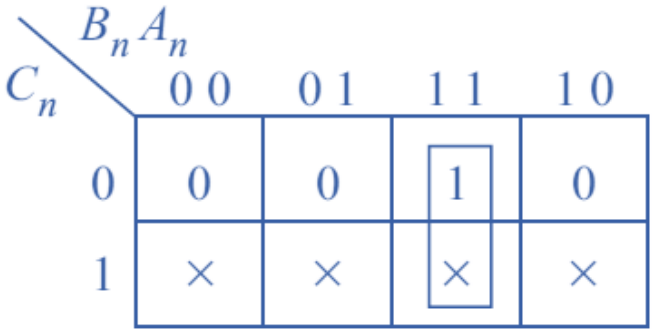
Considered: *JK* Flip-Flop is used and the counter is initialized with one of the valid six states.

$Q_n \rightarrow Q_{n+1}$		J	K
0	0	0	×
0	1	1	×
1	0	×	1
1	1	×	0

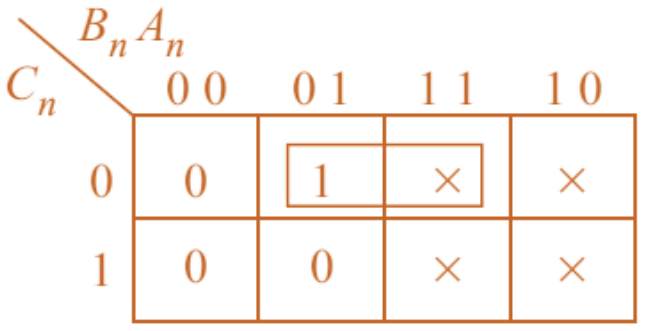


C_n	B_n	A_n	C_{n+1}	B_{n+1}	A_{n+1}	J_C	K_C	J_B	K_B	J_A	K_A
0	0	0	0	0	1	0	×	0	×	1	×
0	0	1	0	1	0	0	×	1	×	×	1
0	1	0	0	1	1	0	×	×	0	1	×
0	1	1	1	0	0	1	×	×	1	×	1
1	0	0	1	0	1	×	0	0	×	1	×
1	0	1	0	0	0	×	1	0	×	×	1

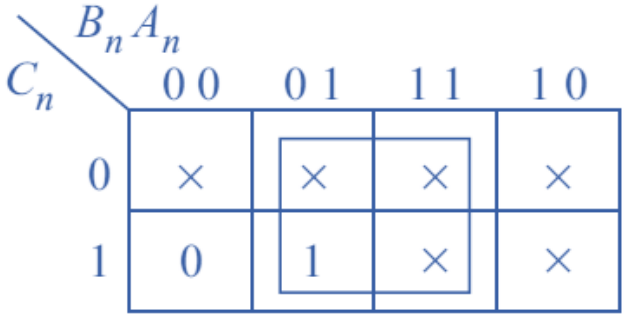
Design Equations from K-Map



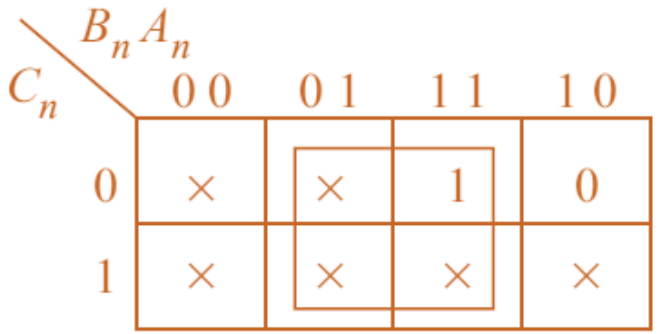
$J_C = B_n A_n$



$J_B = \overline{C}_n A_n$



$K_C = A_n$



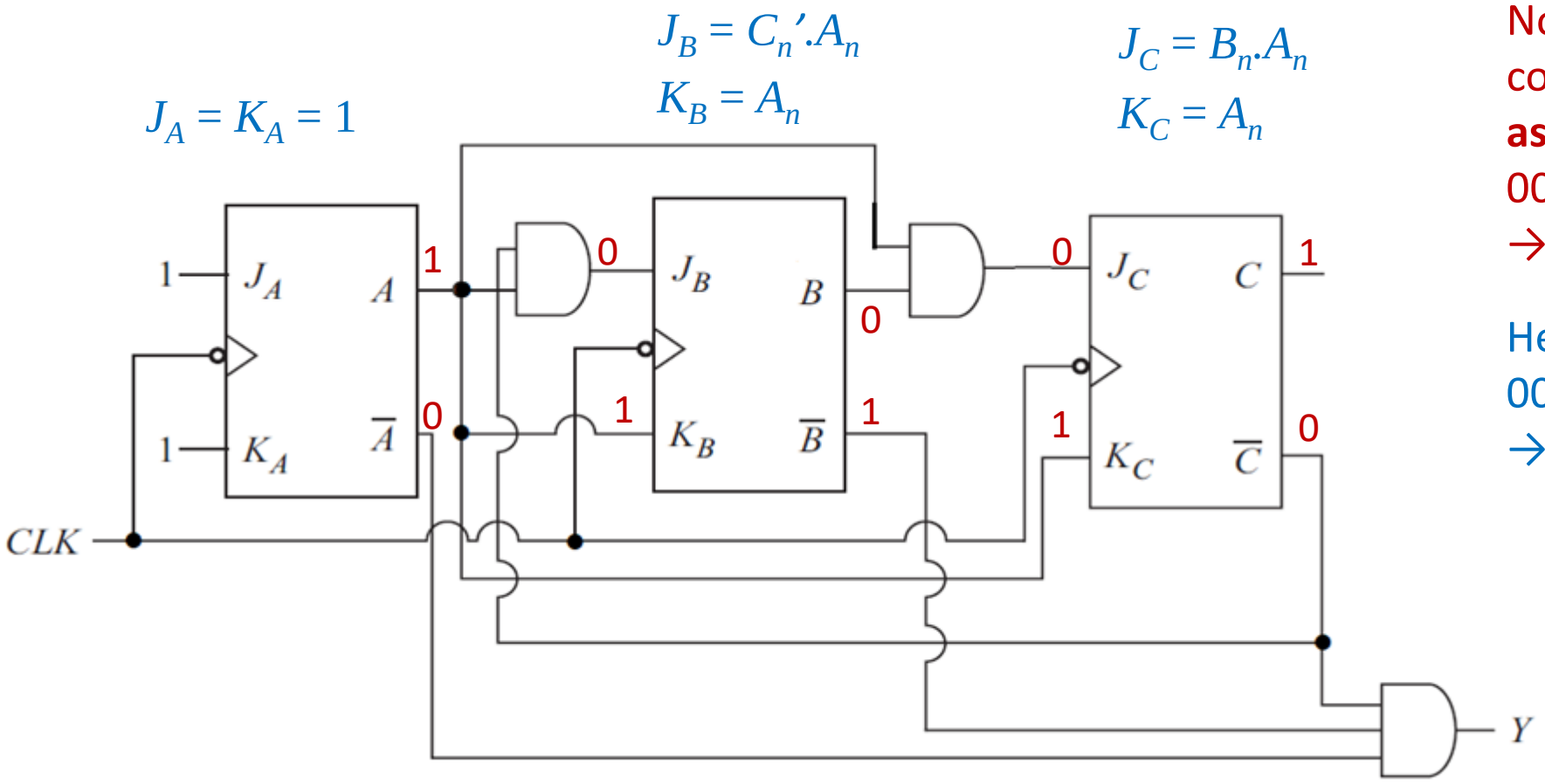
$K_B = A_n$

• • •

J_A	K_A
1	×
×	1
1	×
×	1
1	×
×	1

$J_A = K_A = 1$

Logic Circuit

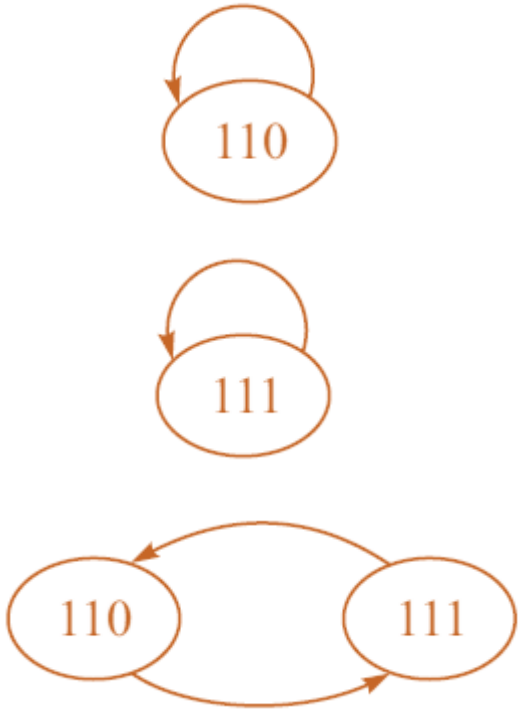


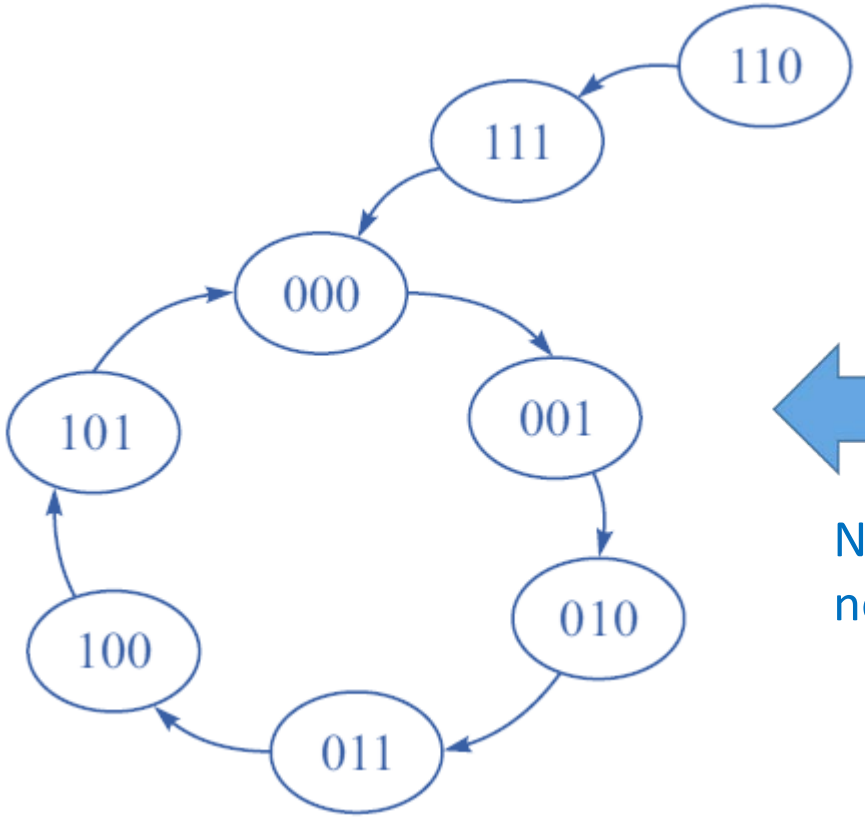
No glitch as was in Mod 6 counter realization using **asynchronous reset**: *CBA*
000→001→010→011→100 →101→**(110)**000

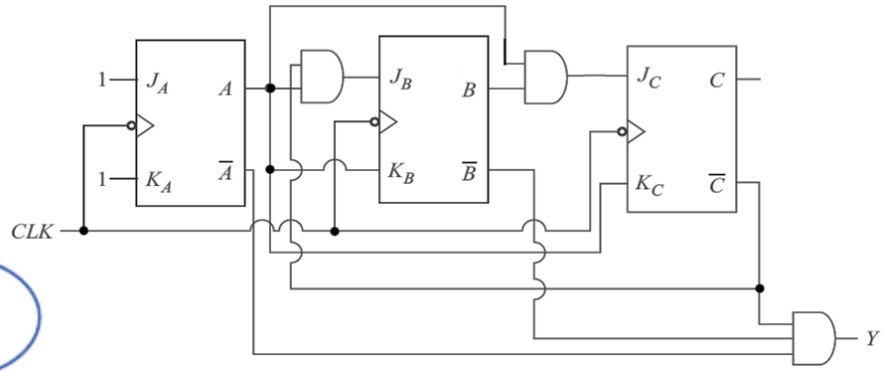
Here:
000→001→010→011 →100→101→000

Role of Invalid States

Possibility of lock-in



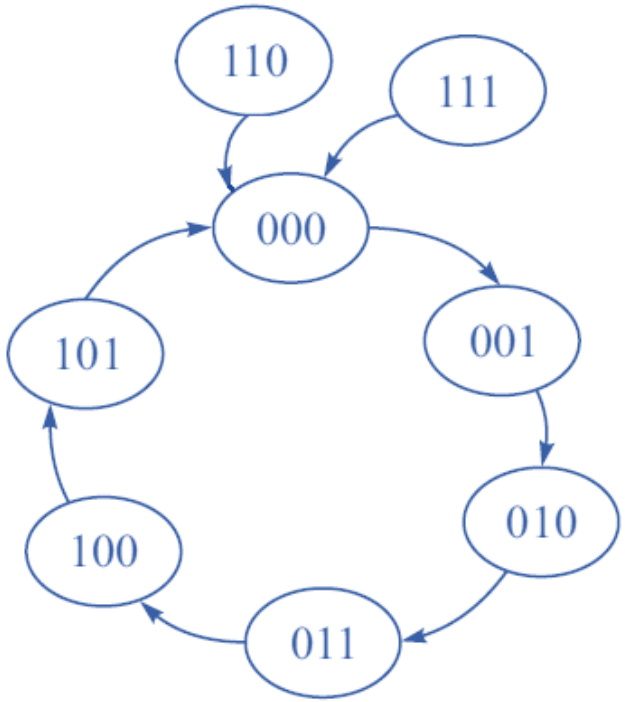




No lock-in but,
not by design

Consideration of Unused States

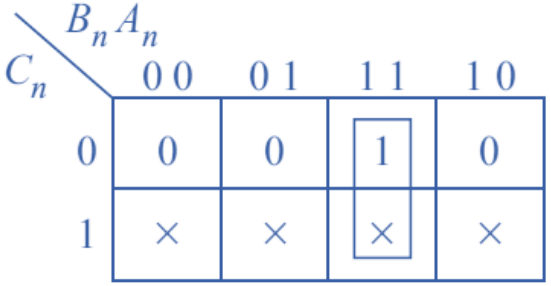
To come from invalid state(s) to valid state(s) and to continue with the counting.



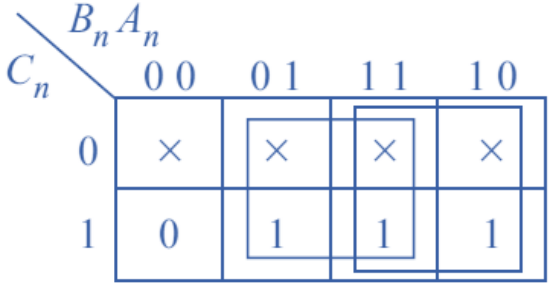
Example:

C_n	B_n	A_n	C_{n+1}	B_{n+1}	A_{n+1}	J_C	K_C	J_B	K_B	J_C	K_C
1	1	0	0	0	0	×	1	×	1	0	×
1	1	1	0	0	0	×	1	×	1	×	1

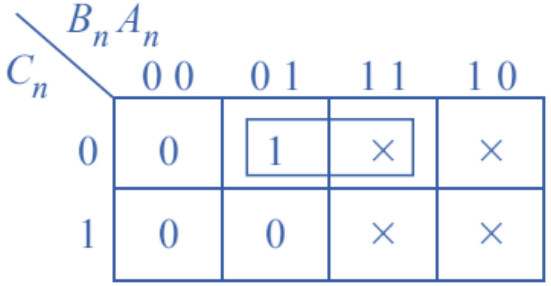
Design Equations from K-Map



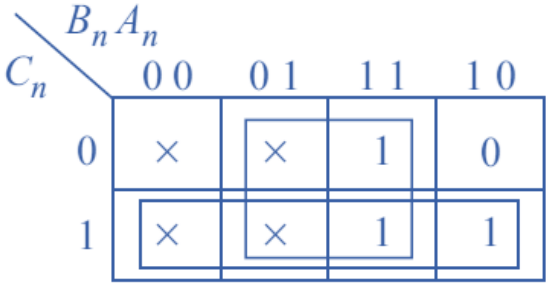
$J_C = B_n A_n$



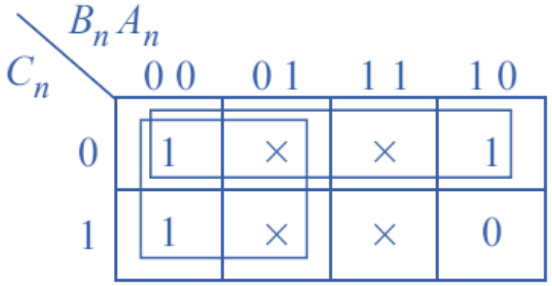
$K_C = A_n + B_n$



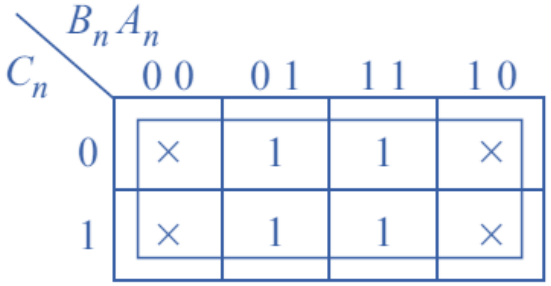
$J_B = \overline{C}_n A_n$



$K_B = A_n + C_n$

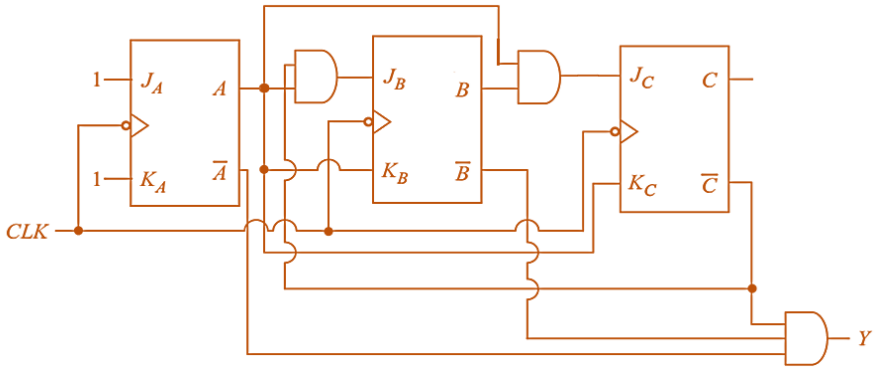


$J_A = \overline{C}_n + \overline{B}_n$

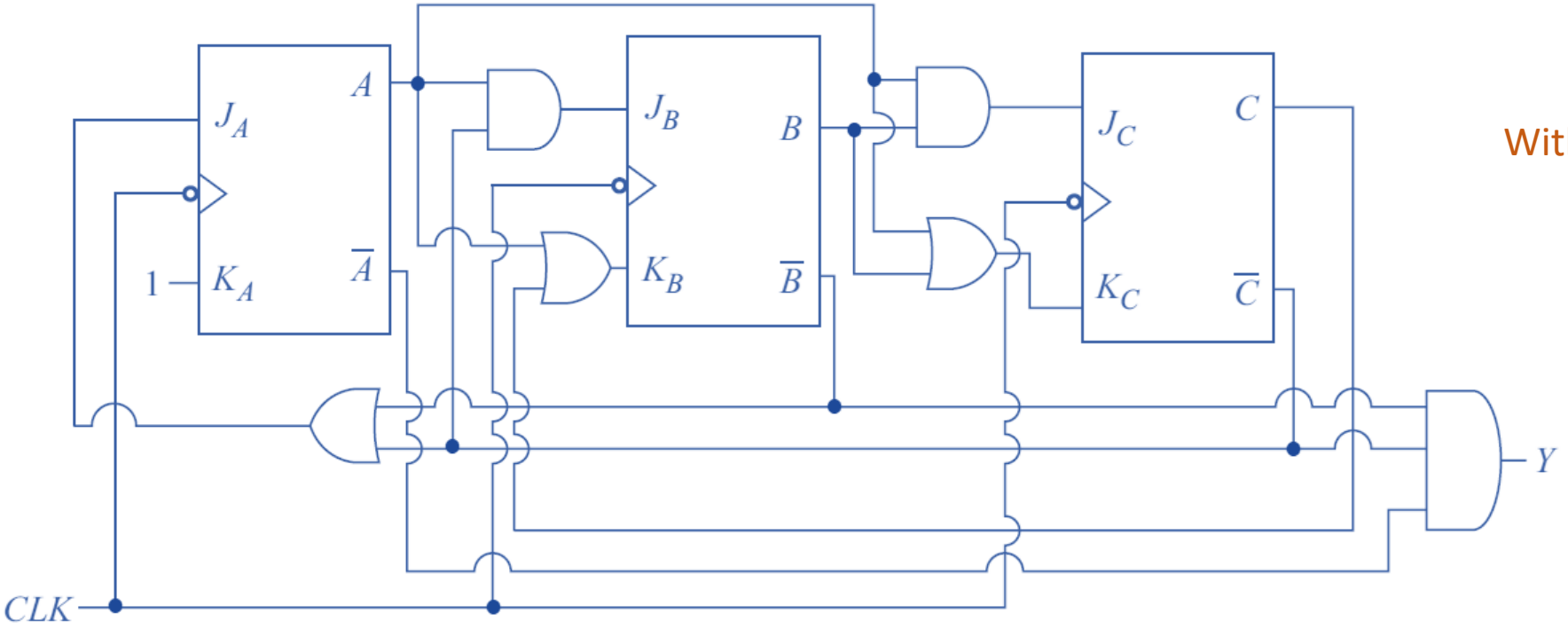


$K_A = 1$

Logic Circuit



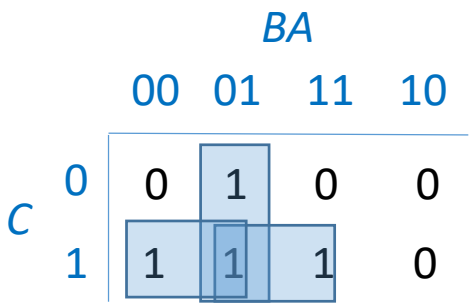
Without considering invalid states



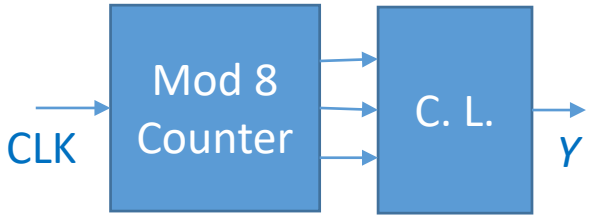
Considering both the invalid states going to valid 000 at next clock trigger

Some Applications

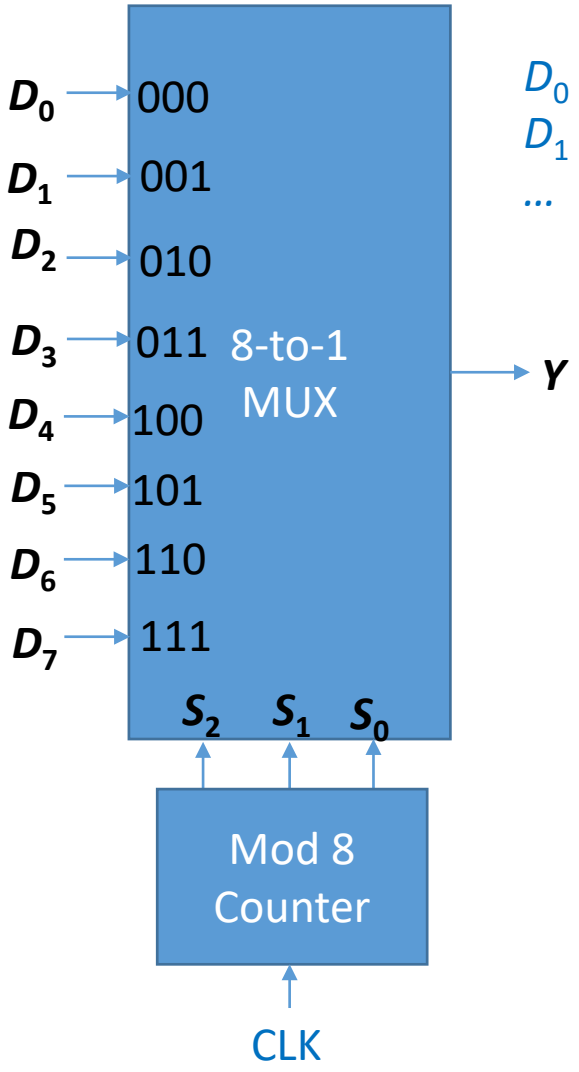
C	B	A	Y
0	0	0	0
0	0	1	1
0	1	0	0
0	1	1	0
1	0	0	1
1	0	1	1
1	1	0	0
1	1	1	1
0	0	0	0



$Y = B'.A + C.B' + C.A$ ⇐ C.L.



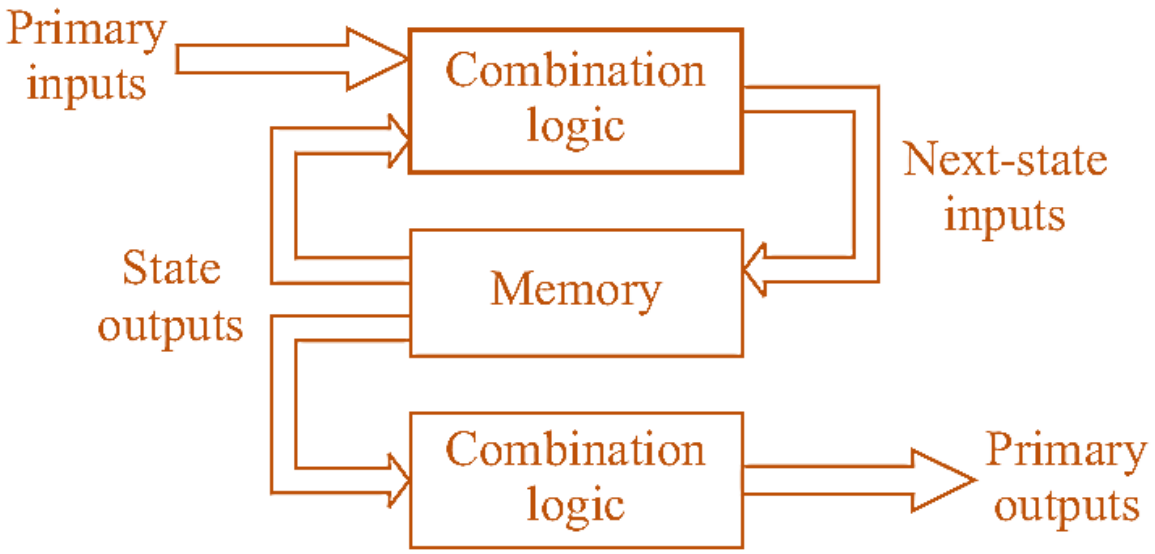
Sequence Generator

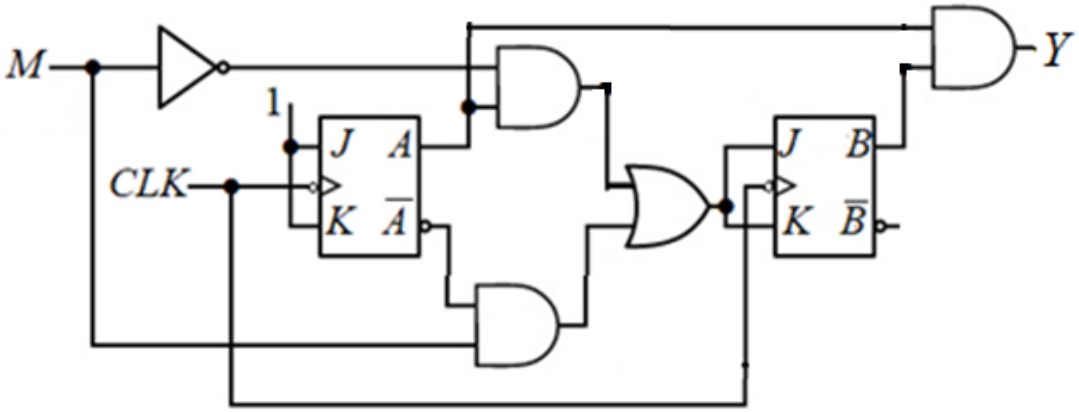


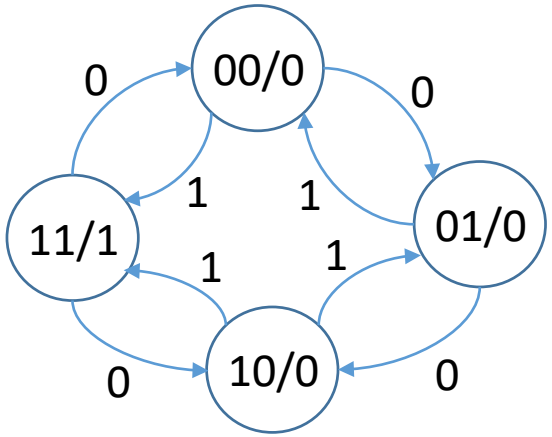
D_0 in 0th, 8th, 16th … clock cycle
D_1 in 1st, 9th, 17th … clock cycle
…

Parallel to Serial:
Time Multiplexing

Moore Model

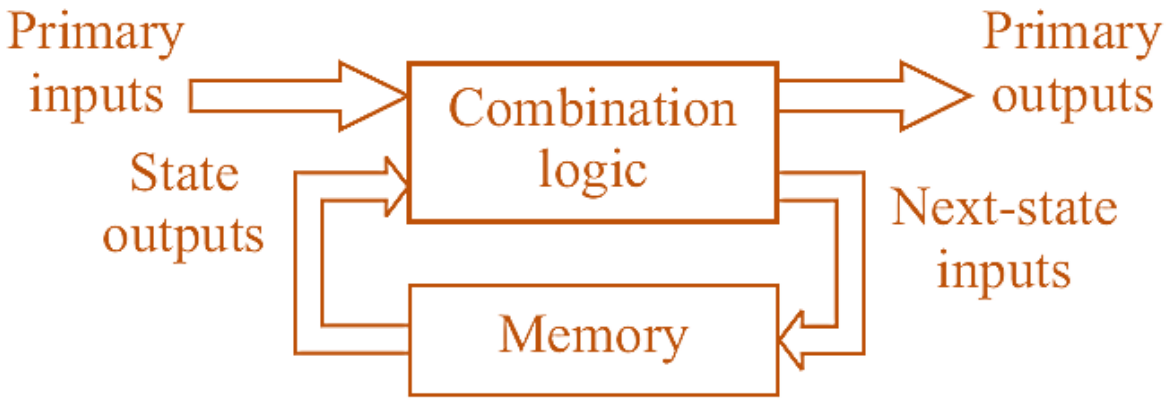


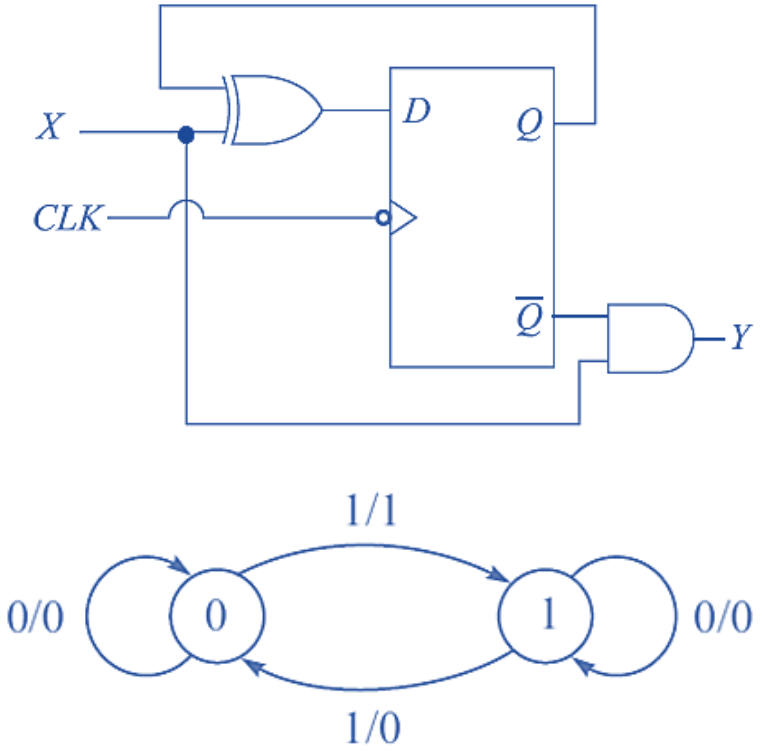




- Input effects internal state and not output directly.
- Output is generated solely from flip-flops / registers.
- Output is synchronized with clock.

Mealy Model



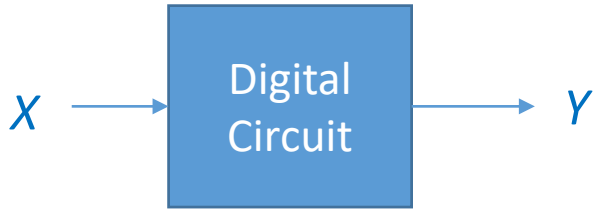


- Input effects internal state and output directly.
- Output is generated from flip-flops / registers and input.
- Output is not synchronized with clock, may change if the input changes during a clock period.
- Input transients / glitches may affect the output

Synthesis with Moore Model

Problem Statement:

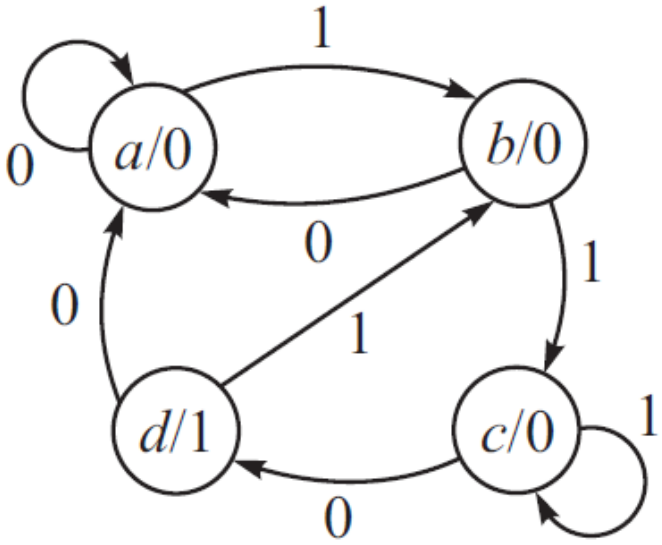
A sequence detector for '110' from a binary data stream is to be designed.



Y is 1 if in the binary data stream, *X* a sequence 110 is detected else, *Y* is 0.

State definition:

a: No bit is correctly decoded (initial state)
b: 1 bit is correctly decoded
c: 2 bits are correctly decoded
d: 3 bits are correctly decoded



State transition diagram

Example:

CLK	0	1	2	3	4	5	6	7	8	9	10
Input	0	1	0	1	1	1	0	1	1	0	0
State	*a*	*a*	*b*	*a*	*b*	*c*	*c*	*d*	*b*	*c*	*d*
Output	0	0	0	0	0	0	0	1	0	0	1

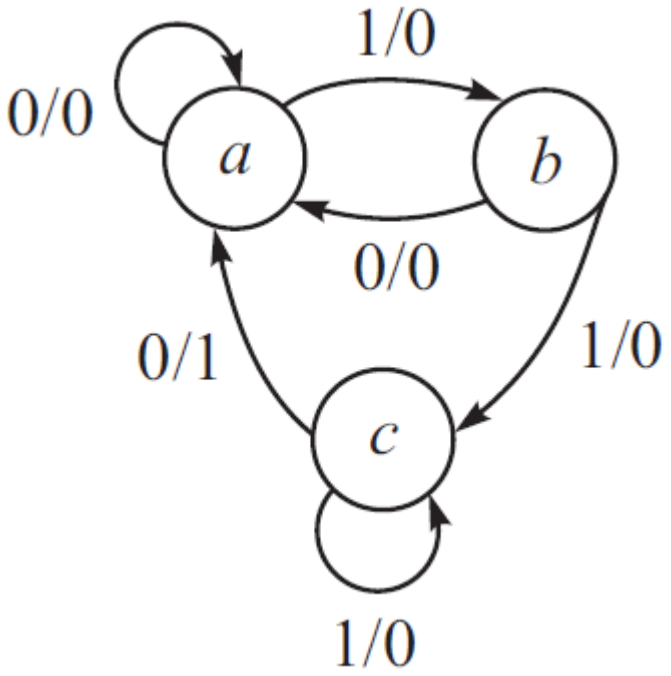
Synthesis with Mealy Model

State definition:

a: No bit is correctly decoded (initial state)
b: 1 bit is correctly decoded
c: 2 bits are correctly decoded

Example:

CLK	0	1	2	3	4	5	6	7	8	9	10
Input	0	1	0	1	1	1	0	1	1	0	0
State	*a*	*a*	*b*	*a*	*b*	*c*	*c*	*a*	*b*	*c*	*a*
Output	0	0	0	0	0	0	1	0	0	1	0

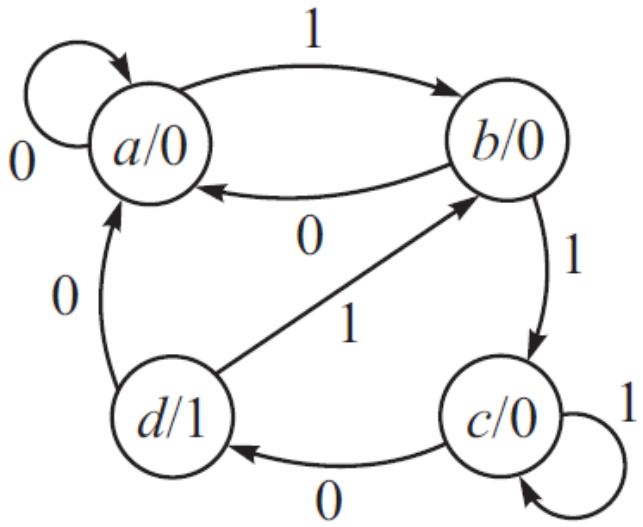


State transition diagram

Problem Statement:

A sequence detector for '110' from a binary data stream is to be designed.

Moore Model: State Assignment



State transition diagram

State assignment:

For *N* states, number of flip-flops required = $\lceil \log_2 N \rceil$ i.e. *ceiling*($\log_2 N$)

Flip-Flop

State	B	A
a	0	0
b	0	1
c	1	0
d	1	1

Alternate assignment (one FF value changes!!)

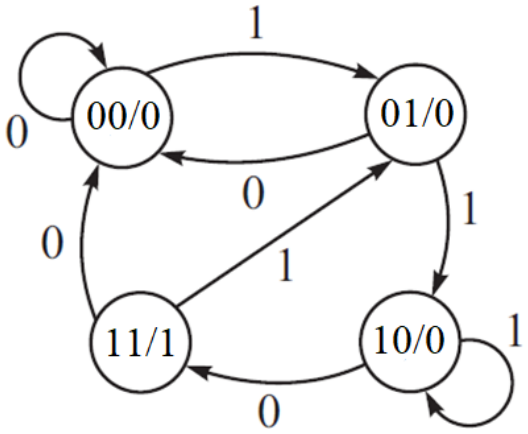
State	B	A
a	0	0
b	0	1
c	1	1
d	1	0

Problem Statement:

A sequence detector for '110' from a binary data stream is to be designed.

Moore Model: State Table

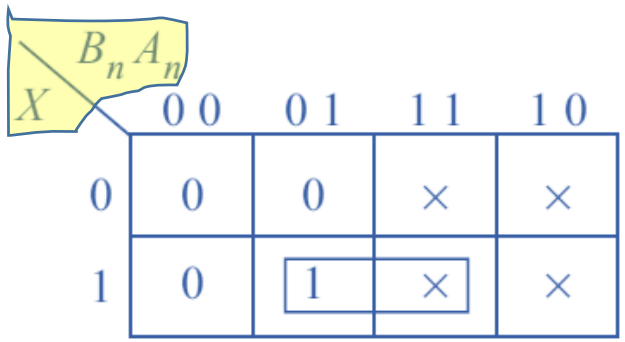
Present State		Present Input	Next State		Output				
B_n	A_n	X_n	B_{n+1}	A_{n+1}	Y_n	J_B	K_B	J_A	K_A
0	0	0	0	0	0	0	×	0	×
0	0	1	0	1	0	0	×	1	×
0	1	0	0	0	0	0	×	×	1
0	1	1	1	0	0	1	×	×	1
1	0	0	1	1	0	×	0	1	×
1	0	1	1	0	0	×	0	0	×
1	1	0	0	0	1	×	1	×	1
1	1	1	0	1	1	×	1	×	0



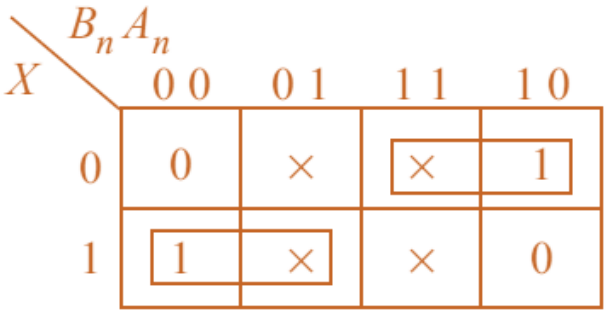
$Q_n \rightarrow Q_{n+1}$		J	K
0	0	0	×
0	1	1	×
1	0	×	1
1	1	×	0

State table

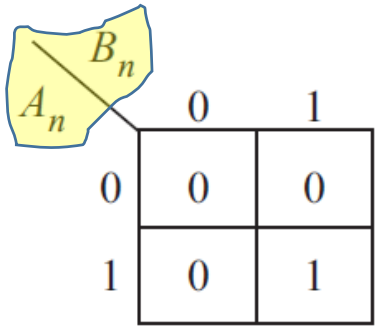
Moore Model: Design Equations



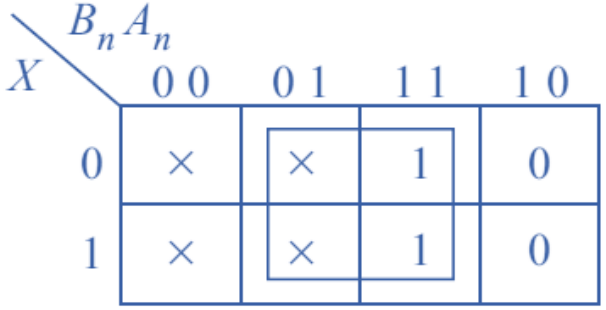
$J_B = X A_n$



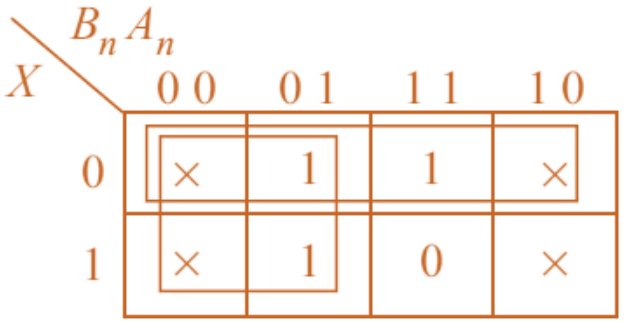
$J_A = \overline{X} B_n + X \overline{B}_n$



$Y = A_n B_n$



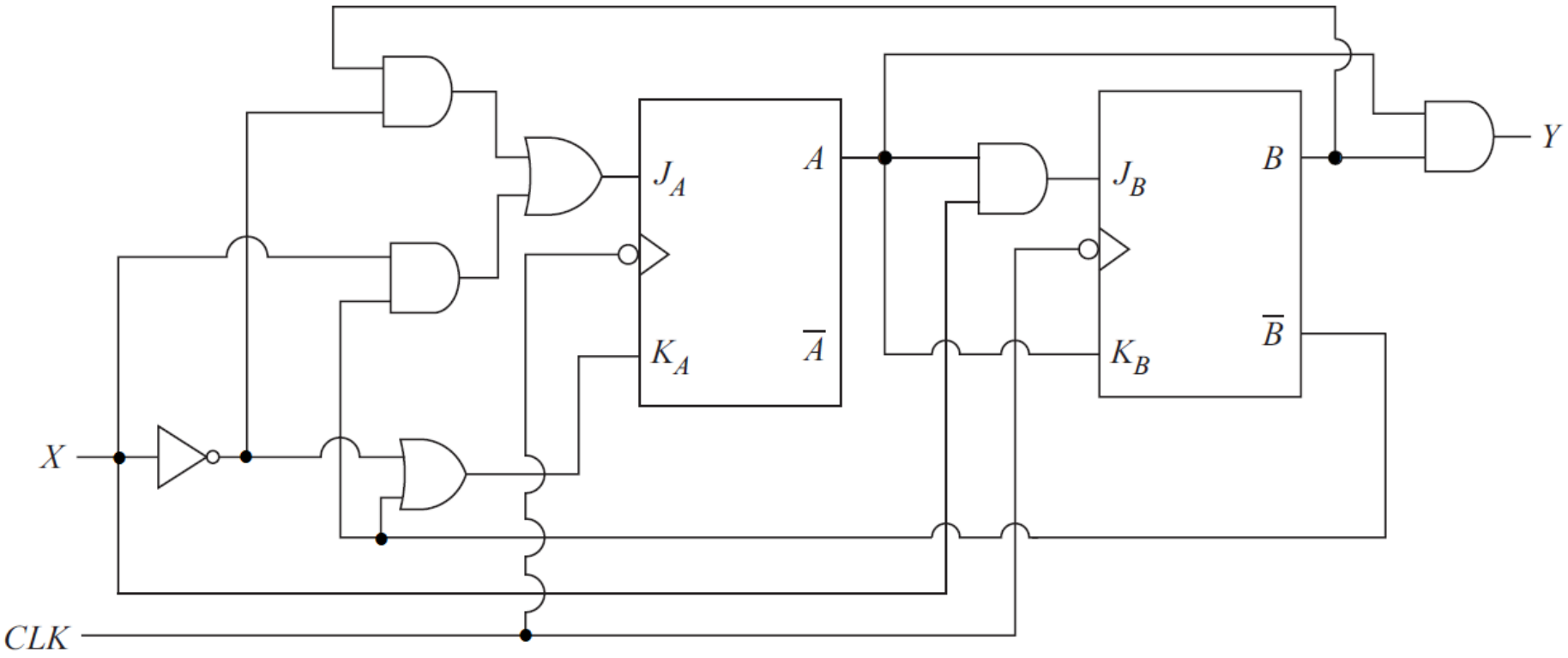
$K_B = A_n$



$K_A = \overline{X} + \overline{B}_n$

Design Equations

Moore Model: Circuit Realization



$$J_B = X A_n$$

$$K_B = A_n$$

$$J_A = \overline{X} B_n + X \overline{B}_n$$

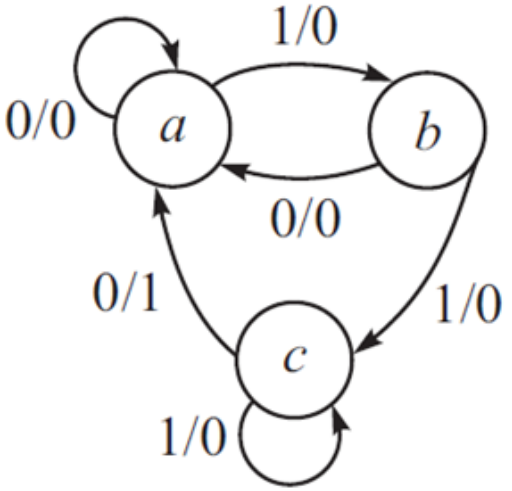
$$K_A = \overline{X} + \overline{B}_n$$

$$Y = A_n B_n$$

Mealy Model: State Assignment & State Table

Continuing with the same design problem: A sequence detector for '110' from a binary data stream is to be designed.
Required, $\lceil \log_2 3 \rceil = \lceil 1.585 \rceil = 2$ Flip-Flops

State	B	A
a	0	0
b	0	1
c	1	0

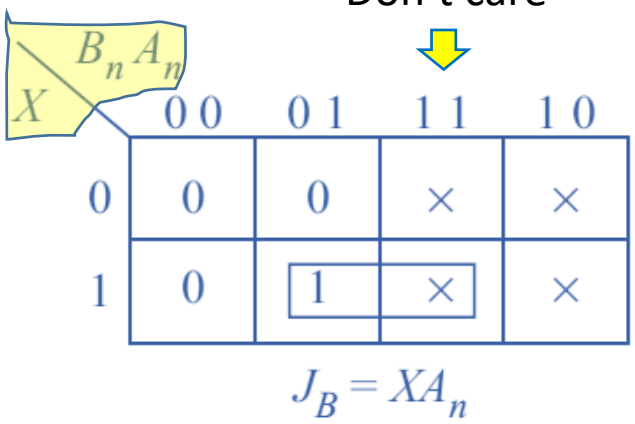


Present State		Present Input	Next State		Present Output				
B_n	A_n	X_n	B_{n+1}	A_{n+1}	Y_n	J_B	K_B	J_A	K_A
0	0	0	0	0	0	0	×	0	×
0	0	1	0	1	0	0	×	1	×
0	1	0	0	0	0	0	×	×	1
0	1	1	1	0	0	1	×	×	1
1	0	0	0	0	1	×	1	0	×
1	0	1	1	0	0	×	0	0	×

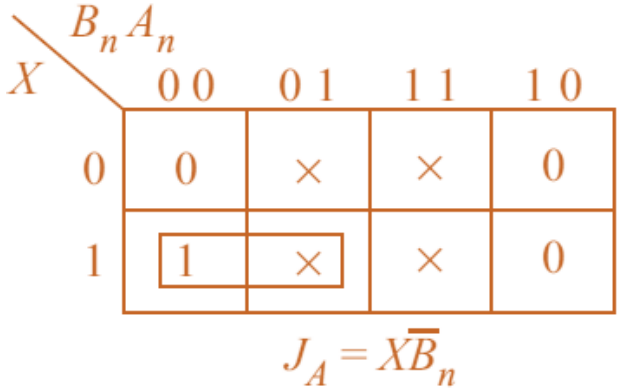
$Q_n \rightarrow Q_{n+1}$		J	K
0	0	0	×
0	1	1	×
1	0	×	1
1	1	×	0

Mealy Model: Design Equations

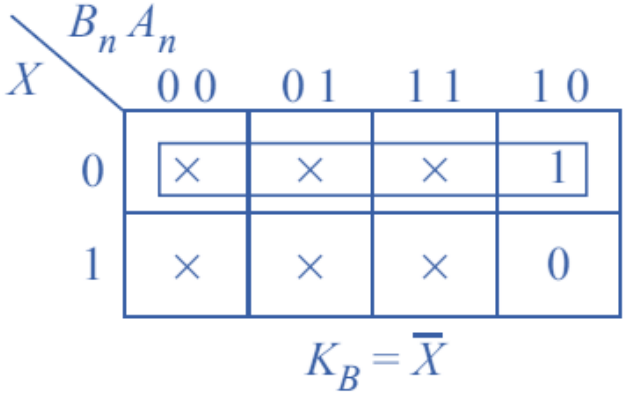
Don't care



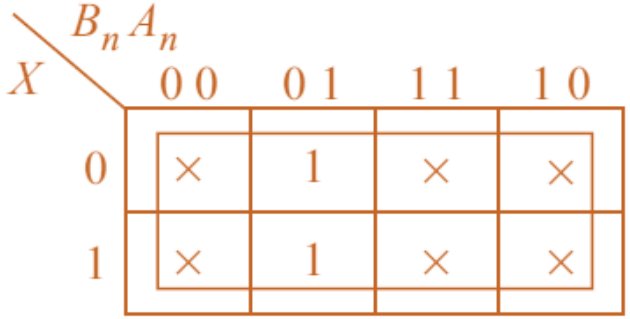
$J_B = XA_n$



$J_A = X\overline{B}_n$

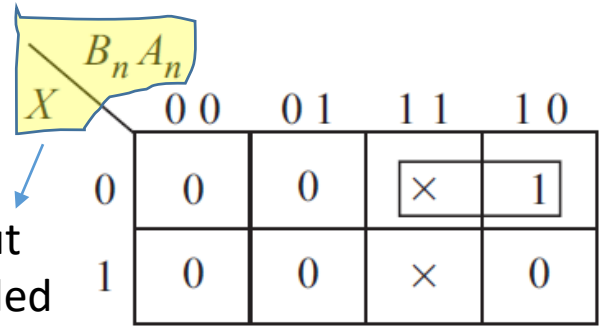


$K_B = \overline{X}$



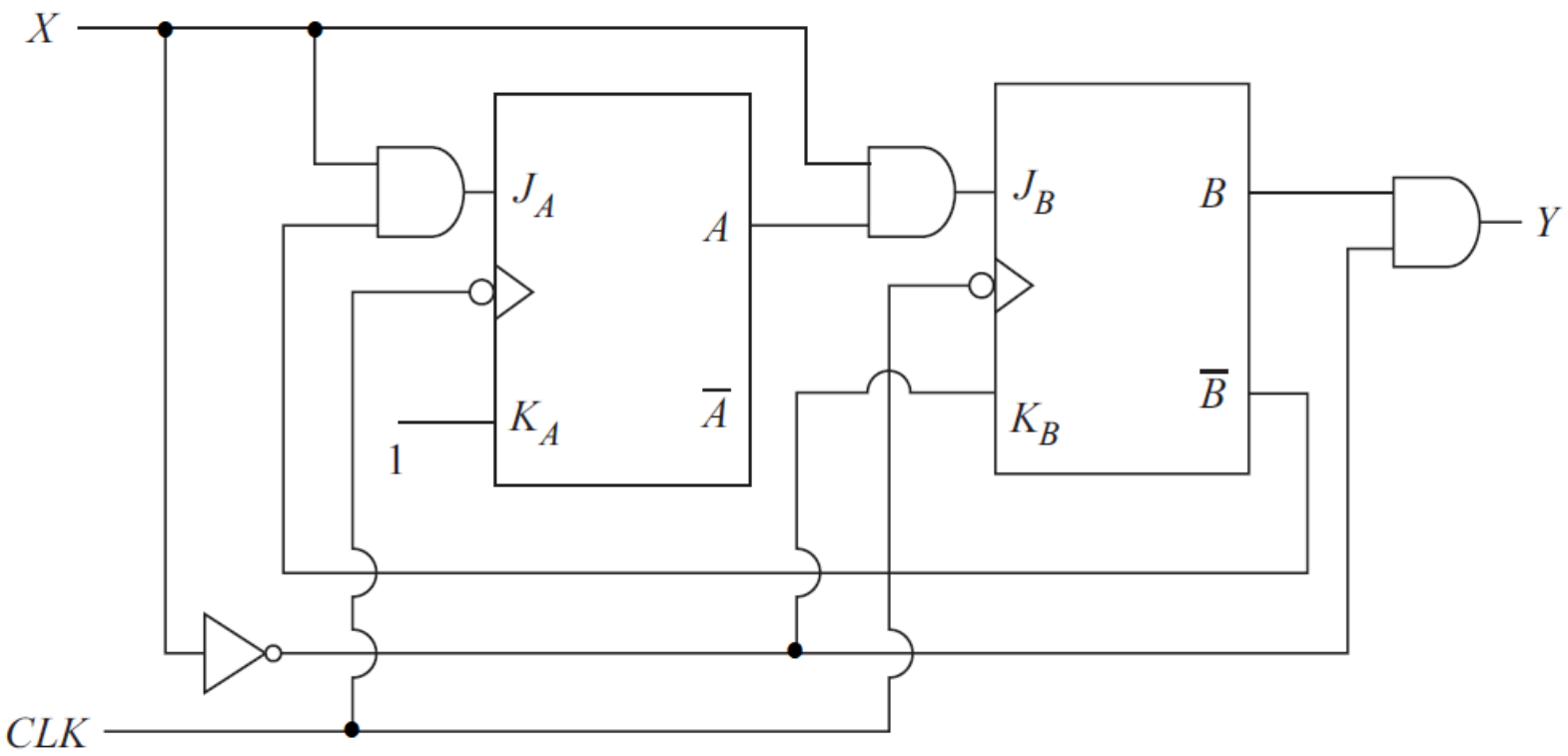
$K_A = 1$

Input included



$Y = \overline{X}B_n$

Mealy Model: Circuit Realization



$$J_B = XA_n$$
$$K_B = \overline{X}$$
$$J_A = X\overline{B}_n$$
$$K_A = 1$$
$$Y = \overline{X}B_n$$

Less complex compared to Moore Model based circuit.

References:

- Donald P. Leach, Albert P. Malvino, and Goutam Saha, Digital Principles & Applications 8e, McGraw Hill
- Texas Instrument's Digital Logic Pocket Data Book (2007)