Department of Electrical & Computer Engineering

COEN 212:
DIGITAL SYSTEMS DESIGN I
Lecture 10: Sequential Circuits
Analysis and Design

Instructor: Dr. Reza Soleymani, Office: EV-5.125,
Telephone: 848-2424 ext.: 4103.

Lecture 10:
Objectives of this lecture

Concordia University
Department of Electrical & Computer Engineering

- In this lecture, we talk about:
  - Analysis of the Sequential Circuits.
  - Design of the Sequential Circuits.

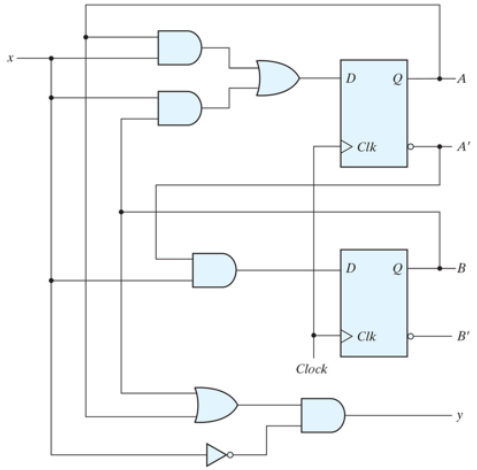
Lecture 10:
Reading for this lecture

- Digital Design by M. Morris R. Mano and Michael D. Ciletti, 6th Edition, Pearson, 2018:
  - Chapter 5 (5.5, 5.7 and 5.8)

Lecture 10:
Analysis of Sequential Circuits:

Concordia University
Department of Electrical & Computer Engineering

- **Analysis:** to describe how a circuit works
- **Example:**



- State equations for this circuit are

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

$$B(t+1)=A'(t)x(t)$$

- and the output $y(t)=[A(t)+B(t)]x'(t)$
- The state-table or state-transition table will have 8 entries, since there are 2 flip-flops resulting in $2^2 = 4$ states and one input resulting in two possibilities.

Lecture 10:
Analysis of Sequential Circuits:

Université Concordia University
Department of Electrical & Computer Engineering

- State-transition table:

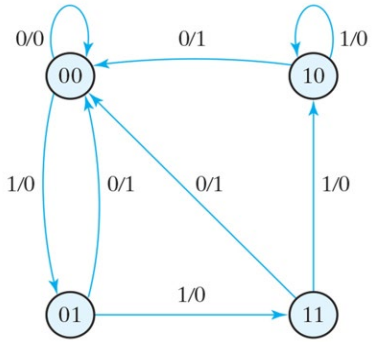
Present state		Input	Next state		Output
$A(t)$	$B(t)$	$x(t)$	$A(t+1)$	$B(t+1)$	$y(t)$
0	0	0	0	0	0
0	0	1	0	1	0
0	1	0	0	0	1
0	1	1	1	1	0
1	0	0	0	0	1
1	0	1	1	0	0
1	1	0	0	0	1
1	1	1	1	0	0

- It can also be drawn as:

Present state		Next state				Output	
		$x=0$		$x=1$		$x=0$	$x=1$
A	B	A	B	A	B	y	y
0	0	0	0	0	1	0	0
0	1	0	0	1	1	1	0
1	0	0	0	1	0	1	0
1	1	0	0	1	0	1	0

Lecture 10:
State Diagram:

- Another way to represent sequential circuits is state diagram: a bubble for each state:



- If the system can move from a state to another there is a line between the two.
- The input that will cause that transition nad the resulting output are shown on the cord.

Lecture 10:
Output equations and FF input equations:

- A sequential circuit has two parts:
  - A memory section consisting of a set of flip flops, and
  - A set of logic gates that form either the outputs or the next state of the circuit.
- The behavior of the circuit is defined in terms of:
  - **output equations** presenting the output as a function of the input and the present state of the circuit.
  - **state equations** or flip-flop input equations presenting the next state as a function of the input and the present state.
- In the sequential circuit shown in the previous slide, we have two flip-flops. So, we have two flip-flop input equations.
- Using D flip-flops:

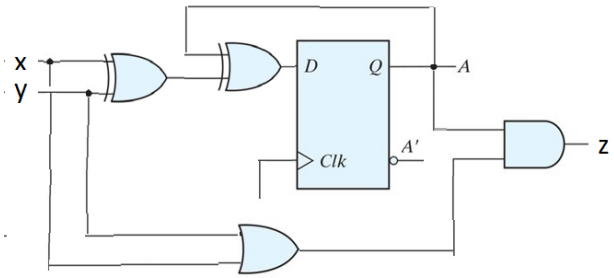
$$D_A = Ax + Bx$$
$$D_B = A'x$$

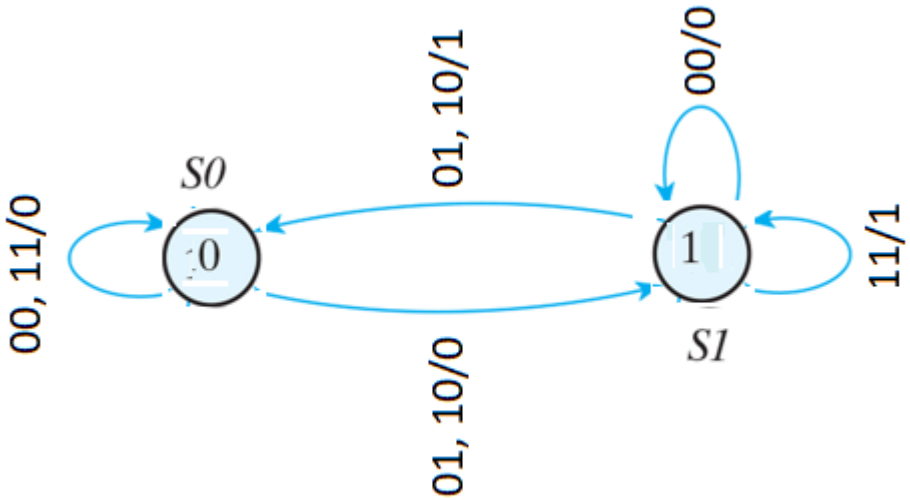
- and the output equation is $y = (A + B)x'$

Lecture 10:
Analysis with D-flip-flops:

Concordia University
Department of Electrical & Computer Engineering

- Example: Consider the circuit shown having 1 D FF.
- FF input equation is $D_A = A \oplus x \oplus y$ and output equation is $z = (x + y)A$
- Showing time explicitly: $D(t) = A(t+1) = A(t) \oplus x(t) \oplus y(t)$ and $z(t) = \big(x(t) + y(t)\big)A(t)$
- Circuit has one FF, so it has two states. The state diagram is shown on the bottom right.





Lecture 10:
Analysis with D-flip-flops:

Concordia University
Department of Electrical & Computer Engineering

- Since there are two inputs, there need to be four arrows leaving each state.
- However, in this case, the arrows for inputs 00 and 11 (also 01 and 10) often coincide and one arrows can be used as a combination of two overlapping arrows.

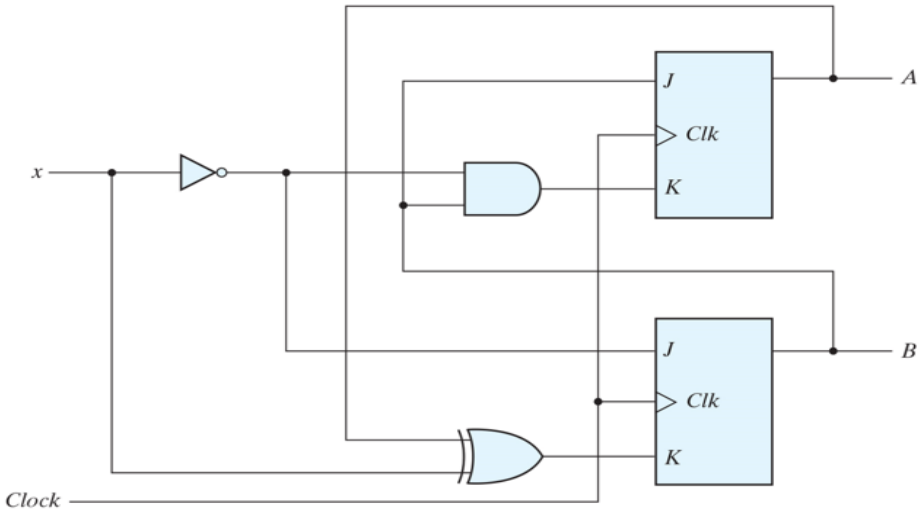
- The state transition table is:

Present state	Inputs		Next state	Output
$A(t)$	$x(t)$	$y(t)$	$A(t+1)$	$z(t)$
0	0	0	0	0
0	0	1	1	0
0	1	0	1	0
0	1	1	0	0
1	0	0	1	0
1	0	1	0	1
1	1	0	0	1
1	1	1	1	1

Lecture 10:
Analysis with JK-flip-flops:

Department of Electrical & Computer Engineering

- Example: Take the following circuit with two JK Flip-flops:



Lecture 10:
Analysis with JK-flip-flops:

Concordia UNIVERSITÉ UNIVERSITY
Department of Electrical & Computer Engineering

- The Analysis involves the following steps:

1) Find the FF input equations. In this case they are:

$$J_A = B \ , \ K_A = Bx'$$

$$J_B = x' , K_B = A \oplus x = A'x + Ax'$$

2) Find the binary values of each input equation.

3) Use the characteristic table or characteristic equation of the JK FF, i.e., $Q(t+1) = JQ' + K'Q$ to find the next state.

4) derive the state transition table.

Note: In this case there is no output. Otherwise we had to consider the output equations.

Lecture 10:
Analysis with JK-flip-flops:

Concordia University
Department of Electrical & Computer Engineering

- Using the above procedure, we derive the following state transition table:

Current state		Input	FF Inputs				Next state	
$A(t)$	$B(t)$	$x(t)$	J_A	K_A	J_B	K_B	$A(t+1)$	$B(t+1)$
0	0	0	0	0	1	0	0	1
0	0	1	0	0	0	1	0	0
0	1	0	1	1	1	0	1	1
0	1	1	1	0	0	1	1	0
1	0	0	0	0	1	1	1	1
1	0	1	0	0	0	0	1	0
1	1	0	1	1	1	1	0	0
1	1	1	1	0	0	0	1	1

- We have used:

$$A(t+1) = J_A A' + K'_A A$$

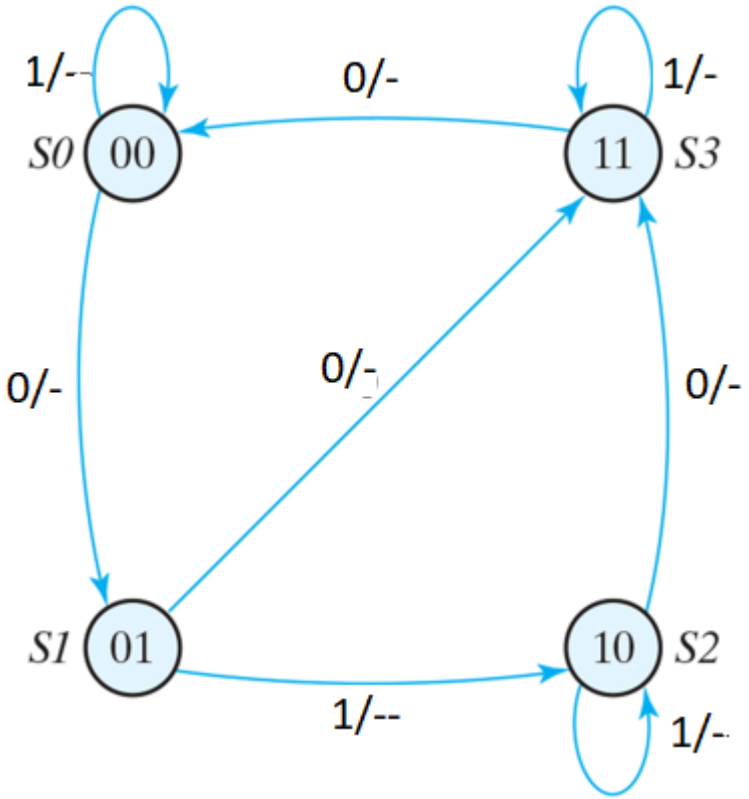
And

$$B(t+1) = J_B B' + K'_B B$$

Lecture 10:
Analysis with JK-flip-flops:

Concordia University
Department of Electrical & Computer Engineering

- The state diagram is:

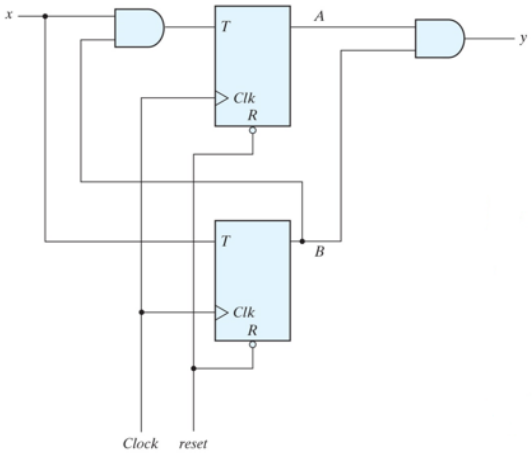


Lecture 10:
Analysis with T-flip-flops:

Concordia UNIVERSITÉ UNIVERSITY
Department of Electrical & Computer Engineering

- When you analyze circuits that have T-FF, use $Q(t+1) = T \oplus Q$.

- **Example:** Consider this circuit with two T-flip-flops:



Lecture 10:
Analysis with T-flip-flops:

Analysis:

1) $T_A = Bx, T_B = x, y = A + B$

2) Find T_A, T_B, for different combinations of $A(t), B(t), x(t)$.

3) $A(t+1) = T_A \oplus A = T_A'A + T_AA' = (Bx)'A + (Bx)A' = AB' + Ax' + A'Bx$ and $B(t+1) = x \oplus B$.

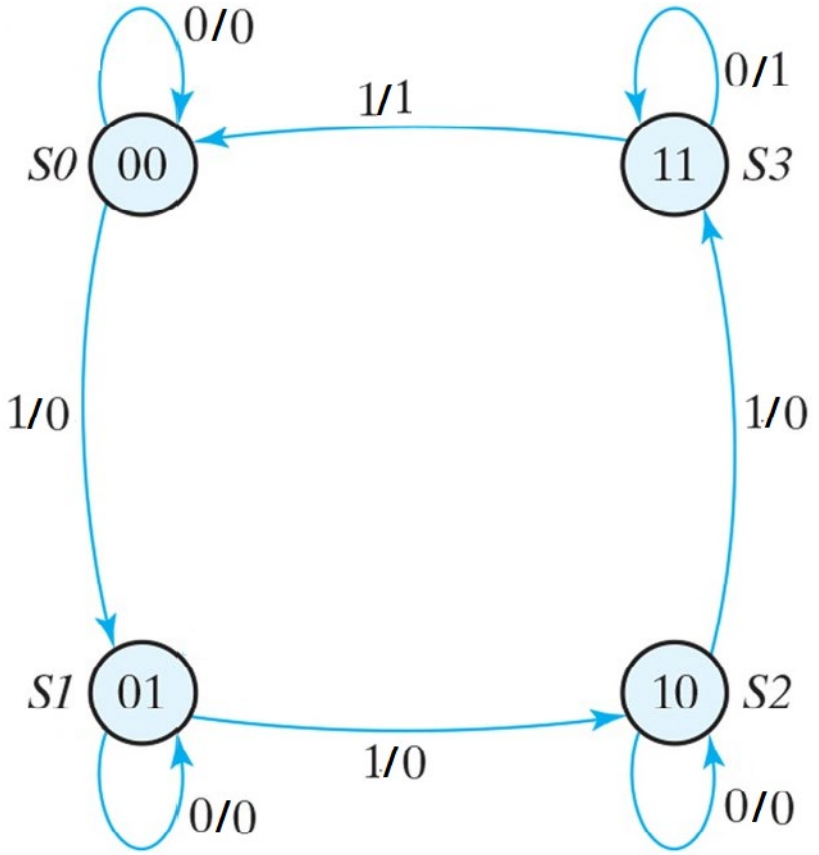
State transition table:

Present state		Input	Next state		Output
$A(t)$	$B(t)$	$x(t)$	$A(t+1)$	$B(t+1)$	$y(t)$
0	0	0	0	0	0
0	0	1	0	1	0
0	1	0	0	1	0
0	1	1	1	0	0
1	0	0	1	0	0
1	0	1	1	1	0
1	1	0	1	1	1
1	1	1	0	0	1

Lecture 10:
Analysis with T-flip-flops:

The state Diagram is:

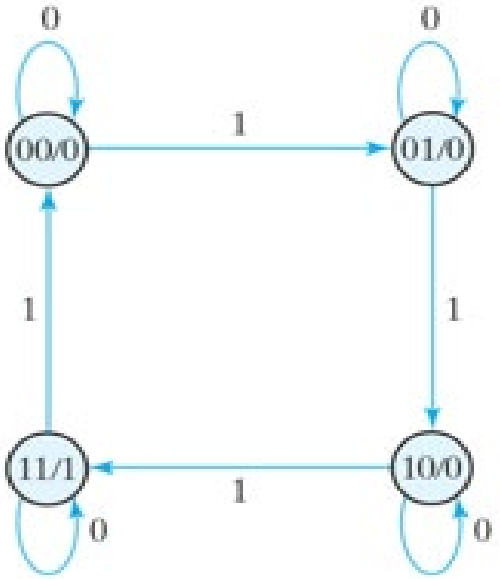
This is an example of Moore Machines



Lecture 10:
Analysis with T-flip-flops:

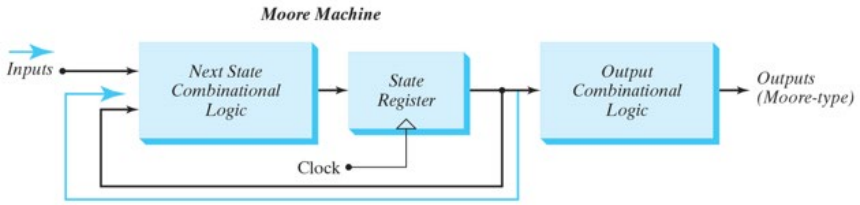
Concordia University
Department of Electrical & Computer Engineering

The outputs can be placed inside the bubbles (states):

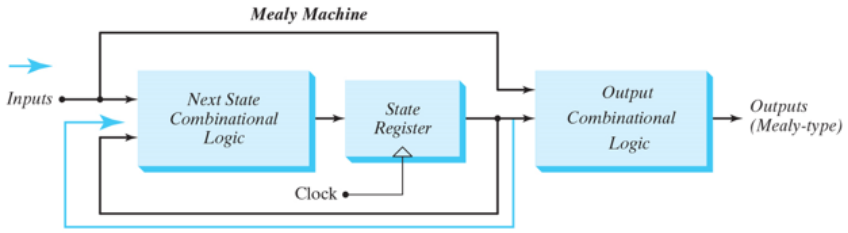


Lecture 10:
Analysis with T-flip-flops:

A Moore Machine or More FSM, looks like this:



The other type of sequential circuits are Mealy Machines:



Lecture 10:
Design of Sequential Circuits:

Université Concordia University
Department of Electrical & Computer Engineering

- We saw that analysis starts with a circuit diagram and ends up with a functional description, e.g., in the form of a state diagram,
- Design or synthesis starts with a functional description and the end result should be a circuit diagram

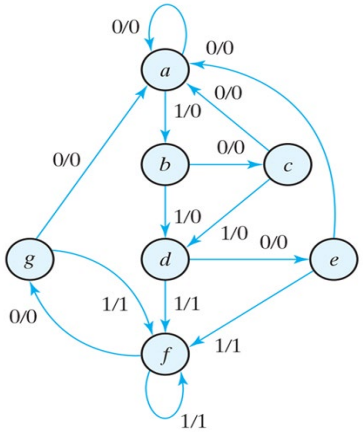
Lecture 10:
State Reduction:

- A functional description, may contain redundant states.
- Redundant states act the same: i.e., for a given input go to the same state and generate the same output.
- State reduction is used to combine redundant states.
- Remember that m flip-flops give us up to 2^m states. Reducing the number of states, can, possibly, reduce the number of FFs.
- A circuit with 6 states needs 3 FFs. Reducing the number of states to 4 we only need 2 FFs. But going from 8 to 5, there is no saving.
- In reducing the number of states, usually, the following fact is used:
- Two states are equivalent if, for every input, they generate the same output and transition to the same next state or to an equivalent state.

Lecture 10:
State Reduction: Example

Concordia University
Department of Electrical & Computer Engineering

- Example:



State Transition Table:

Present state	Next state		Output	
	$x = 0$	$x = 1$	$x = 0$	$x = 1$
a	a	b	0	0
b	c	d	0	0
c	a	d	0	0
d	e	f	0	1
e	a	f	0	1
f	g	f	0	1
g	a	f	0	1

- g and e are equivalent. Remove g.

Present state	Next state		Output	
	$x = 0$	$x = 1$	$x = 0$	$x = 1$
a	a	b	0	0
b	c	d	0	0
c	a	d	0	0
d	e	f	0	1
e	a	f	0	1
f	e	f	0	1

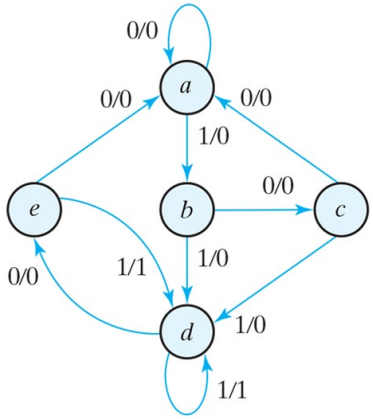
Lecture 10:
State Reduction: Example

Concordia University
Department of Electrical & Computer Engineering

- The reduced-state Transition Table:

Present state	Next state		Output	
	$x = 0$	$x = 1$	$x = 0$	$x = 1$
a	a	b	0	0
b	c	d	0	0
c	a	d	0	0
d	e	d	0	1
e	a	d	0	1

- The reduced state diagram is:



Lecture 10: State Assignment

Concordia University
Department of Electrical & Computer Engineering

- We need to assign binary values to states.
- We use binary values from 0 to 2^{m-1} to states.
- If the number of states s is not a power of 2, we can use the first s numbers counting from 0 to $s-1$. For example, for the above 5-state circuit, we may use 000, 001, 010, 011, and 100.
- **Gray code** can also be used. In a Gray code any two consecutive numbers differ in only one bits, this may allow some simplification in logic design.
- Third option **one-hot** assignment: use s bits to represent each state, only one bit is equal to 1 in each state index, i.e., one flip-flop per state. This makes the design of combinational circuit trivial, but, results in waste of flip-flops and is only wise if there are lots of flip-flops on the chip.

Lecture 10: State Assignment

Concordia University
Department of Electrical & Computer Engineering

- These three state assignment schemes are shown in this state transition table:

state	binary	Gray code	One-hot
a	0 0 0	0 0 0	0 0 0 0 1
b	0 0 1	0 0 1	0 0 0 1 0
c	0 1 0	0 1 1	0 0 1 0 0
d	0 1 1	0 1 0	0 1 0 0 0
e	1 0 0	1 1 0	1 0 0 0 0

- Reduced-state transition table using binary assignment:

Present state	Next state		Output	
	$x = 0$	$x = 1$	$x = 0$	$x = 1$
0 0 0	0 0 0	0 0 1	0	0
0 0 1	0 1 0	0 1 1	0	0
0 1 0	0 0 0	0 1 1	0	0
0 1 1	1 0 0	0 1 1	0	1
1 0 0	0 0 0	0 1 1	0	1

Lecture 10:
Design Procedure

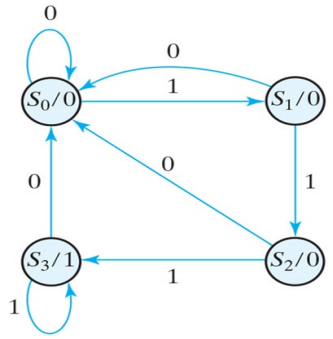
A design process consists of the following tasks:

1. Translating the word description of the circuit into a state diagram.
2. Reducing the number of states if possible.
3. Assigning binary values to the states.
4. Obtaining the binary-coded state table.
5. Choosing the type of flip-flops to be used.
6. Writing down the flip-flop input equations and output equations.
7. Drawing the circuit diagram.

Lecture 10:
Design Procedure: Example

Department of Electrical & Computer Engineering

- **Example:** Design a circuit that detects the occurrence of three or more consecutive ones.



- State Transition Table:

Present state		Input	Next state		Output
A	B	x	A	B	y
0	0	0	0	0	0
0	0	1	0	1	0
0	1	0	0	0	0
0	1	1	1	0	0
1	0	0	0	0	0
1	0	1	1	1	0
1	1	0	0	0	1
1	1	1	1	1	1

Lecture 10:
Design using D Flip-flop

Université Concordia University
Department of Electrical & Computer Engineering

- Assume that we choose D flip-flops.
- Then the flip-flop input equations are:

$$D_A(A, B, x) = A(t+1) = \sum(3,5,7)$$

and,

$$D_B(A, B, x) = B(t+1) = \sum(1,5,7)$$

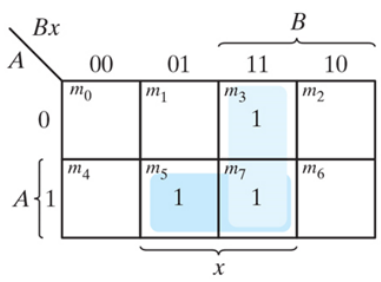
The output equation is

$$y(A, B, x) = \sum(6,7).$$

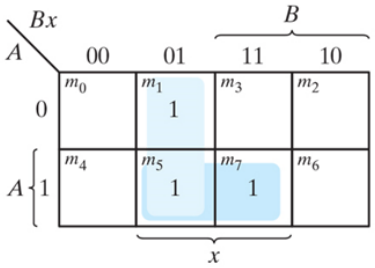
Lecture 10: Design using D Flip-flop

Department of Electrical & Computer Engineering

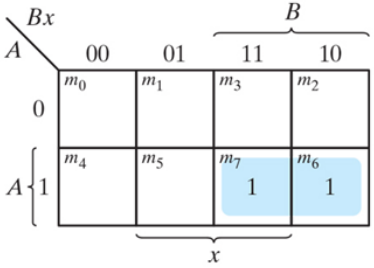
- Using the K-maps we find D_A, D_B, and y:



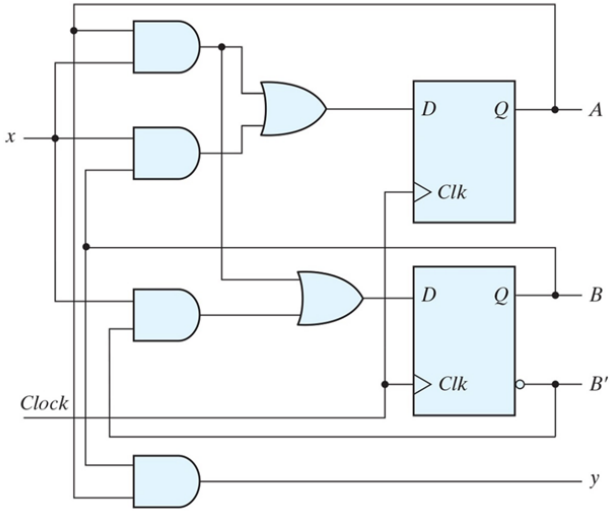
$D_A = Ax + Bx$



$D_B = Ax + B'x$



$y = AB$



Lecture 10: Excitation Tables

Concordia University
Department of Electrical & Computer Engineering

- An **excitation table** is a table that determines the value of the input equations for each state transition. Such a table is called an excitation table.
- For D flip-flops are simple and we don't need an excitation table.
- JK or T flip-flops are more tricky.

Excitation Table: a) for a JK FF:

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

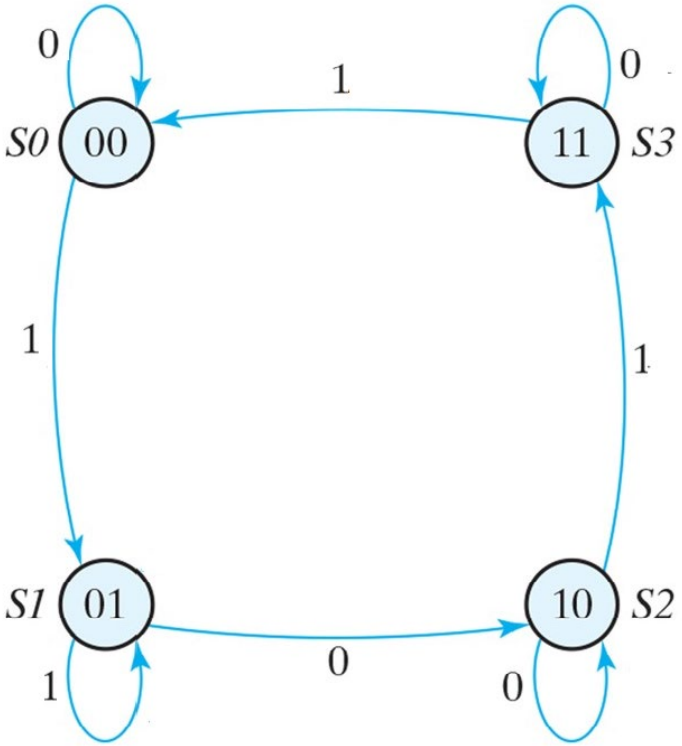
b) For a T FF:

$Q(t)$	$Q(t+1)$	T
0	0	0
0	1	1
1	0	1
1	1	0

Lecture 10: Synthesis with JK flip-flops

Concordia University
Department of Electrical & Computer Engineering

- Example: Design a circuit for this state diagram using JK FF:



Lecture 10:
Synthesis with JK flip-flops

Concordia University
Department of Electrical & Computer Engineering

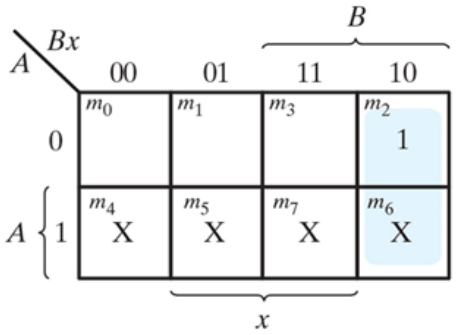
- State Transition Table:

Present state		Input	Next state		FF Inputs			
A	B	x	A	B	J_A	K_A	J_B	K_B
0	0	0	0	0	0	X	0	X
0	0	1	0	1	0	X	1	X
0	1	0	1	0	1	X	X	1
0	1	1	0	1	0	X	X	0
1	0	0	1	0	X	0	0	X
1	0	1	1	1	X	0	1	X
1	1	0	1	1	X	0	X	0
1	1	1	0	0	X	1	X	1

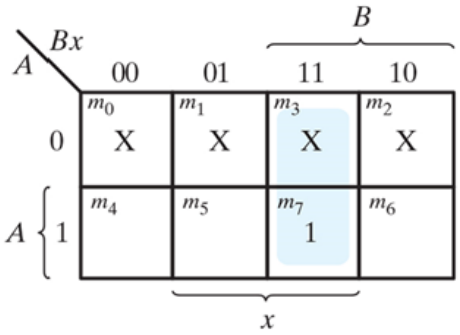
Lecture 10: Synthesis with JK flip-flops

Department of Electrical & Computer Engineering

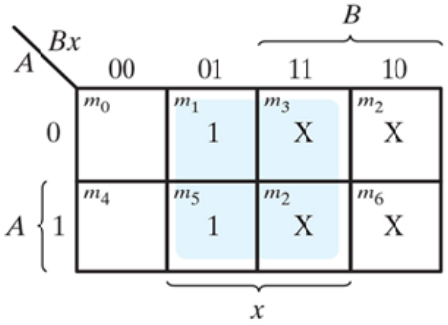
- Example: Design a circuit using JK flip-flop for:



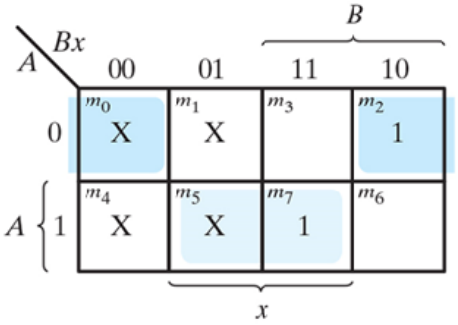
$J_A = Bx'$



$K_A = Bx$



$J_B = x$

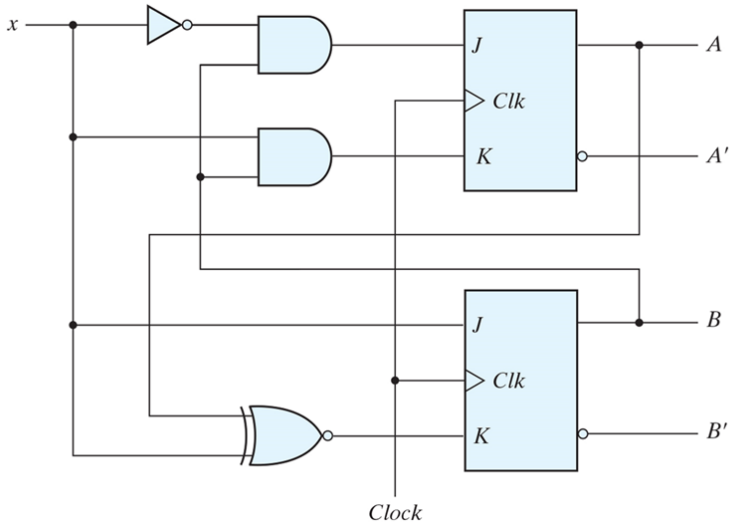


$K_B = (A \oplus x)'$

Lecture 10: Synthesis with JK flip-flops

Concordia University
Department of Electrical & Computer Engineering

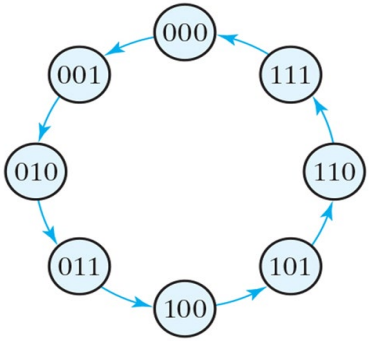
- The Logic Diagram is:



Lecture 10: Synthesis with T flip-flops

Department of Electrical & Computer Engineering

- State Diagram of a 3-bit Counter:



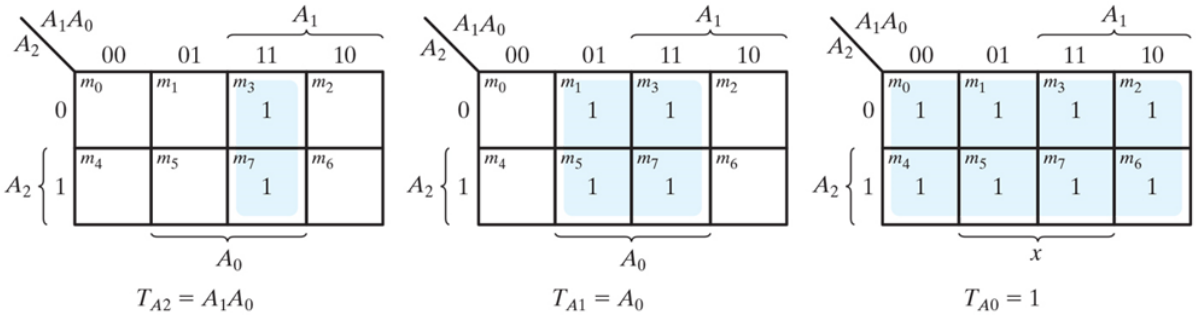
- State Transition Table:

Present state			Next state			FF inputs		
A_2	A_1	A_0	A_2	A_1	A_0	T_{A_2}	T_{A_1}	T_{A_0}
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	1	1	0	0	1	1
1	1	0	1	1	1	0	0	1
1	1	1	0	0	0	1	1	1

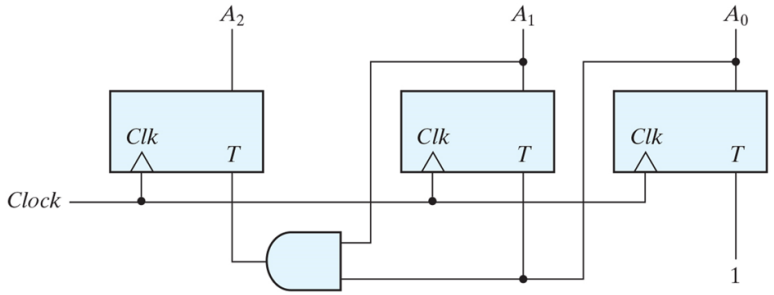
Lecture 10: Synthesis with T flip-flops

Department of Electrical & Computer Engineering

- FF input equations:



- The Logic Diagram:



Lecture 10: Knowledge Check

Université Concordia University
Department of Electrical & Computer Engineering

- **Question 1:** A counter counts from 0 to 17. How many FFs do we need to implement it?

a) 18, b) 5, c) 3, d) 4

- **Question 2:** A counter with 6 FFs can count up to:

- a) 16, b) 32, c) 63, d) 31