Department of Electrical & Computer Engineering

COEN 212:
DIGITAL SYSTEMS DESIGN I
Lecture 11: Registers and Counters

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

Lecture 11:
Objectives of this lecture

Concordia University
Department of Electrical & Computer Engineering

- In this lecture, we talk about:
  - Different types of Registers.
  - Implementation of registers and
  - Different Types of Counters.
  - Implementation of Counters.
  - Sample Applications.

Lecture 11:
Reading for this lecture

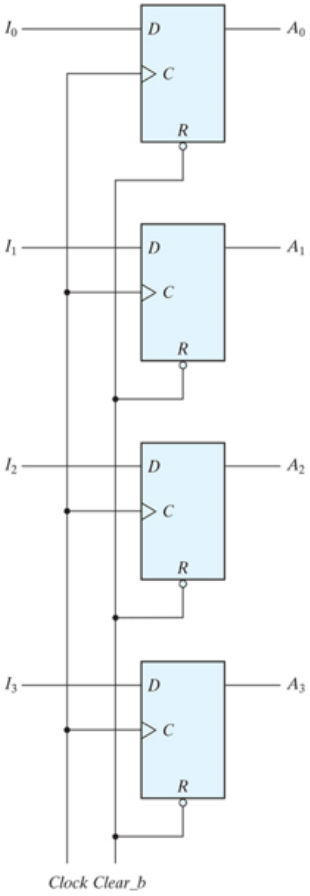
Department of Electrical & Computer Engineering

- Digital Design by M. Morris R. Mano and Michael D. Ciletti, 6th Edition, Pearson, 2018:
  - Chapter 6

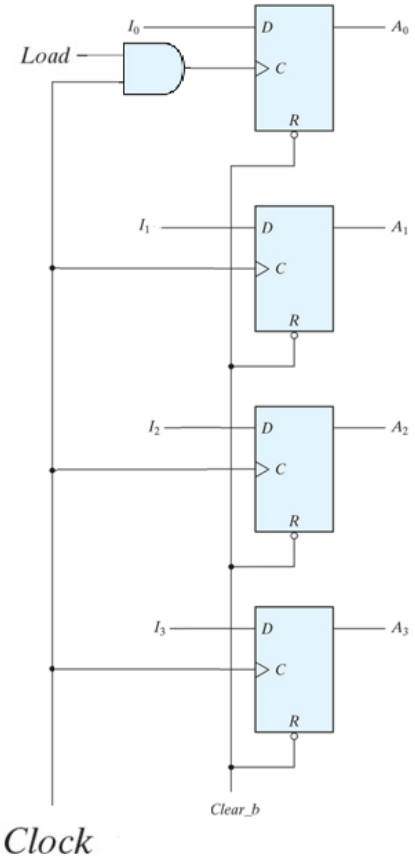
Lecture 11: Registers:

Université Concordia University
Department of Electrical & Computer Engineering

- 4-bit register:



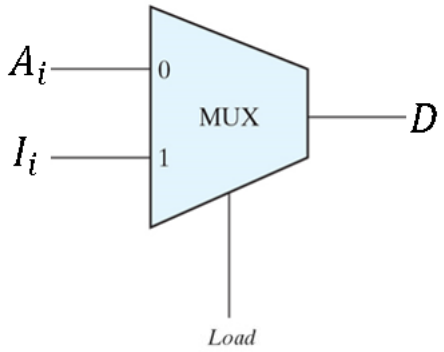
4-bit register with Load control



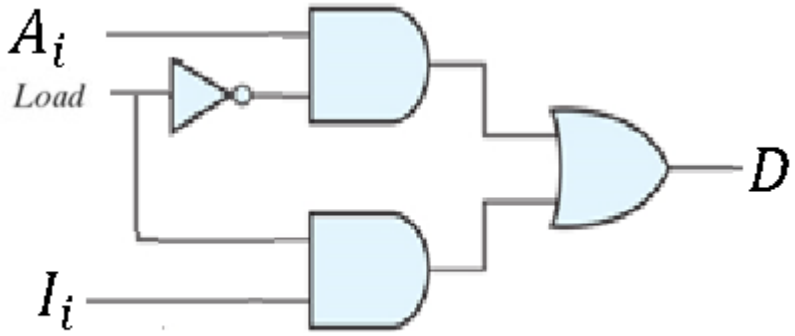
Lecture 11: Registers:

Department of Electrical & Computer Engineering

- MUX for Load Control:



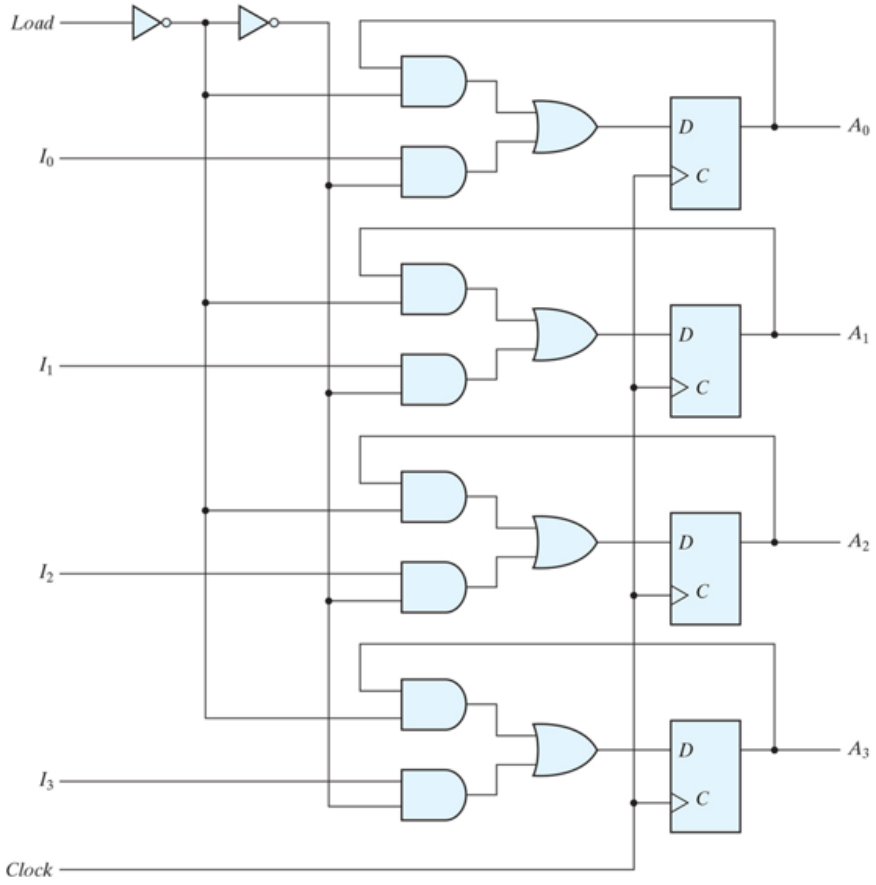
- Implementation of the MUX:



Lecture 11: Registers:

Concordia University
Department of Electrical & Computer Engineering

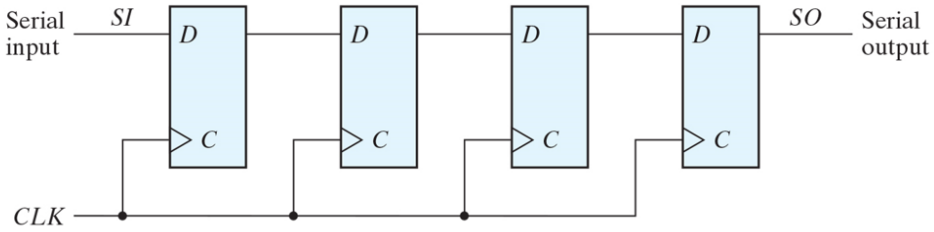
- 4-bit register with Load:



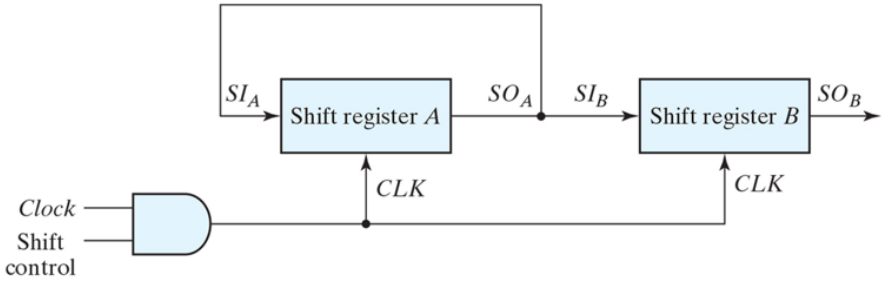
Lecture 11:
Shift Registers:

Concordia University
Department of Electrical & Computer Engineering

- 4-bit Shift Register:

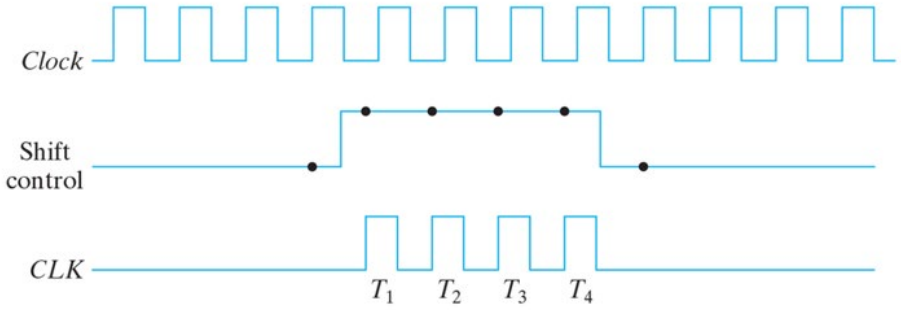


- Serial Transfer:



Lecture 11:
Register to Register Serial Transfer:

- Serial Transfer:



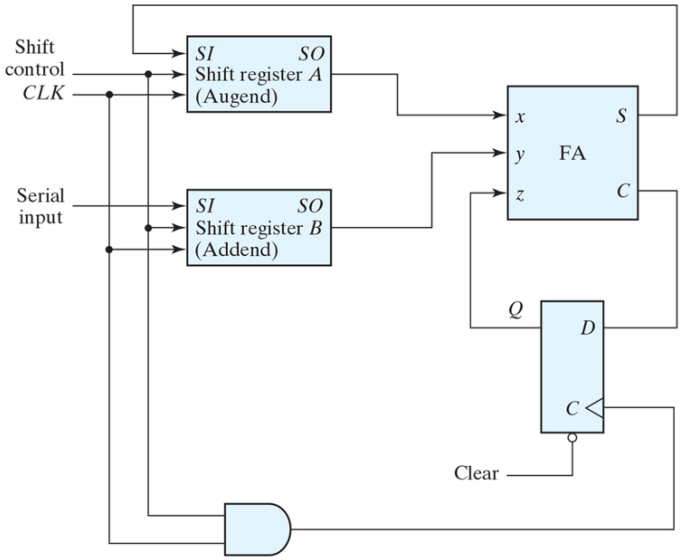
- Content of registers:

Timing Pulse	Shift Register A	Shift Register B
Initial value	1011	0010
After T_1	1101	1001
After T_2	1110	1100
After T_3	0111	0110
After T_4	1011	1011

Lecture 11:
Serial Addition:

Concordia University
Department of Electrical & Computer Engineering

- Serial Adder:



Lecture 11:
Serial Addition: JK FF IMplementation

Concordia University
Department of Electrical & Computer Engineering

- State Transition Table:

Present state	Inputs		Next state	Output	FF Inputs	
Q	x	y	Q	S	J_Q	K_Q
0	0	0	0	0	0	X
0	0	1	0	1	0	X
0	1	0	0	1	0	X
0	1	1	1	0	1	X
1	0	0	0	1	X	1
1	0	1	1	0	X	0
1	1	0	1	0	X	0
1	1	1	1	1	X	0

- FF input equations:

$$J_Q = xy$$
$$K_Q = x'y' = (x + y)'$$

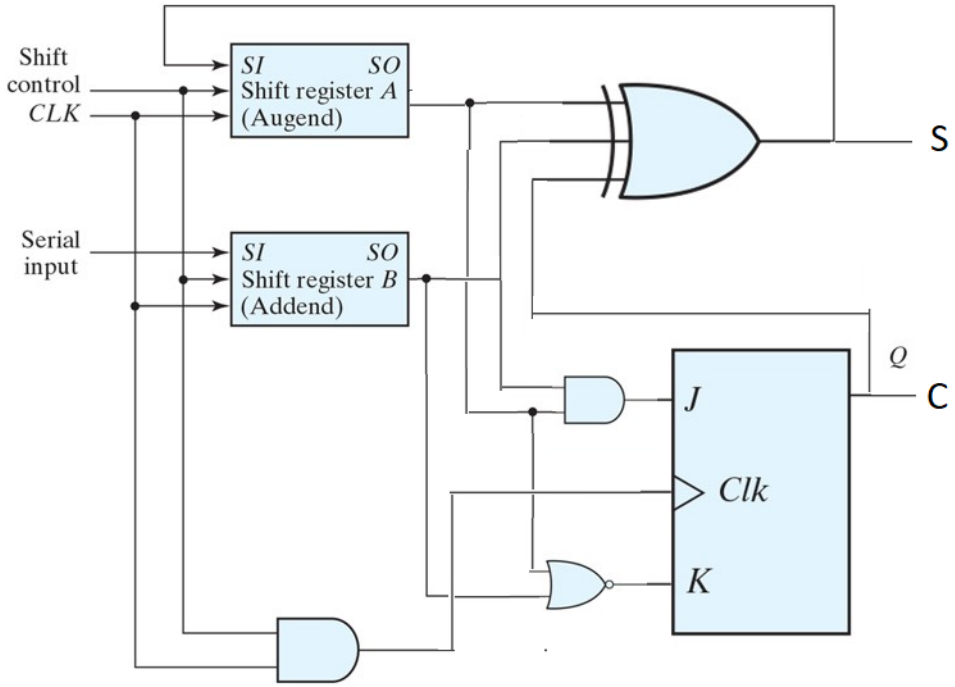
The output equation:

$$S = x \oplus y \oplus Q$$

Lecture 11:
Serial Addition: JK FF Implementation

Concordia University
Department of Electrical & Computer Engineering

- Serial adder circuit diagram:



Lecture 11:
Universal Shift Register

- A Universal SR can:
  - Remain unchanged,
  - Load serially and shift right,
  - Load serially and shift left,
  - Load in parallel.

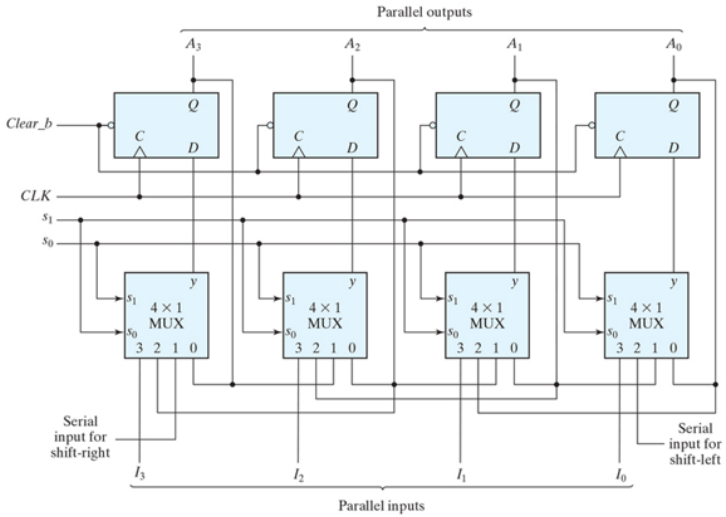
- We need 4 control signals (2 control bits):

Control bits		operation
s_1	s_0	
0	0	No change
0	1	Shift right
1	0	Shift left
1	1	Parallel load

Lecture 11:
Universal Shift Register

Université Concordia University
Department of Electrical & Computer Engineering

- The circuit for Universal shift Register:



Lecture 11: Counters:

Concordia University
Department of Electrical & Computer Engineering

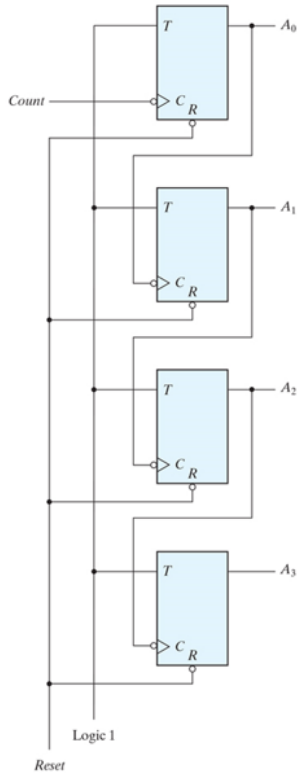
- A counter is a register that changes in a certain order.

- Examples:
  - A binary counter counts from 0 to 2^n-1.
  - A BCD counter counts from 0 to 9 (0000 to 1001).
- There are two types of counters:
  - Ripple counters
  - Synchronous counters
- In a synchronous counter, state changes are dictated by a common clock.
- In a ripple counter, the output of one flip-flop has effect on the next state of the following flip-flop.

Lecture 11:
Binary Ripple Counter: with T FF

Concordia UNIVERSITÉ UNIVERSITY
Department of Electrical & Computer Engineering

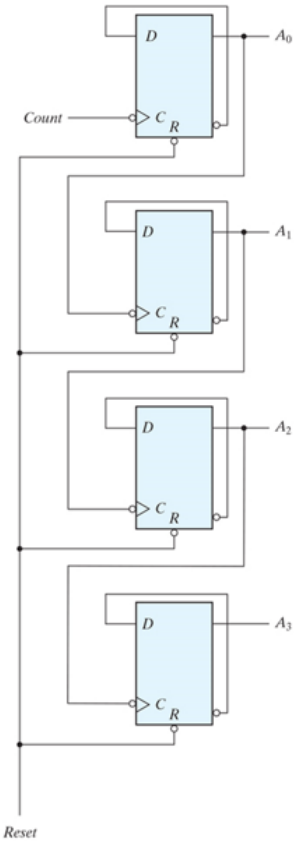
- A 4-bit ripple counter:



Lecture 11:
Binary Ripple Counter: with D FF

Department of Electrical & Computer Engineering

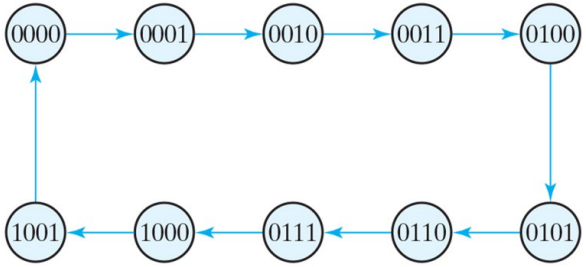
- A 4-bit ripple counter with D FFs:



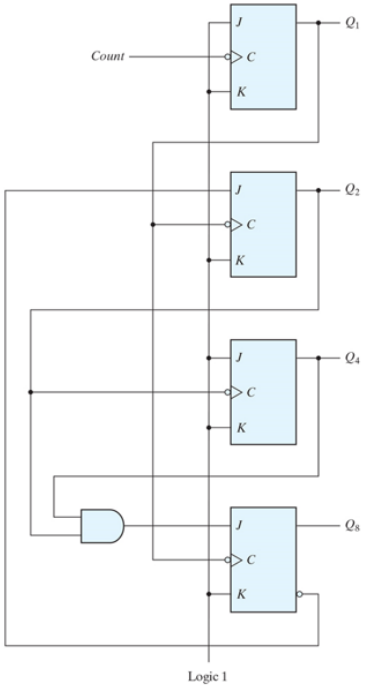
Lecture 11:
BCD Ripple Counter:

Department of Electrical & Computer Engineering

- State Diagram of a BCD Ripple Counter:



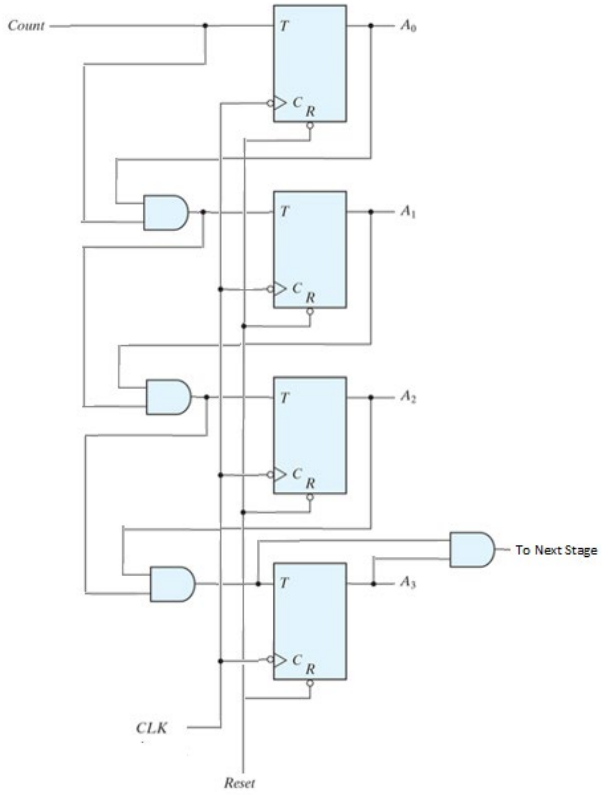
Circuit Diagram:



Lecture 11:
Synchronous Counters:

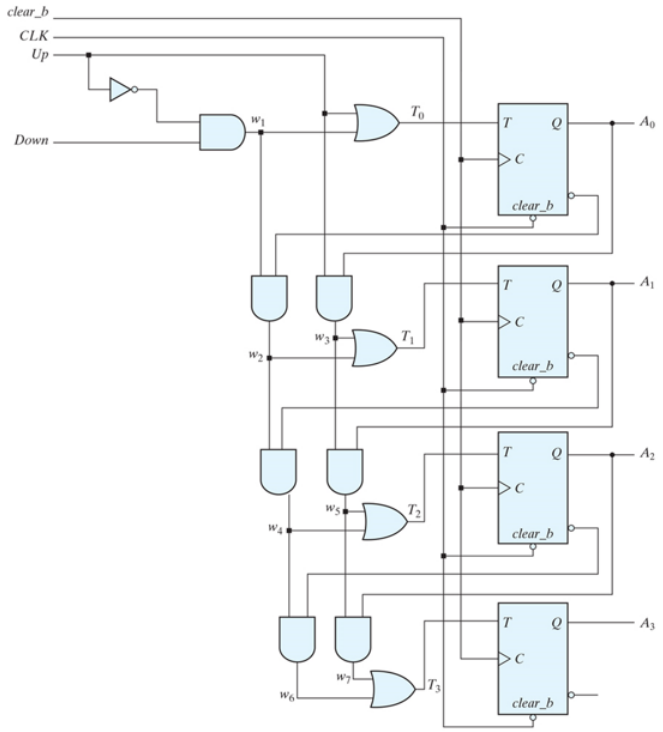
Department of Electrical & Computer Engineering

- State Diagram of a Synchronous Binary Counter:



Lecture 11:
Up-Down counters:

- A 4-bit up/down counter:



Lecture 11:
Synchronous BCD counter:

Department of Electrical & Computer Engineering

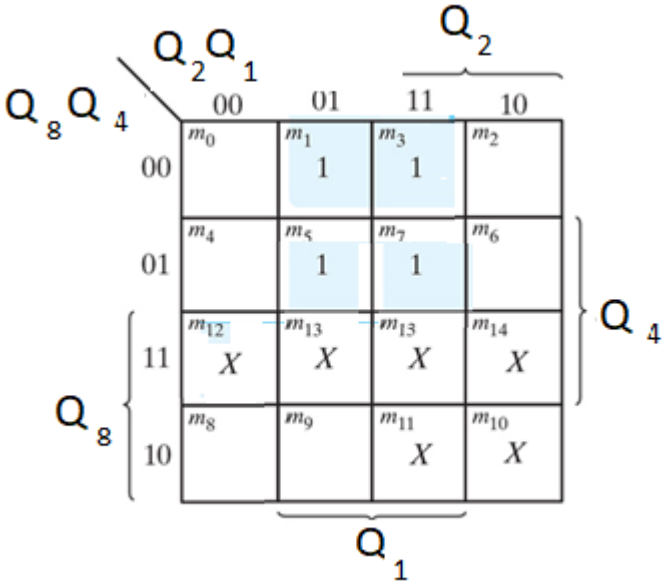
- State transition table of a 4-bit up/down counter:

Present state				Next state				Output	FF inputs			
Q_8	Q_4	Q_2	Q_1	Q_8	Q_4	Q_2	Q_1	y	T_8	T_4	T_2	T_1
0	0	0	0	0	0	0	1	0	0	0	0	1
0	0	0	1	0	0	1	0	0	0	0	1	1
0	0	1	0	0	0	1	1	0	0	0	0	1
0	0	1	1	0	1	0	0	0	0	1	1	1
0	1	0	0	0	1	0	1	0	0	0	0	1
0	1	0	1	0	1	1	0	0	0	0	1	1
0	1	1	0	0	1	1	1	0	0	0	0	1
0	1	1	1	1	0	0	0	0	1	1	1	1
1	0	0	0	1	0	0	1	0	0	0	0	1
1	0	0	1	0	0	0	0	1	1	0	0	1

Lecture 11:
Synchronous BCD counter:

Concordia University
Department of Electrical & Computer Engineering

- K-map for T_2:



FF Input Equations:

$$T_1 = 1,$$

$$T_2 = Q_8' Q_1,$$

$$T_4 = Q_2 Q_1$$

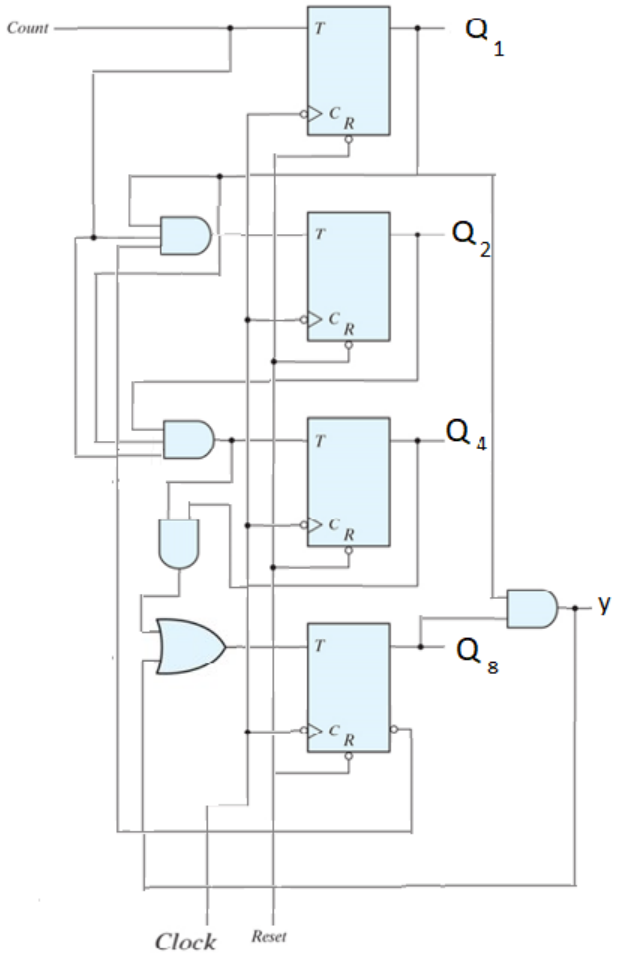
$$T_8 = Q_8 Q_1 + Q_4 Q_2 Q_1$$

and the output equation: $y = Q_8 Q_1$.

Lecture 11:
Synchronous BCD counter:

Concordia University
Department of Electrical & Computer Engineering

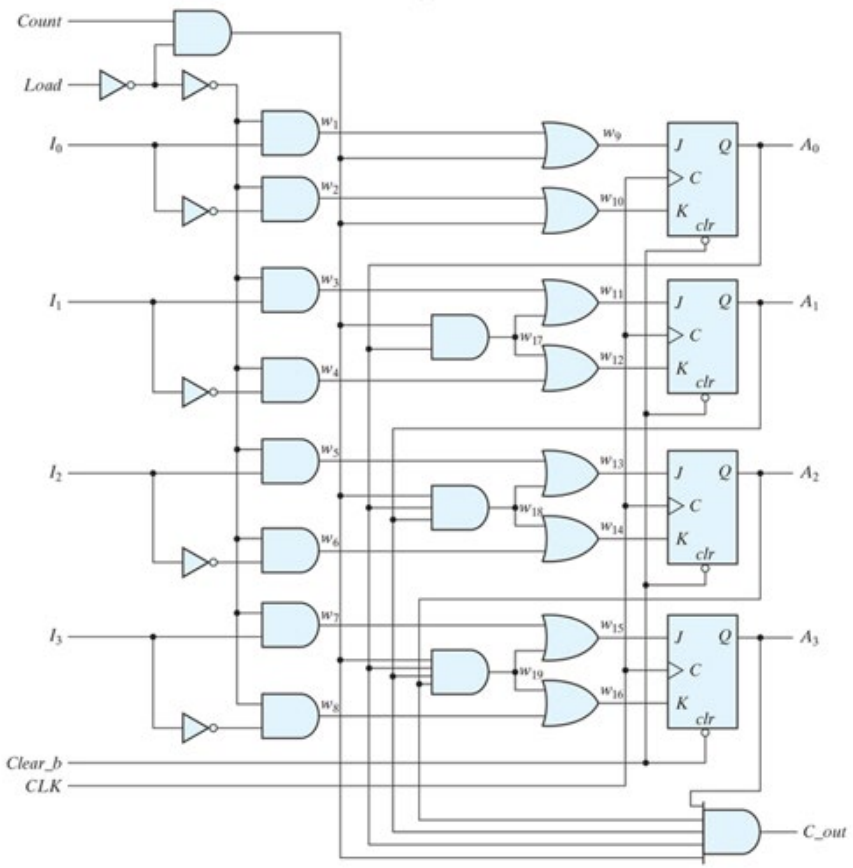
- The circuit for a BCD Counter with T Flip-flops:



Lecture 11:
Counters with Parallel Load:

Concordia UNIVERSITY
Department of Electrical & Computer Engineering

- Binary Counter with Parallel Load:



Lecture 11:
Counters with Unused States:

Université Concordia University
Department of Electrical & Computer Engineering

- Binary Counter with n flip-flops can have 2^n states.
- We may not need all these states.
- We have already seen the example of BCD counter that uses only 10 out of $2^4 = 16$ possible states.
- We can treat the unused states:
  - as don't care condition or
  - some specific next states. Since,

- **Example:** consider a counter following the sequence 000, 001, 010, 100, 101, 110. That is, the counter skips 011 and 111.

Lecture 11:
Counters with Unused States:

Concordia University
Department of Electrical & Computer Engineering

- **Example:** consider a counter following the sequence 000, 001, 010, 100, 101, 110. That is, the counter skips 011 and 111.
- The state table for this counter is:

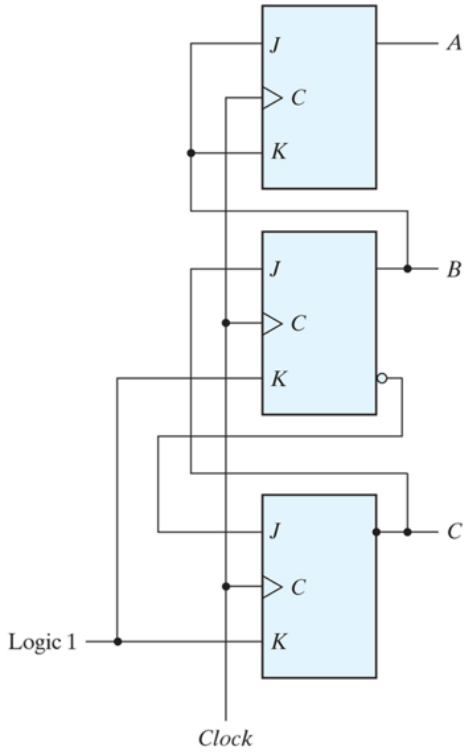
Present state			Next state			Flip-flop inputs					
A	B	C	A	B	C	J_A	K_A	J_B	K_B	J_C	K_C
0	0	0	0	0	1	0	X	0	X	1	X
0	0	1	0	1	0	0	X	1	X	X	1
0	1	0	1	0	0	1	X	X	1	0	X
1	0	0	1	0	1	X	0	0	X	1	X
1	0	1	1	1	0	X	0	1	X	X	1
1	1	0	0	0	0	X	1	X	1	0	X

- We have $J_A = K_A = B$, $J_B = C$, $K_B = 1$, $J_C = B'$, and $K_C = 1$.

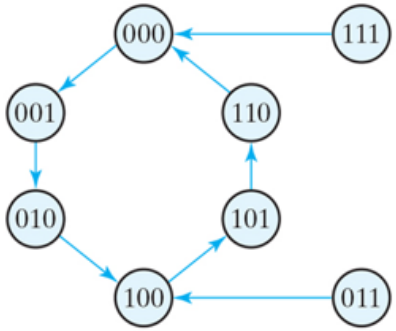
Lecture 11:
Counters with Unused States:

Department of Electrical & Computer Engineering

- The circuit diagram is:



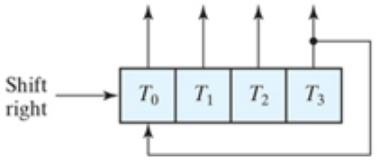
State Diagram:

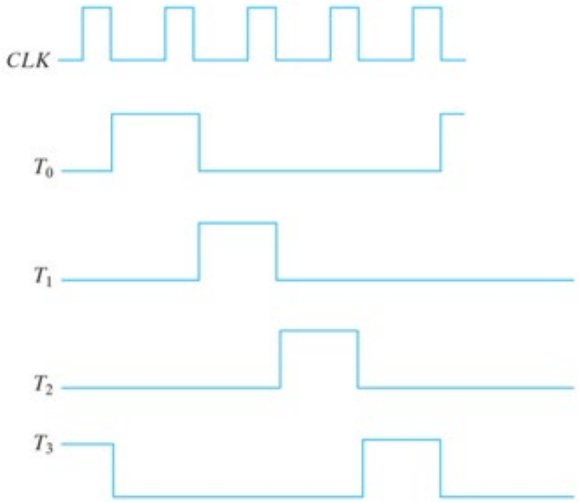


Lecture 11:
Ring Counter:

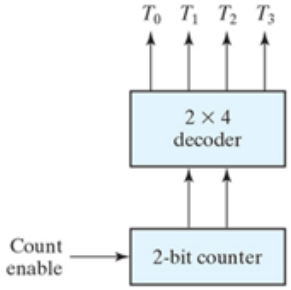
Concordia University
Department of Electrical & Computer Engineering

- A Ring Counter counts: 1000, 0100, 0010, 0001, 1000, …
- At any time only one FF is on.
- It can be used to control a sequence of repetitive operations.
- A Ring Counter can be implemented by connecting the output of a shift register to its input.



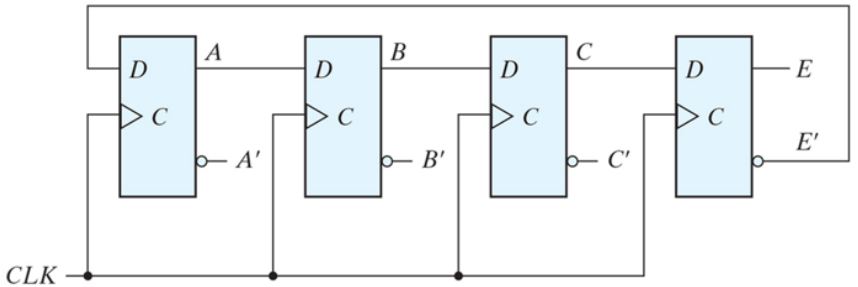


Two FF implementation:



Lecture 11:
Johnson Counter:

- Schematic of
- switch-tail Ring Counter:



- A Johnson counter is a switch tail ring with outputs (or inverted outputs) of flip-flops combined (by AND) to form 2k timing signals.
- The rule for forming AND gate inputs is straightforward:
  - For all zero pattern the inverted outputs of the first and last flip-flop are used.
  - For all one pattern the regular outputs of the extreme flip-flops are used.
  - For other patterns, the first two alternating bits, either 01 or 10 are used.

Lecture 11:
Johnson Counter:

- The count sequence for the 4-bit tail-switch ring counter:

Sequence number	(flip-flop outputs) A	B	C	E	AND gate required for output
1	0	0	0	0	$A'E'$
2	1	0	0	0	AB'
3	1	1	0	0	BC'
4	1	1	1	0	CE'
5	1	1	1	1	AE
6	0	1	1	1	$A'B$
7	0	0	1	1	$B'C$
8	0	0	0	1	$C'E$

- So, in this example: we use$A'E'$ for 0000, AE for 1111, AB' for 1000, BC' for1100. Similarly $A'B$ for 0111 and $B'C$ for 0011 and so on.
- When Johnson counter goes to an unused state, it goes from one invalid state to another. To avoid this, we may disconnect the output of B flip-flop from the input D of C flip-flop and instead feed C flip-flop with $D_C = (A + C)B$.

Lecture 11: Knowledge Check

Concordia University
Department of Electrical & Computer Engineering

- **Question 1:** Minimum number of Flip-flops needed to implement a counter that counts up to 25 is:
- a) 25, b) 5, c) 6, d) 8

- **Question 2:** How many FF's do we need if we use ring counter:
- a) 5, b) 8, c) 25, d) 12

- **Question 3:** Compared to parallel adder, a Serial adder:
- a) is faster, b) is slower, c) needs more gates, d) both a and c