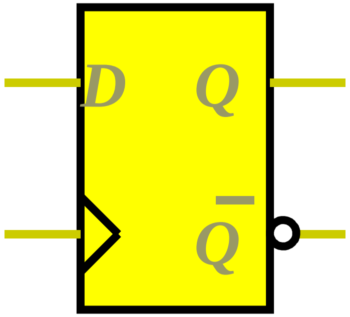
Digital Electronics and Microprocessors

Class 10

CHHAYADEVI BHAMARE

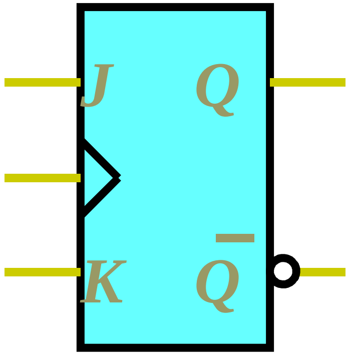
Flip-Flop Applications

- Examples of applications:
  - Counting
  - Storing binary data
  - Transferring binary data between locations
- Many FF applications are categorized as sequential, which means that the output follows a predetermined sequence of states.



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

$Q(t+1) = D$



J	K	$Q(t+1)$
0	0	$Q(t)$
0	1	0
1	0	1
1	1	$Q'(t)$

$Q(t+1) = JQ' + K'Q$

Frequency Division and Counting

- FFs are often used to divide a frequency. Here the output frequency is 1/8th the input (clock) frequency.
- The same circuit is also acting as a binary counter. The outputs will count from 000_2 to 111_2 or 0_{10} to 7_{10}
- The number of states possible in a counter is the modulus or MOD number. The figure is a MOD-8 (2^3) counter. If another FF is added it would become a MOD-16 (2^4) counter.

J-K flip-flops wired as a three-bit binary counter (MOD-8).

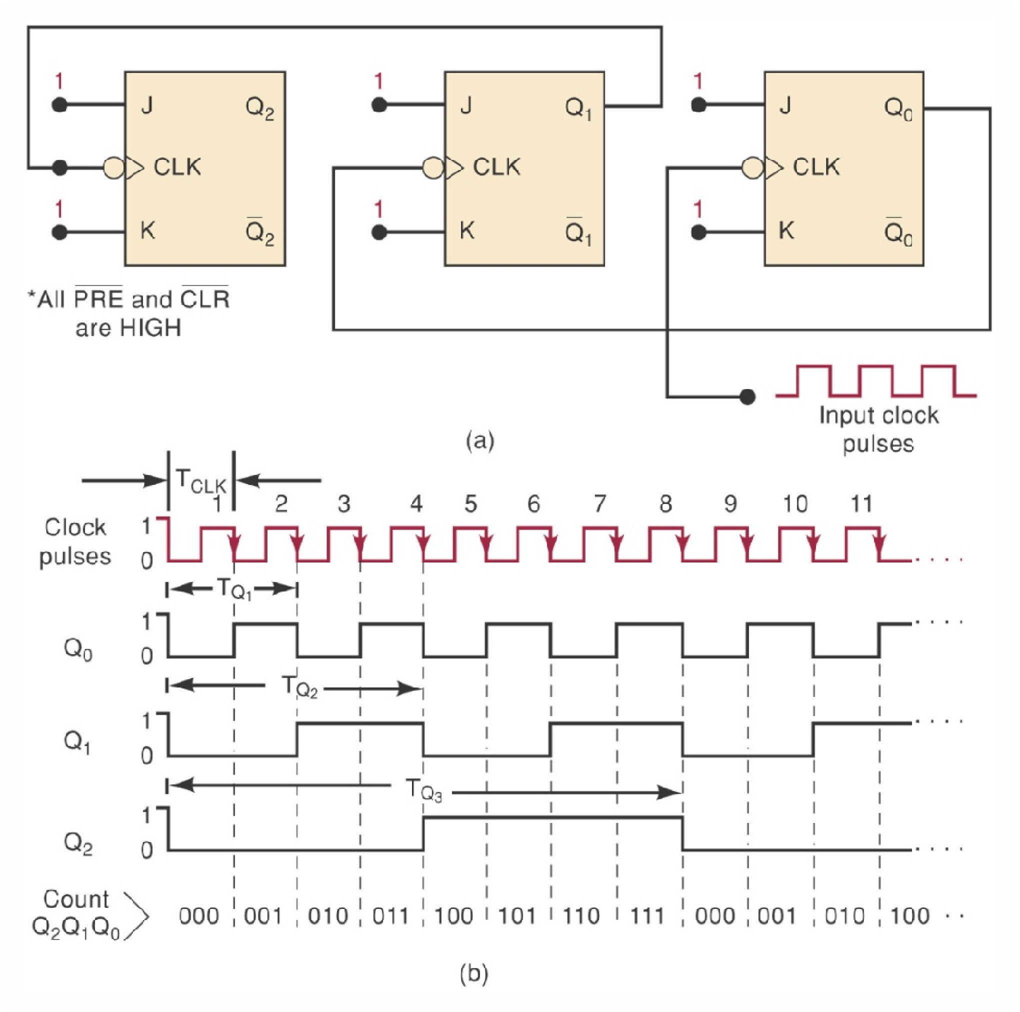
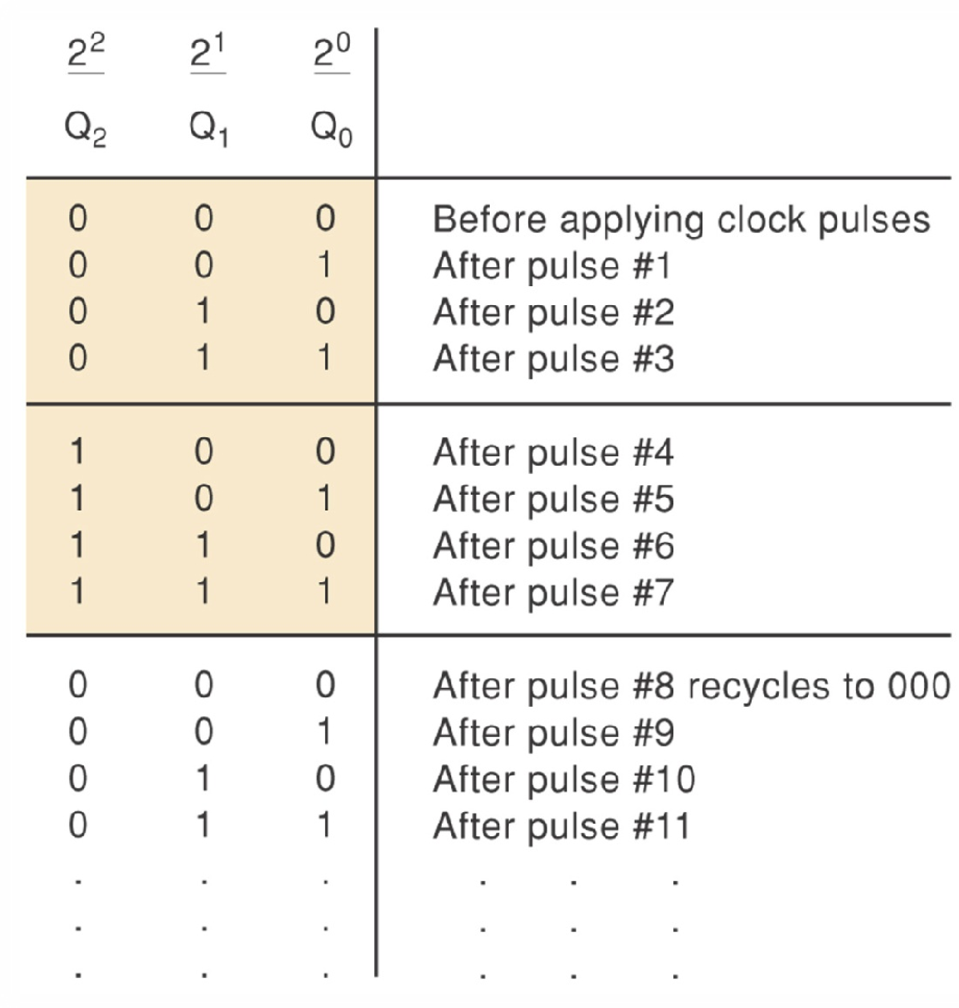
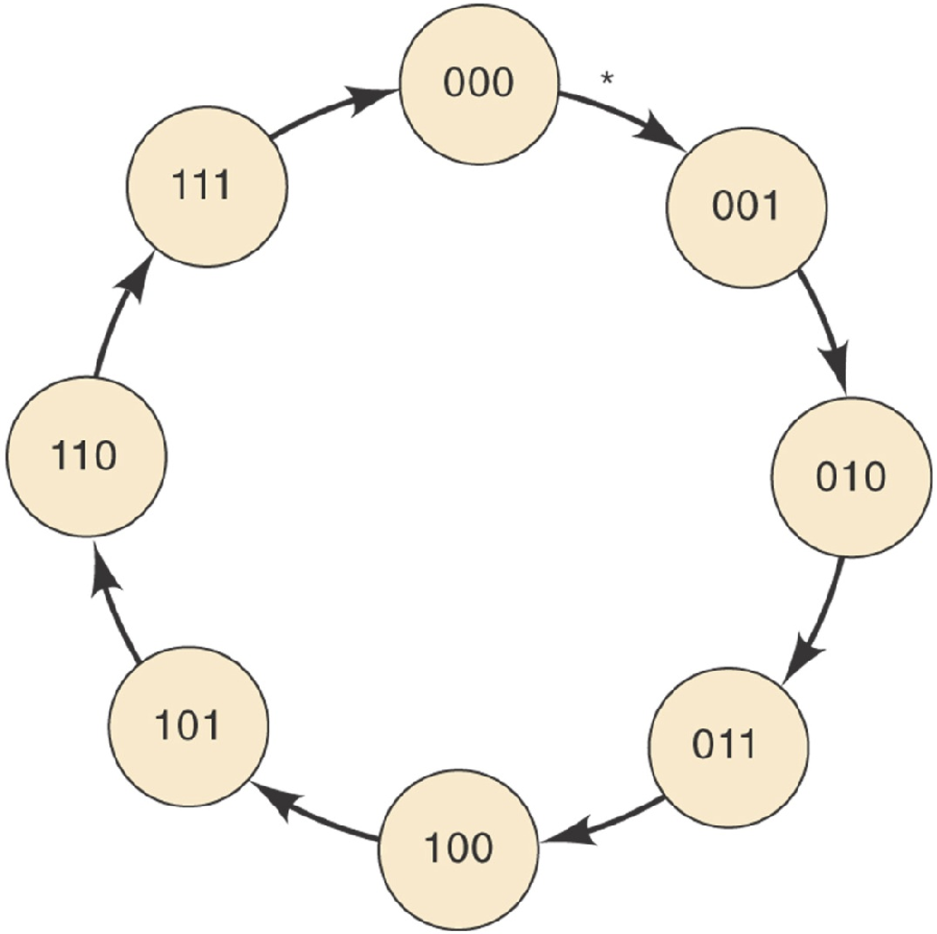


Table of flip-flop states shows binary counting sequence.



2^2	2^1	2^0	
Q_2	Q_1	Q_0	
0	0	0	Before applying clock pulses
0	0	1	After pulse #1
0	1	0	After pulse #2
0	1	1	After pulse #3
1	0	0	After pulse #4
1	0	1	After pulse #5
1	1	0	After pulse #6
1	1	1	After pulse #7
0	0	0	After pulse #8 recycles to 000
0	0	1	After pulse #9
0	1	0	After pulse #10
0	1	1	After pulse #11
.	.	.	. . .
.	.	.	. . .
.	.	.	. . .

State transition diagram shows how the states of the counter flip-flops change with each applied clock pulse.



* **Note:** each arrow represents the occurrence of a clock pulse

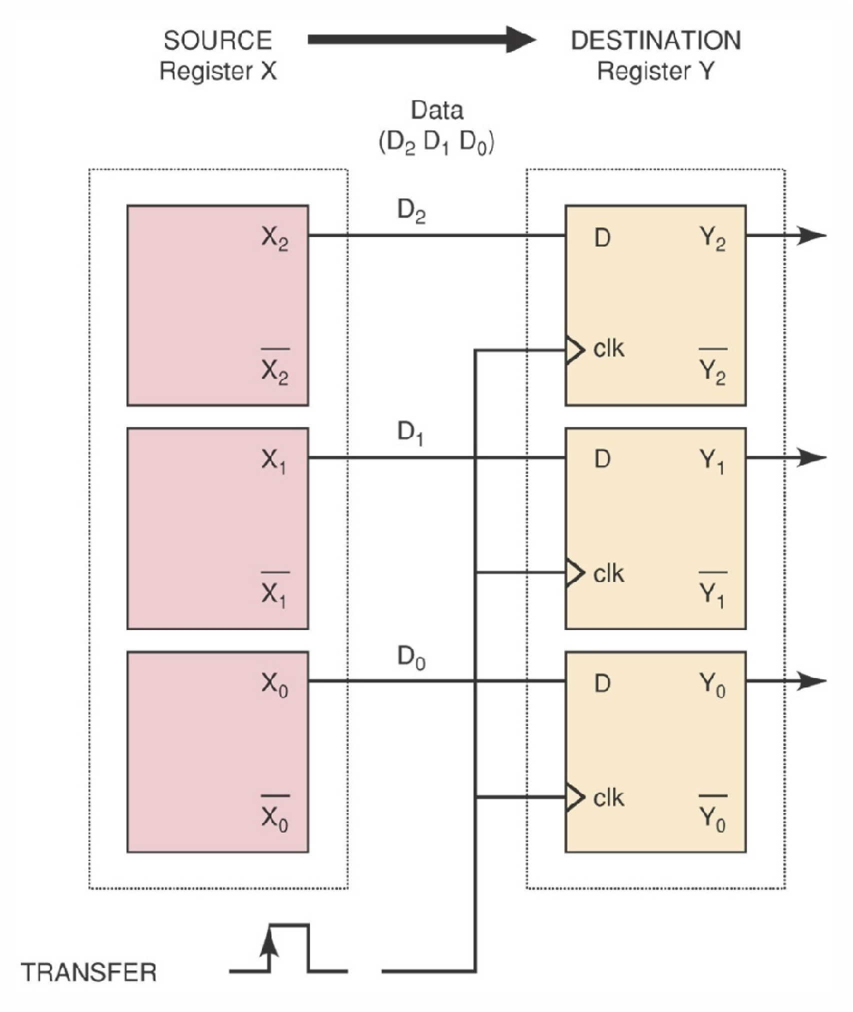
Data Storage and Transfer

- FFs are commonly used for storage and transfer of data in binary form.
- Groups of FFs used for storage are registers.
- Data transfers take place when data is moved between registers or FFs.
- Synchronous transfers take place at PGT or NGT of clock.

Data Storage and Transfer

- Asynchronous transfers are controlled by PRE and CLR inputs.
- Transferring the bits of a register simultaneously is a parallel transfer.
- Transferring the bits of a register a bit at a time is a serial transfer.

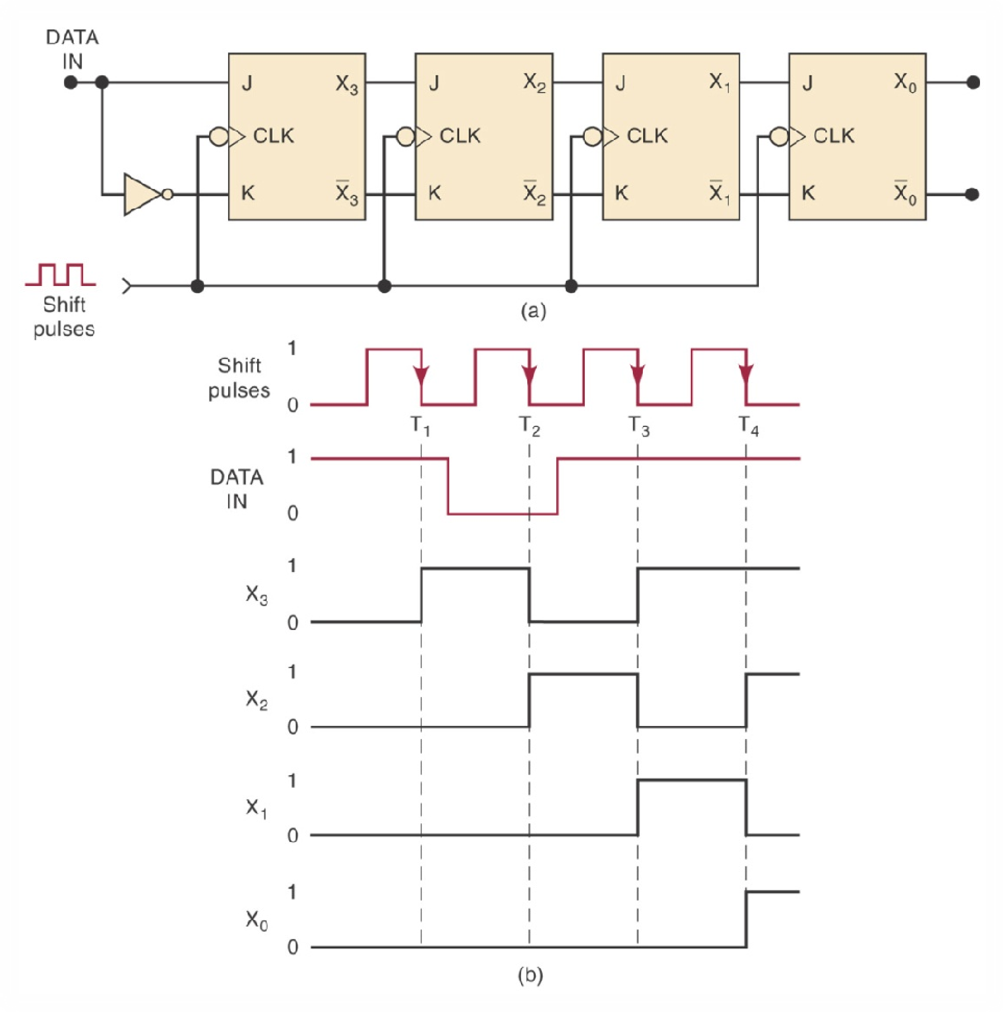
Parallel transfer of contents of register *X* into register *Y*.



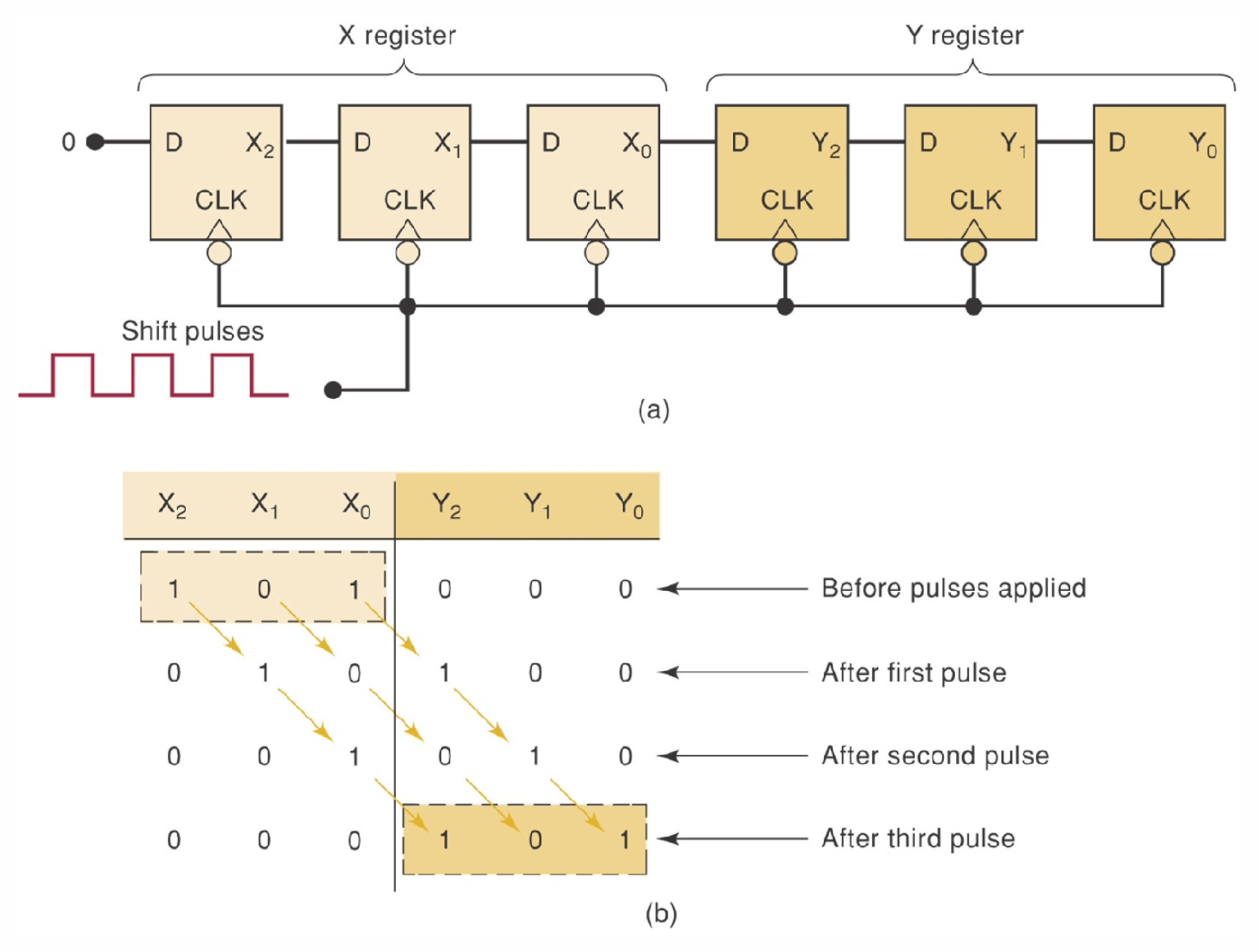
Serial Data Transfer: Shift Registers

- When FFs are arranged as a shift register, bits will shift with each clock pulse.
- The direction of data shifts will depend on the circuit requirements and the design.

Four-bit shift register.



Serial transfer of information from *X* register into *Y* register.



(a)

X_2	X_1	X_0	Y_2	Y_1	Y_0	
1	0	1	0	0	0	Before pulses applied
0	1	0	1	0	0	After first pulse
0	0	1	0	1	0	After second pulse
0	0	0	1	0	1	After third pulse

(b)

Serial Data Transfer: Shift Registers

- □ Parallel transfers – register contents are transferred simultaneously with a single clock cycle.
- □ Serial transfers – register contents are transferred one bit at a time, with a clock pulse for each bit.
- □ Serial transfers are slower, but the circuitry is simpler. Parallel transfers are faster, but circuitry is more complex.
- □ Serial and parallel are often combined to exploit the benefits of each.

More on Counter

Categories of counters

1. Ripple counters

   The flip-flop output transition serves as a source for triggering other flip-flops

   ⇒ no common clock pulse (not synchronous)

2. Synchronous counters:

   The CLK inputs of all flip-flops receive a

   common clock

Synchronous/Asynchronous Counter

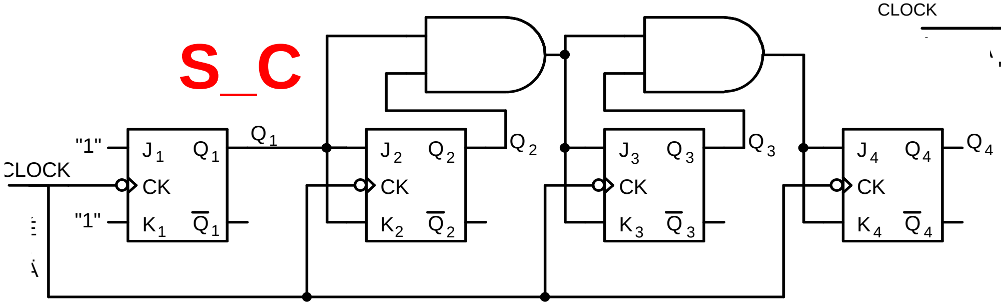
Synchronous counter:

All flip-flops in a synchronous counter receive the same clock pulse and so change state simultaneously.

Asynchronous (Ripple) counter:

Flip-flops transitions ripple through from one flip-flop to the next in sequence until all flip-flops reach a new stable value (state). Each single flip-flop stage divides the frequency of its input signal by two.

S_C

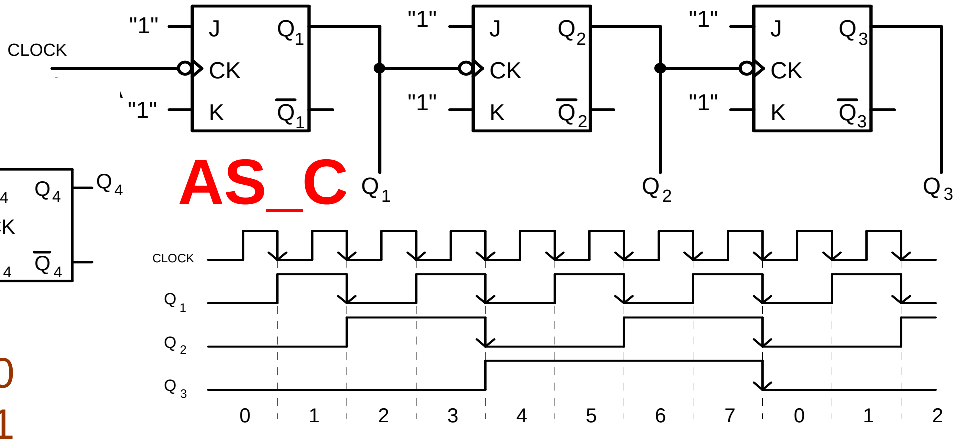


0 0 0 0

0 0 0 1

0 0 1 0

AS_C



What good are counters?

- Counters can act as simple clocks to keep track of "time."
- You may need to record how many times something has happened.
  - How many bits have been sent or received?
  - How many steps have been performed in some computation?
- All processors contain a program counter, or PC.
  - Programs consist of a list of instructions that are to be executed one after another (for the most part).
  - The PC keeps track of the instruction currently being executed.
  - The PC increments once on each clock cycle, and the next program instruction is then executed.

Some important terms

- Modulus of a counter:-the number of states trough which a counter sequences before repeating.
- Full sequence counter.
- Truncated sequence counter.

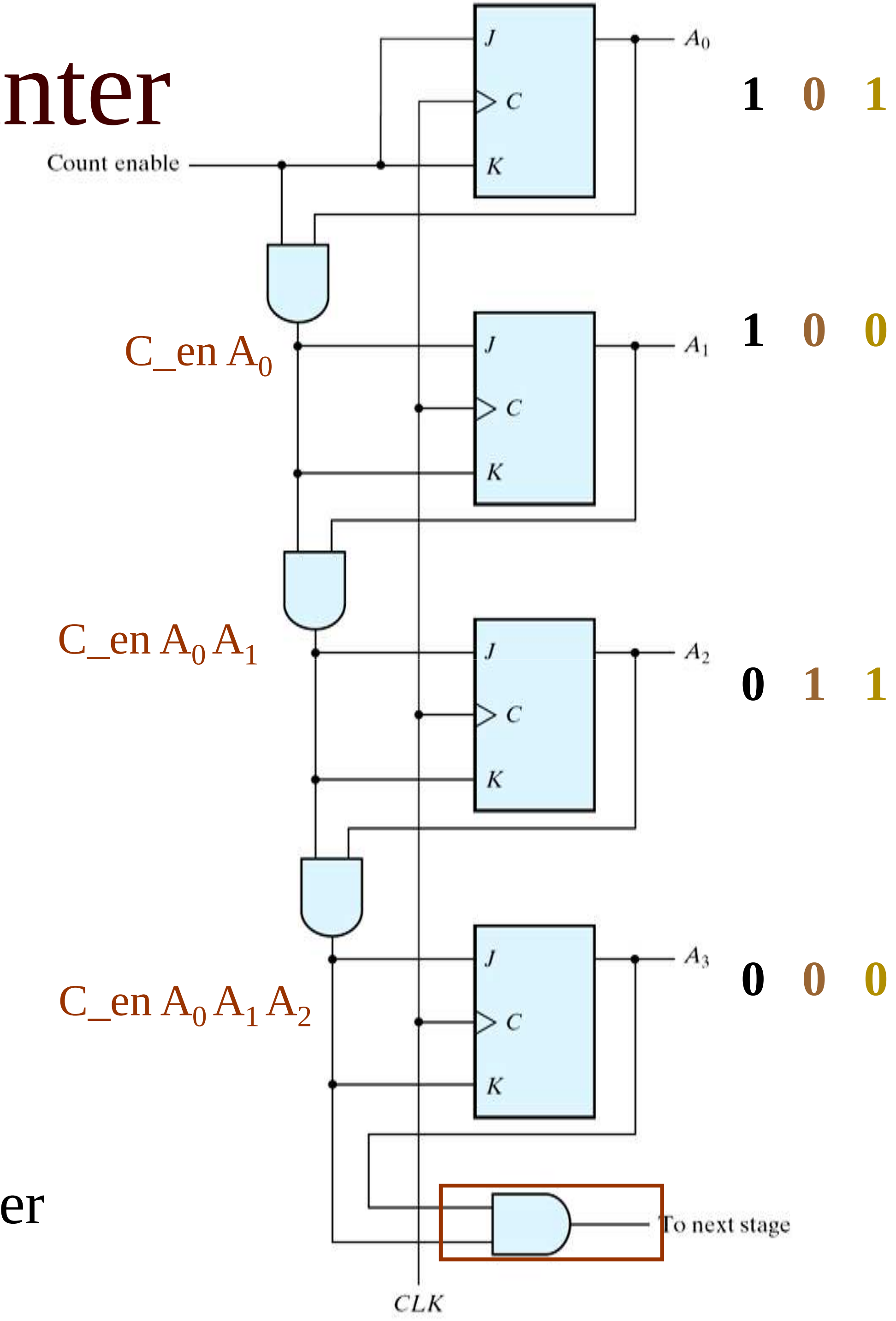
Counters

- Analysis of counter:-counter circuit is given analyzing its count sequence.
- Design of a counter:-counter behavior will be given, then designing a circuit accordingly to get expected counter behavior.

Analysis of a counter

4-bit Binary Counter

A3	A2	A1	A0
0	0	1	1
0	1	0	0



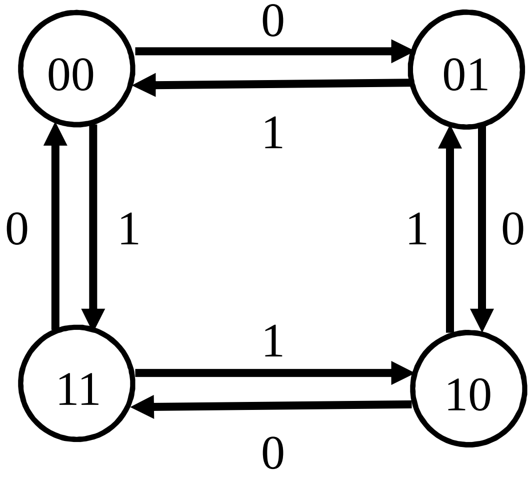
Four-bit synchronous binary counter

Design of a counter

Design of a synchronous 2 bit up/down counter using D flip flops

- □ Determine the number of flip-flops
- □ Draw state diagram
- □ Get excitation table
- □ Obtain the minimal expressions for inputs using K-map
- □ Draw the logic diagram from the expression.

The complete state diagram and table



Present State		Inputs	Next State	
Q_1	Q_0	X	Q_1	Q_0
0	0	0	0	1
0	0	1	1	1
0	1	0	1	0
0	1	1	0	0
1	0	0	1	1
1	0	1	0	1
1	1	0	0	0
1	1	1	1	0

D flip-flop inputs

- If we use D flip-flops, then the D inputs will just be the same as the desired next states.
- Equations for the D flip-flop inputs are shown at the right.

Present State		Inputs	Next State		Excitation IP	
Q_1	Q_0	X	Q_1	Q_0	D1	D0
0	0	0	0	1	0	1
0	0	1	1	1	1	1
0	1	0	1	0	1	0
0	1	1	0	0	0	0
1	0	0	1	1	1	1
1	0	1	0	1	0	1
1	1	0	0	0	0	0
1	1	1	1	0	1	0

$$D_1 = Q_1 \oplus Q_0 \oplus X$$

$$D_0 = Q_0'$$

Flip-Flop Excitation Tables

Excitation Table:- the excitation table specifies the values for the flip-flop input signals needed to cause the transitions in the transition table.

Present State	Next State	F.F. Input
$Q(t)$	$Q(t+1)$	D
0	0	0
0	1	1
1	0	0
1	1	1

Present State	Next State	F.F. Input	
$Q(t)$	$Q(t+1)$	J	K
0	0	0	x
0	1	1	x
1	0	x	1
1	1	x	0

0 0 (No change)
0 1 (Reset)

1 0 (Set)
1 1 (Toggle)

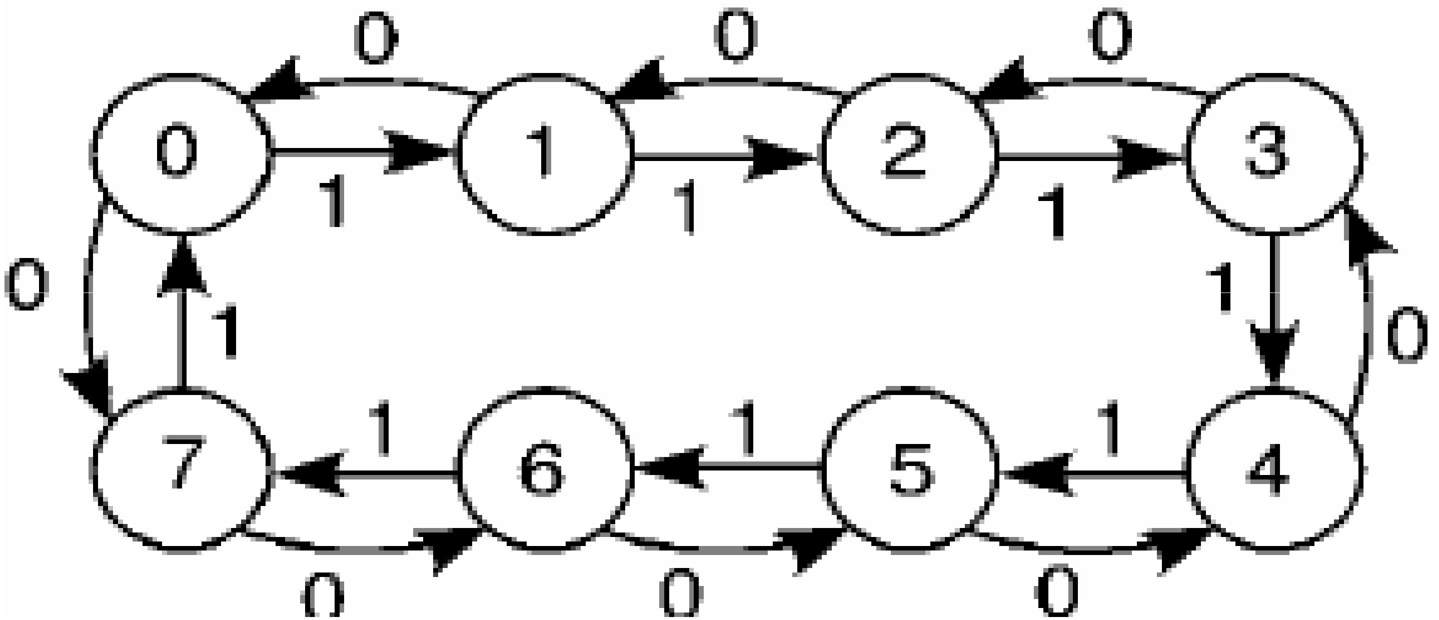
0 1 (Reset)
1 1 (Toggle)

0 0 (No change)
1 0 (Set)

Design of a synchronous 3 bit up/down counter using JK flip flops

- Determine the number of flip-flops
- Draw state diagram
- Get excitaion table
- Obtain the minimal expressions for inputs
- Draw the logic diagram

- State diagram



Excitation table

Present state			Mode	Next state			Required inputs					
Q3	Q2	Q1	M	Q3	Q2	Q1	J3	K3	J2	K2	J1	K1
0	**0**	**0**	**0**	**1**	**1**	**1**	**1**	**X**	**1**	**X**	**1**	**X**
0	0	0	1	0	0	1	0	X	0	X	1	X
0	0	1	0	0	0	0	0	X	0	X	X	1
0	0	1	1	0	1	0	0	X	1	X	X	1
0	1	0	0	0	0	1	0	X	X	1	1	X
0	1	0	1	0	1	1	0	X	X	0	1	X
0	1	1	0	0	1	0	0	X	X	0	X	1
0	1	1	1	1	0	0	1	X	X	1	X	1
1	0	0	0	0	1	1	X	1	1	X	1	X
1	0	0	1	1	0	1	X	0	0	X	1	X
1	0	1	0	1	0	0	X	0	0	X	X	1
1	0	1	1	1	1	0	X	0	1	X	X	1
1	1	0	0	1	0	1	X	0	X	1	1	X
1	1	0	1	1	1	1	X	0	X	0	1	X
1	1	1	0	1	1	0	X	0	X	0	X	1
1	1	1	1	0	0	0	X	1	X	1	X	1

* PS=Present state   *NS= Next state

Present State	Next State	F.F. Input	
$Q(t)$	$Q(t+1)$	J	K
0	0	0	x
0	**1**	**1**	**x**
1	0	x	1
1	1	x	0

Q3(PS)Q3(NS) = 01 then from excitation table of JK flip flop J3K3 should be 1X(J3=1 K3=X)
Q2(PS)Q2(NS)= 01 then from excitation table of JK flip flop J2K2 should be 1X(J2=1 K2=X)
Similarly fill all the rows of excitation inputs for all states of counter

K-maps for excitations of synchronous 3 bit U/D counter

Q1M

Q3Q2

1			
		1	
X	X	X	X
X	X	X	X

J3=Q2’Q1’M’+Q2Q1M

Q1M

Q3Q2

X	X	X	X
X	X	X	X
		1	
1			

K3=Q2’Q1’M’+Q2Q1M

Q1M

Q3Q2

1		1	
X	X	X	X
X	X	X	X
1		1	

J2=Q1’M’+Q1M

J1=1

Q1M

Q3Q2

X	X	X	X
1		1	
1		1	
X	X	X	X

K2=Q1’M’+Q1M

k1=1

Draw the final circuit of 3bit UP/DOWN counter using JK flip flop

□ Work for you