

UNIT - IV

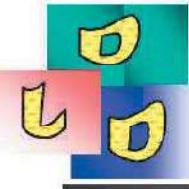
SYNCHRONOUS SEQUENTIAL LOGIC

P. Gopala Krishna

UNIT-IV/DIGITAL LOGIC DESIGN/IT II-I Sem/GRIET

P. Gopala Krishna

1



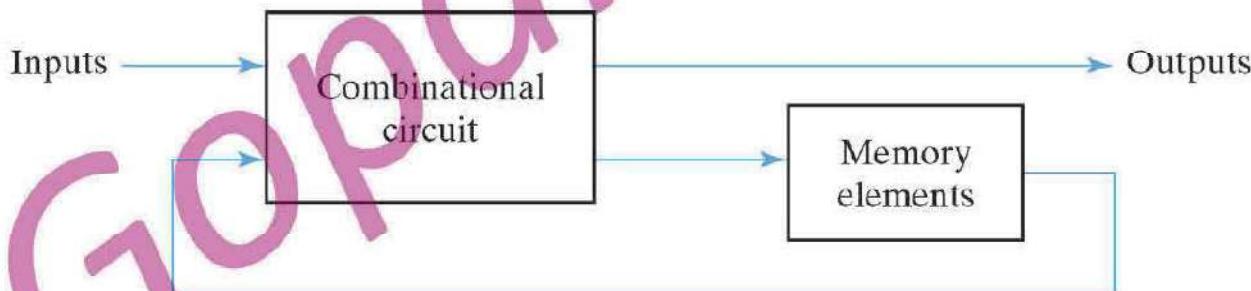
Sequential Circuits

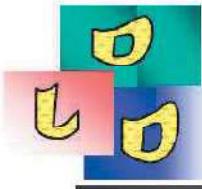
- The digital circuits considered thus far have been combinational, where the outputs are entirely dependent on the current inputs.
- Although every digital system is likely to have combinational circuits, most systems encountered also include storage elements, which require that the system be described in terms of sequential logic.



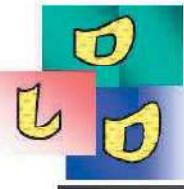
Block Diagram of a sequential circuit

- A sequential circuit consists of a combinational circuit to which storage elements are connected to form a feed back path.
- The storage elements are devices capable of storing binary information.
- The binary information stored in these elements at any given time defines the state of the sequential circuit at that time.



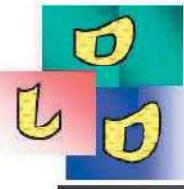


- The sequential circuits receives binary information from external inputs.
- These inputs together with the present state of the storage elements, determine the binary value of the outputs.
- So, the outputs in sequential circuit are a function not only of the inputs, but also of the present state of the storage elements.
- The next state of the storage element is also a function of external inputs and the present state.

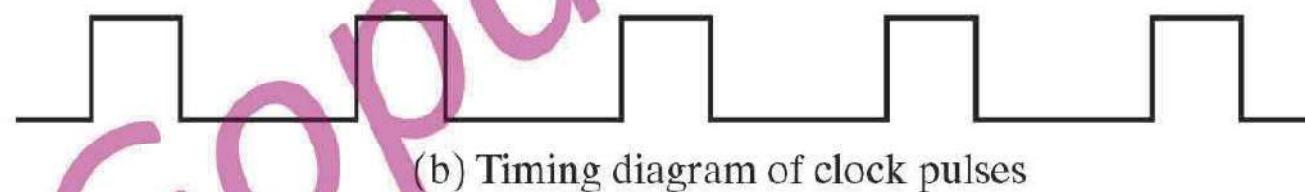
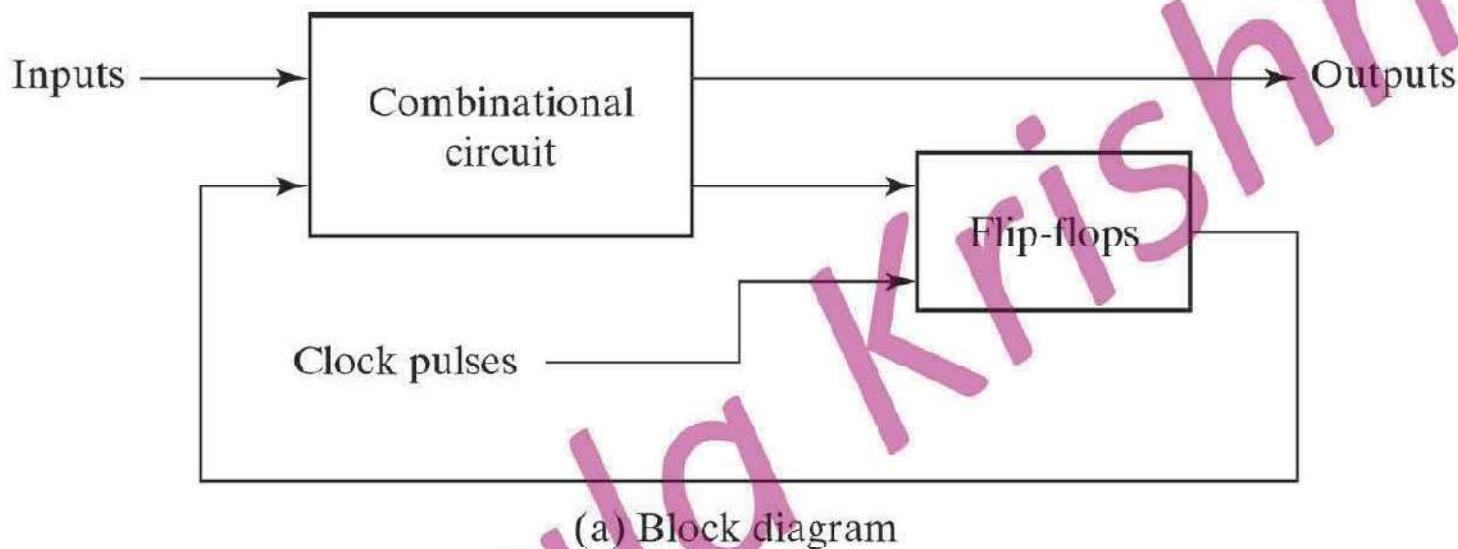


Types of sequential circuits

- There are two main types of sequential circuits and their classification depends on the timing of their signals.
- A Synchronous sequential circuit is a system whose behavior can be defined from the knowledge of its signals at discrete instants of time.
- The behavior of Asynchronous sequential circuit depends upon the input signals at any instant of time and the order in which the inputs change.
- Synchronous sequential circuits that use clock pulses in the inputs of storage elements are called clocked sequential circuits.
- The storage elements used in clocked sequential circuits are called **flip flops**.



Synchronous clocked sequential circuits

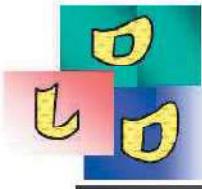


(b) Timing diagram of clock pulses



Latches

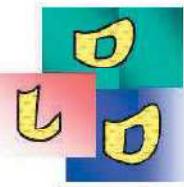
- The most basic types of flip flops operate with signal levels and are referred to as latches.
- Although latches are useful for storing information and for the design of asynchronous sequential circuits, they are not practical for use in synchronous sequential circuits.



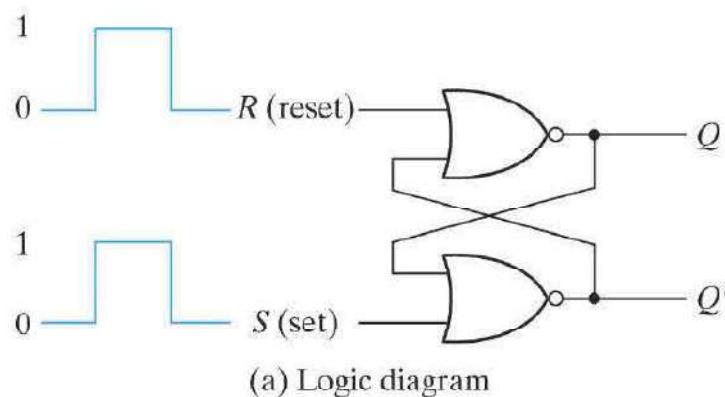
SR Latch

- The SR Latch is a circuit with two cross coupled NOR gates or two cross coupled NAND gates.
- Under normal conditions, both inputs of the latch remain at 0 unless the state has to be changed.

P. Gopala Krishna



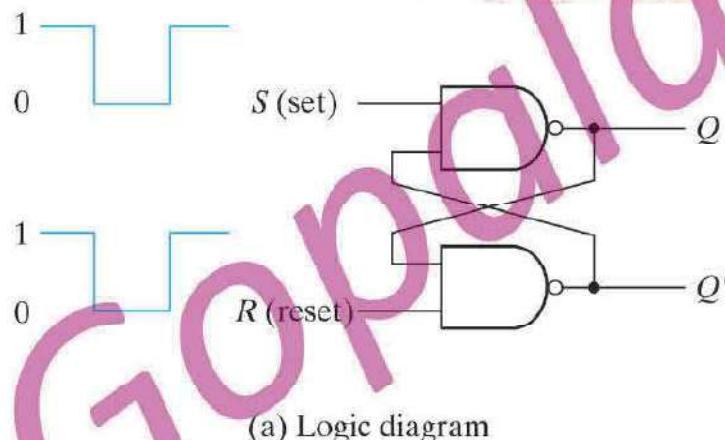
SR Latch using NOR & NAND Gates



(b) Function table

S	R	Q	Q'
1	0	1	0
0	0	1	0
0	1	0	1
0	0	0	1
1	1	0	0

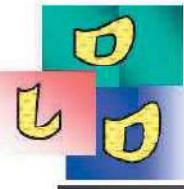
(after S = 1, R = 0)
(after S = 0, R = 1)



(b) Function table

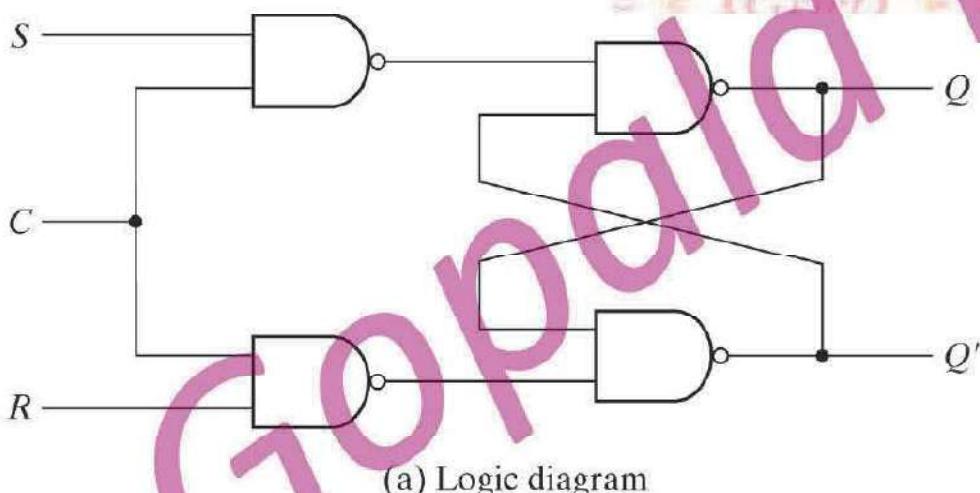
S	R	Q	Q'
1	0	0	1
1	1	0	1
0	1	1	0
1	1	1	0
0	0	1	1

(after S = 1, R = 0)
(after S = 0, R = 1)



SR Latch with control input

- The operation of the basic SR Latch can be modified by providing an additional control input that determines when the state of the latch can be changed.



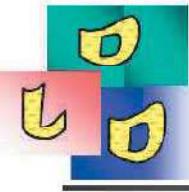
C	S	R	Next state of Q
0	X	X	No change
1	0	0	No change
1	0	1	$Q = 0$; Reset state
1	1	0	$Q = 1$; set state
1	1	1	Indeterminate

(b) Function table

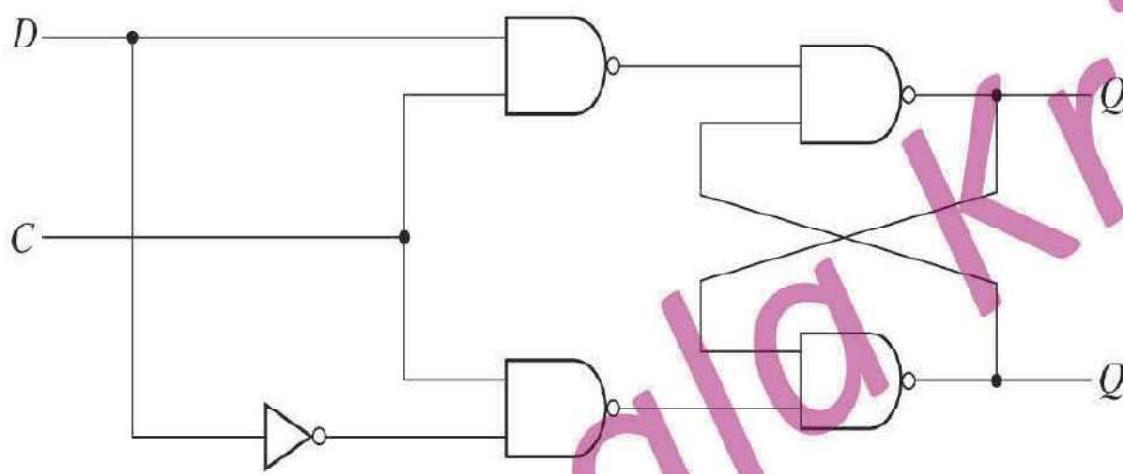


D Latch

- One way to eliminate the undesirable condition of the indeterminate state in the SR latch is to ensure that inputs S and R never equal to 1 at the same time.
- This done in D latch. This latch has only two inputs: D and C. The D input goes directly to the S input and its complement is connected to the R input.
- As long as the control input is at 0, the circuit cannot change its state regardless of the value of D. The D input is sampled when the value of the control input is 1.



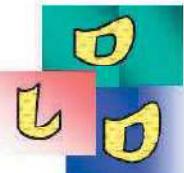
D Latch



(a) Logic diagram

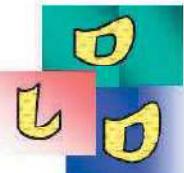
C	D	Next state of Q
0	X	No change
1	0	$Q = 0$; Reset state
1	1	$Q = 1$; Set state

(b) Function table



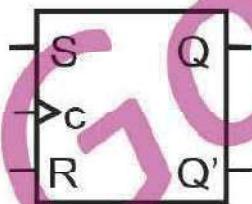
Flip Flops

- The storage elements used in clocked sequential circuits are called **flip flops**.
- A flip flop is a binary cell capable of storing one bit of information.
- It has two outputs, one for the normal value and one for the complement values of the bit stored in it.
- A flip flop maintains a binary state until directed by a clock pulse to switch states.
- The difference among various types of flip flops are in the number of inputs they possess and in the manner in which the inputs affect the binary state. The most common types of flip flops are:
 - SR Flip Flop
 - D Flip Flop
 - JK Flip Flop
 - T Flip Flop

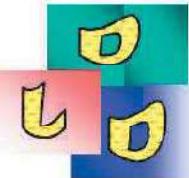


SR Flip Flop

- The SR Flip Flop has three inputs, labeled S (for set), R (for Reset), and C (for clock).
- It has an output Q and sometimes the flip flop has a complemented output, which is indicated with a small circle at the other output terminal.
- There is an arrowhead-shaped symbol in front of the letter C to designate a dynamic input this indicates the fact that the flip flop responds to a positive transition of the input clock signal.

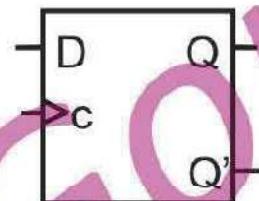


S	R	Q(t+1)	
0	0	Q(t)	No change
0	1	0	Clear to 0
1	0	1	Set to 1
1	1	?	Indeterminate

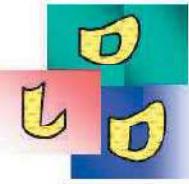


D Flip Flop

- The D flip flop is a slight modification to the SR Flip Flop.
- An SR Flip Flop is converted to D flip flop by inserting an inverter between S and R and assigning the Symbol D to the single input.
- The D input is sampled during the occurrence of a clock transition from 0 to 1.
- If $D=1$, the output of the flip flop goes to 1 state, but if $D=0$, the output of the flip flop goes to the 0 state.

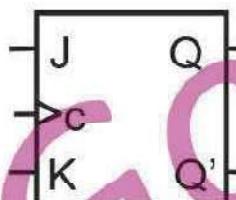


D	Q(t+1)	
0	0	Clear to 0
1	1	Set to 1

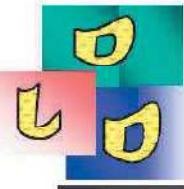


JK Flip Flop

- A JK flip flop is a refinement of the SR flip flop in that the indeterminate condition of the SR flip flop is defined in the JK type. Inputs J and K behave like inputs S and R to set and clear the flip flop, respectively. When the input J and K are both equal to 1, a clock transition switches the outputs of the flip flop to their complement state.

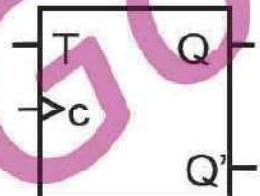


J	K	Q(t+1)	
0	0	Q(t)	No change
0	1	0	Clear to 0
1	0	1	Set to 1
1	1	$Q^1(t)$	Complement

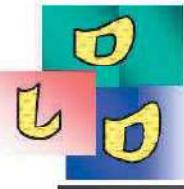


T Flip Flop

- This flip flop is obtained from JK flip flop when inputs J and K are connected to provide a single input designated by T.
- The T flip flop therefore has only two conditions. When $T=0$ a clock transition does not change the state of the flip flop. When $T=1$ a clock transition complements the state of the flip flop.

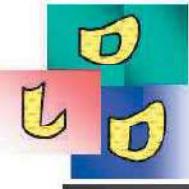


T	Q(t+1)	
0	Q(t)	NO CHANGE
1	Q'(t)	COMPLEMENT



Characteristic Equations

- The logical properties of a flip flop as described in the characteristic table can be expressed also algebraically with a characteristic equation.
 - For D flip flop ---- $Q(t+1)=D$
 - For JK flip flop --- $Q(t+1)=JQ^1+K^1Q$
 - For T flip flop ---- $Q(t+1)= TQ^1+T^1Q$
- Where Q is the value of the flip flop output prior to the application of a clock edge.



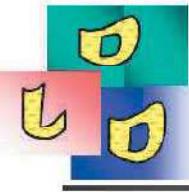
Analysis Procedure

- The behavior of a clocked sequential circuit is determined from the inputs, the outputs, and the state of its flip flops.
- The outputs and next state are both a function of the inputs and the present state.
- The analysis of a sequential circuit consists of obtaining a table or a diagram for the time sequence of inputs, outputs and internal states.
- It is also possible to write Boolean Expressions that describe the behavior of the sequential circuits.

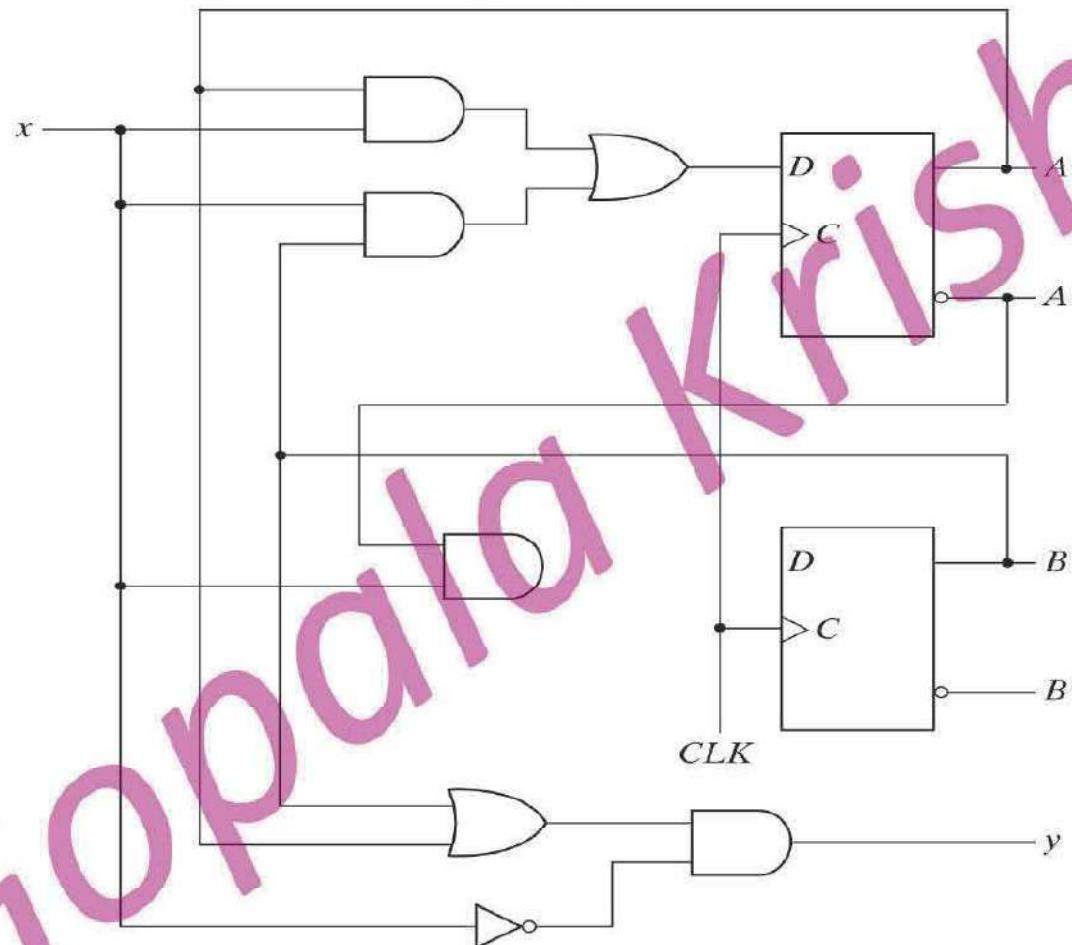


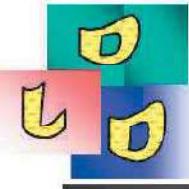
State Equations

- The behavior of a clocked sequential circuit can be described algebraically by means of state equations.
- A state equation specifies the next state as a function of the present state and inputs.
- A state equation is an algebraic expression that specifies the condition for a flip flop state transition.

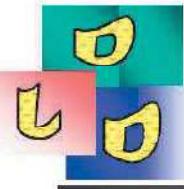


Example 1



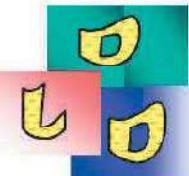


- $A(t+1)=A(t)x(t) + B(t)x(t)$
- $B(t+1)=A^1(t)x(t)$
- $y(t)=[A(t)+B(t)]x^1(t)$
- The left side of the equation with $(t+1)$ denotes the next state of the flip flop one clock edge later. The right side of the equation is a Boolean expression that specifies the present state and input conditions that make the next equal to 1.
- They can be formally written as:
- $A(t+1)=Ax + Bx$
- $B(t+1)=A^1x$
- $y(t)=[A+B]x^1$



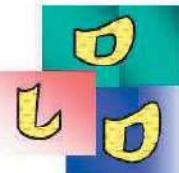
State Table

- The time sequence of inputs, outputs and flip flop states can be enumerated in a state table.
- The table consists of four sections named present state, input, next state, and output.
- The present state section shows the states of flip flops A and B at any given time t.
- The input section gives the value of input for each possible present state.
- The next state section shows the states of the flip flops one clock cycle later at time t+1.
- The derivation of a state table requires listing all possible binary combinations of present state and inputs.
- The next state values are then determined from the logic diagram or from the state equations.



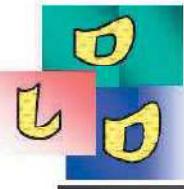
State Table

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	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



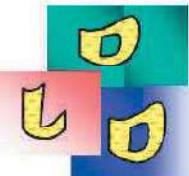
Another form of State Table

Present State		Next State				Output	
A	B	X=0		X=1		X=0	X=1
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



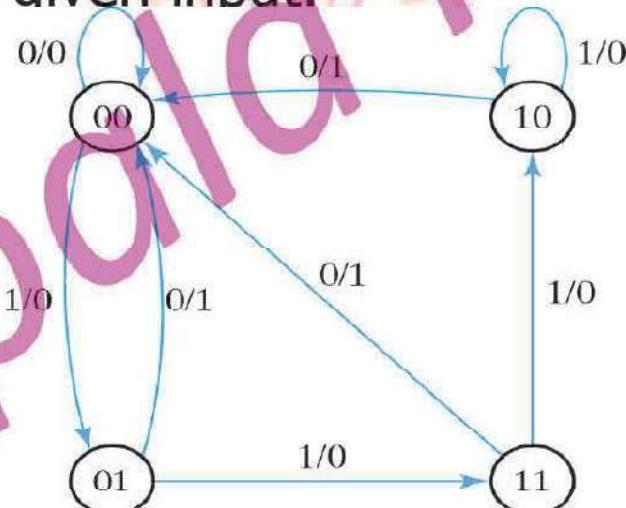
State Diagram

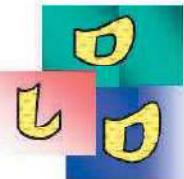
- The information available in a state table can be represented graphically in the form of a state diagram.
- In this representation a state is represented by a circle, and the transitions between states are indicated by directed lines connecting the circles.
- The state diagram provides the same information as the state table and is obtained directly from state table.



Example 2

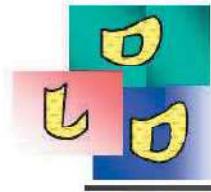
- The binary number inside each circle identifies the state of the flip flops.
- The directed lines are labeled with two binary numbers separated by slash.
- The input value during the present state is listed first and the number after the slash gives the output during the present with the given input.



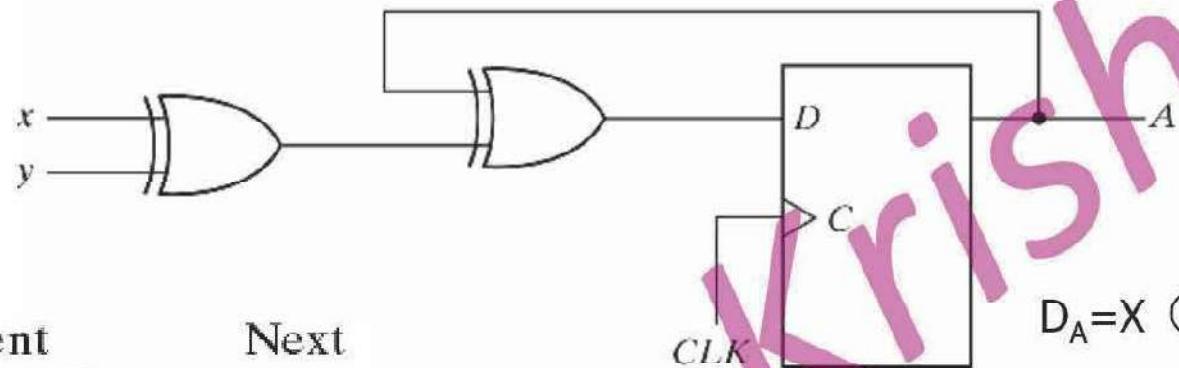


Flip Flop Input Equations

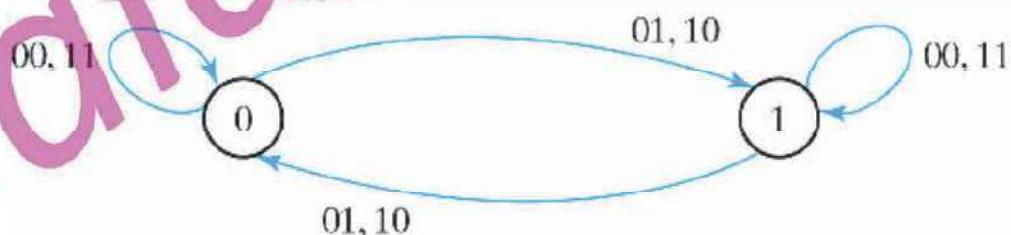
- The logic diagram of a sequential circuit consists of flip flops and gate.
- The part of the combinational circuit that generates external outputs is described algebraically by set of boolean functions named **output equations**.
- The part of the circuit that generates inputs to flip flops is described algebraically by set of boolean functions named **flip flop input equations** or **excitation equations**.
- We adopt the convention of using the flip flop input symbol to denote the input equation variable and a subscript to designate the name of the flip flop output.
- For example, $D_Q = x+y$, the input equation specifies an OR gate with inputs x and y connected to the D input of a flip flop whose output is labeled with the symbol Q.

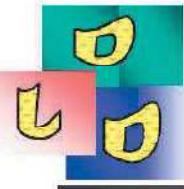


Analysis with D Flip-Flops



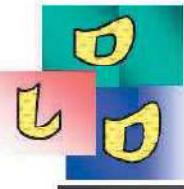
Present state	Inputs		Next state
A	x	y	A
0	0	0	0
0	0	1	1
0	1	0	1
0	1	1	0
1	0	0	1
1	0	1	0
1	1	0	0
1	1	1	1





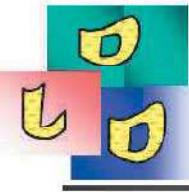
Analysis with JK Flip Flops

- A state table consists of four sections: present state, inputs, next state, and outputs.
- The first two are obtained by listing all the binary combinations. The output section is determined from the output equations.
- The next state values are evaluated from the state equations.
- For a D type flip flop, the state equation is the same as the input equation.
- When other types of flip flops like JK or T are used, it is necessary to refer to their characteristic table or characteristic equation.

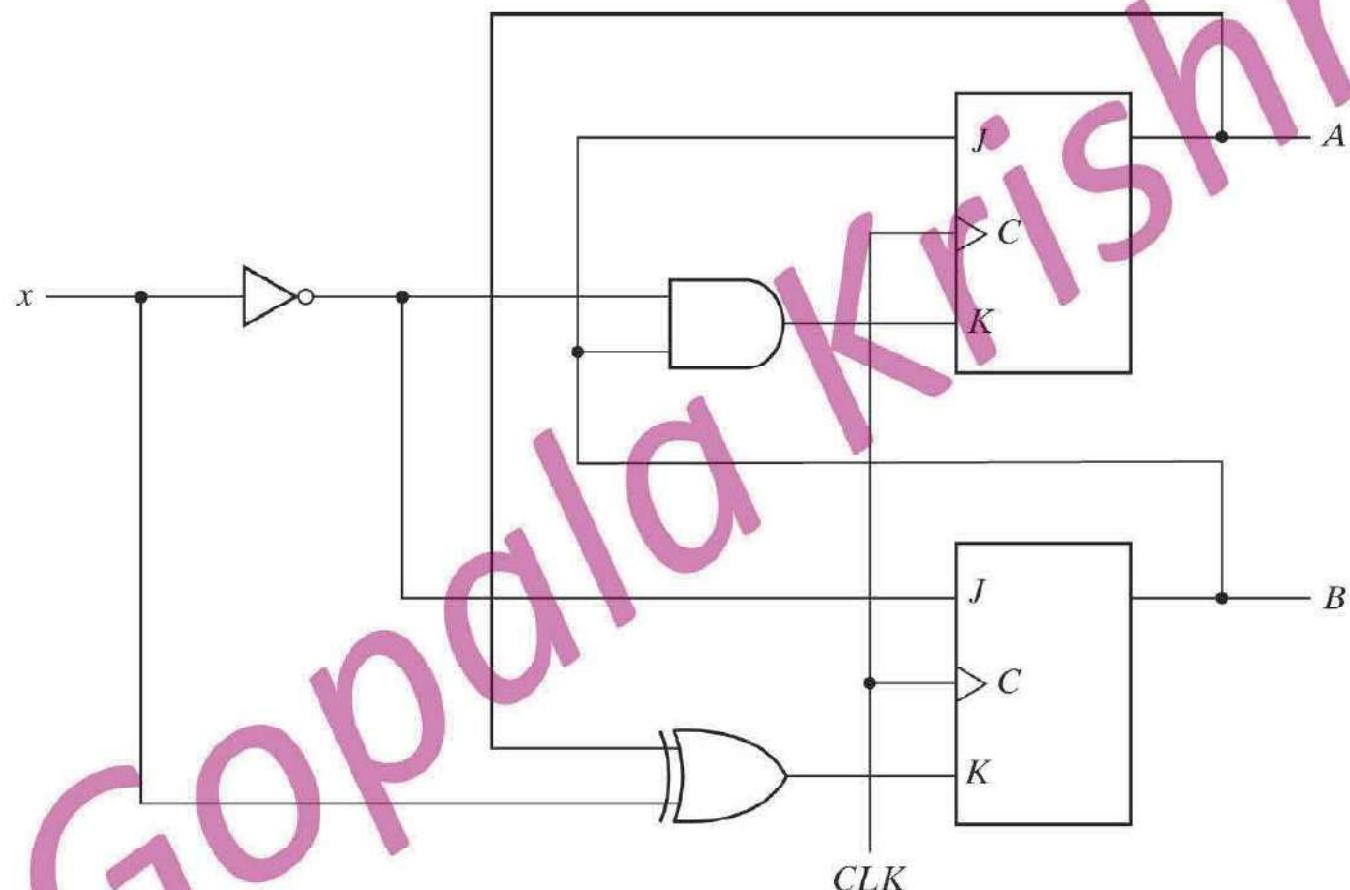


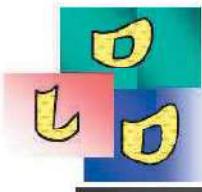
Next state values-characteristic table

- The next state values of a sequential circuit that uses flip flops such JK or T can be derived using characteristic table by the following procedure:
 - Determine the flip flop input equations in terms of the present state and input variables.
 - List the binary values of each input equation.
 - Use the corresponding flip flop characteristic table to determine the next state values in the state table.



Example 3

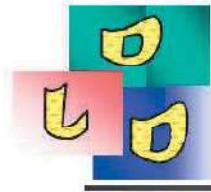




- The sequential circuit contains two JK flip flops A and B and one input x, and has no outputs and therefore the state table does not contain an output column.
- The circuit can be specified by the flip flop input equations:

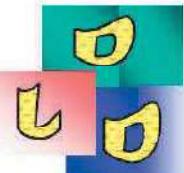
$$J_A = B \quad K_A = Bx^1$$

$$J_B = x^1 \quad K_B = A^1X + AX^1$$



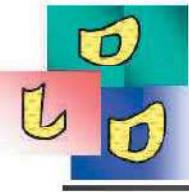
Present state		Input	Next state		Flip Flop Input Equations			
A	B	X	A	B	J_A	K_A	J_B	K_B
0	0	0	0	1	0	0	1	0
0	0	1	0	0	0	0	0	1
0	1	0	1	1	1	1	1	0
0	1	1	1	0	1	0	0	1
1	0	0	1	1	0	0	1	1
1	0	1	1	0	0	0	0	0
1	1	0	0	0	1	1	1	1
1	1	1	1	1	1	0	0	0



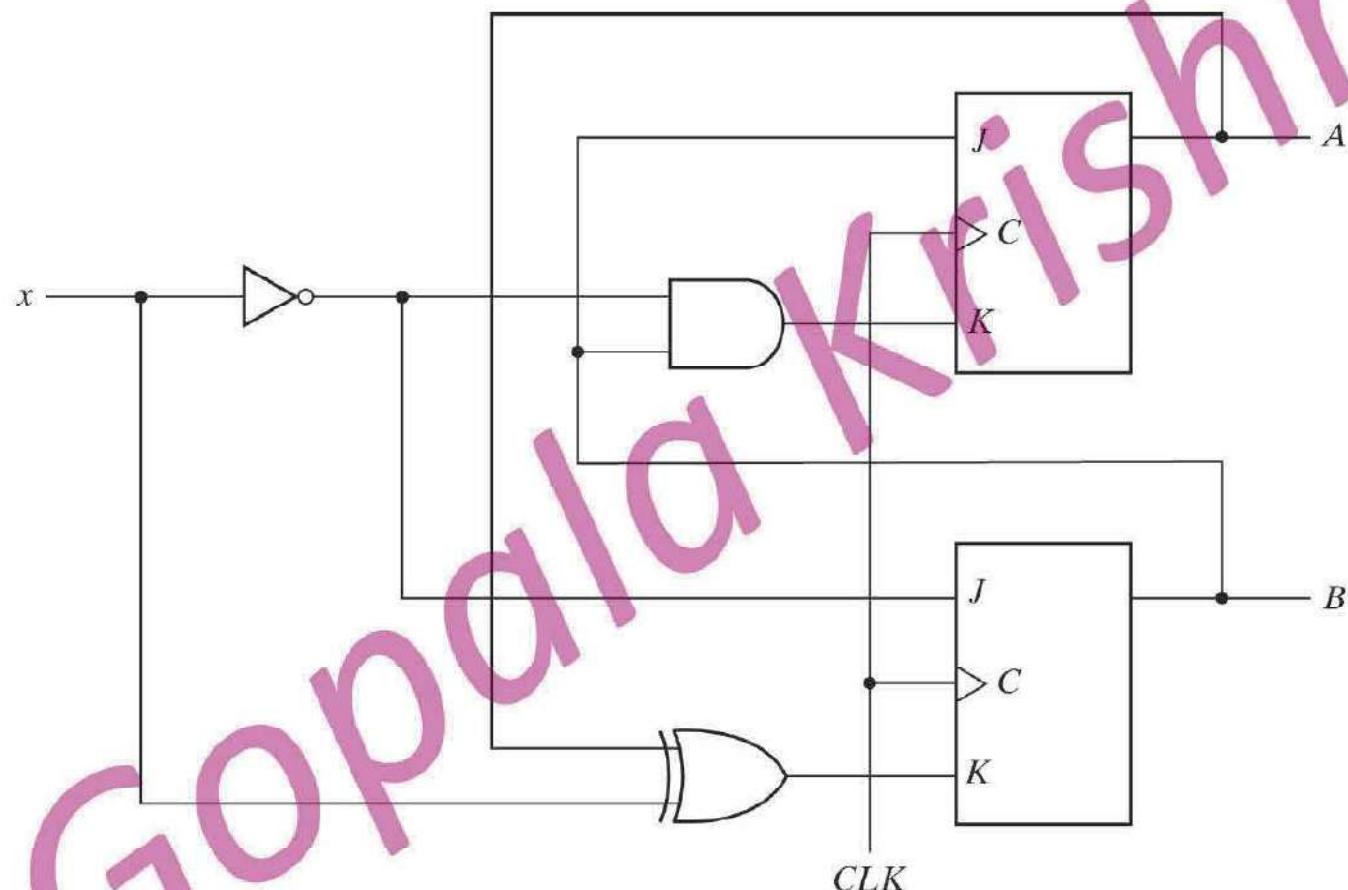


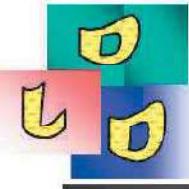
Next state values-characteristic equation

- The next state values of a sequential circuit that uses flip flops such JK or T can be derived using characteristic equation by the following procedure:
 - Determine the flip flop input equations in terms of the present state and input variables.
 - Substitute the input equations into the flip flop characteristic equation to obtain the state equations.
 - Use the corresponding state equations to determine the next state values in the state table.



Example 4

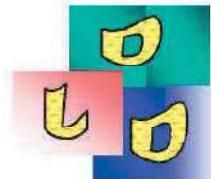




- The circuit can be specified by the flip flop input equations:
$$\begin{array}{ll} J_A = B & K_A = Bx^1 \\ J_B = x^1 & K_B = A^1X + AX^1 \end{array}$$
- The characteristic equation of JK Flip Flop is $Q(t+1) = JQ^1 + K^1Q$.
- The characteristic equations for the flip flops in the circuit are obtained by substituting the names of the flip flops instead of Q in the characteristic equation.
- So, we obtain:
$$\begin{array}{l} A(t+1) = JA^1 + K^1A \\ B(t+1) = JB^1 + K^1B \end{array}$$
- Substituting the values of JA and KA from the input equations, we obtain the state equation for A:

$$A(t+1) = JA^1 + K^1A = BA^1 + (Bx^1)^1A = A^1B + AB^1 + Ax$$

$$B(t+1) = JB^1 + K^1B = x^1B^1 + (A^1X + AX^1)^1B = B^1x^1 + ABx + A^1Bx^1$$

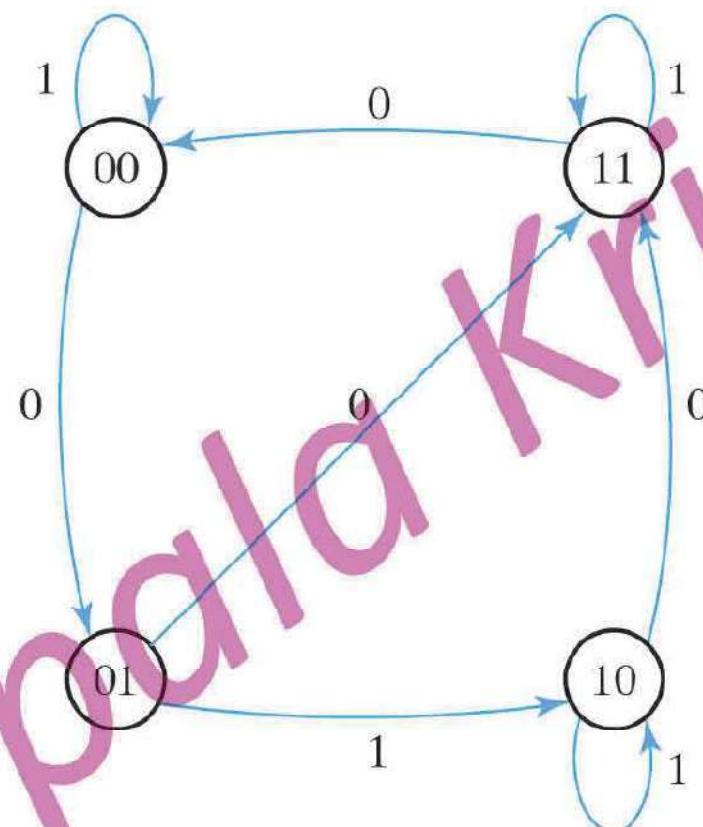


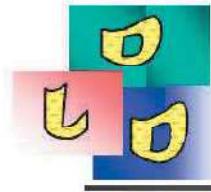
Example 4 - State Table

Present state		Input	Next state	
A	B	X	A(t+1)	B(t+1)
0	0	0	0	1
0	0	1	0	0
0	1	0	1	1
0	1	1	1	0
1	0	0	1	1
1	0	1	1	0
1	1	0	0	0
1	1	1	1	1



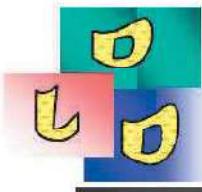
Example 4-State Diagram



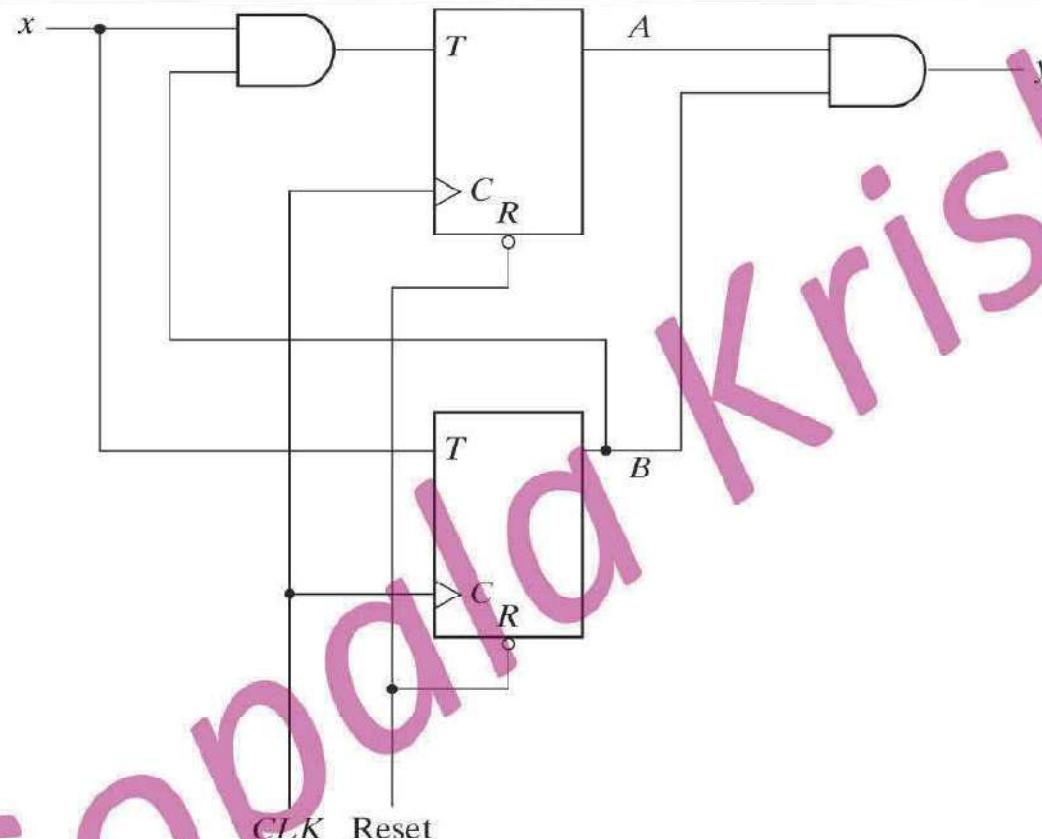


Analysis with T Flip Flops

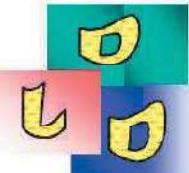
- The analysis of a sequential circuit with T flip flop follows the same procedure outlined for JK flip flops. The next state values in the state table can be obtained either by using the characteristic table or characteristic equation.
- The characteristic equation for T flip flop is:
$$Q(t+1) = TQ^1 + \bar{T}Q$$



Example 5



P. Gopala Krishna



- The sequential circuit contains two T flip flops A and B and one input x and one output y.
- The circuit can be specified by the flip flop input equations and output equation:

$$T_A = Bx$$

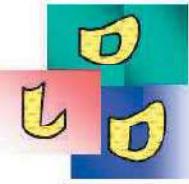
$$T_B = x$$

$$y = AB$$

- The values for y are obtained from the output equation.
- The values for the next state can be derived from the state equations by substituting T_A and T_B in the characteristic equations, yielding

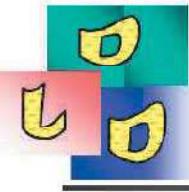
$$A(t+1) = T^1 A + T A^1 = (Bx)^1 A + (Bx) A^1 = AB^1 + Ax^1 + A^1 Bx$$

$$B(t+1) = x \oplus B$$

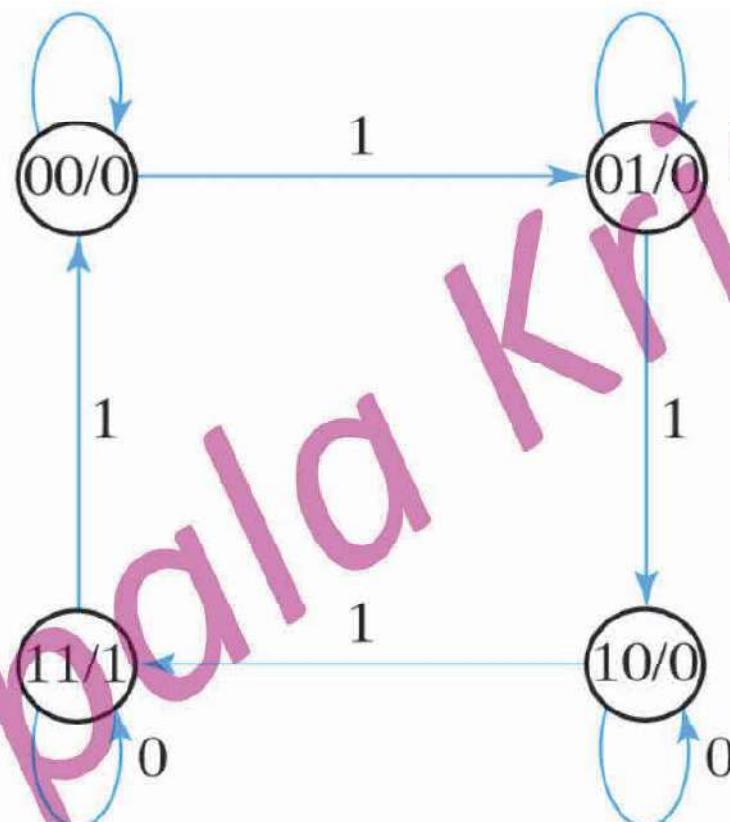


Example 5 - State table

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	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	



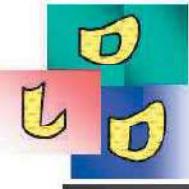
Example 5 – State Diagram





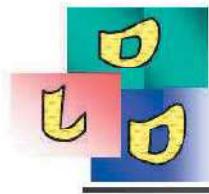
Mealy and Moore Models

- The most general model of a sequential circuit has inputs, outputs, and internal states.
- There are two models of the sequential circuits:
 - Mealy model
 - Moore model

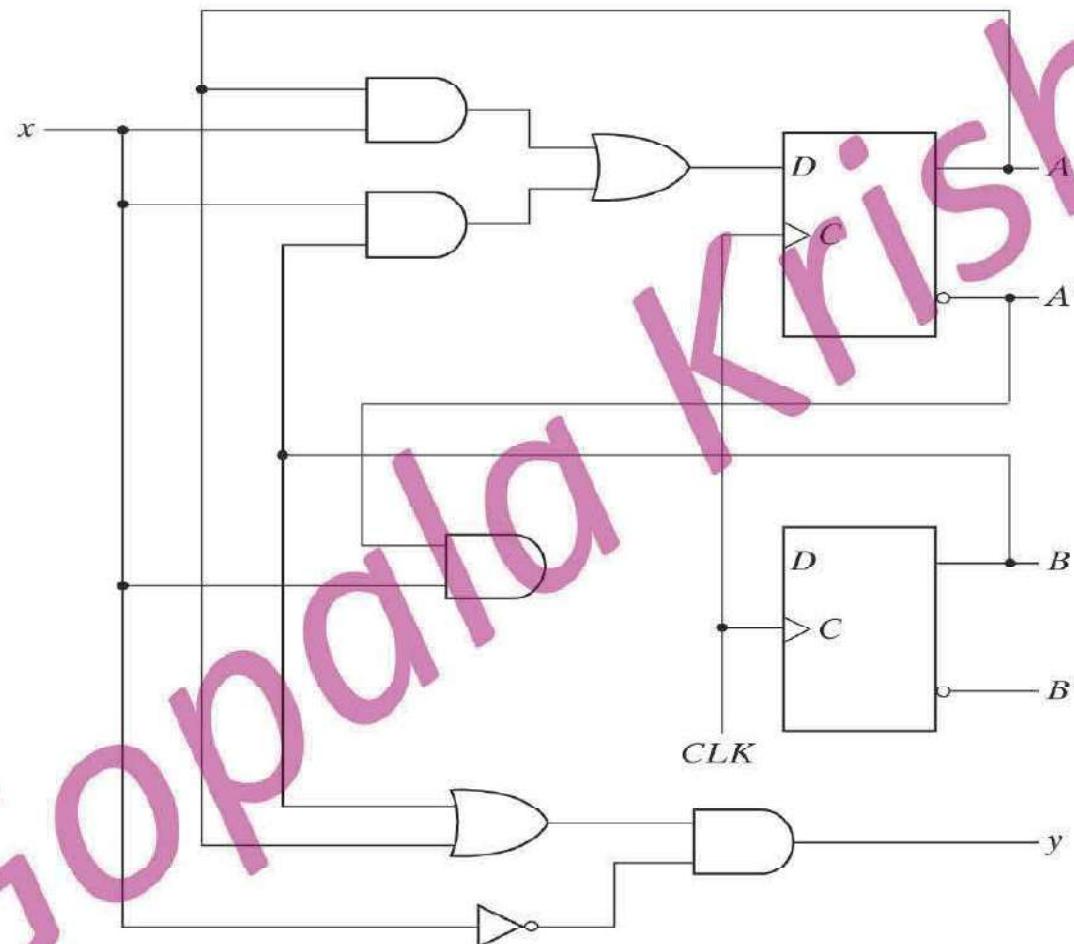


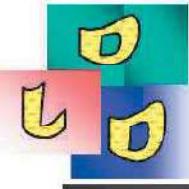
Mealy Machine

- In the Mealy model, the output is a function of both the present state and input.
- The sequential circuit can also be referred as Finite State Machine abbreviated as FSM.
- The Mealy model of a sequential circuit is referred as Mealy FSM or Mealy Machine.



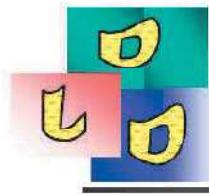
Mealy Machine - Example



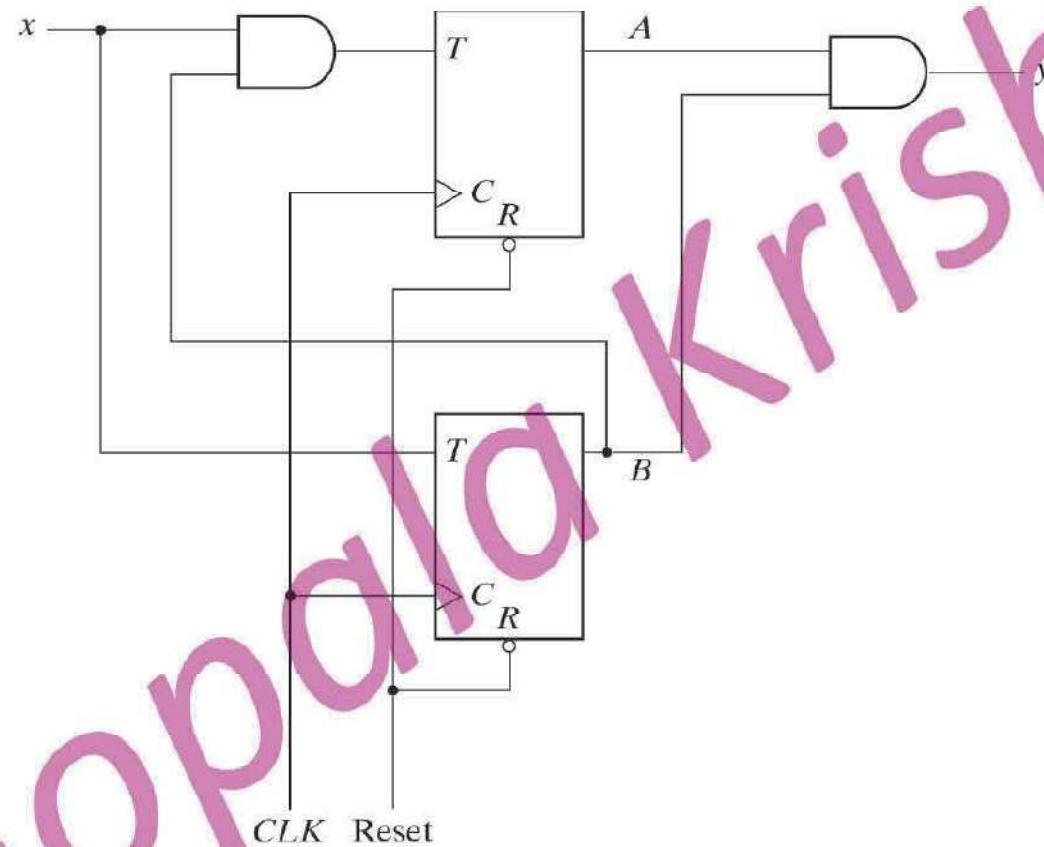


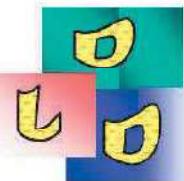
Moore Machine

- In the Moore model, the output is a function of only the present state.
- The sequential circuit can also be referred as Finite State Machine abbreviated as FSM.
- The Moore model of a sequential circuit is referred as Moore FSM or Moore Machine.



Moore Machine - Example



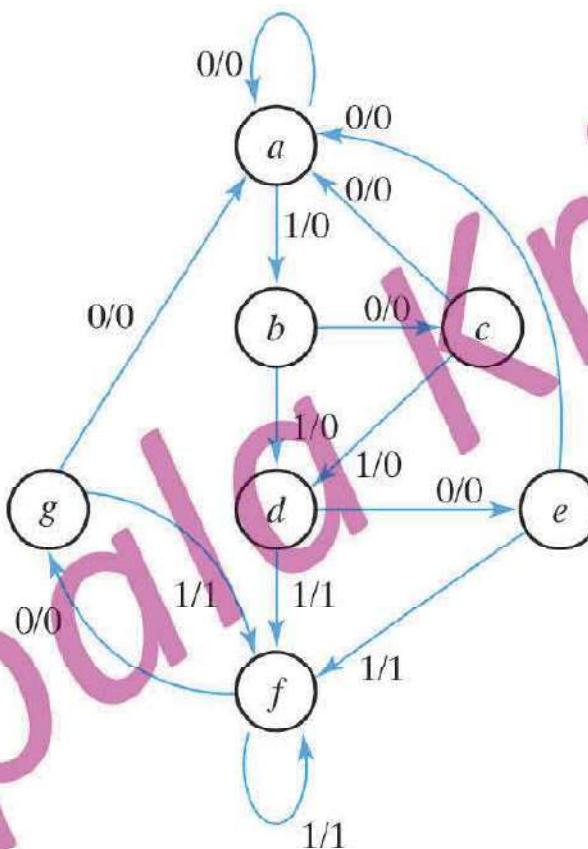


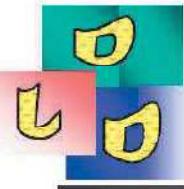
State Reduction and Assignment

- The reduction of the number of flip flops in a sequential circuit is referred to as the state reduction problem.
- State reduction algorithms are concerned with procedures for reducing the number of states in a state table, while keeping the external input output requirements unchanged.
- Since m flip flops produce 2^m states, a reduction in the number of states may result in a reduction in the number of flip flops.
- An unpredictable effect in reducing the number of flip flops is that sometimes the equivalent circuit may require more combinational gates.



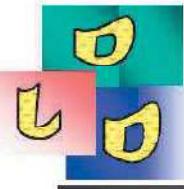
State Reduction-Example



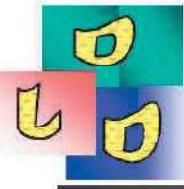


- We start with a sequential circuit whose specification is given in the state diagram.
- Here, the only input output sequence are important; the internal states are used merely to provide the required sequences.
- For this reason the states are marked with letter symbols.
- There are infinite number of input sequences that may be applied to the circuit. Each result in a unique output sequence.
- For example, consider the input sequence: 01010110100

State	a	a	b	c	d	e	f	f	g	f	g	a
input	0	1	0	1	0	1	1	0	1	0	0	
output	0	0	0	0	0	1	1	0	1	0	0	

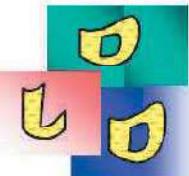


- In each column, we have the present state, input value, and output value. The next state is written on top of the next column.
- Now let us assume that we have found a sequential circuit whose state diagram has less than seven states and we wish to compare it with the circuit.
- The problem of state reduction is to find ways of reducing the number of states in a sequential circuit without altering the input-output relationship.



State 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



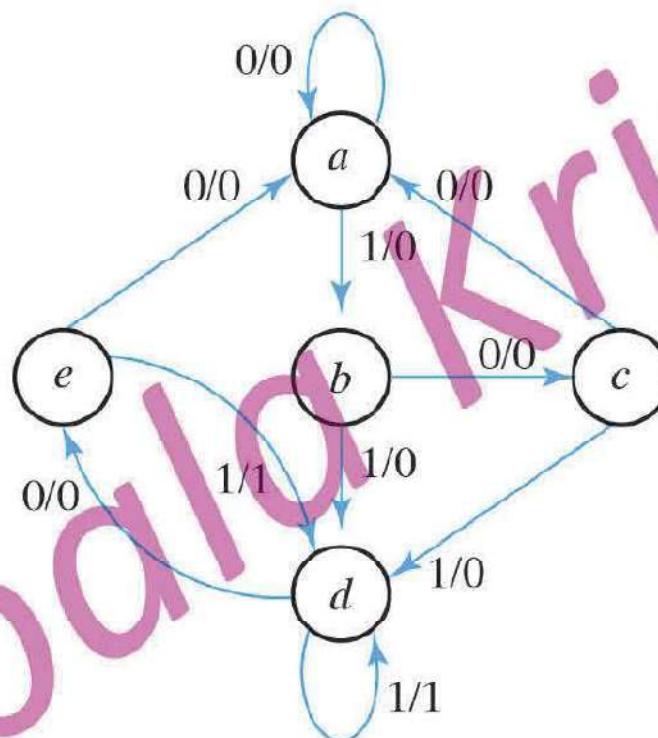
Reduced tables

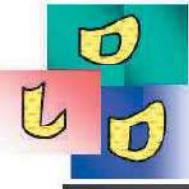
Present state	Next ststate		Output	
	X=0	x=1	X=0	X=1
A	A	B	0	0
B	C	D	0	0
C	A	D	0	0
D	E	F	0	1
E	A	F	0	1
F	E	F	0	1

Present state	Next ststate		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



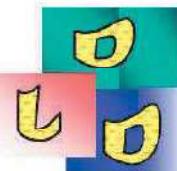
Reduced State Diagram





State Assignment

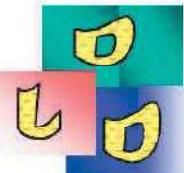
- In order to design a sequential circuit with physical components, it is necessary to assign coded binary values to the states.
- For a circuit with m states, the codes must contain n bits where $2^n \geq m$.
- For example, with three bits it is possible to assign codes to eight states denoted by binary numbers 000 through 111.



Three Possible Binary State Assignments

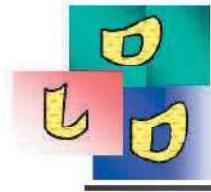
State	Assignment 1 Binary	Assignment 2 Gray Code	Assignment 3 One-hot
A	000	000	00001
B	001	001	00010
C	010	011	00100
D	011	010	01000
E	100	110	10000

Present state	Next state		Output	
	X=0	x=1	X=0	X=1
000	000	001	0	0
001	010	011	0	0
010	000	011	0	0
011	100	101	0	1
100	000	011	0	1



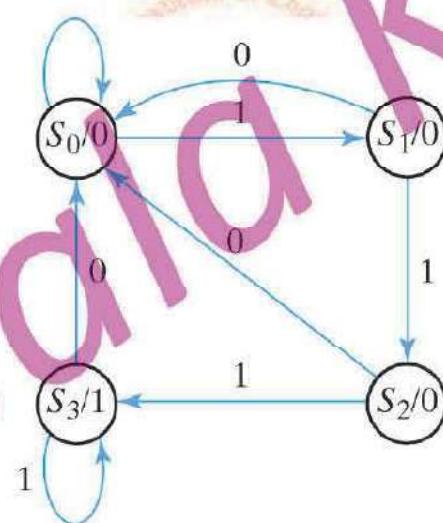
Design Procedure

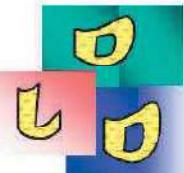
- The design of a clocked sequential circuit starts from a set of specifications and culminates in a logic diagram or a list of Boolean functions.
- The procedure for designing synchronous sequential circuits can be as follows:
 - From the word description of the desired operation, derive a state diagram for the circuit.
 - Reduce the number of states if necessary.
 - Assign binary values to the states.
 - Obtain the binary coded state table.
 - Choose the type of flip flops to be used.
 - Derive the simplified flip flop input equations and output equations.
 - Draw the logic diagram.



Design Procedure-Example

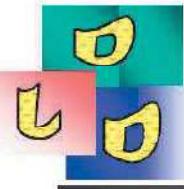
- Design a circuit that detects three or more consecutive 1's in a string of bits coming through an input line.





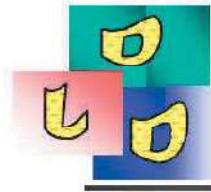
State Table for Sequence Detector

Present state		Input	Output	Next state	
A	B	X	Y	A	B
0	0	0	0	0	0
0	0	1	0	0	1
0	1	0	0	0	0
0	1	1	0	1	0
1	0	0	0	0	0
1	0	1	0	1	1
1	1	0	1	0	0
1	1	1	1	1	1



Synthesis using D Flip Flops

- Once the state table is derived with a straight forward binary assignment, we choose required number of flip flops.
- Here we choose two flip flops to represent four states and label their outputs as A and B.
- The characteristic equation of the D Flip Flop is $Q(t+1)=D_Q$ which means that the next state values in the state table specify the D input condition for the flip flop.
- So, the flip flop input equations can be obtained from the next state columns of A and B and expressed in sum of minterms as:
 - $A(t+1)=D_A(A,B,X)=(3,5,7)$
 - $B(t+1)=D_B(A,B,X)=(1,5,7)$
 - $Y(A,B,X)=(6,7)$



Maps for Sequence Detector

Bx	00	01	<u>11</u>	10
A			1	
0				
A	1		1	
x				

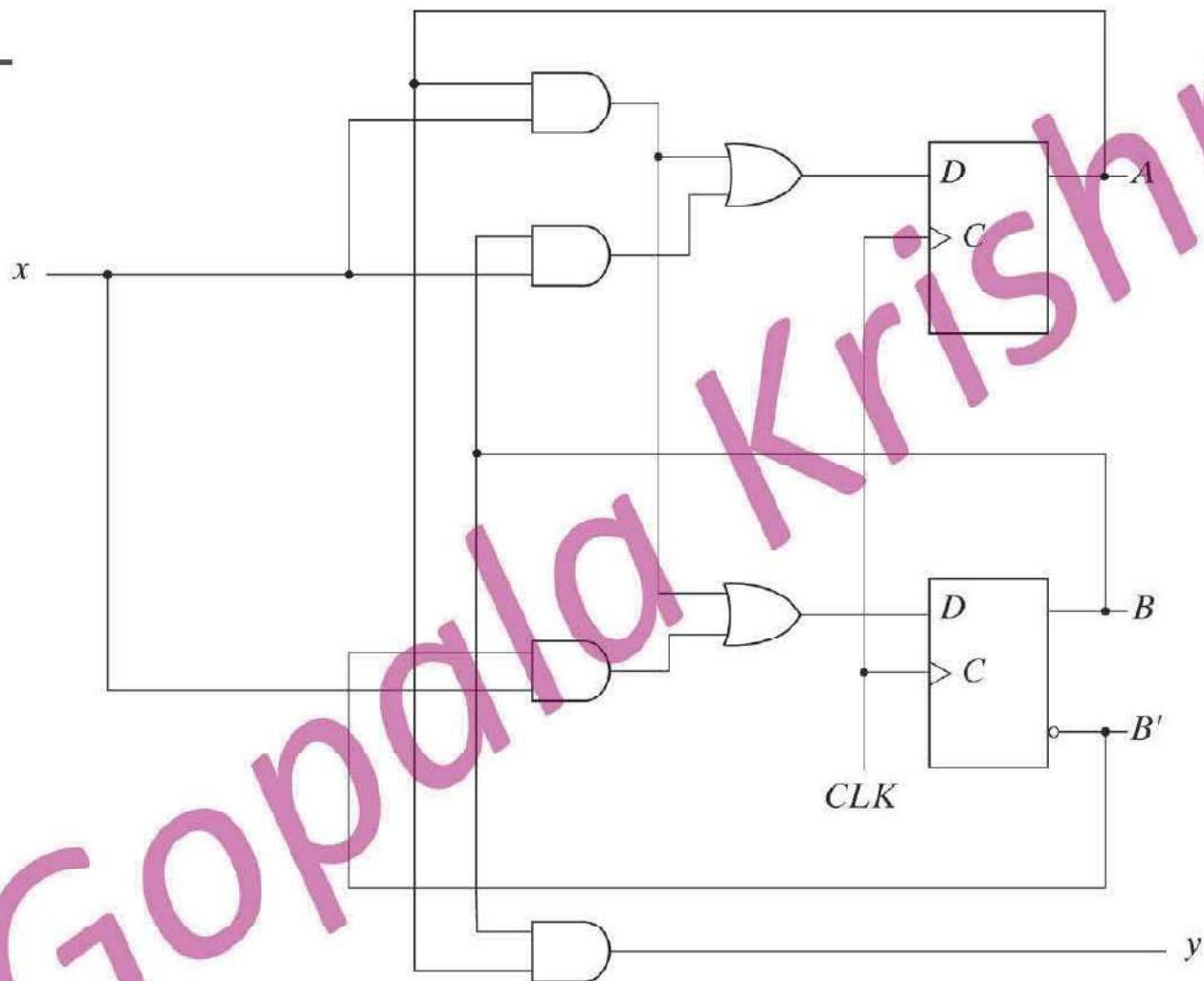
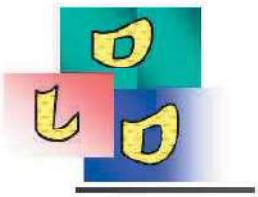
$$D_A = Ax + Bx$$

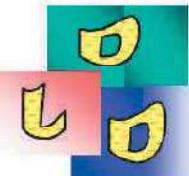
		1	
		1	
		1	

$$D_B = Ax + B'x$$

	1	1	

$$y = AB$$

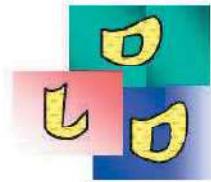




Flip Flop Excitation Table

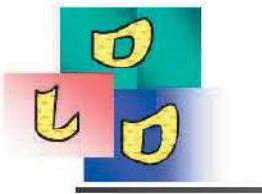
Q(t)	Q(t+1)	J	K
0	0	0	X
0	1	1	X
1	0	X	1
1	1	X	0

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



Synthesis using JK Flip Flops

Present state		Input	Next state		Flip Flop Input Equations			
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



		Bx	00	01	11	B	10
		A	0				
A	1						
		X	X	X		1	X
					x		

$$J_A = Bx'$$

		Bx	00	01	11	B	10
		A	0				
A	1						
		1		X		X	X
					x		

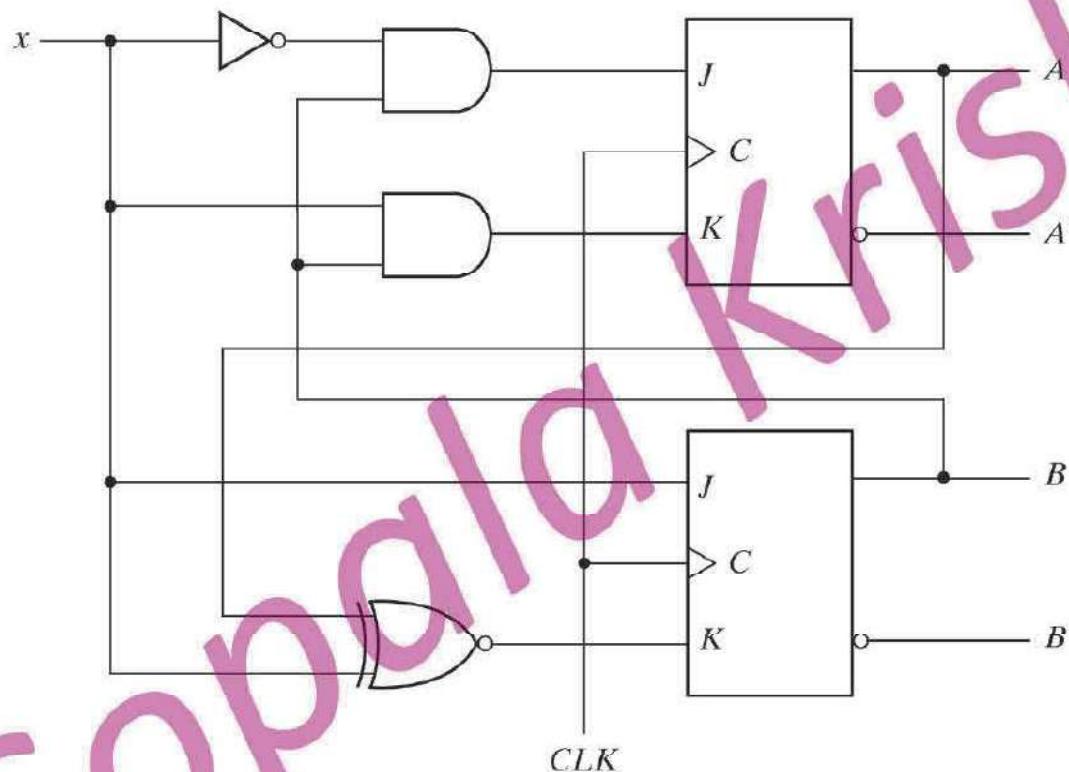
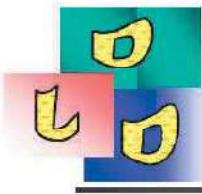
$$J_B = x$$

		Bx	00	01	11	B	10
		A	0				
A	1						
		X	X		X	X	X
					x		

$$K_A = Bx$$

		Bx	00	01	11	B	10
		A	0				
A	1						
		X	X			1	
					x		

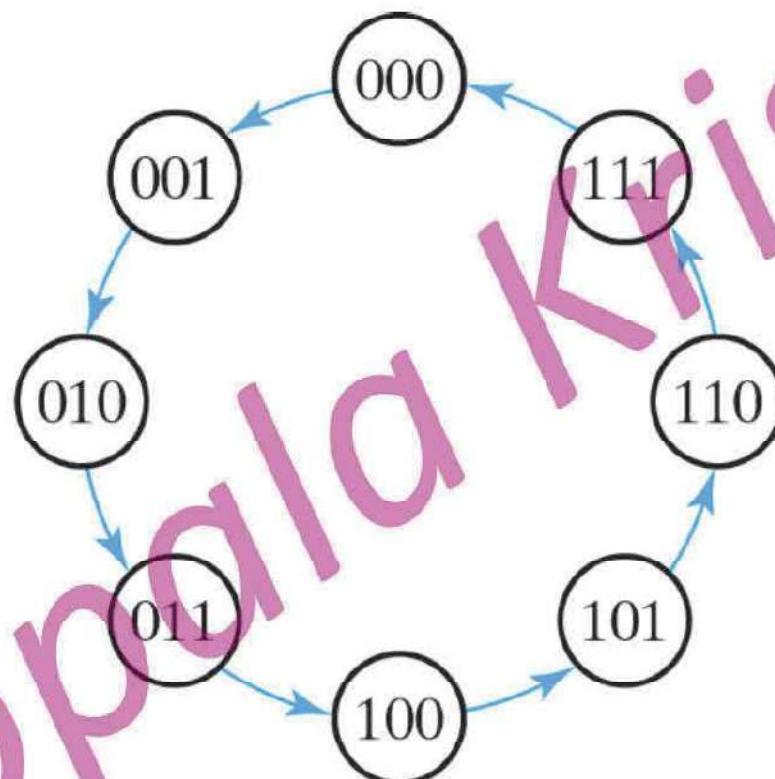
$$K_B = (A \oplus x)'$$



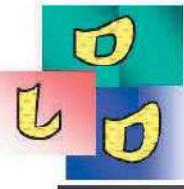
P. Gopala Krishna



Synthesis Using T Flip Flops

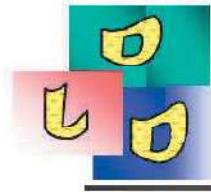


P. Gopala Krishna

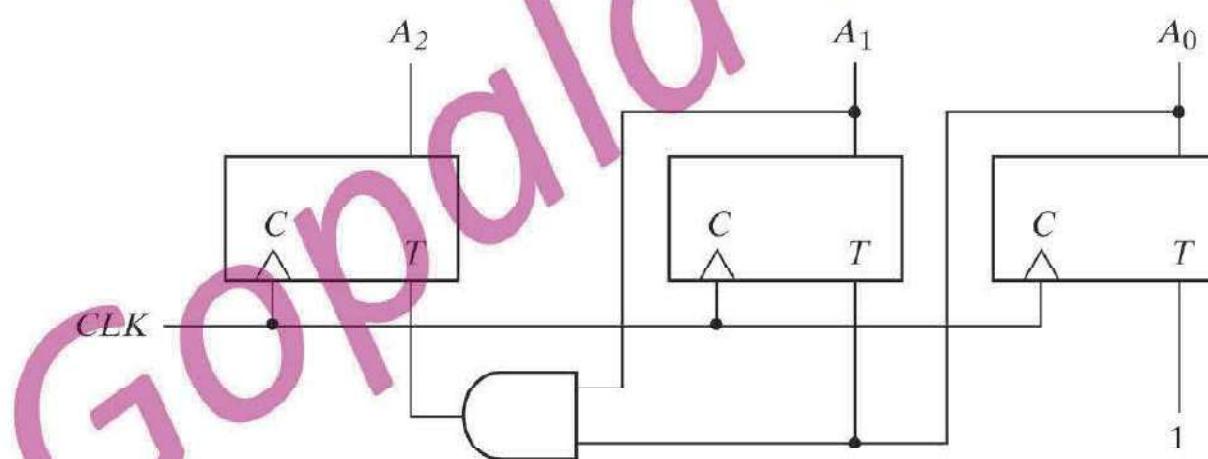
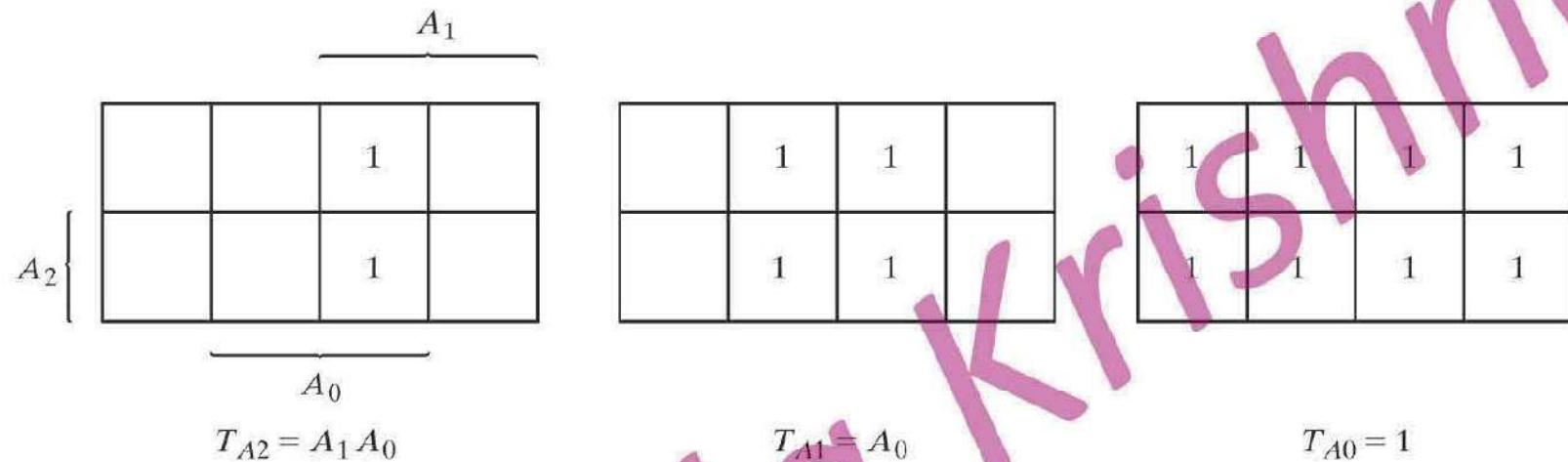


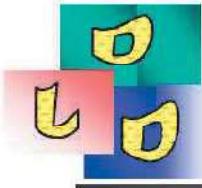
State Table

Present state			Next state			Flip Flop Inputs		
A2	A1	A0	A2	A1	A0	T _{A2}	T _{A1}	T _{A0}
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



3-bit Counter Design





UNIT - IV

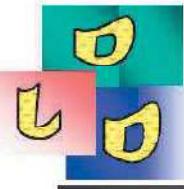
REGISTERS AND COUNTERS

P. Gopala Krishna

UNIT-IV/DIGITAL LOGIC DESIGN/IT II-I Sem/GRIET

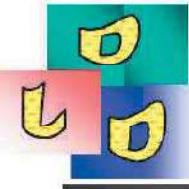
P. Gopala Krishna

72

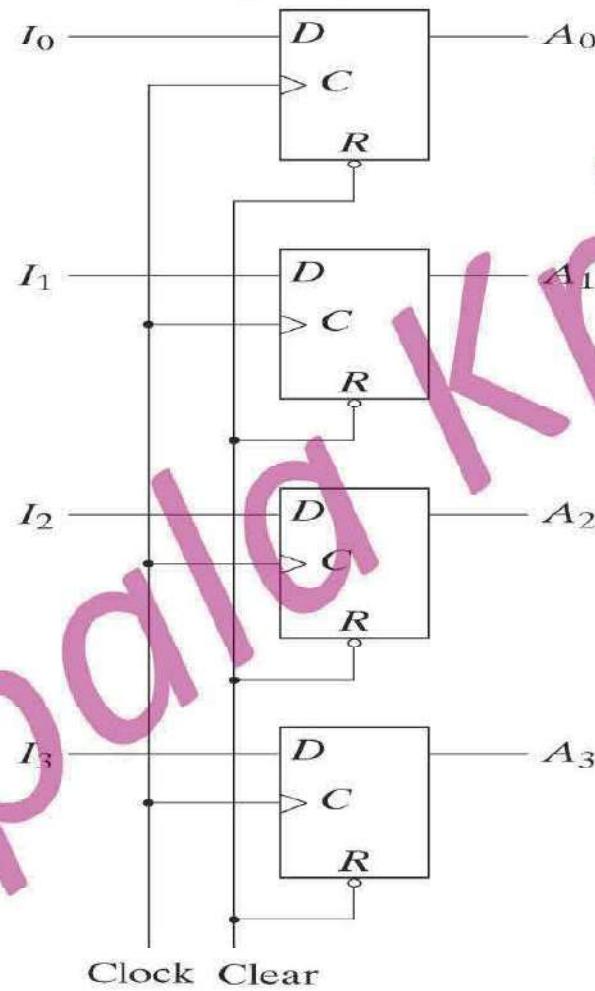


Registers

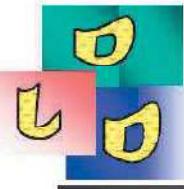
- A register is a group of flip flops.
- Each flip flop is capable of storing one bit of information.
- An n bit register consists of a group of n flip flops capable of storing n bits of binary information.
- In addition to the flip flops, a register may have combinational gates that perform certain data processing tasks.
- A register consists of a group of flip flops and gates that affect their transition.
- The flip flops hold the binary information and the gates determine how the information is transferred into the register.



4-Bit Register

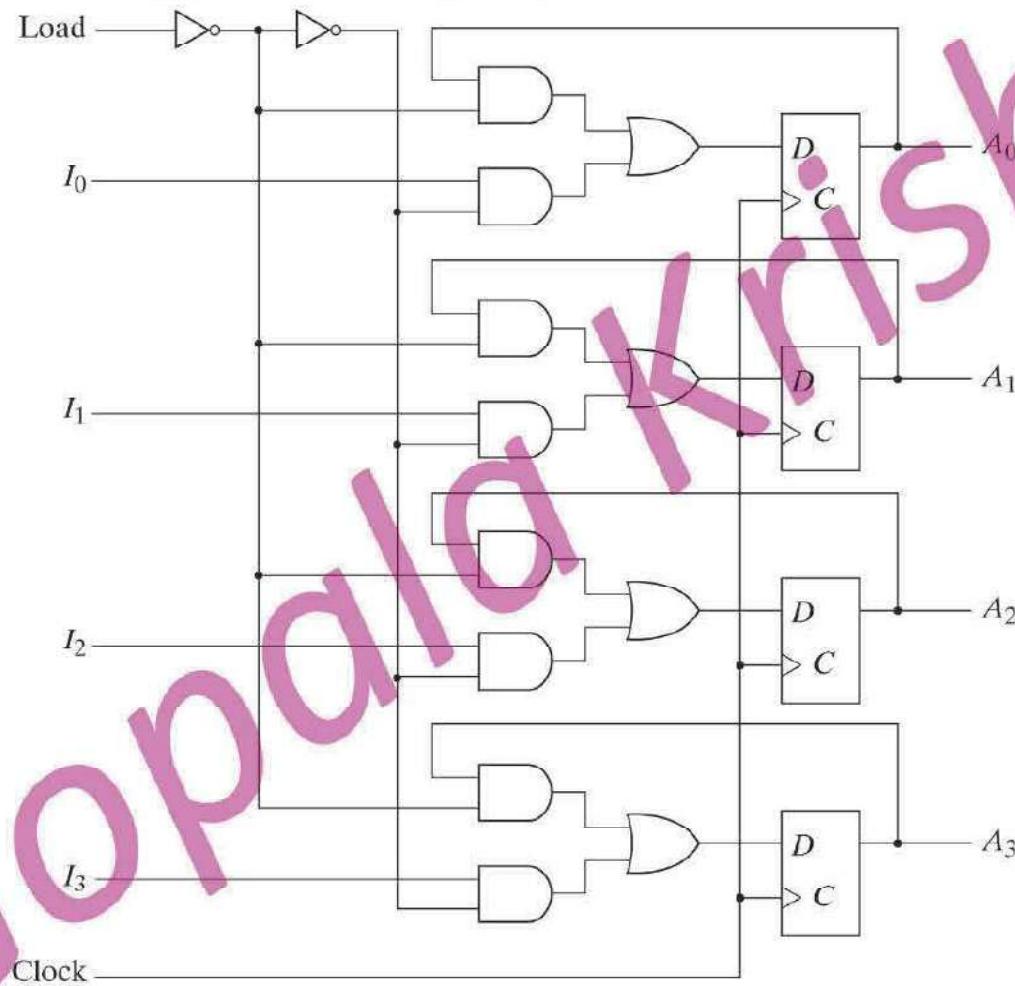
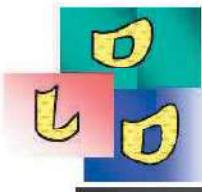


P. Gopala Krishna

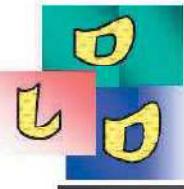


Register with Parallel Load

- The transfer of new information into a register is referred to as loading the register.
- If all the bits of the register are loaded simultaneously with a common clock pulse, we say that the loading is done in parallel.
- For a register with parallel load, the load input determines whether the next pulse will accept new information or leave the information in the register intact.
- The transfer of information from the data inputs or the outputs of the register is done simultaneously with all four bits in response to a clock edge.

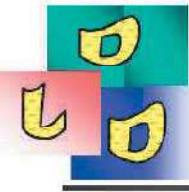


P. Gopala Krishna

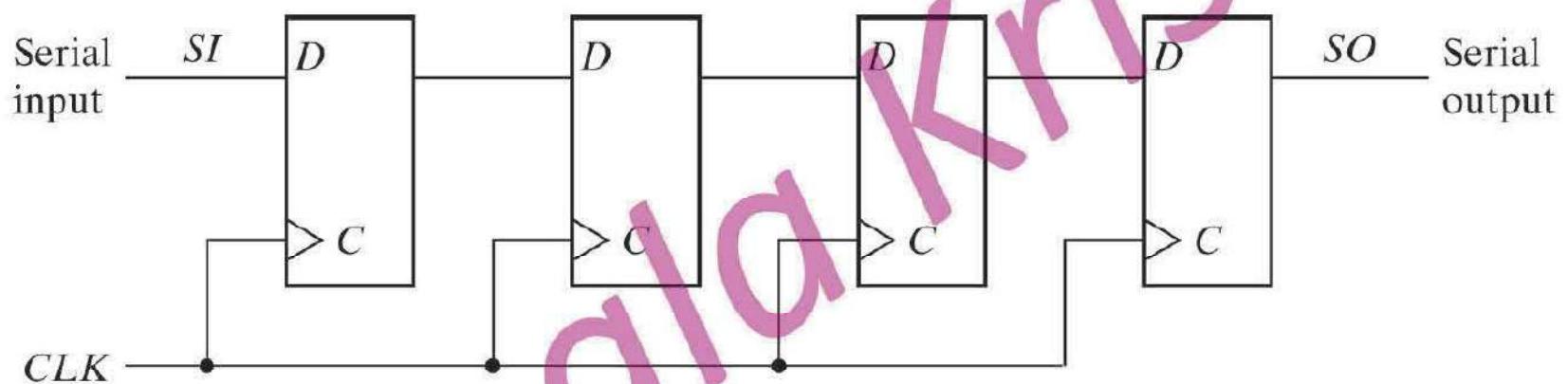


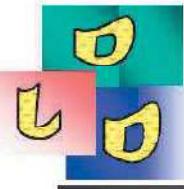
Shift Registers

- A register capable of shifting its binary information in one or both directions is called a shift register.
- The logical configuration of a shift register consists of a chain of flip flops in cascade, with the output of one flip flop connected to the input of the next flip flop.
- All flip flops receive common clock pulses, which activate the shift from one stage to the next.



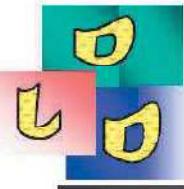
4-bit Shift Register



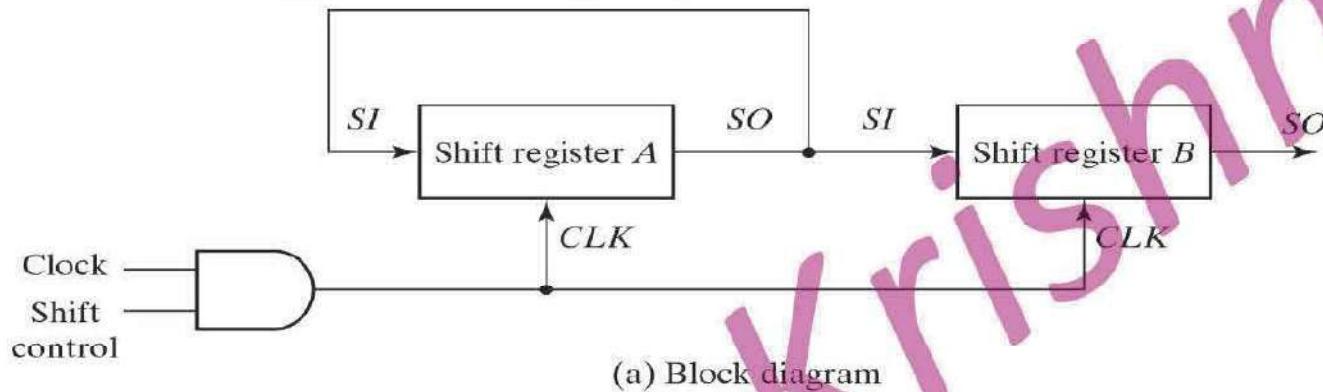


Serial Transfer

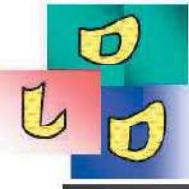
- A digital system is said to operate in a serial mode when information is transferred and manipulated one bit at a time.
- Information is transferred one bit at a time by shifting the bits out of the source register into the destination register. This is in contrast to parallel transfer where all the bits of the register are transferred at the same time.
- To attain serial transfer, the source and destination register are treated as shift registers with the serial output of source register is connected to the serial input of the destination register.



Serial Transfer



Timing Pulse	Shift Register A				Shift Register B			
Initial value	1	0	1	1	0	0	1	0
After T1	1	1	0	1	1	0	0	1
After T2	1	1	1	0	1	1	0	0
After T3	0	1	1	1	0	1	1	0
After T4	1	0	1	1	1	0	1	1



Serial Adder

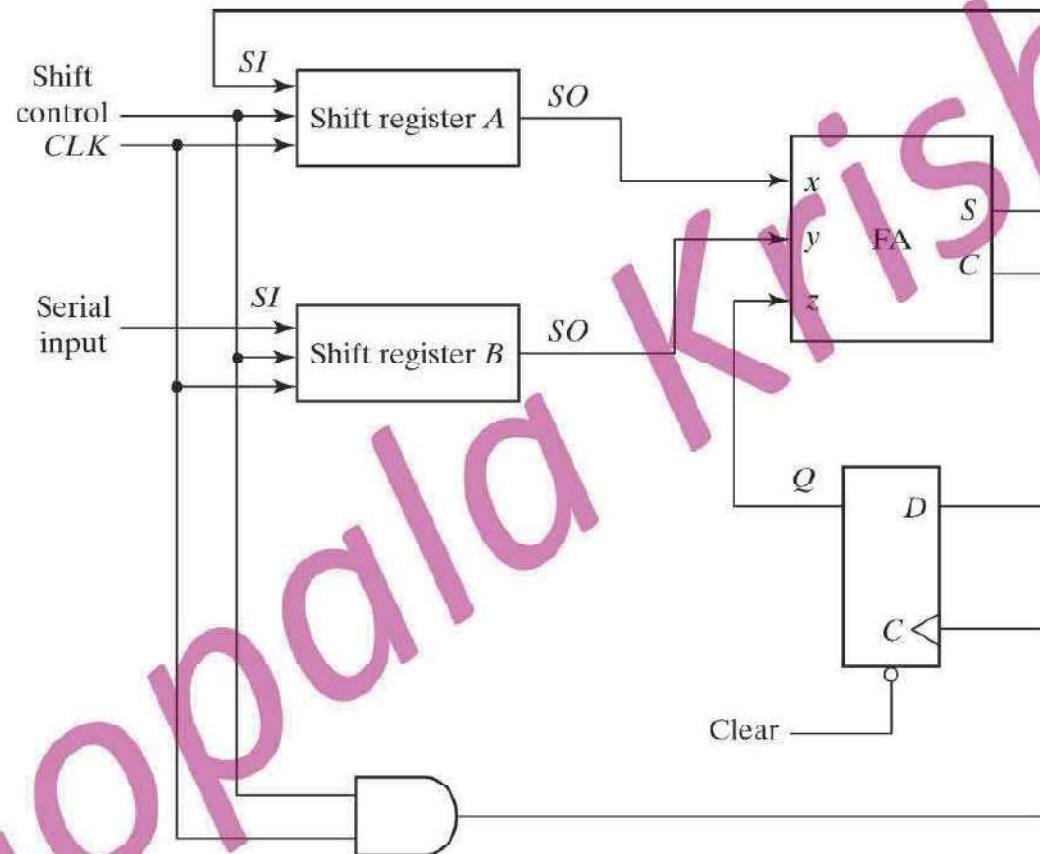
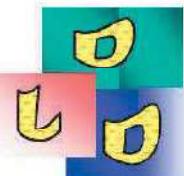


Fig. 6-5 Serial Adder



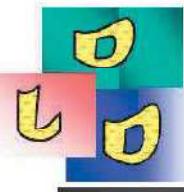
State table for Serial Adder using JK Flip Flop

Present state		Inputs		Next state		Output		Flip Flop 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		

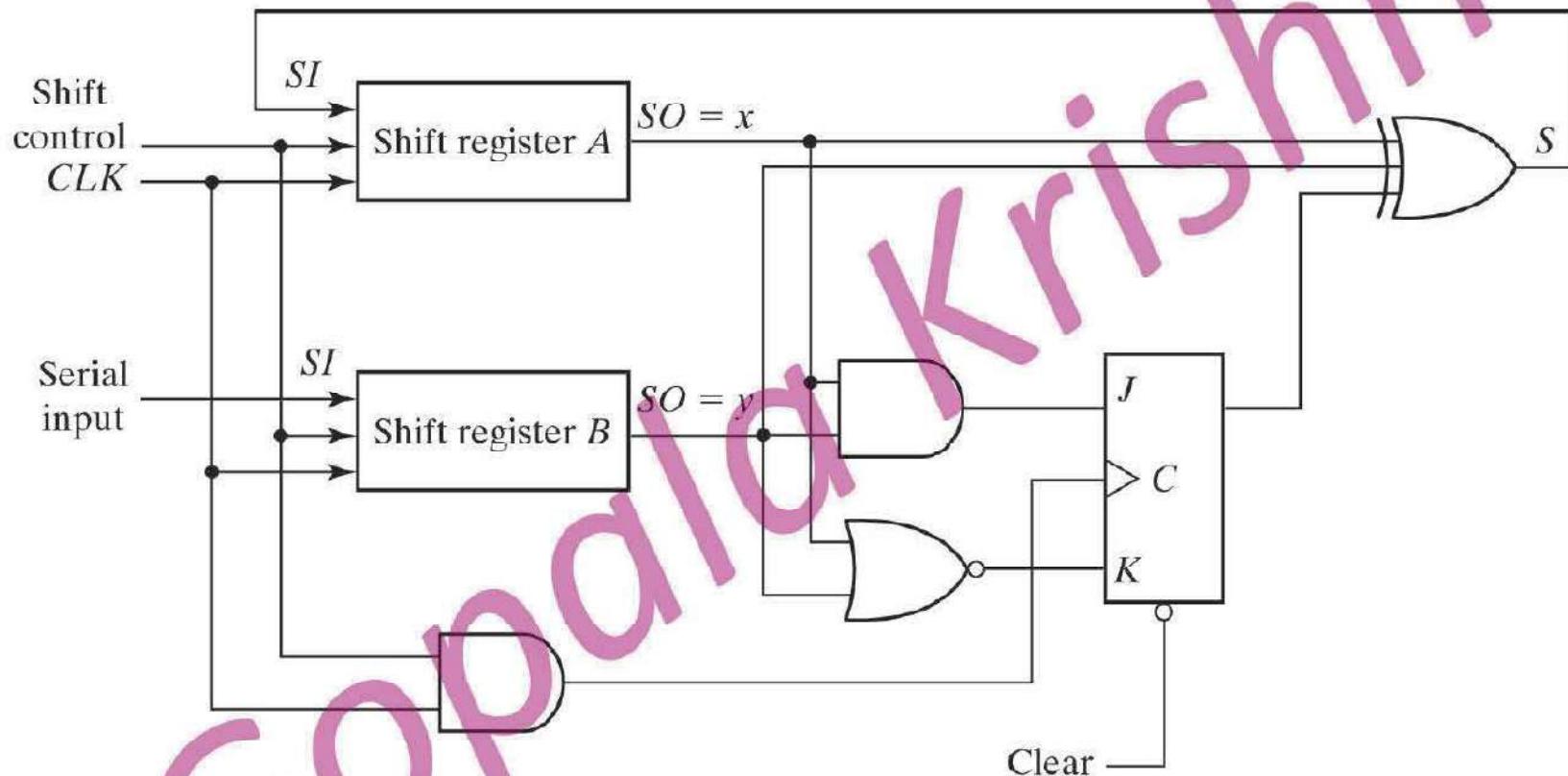
$$J_Q = XY$$

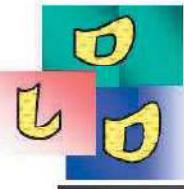
$$K_Q = X^1 Y^1 = (X+Y)^1$$

$$S = X \oplus Y \oplus Q$$



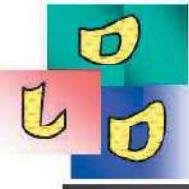
Serial Adder using JK Flip Flop



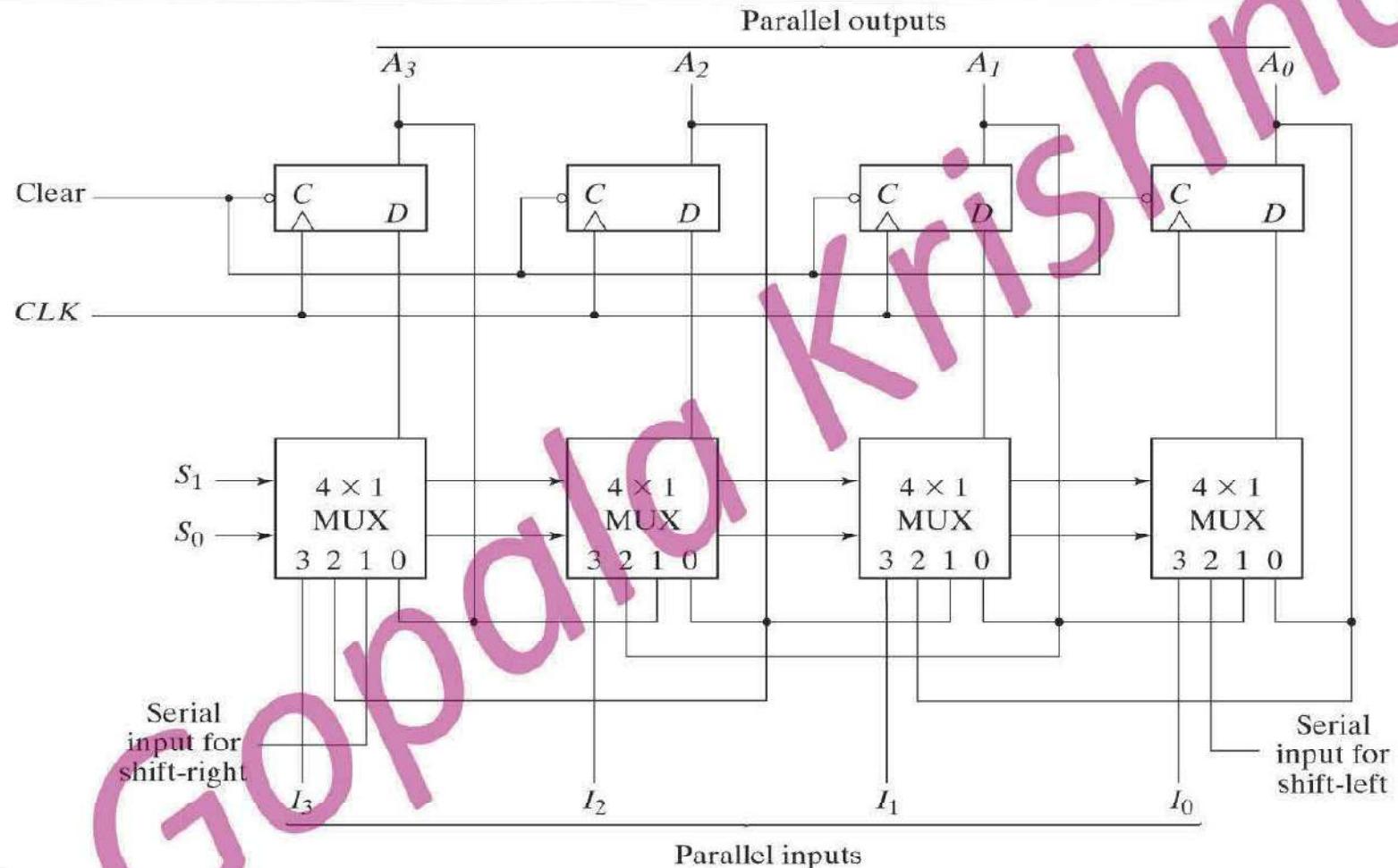


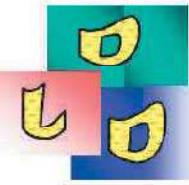
Universal Shift Register

- The most general shift register has the following capabilities:
 1. A clear control to clear the register to 0.
 2. A clock input to synchronize the operation.
 3. A shift right control to enable the shift right operation and the serial input and output lines associated with the shift right.
 4. A shift left control to enable the shift left operation and the serial input and output lines associated with the shift left.
 5. A parallel load control to enable a parallel transfer and the n input lines associated with the parallel transfer.
 6. N parallel output lines.
 7. A control state that leaves the information in the register unchanged in the presence of the clock.



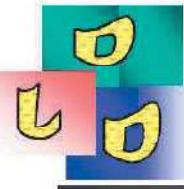
Universal Shift Register





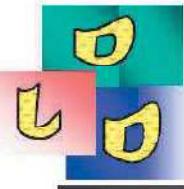
Function table for Universal Shift Register

Mode Control		Register Operation
S1	S0	
0	0	No change
0	1	Shift right
1	0	Shift left
1	1	Parallel load



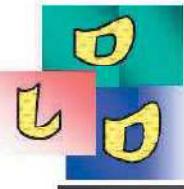
Counters

- A register that goes through a prescribed sequence of states upon the application of input pulses is called a counter.
- The input pulses may be clock pulses or they may originate from some external source and may occur at a fixed interval of time or random.
- A counter that follows the binary number sequence is called binary counter. An n bit binary counter have n flip flops and can count in binary from 0 to $2^n - 1$.
- Counters are available in two categories:
 - Ripple Counters: In a ripple counter, the flip flop output transition serves as a source for triggering other flip flops.
 - Synchronous counters: In a synchronous counter, the C inputs of all flip flops receive common clock.

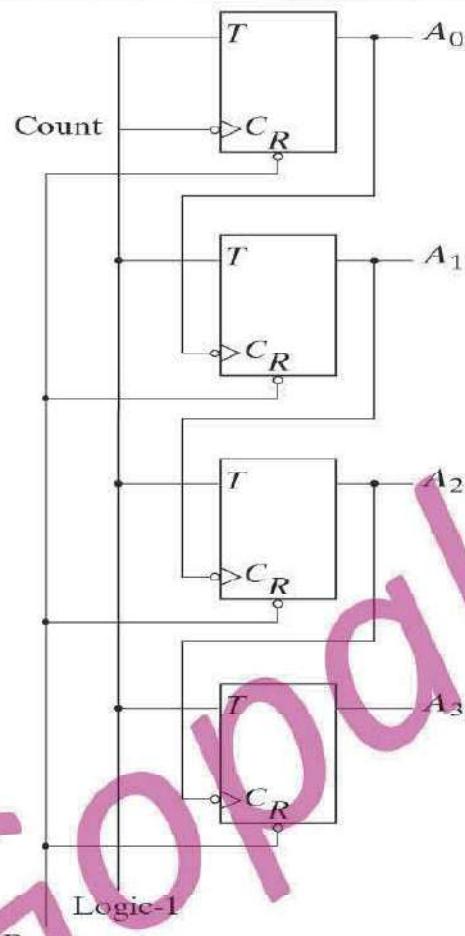


Binary Ripple Counter

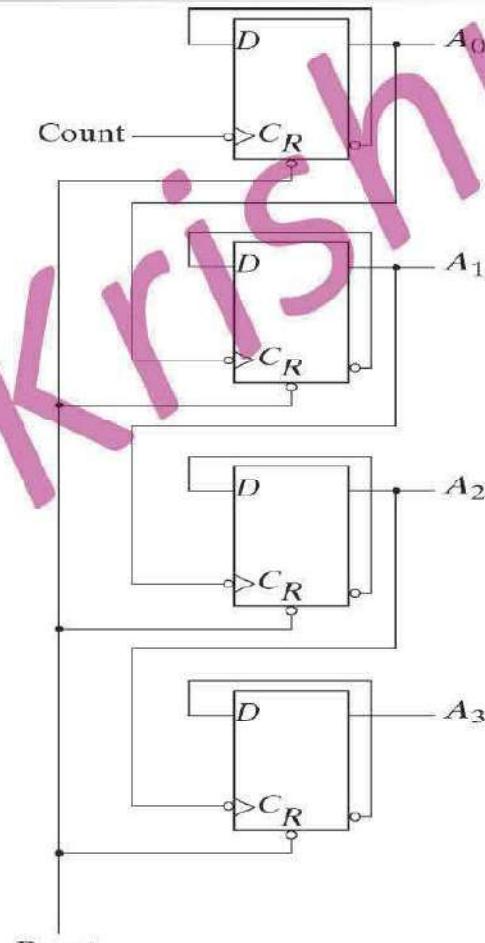
- A binary ripple counter consists of a series connection of complementing flip-flops, with the output of each flip flop connected to the c input of the next higher order flip flop.
- The flip flop holding the least significant bit receives the incoming count pulses.
- The counter is connected with the D type or T type flip flops.
- The output of each flip flop is connected to the C input of the next flip flop in sequence.



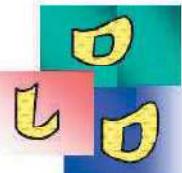
Binary Ripple Counter with T and D Flip Flops



(a) With T flip-flops

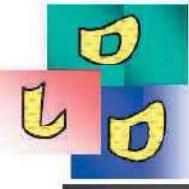


(b) With D flip-flops



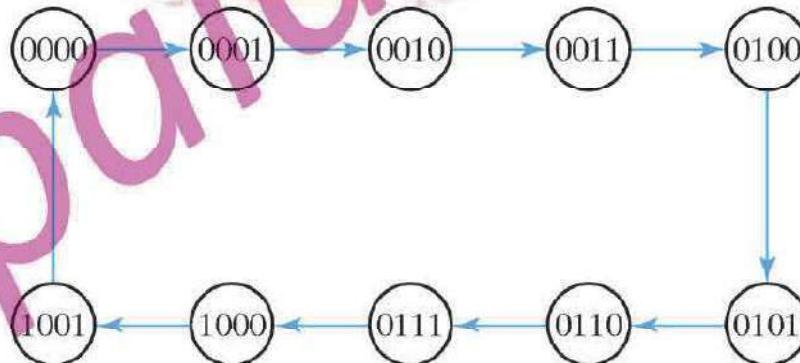
Binary Count Down Counter

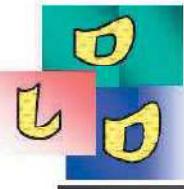
- A binary counter with a reverse count is called a binary count down counter.
- In a down ward counter is decremented by 1 with every input count pulse.
- A bit in the sequence is complemented if its previous least significant bit goes from 0 to 1.
- Therefore the diagram of a binary count down counter looks like a binary ripple counter provided all flip flops trigger on the positive edge of the clock.
- If negative edge triggered flip flops are used, then the C input of each flip flop is connected to the complement of the output of the previous flip flop.



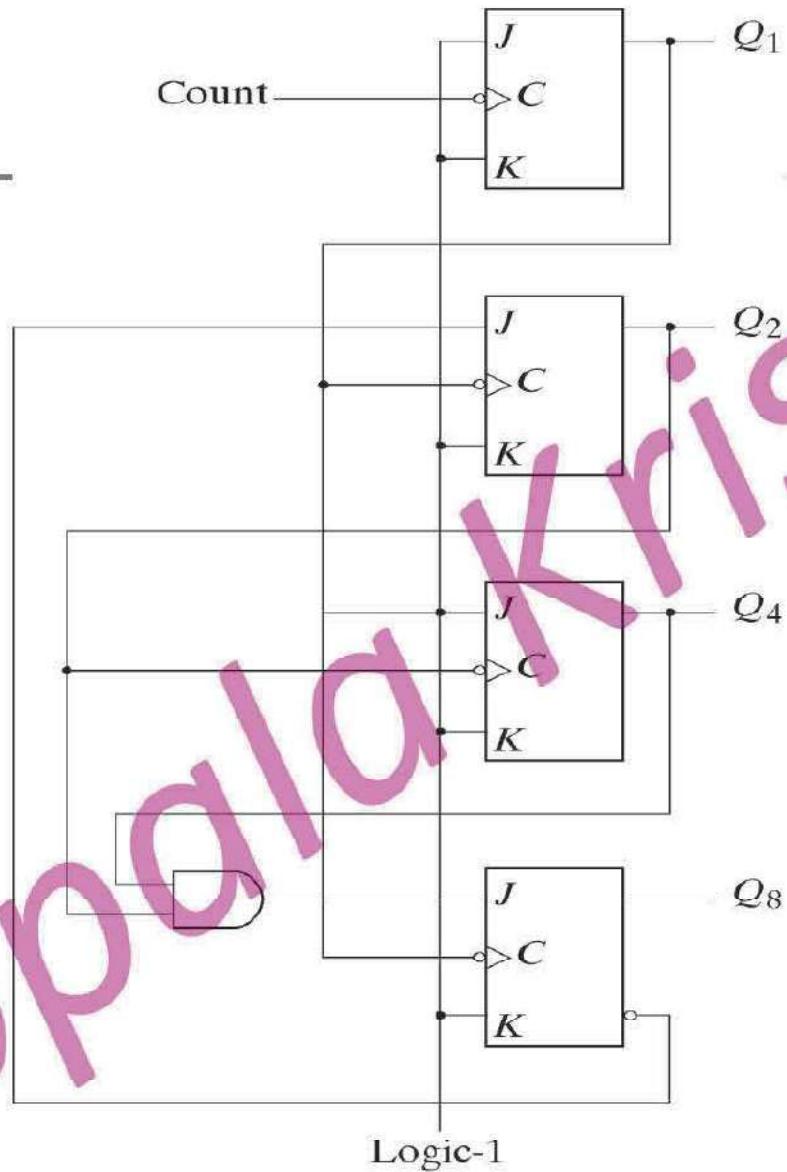
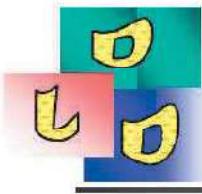
BCD Ripple Counter

- A decimal counter follows a sequence of ten states and return to 0 after the count of 9.
- Such a counter must have at least four flip flops to represent each decimal digit, since a decimal digit represented by a binary code with at least four bits.
- If a BCD Code is used, the sequence of states are as follows:

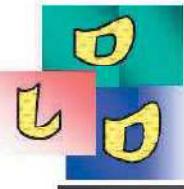




- To verify the conditions result in the sequence required by BCD ripple counter, it is necessary to verify that the flip flop transitions indeed follows a sequence of states as specified by the state diagram.
- Q1 changes state after every clock pulse.
- Q2 complements every time Q1 goes from 1 to 0 as long as Q8 is 0.
- When Q8 becomes 1, Q2 remains at 0.
- Q4 complements every time Q2 goes from 1 to 0.
- Q8 remains at 0 as long as Q2 or Q4 is 0.
- When both Q2, Q4 become 1, Q8 complements when Q1 goes from 1 to 0.
- Q8 is cleared on the next transition of Q1.

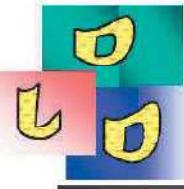


P. Gopala Krishna



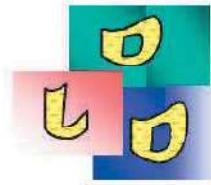
Synchronous Counters

- Synchronous Counters are different from ripple counters in that clock pulses are applied to the inputs of all flip flops.
- A common clock triggers all flip flops simultaneously rather than one at a time.
- The decision whether a flip flop is to be complemented or not is determined from the values of the data inputs such as T or J and K at the time of clock edge.

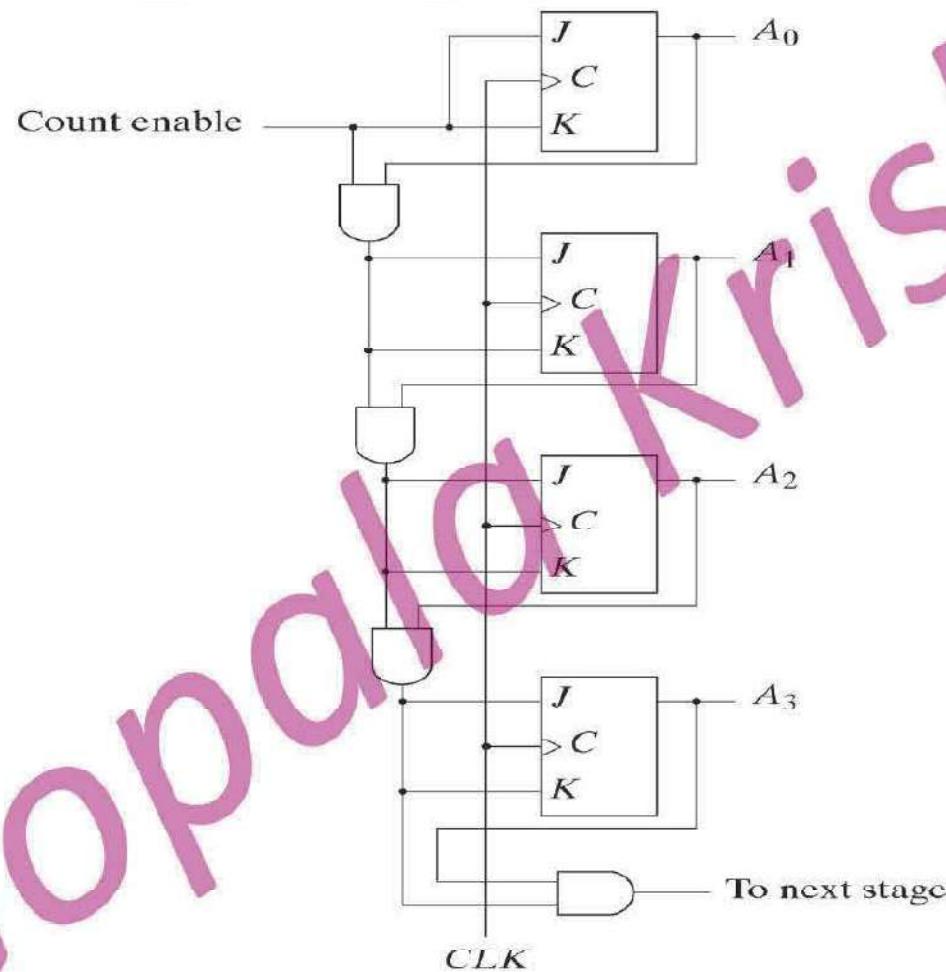


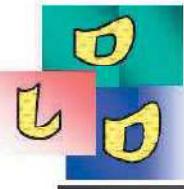
Binary Counter

- In a Synchronous Binary Counter, the flip flop in the least significant position is complemented with every pulse.
- A flip flop in any other position is complemented when all the bits in the lower significant positions are equal to 1.
- Synchronous Binary Counters have a regular pattern and can be constructed with complementing flip flops and gates.
- The Synchronous Counters can be triggered with either positive or the negative clock edge.



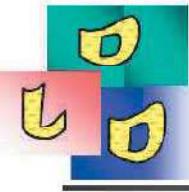
4-bit Synchronous Binary Counter



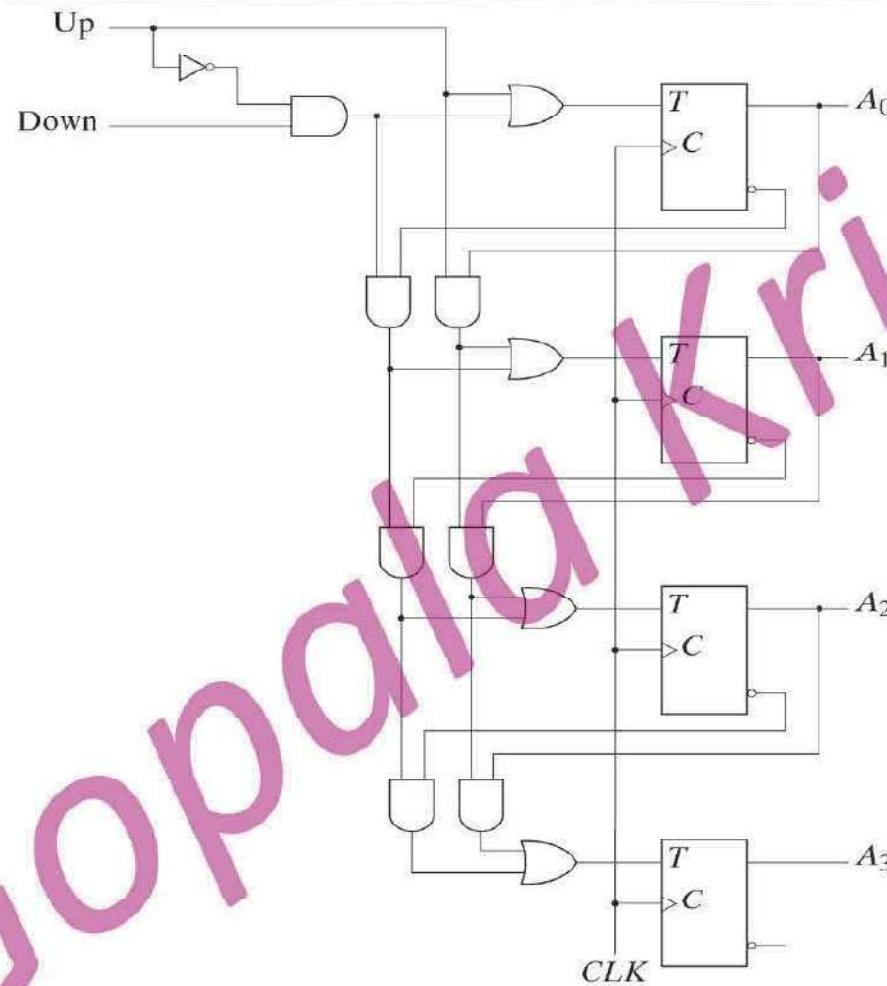


Up-Down Binary Counter

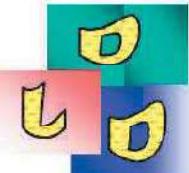
- In a Synchronous count down Binary Counter, the flip flop in the least significant position is complemented with every pulse.
- A flip flop in any other position is complemented when all the bits in the lower significant positions are equal to 0.
- A count down binary counter can be constructed by connecting the complemented outputs of the previous flip flops to the AND gates.
- The two operations can be combined in one circuit to form a counter capable of counting either up or down.



4-bit Up Down Binary Counter



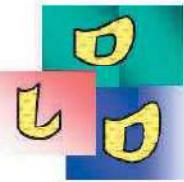
P. Gopala Krishna



BCD Counter

- A BCD counter counts in binary coded decimal from 0000 to 1001 and back to 0000.
- Because of the return to 0 after a count of 9, a BCD counter does not have a regular pattern as in straight binary counter.
- The circuit for a BCD Counter can be obtained by normal design procedure.
- The flip flop input equations can be simplified by means of maps. The unused states for minterms 10 to 15 are taken as don't care conditions.

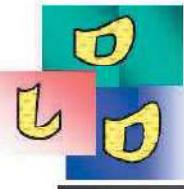
- $T_{Q_1} = 1$
- $T_{Q_2} = Q_8 \bar{Q}_1$
- $T_{Q_4} = Q_2 Q_1$
- $T_{Q_8} = Q_8 Q_1 + Q_4 Q_2 Q_1$
- $Y = Q_8 Q_1$



BCD Counter-State Table

Present State				Next State				Output	Flip Flop Inputs			
Q8	Q4	Q2	Q1	Q8	Q4	Q2	Q1	Y	TQ8	TQ4	TQ2	TQ1
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	0	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

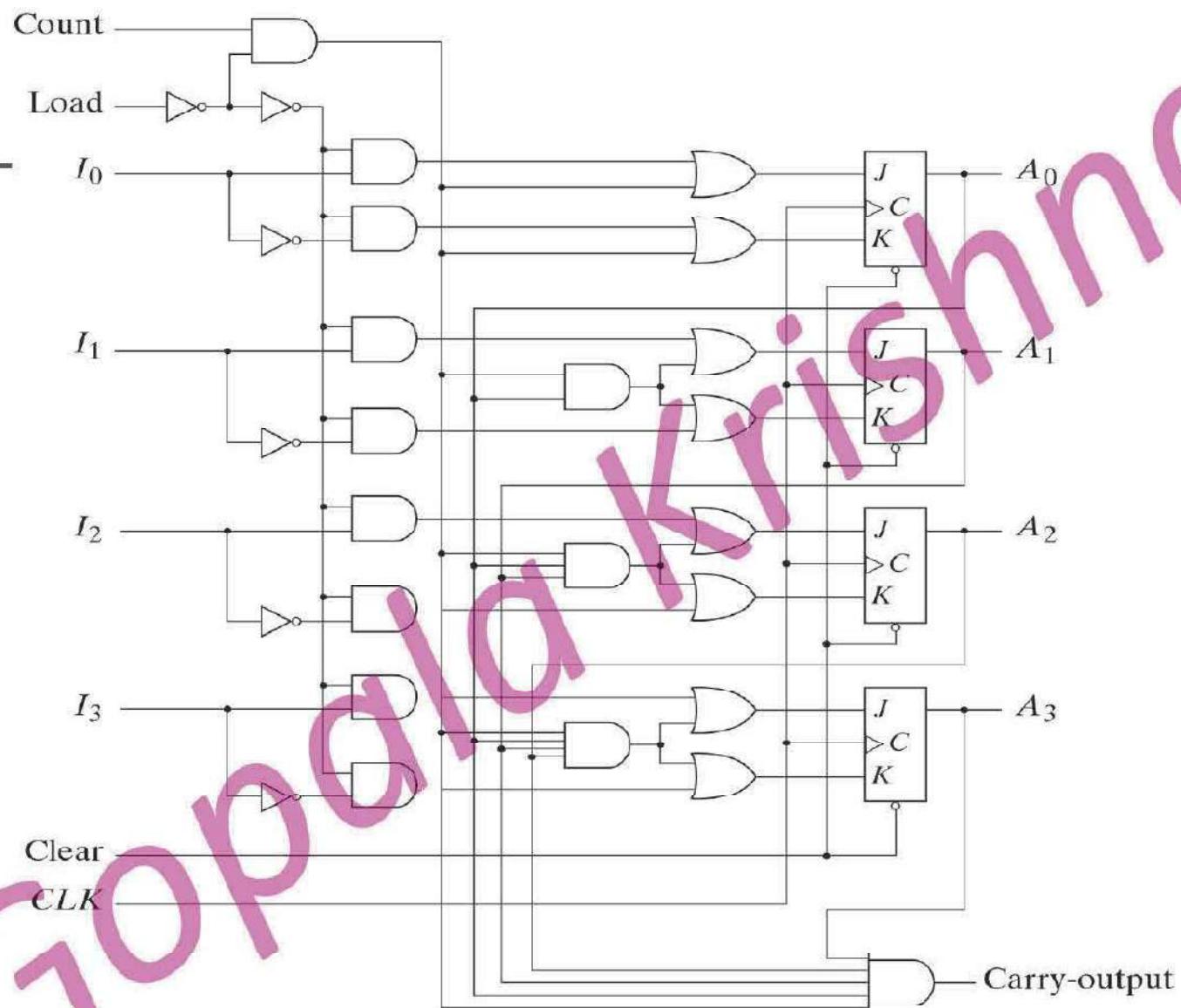
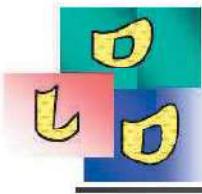
- $T_{Q_1} = 1$ $T_{Q_2} = Q_8 \bar{Q}_1 Q_1$ $T_{Q_4} = Q_2 Q_1$ $T_{Q_8} = Q_8 Q_1 + Q_4 Q_2 Q_1$ $Y = Q_8 Q_1$

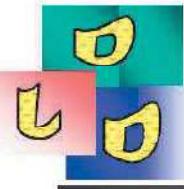


Binary Counter with Parallel Load

- Counters employed in digital systems quite often require a parallel load capability for transferring an initial binary number into the counter prior to the count operation.

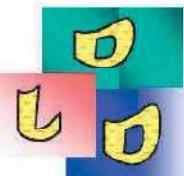
Clear	Clk	Load	Count	function
0	X	X	X	Clear to 0
1	1	1	X	Load inputs
1	1	0	1	Count next binary state
1	1	0	0	No change





Counter with Unused States

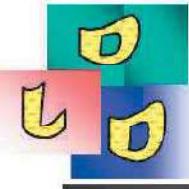
- A circuit with n flip flops has 2^n binary states. There are occasions when a sequential circuit uses less than this maximum number of states.
- States that are not used in specifying the sequential circuit are not listed in the state table.
- When simplifying the input equations, the unused states may be treated as don't care conditions or may be assigned specific next states.
- Outside interference may cause the circuit to enter one of the unused states. In that case, it is necessary to ensure that the circuit eventually goes into one of the valid states so it can resume normal operation.
- Otherwise, if the sequential circuit circulates among unused states, there will be no way to bring it back to its intended sequence of state transitions.



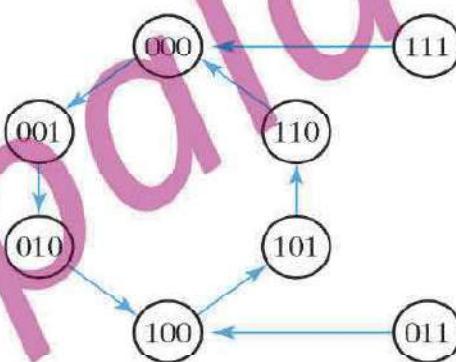
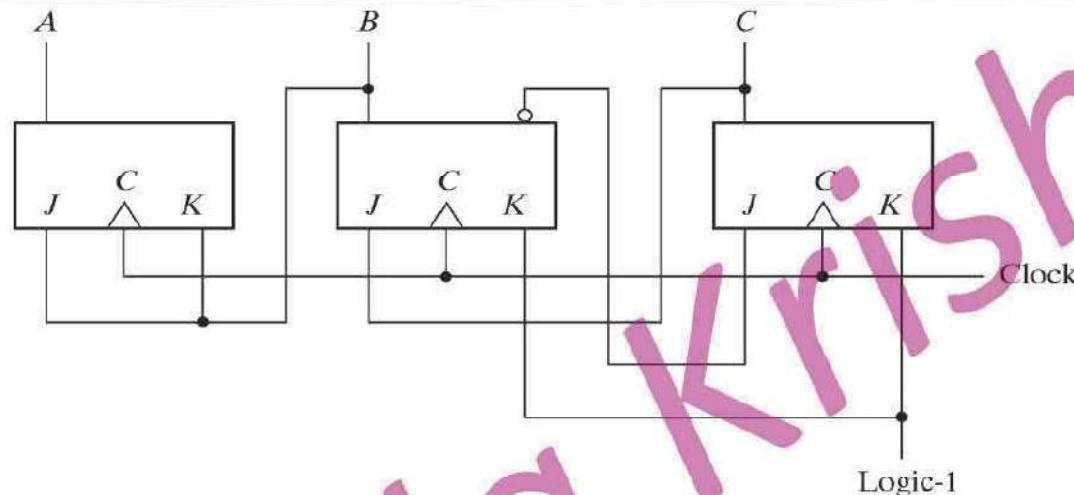
Counter with Unused States-Example

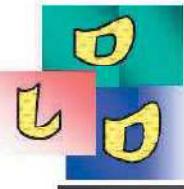
Present state			Next state			Flip Flop Inputs					
A	B	C	A	B	C	J_A	K_A	J_B	K_B	J_C	K_C
0	0	0	0	0	1	0	X	0	X	1	X
0	0	1	0	1	0	0	X	1	X	X	1
0	1	0	1	0	0	1	X	X	1	0	X
1	0	0	1	0	1	X	0	0	X	1	X
1	0	1	1	1	0	X	0	1	X	X	1
1	1	0	0	0	0	X	1	X	1	0	X

- $J_A = B$ $J_B = C$ $J_C = B^1$
- $K_A = B$ $K_B = 1$ $K_C = 1$



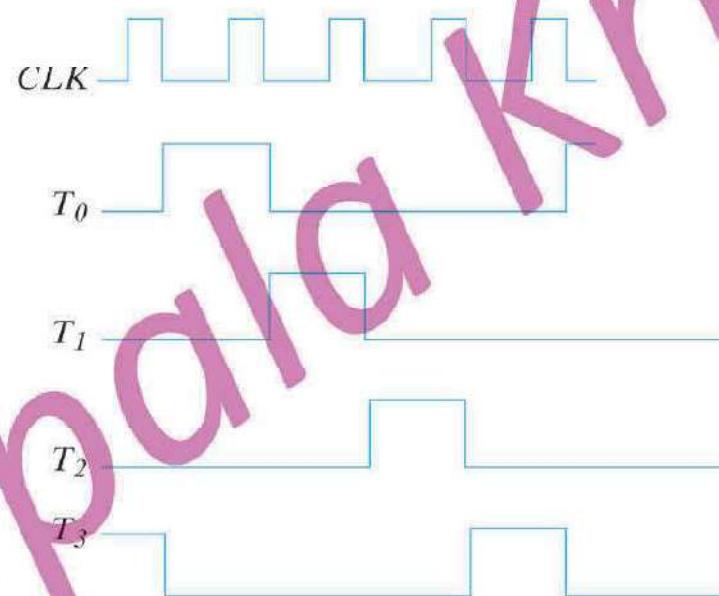
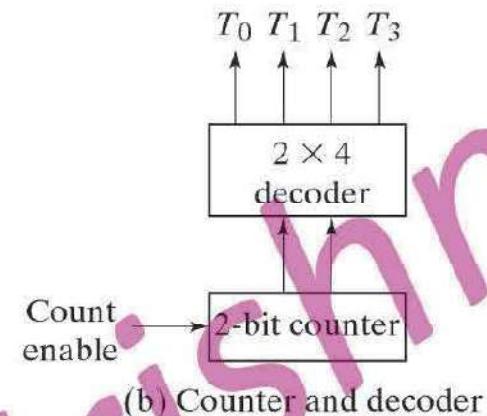
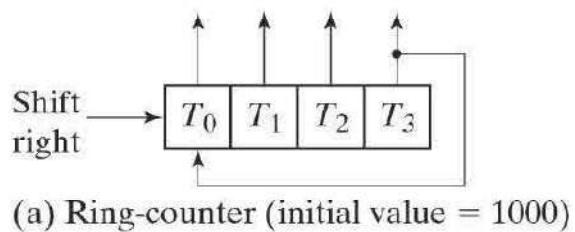
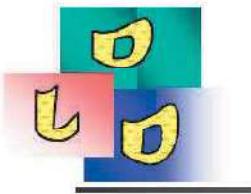
Counter with Unused States

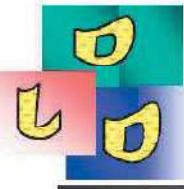




Ring Counter

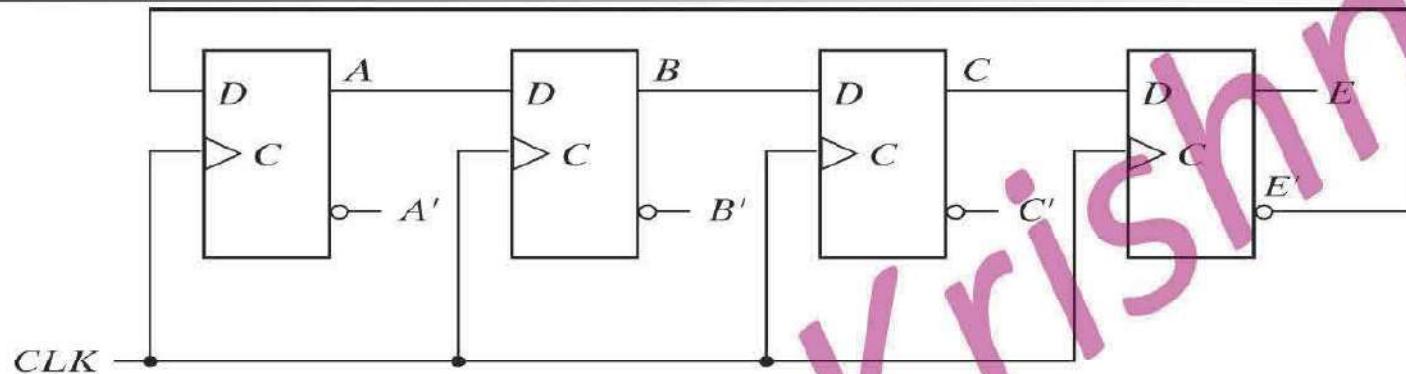
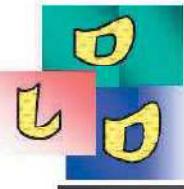
- Timing signals that control the sequence of operations in a digital system can be generated with a shift register or a counter with a decoder.
- A ring counter is a circular shift register with only one flip flop being set at any particular time, all others are cleared.
- The single bit is shifted from one flip flop to the next to produce the sequence of timing signals.
- The timing signals can be generated also by a 2-bit counter that goes through four distinct states.
- To generate 2^n timing signals, we need either a shift register with 2^n flip flops or an n bit binary counter together with an n to 2^n line decoder.





Johnson Counter

- Timing signals can be generated using a combination of shift register and decoder.
- In this way the number of flip flops is less than ring counter, and the decoder requires only 2-input gates. This combination is called Johnson Counter.
- A k-bit ring counter circulates a single bit among the flip flops to provide k distinguishable states.
- The number of states can be doubled if the shift register is connected as a switch tail ring counter.
- A switch tail ring counter is a circular shift register with the complement output of the last flip flop is connected to the input of the first flip flop



(a) Four-stage switch-tail ring counter

Sequence number	Flip-flop outputs			
	A	B	C	E
1	0	0	0	0
2	1	0	0	0
3	1	1	0	0
4	1	1	1	0
5	1	1	1	1
6	0	1	1	1
7	0	0	1	1
8	0	0	0	1

(b) Count sequence and required decoding