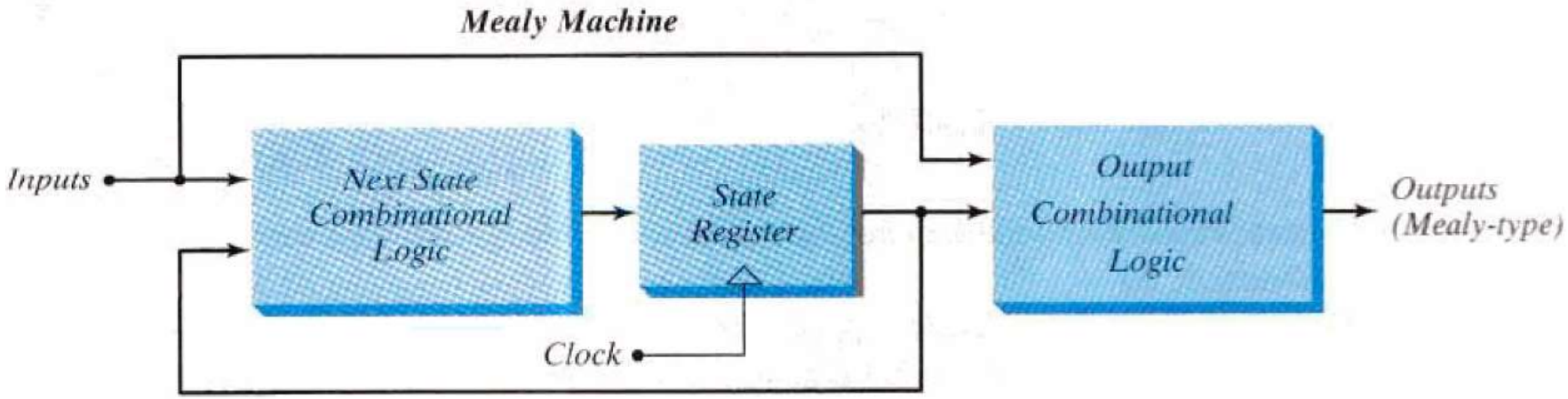
Design of Sequential Circuits

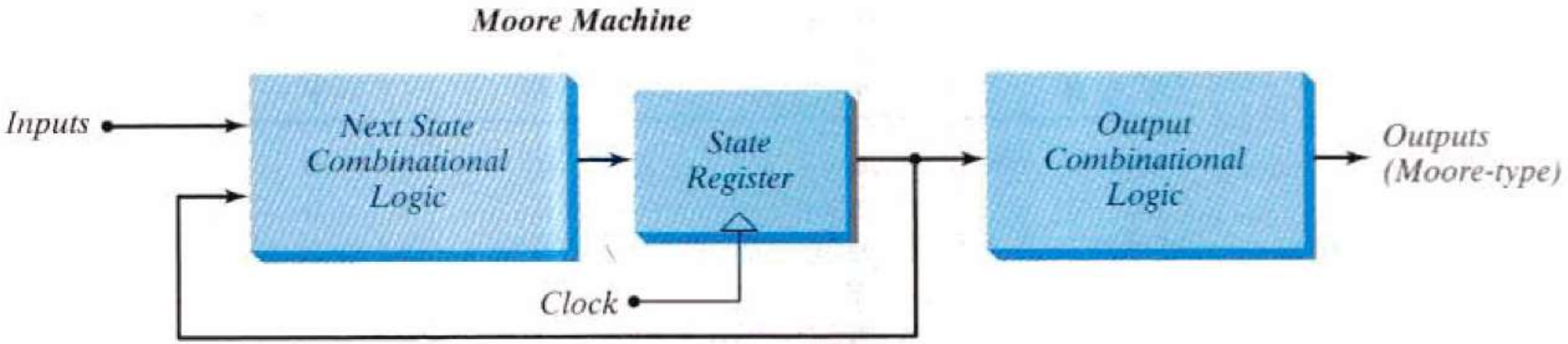
By Engr. Rimsha

Mealy and Moore Models

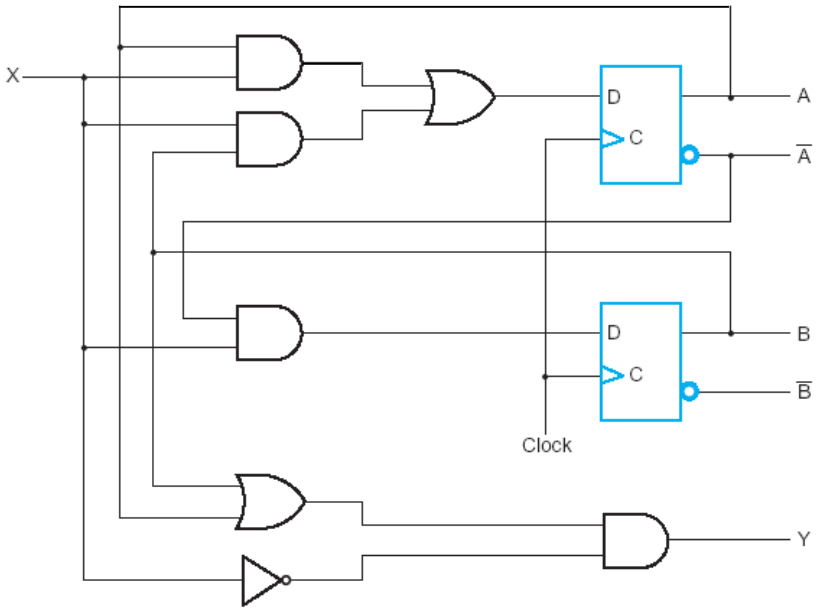
- The Mealy and Moore models differ in the way the output is generated.
  - In the Mealy model, the output is a function of both the present state and input, referred to as a Mealy finite state machine (FSM) or Mealy machine.
  - In the Moore model, the output is a function of the present state only, referred to as a Moore FSM or Moore machine.

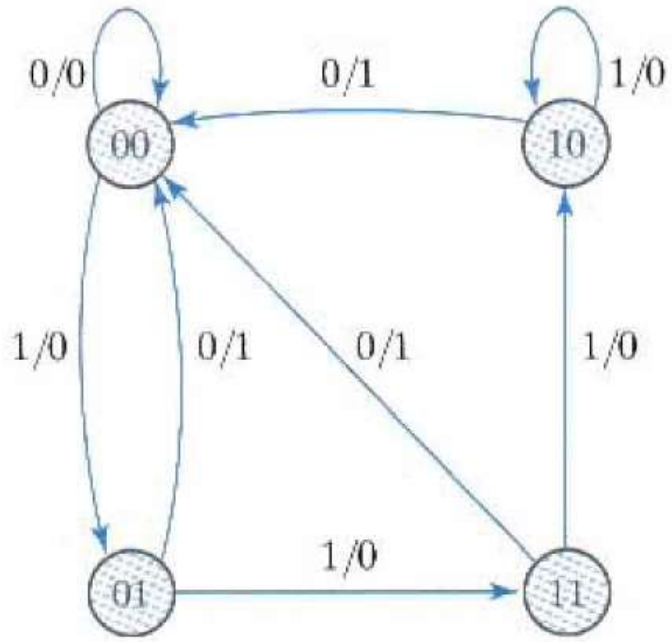
Mealay and Moore Circuits





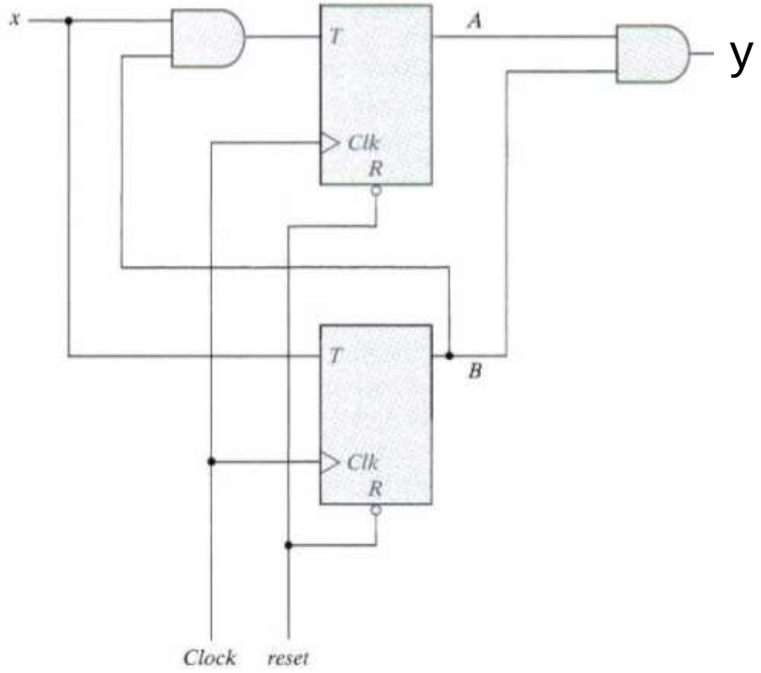
Example Mealy Model

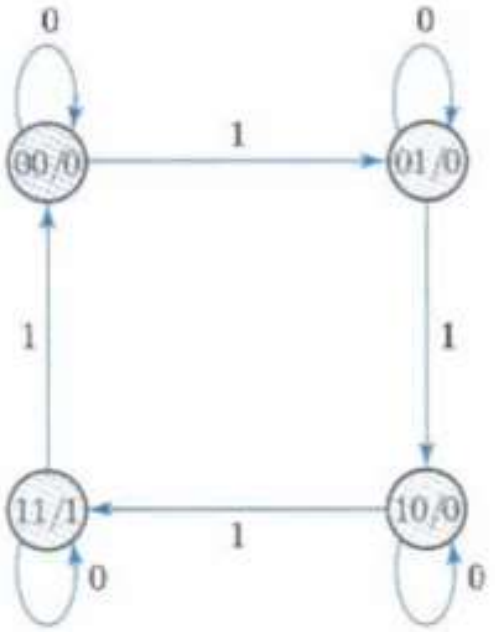




- Output y is a function of both input x and the present state of A and B.

Example Moore Model





- The output is a function of the present state only.

Notes on Mealy and Moore

- In the Moore model, the outputs of the sequential circuit are synchronized with the clock because they depend on only flip flop outputs that are synchronized with the clock.
- In the Mealy model, the outputs may change if the inputs change during the clock cycle and the outputs may have momentary false values because of the delay encountered from the time that the inputs change and the time that the flip flop outputs change.
  - To synchronize a Mealy type circuit, the inputs of the sequential circuit must be synchronized with the clock and the outputs must be sampled only during the clock edge.

Design of Sequential Circuits

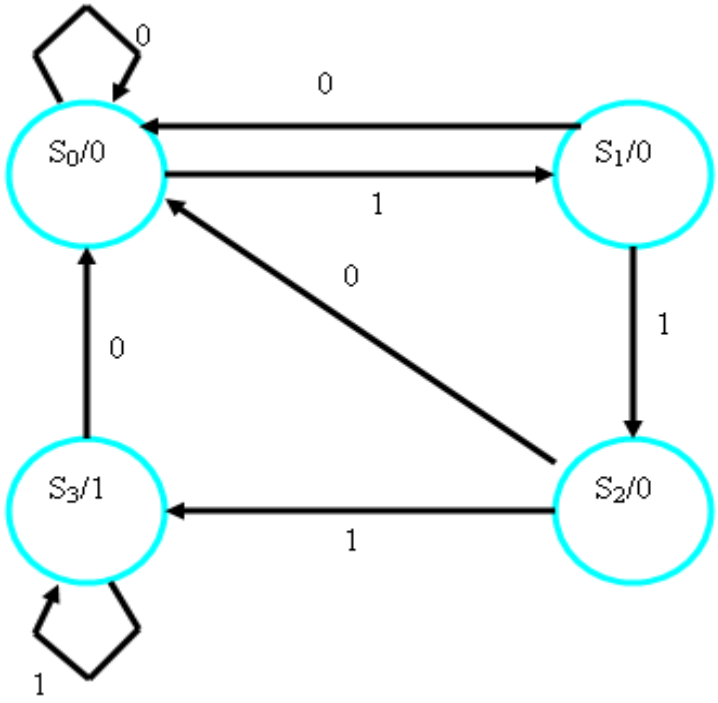
- Steps involved in design
  - Derive a state diagram
  - Reduce the number of states
  - Assign binary values
  - Obtain the binary coded state table
  - Choose the type of flip flops to be used
  - Derive the simplified flip flop input equations and output equations
    - Exitation Table?
      - » It relates present state, next state to inputs of flip flop
  - Draw the logic diagram

Design Statement

- Design a circuit that detects a sequence of three or more consecutive 1's in a string of bits coming through an input line (i.e., the input is a serial bit stream)

A Sequence Detector

- The following FSM detects a sequence of three ones.



Synthesizing Using D Flip Flops

- The next step is to create a state table and then select two D flip flops to represent the four states, labeling their outputs as A and B.
- There is one input, x, and one output, y, representing the input sequence and the output value respectively.
- Remember that the characteristic equation of the D flip flop is
  - $Q(t + 1) = D_Q$
  - This means that the next-state values in the state table specify the D input condition for the flip flop.

State Table for Sequence Detector

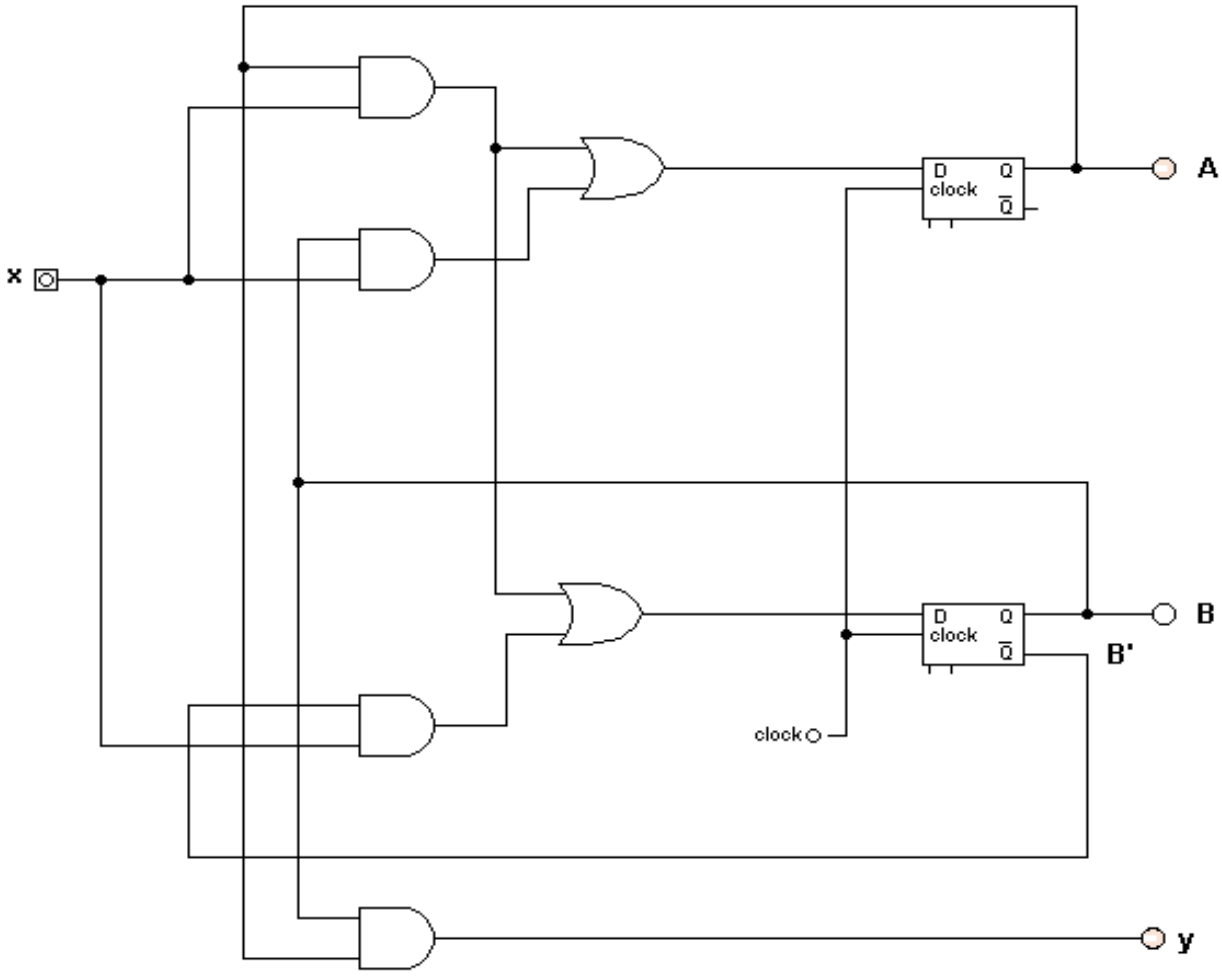
Present State			Input		Next State			Output
A	B		x		A	B		y
0	0		0		0	0		0
0	0		1		0	1		0
0	1		0		0	0		0
0	1		1		1	0		0
1	0		0		0	0		0
1	0		1		1	1		0
1	1		0		0	0		1
1	1		1		1	1		1

- **Input equations can be obtained directly from the table using minterms:**
  - **$A(t + 1) = D_A(A, B, x) = \sum(3, 5, 7)$**
  - **$B(t + 1) = DB(A, B, x) = \sum(1, 5, 7)$**
  - **$y(A, B, x) = \sum(6, 7)$**

Boolean Minimization

- K-Maps can be used to minimize the input equations, resulting in
  - $D_A = Ax + Bx$
  - $D_B = Ax + B'x$
  - $Y = AB$

Logic Diagram



Excitation Table

- The design of sequential circuits other than D type flip flops is complicated by the fact that input equations must be derived indirectly from the state table.
  - It is necessary to derive a functional relationship between the state table and the input equations.
- During the design, we usually know the transition from present to next state but we need to find the flip flop input conditions that will cause the required transition.
  - We need a table that lists the required inputs for a given change of state, called an excitation table.

Excitation Tables

JK Flip Flop			
J	**K**	**Q(t + 1)**	
0	0	Q(t)	No change
0	1	0	Reset
1	0	1	Set
1	1	Q'(t)	Complement

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

T Flip Flop		
T	**Q(t + 1)**	
0	Q(t)	No change
1	Q'(t)	Complement

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

Synthesis Using JK Flip Flops

- Synthesis of circuits with JK flip flops is the same as with D flip flops except that the input equations must be evaluated from the present-state to the next-state transition derived from the excitation table.

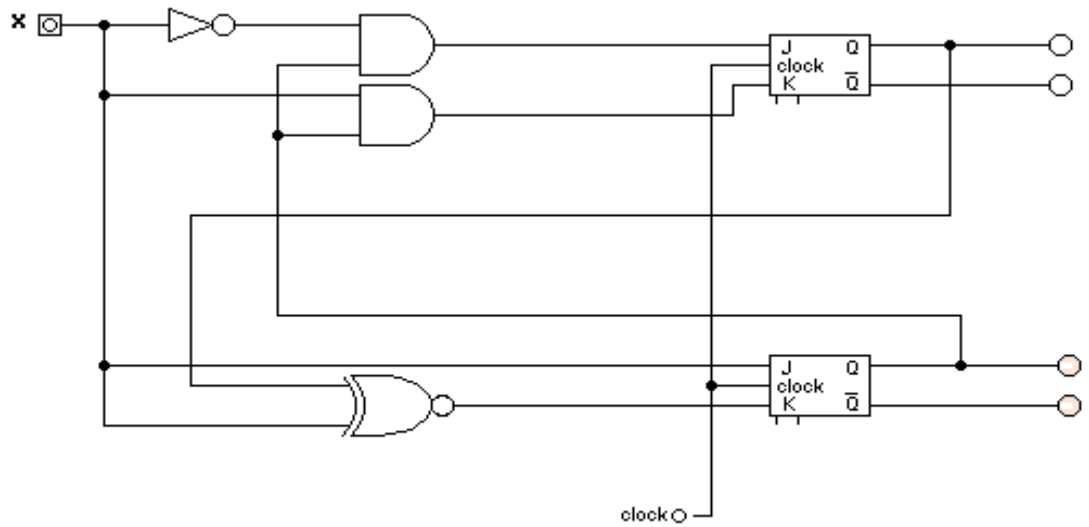
Example JK Synthesis

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

Present State		Input	Next State		Flip-Flop Inputs			
A	B	x	A	B	J_A	K_A	J_B	K_B
0	0	0	0	0	0	X	0	X
0	0	1	0	1	0	X	1	X
0	1	0	1	0	1	X	X	1
0	1	1	0	1	0	X	X	0
1	0	0	1	0	X	0	0	X
1	0	1	1	1	X	0	1	X
1	1	0	1	1	X	0	X	0
1	1	1	0	0	X	1	X	1

JK Synthesis Logic

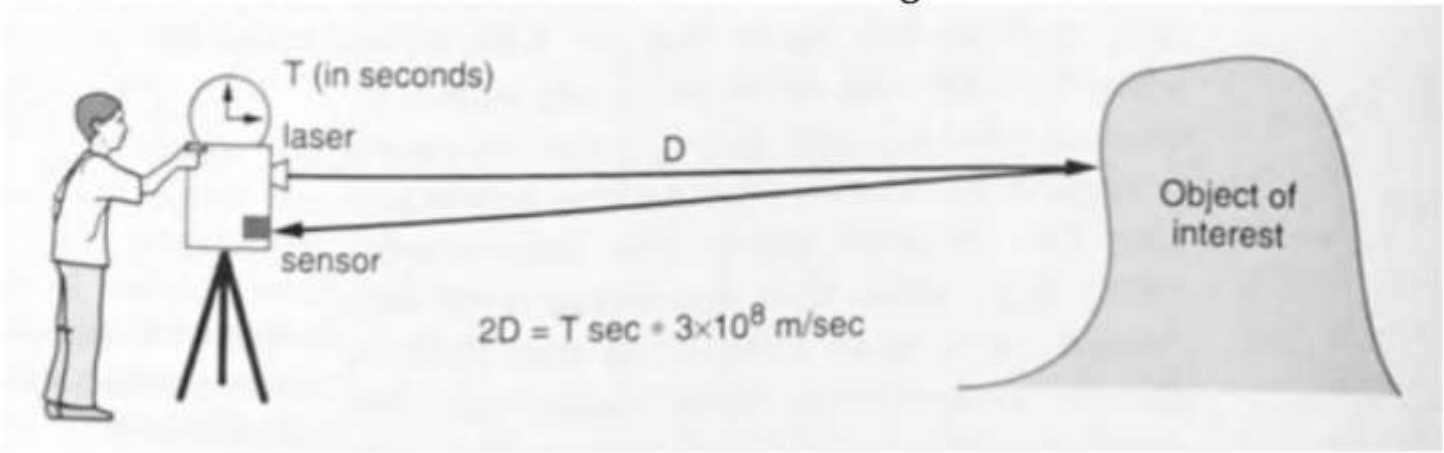
- By using K-maps we can minimize the flip flop input equations.



- Design of Sequential Circuits
  - Example to understand the sequential design

Design Statement

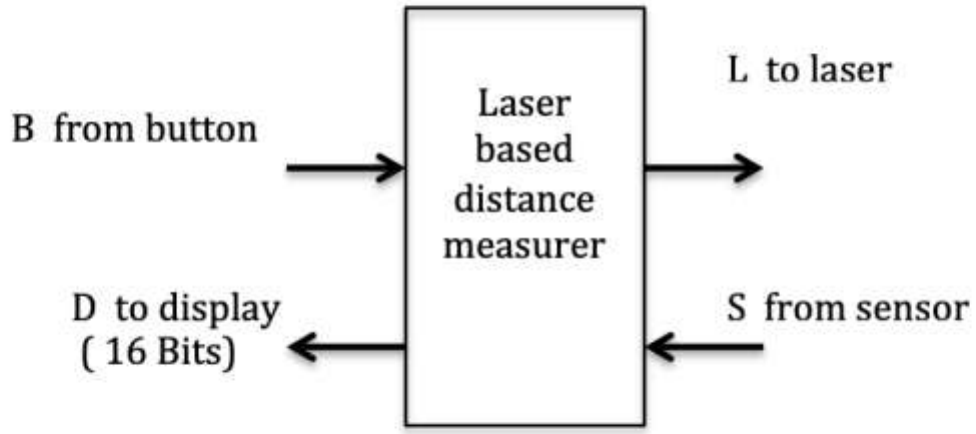
In laser based distance measurement, a laser is aimed at the object of interest. The laser is briefly turned on, and a timer is started. The laser light, traveling at the speed of light, travels to the object and reflects back. A sensor detects the reflection of the laser light, causing the timer to stop.
Laser based distance measurement is illustrated in figure below



After the reflection is detected the system should use the amount of time passed since the laser was pulsed to compute the distance to the object of interest. The system should then return to waiting for the user to press the button so that a new measurement can be taken.

Design Statement Contd..

Design a processor to control the laser and the timer and to compute distances up to 2000 meters. A block diagram is shown below



The system has a bit input B, which equals 1 when the user presses a button to start the measurement. Another bit input S comes from the sensor, and is 1 when the reflected laser is detected. The bit L controls the laser, turning the laser on when L is 1. Finally N-bit output D indicate the distance in binary, in units of meters. Let us make D 16 bits. The system uses 300 MHz clock (every cycle corresponds to 1 m) and active low reset signal.

THE END