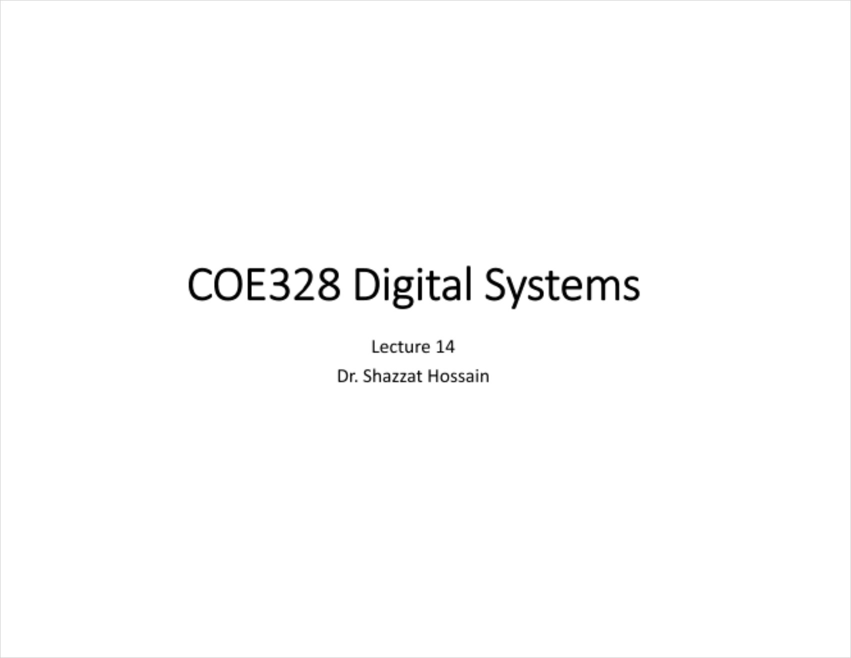
Printout

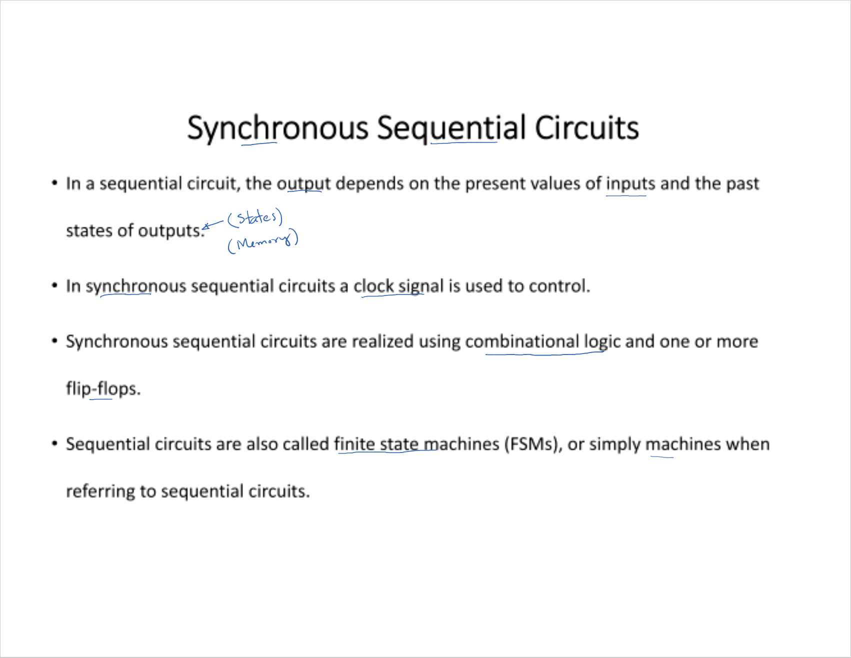
Friday, November 4, 2022 4:03 PM



COE328 Digital Systems

Lecture 14

Dr. Shazzat Hossain



Synchronous Sequential Circuits

- In a sequential circuit, the output depends on the present values of inputs and the past states of outputs. ← (States) (Memory)
- In synchronous sequential circuits a clock signal is used to control.
- Synchronous sequential circuits are realized using combinational logic and one or more flip-flops.
- Sequential circuits are also called finite state machines (FSMs), or simply machines when referring to sequential circuits.

Structure of Sequential Circuit

- The circuit shown below has a set of primary inputs, W , and produces a set of outputs, Z.
- The values of the outputs of the flip-flops are referred to as the state, Q, of the circuit.
- Under control of the clock signal, the flip-flop outputs change their state as determined by the combinational logic that feeds the inputs of these flip-flops. Thus, the circuit moves from one state to another.
- To ensure that only one transition from one state to another takes place during one clock cycle, the flip-flops have to be of the edge-triggered type.

T, D, J-K

Moore Circuit: Sequential circuits whose outputs depend only on the state of the circuit are of Moore type.
Mealy Circuit: Sequential circuits whose outputs depend on both the state and the primary inputs are of Mealy type.
These names are in honor of Edward Moore and George Mealy, who investigated the behavior of such circuits in the 1950s.

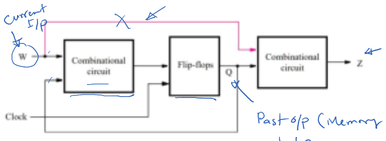


Figure 8.1 The general form of a sequential circuit.

Past o/p (Memory)
state

output

- * state diagram
- * state Table
- * K-map
- * logic function
- * Implement

Design Steps of Sequential Circuits

- Design steps for FSM:

Step#1: Determine how many states are possible

Step#2: Which transitions are possible from one state to another state

Step#3: Begin from a state (when power is turned on or reset is used)

Suppose that we wish to design a circuit that meets the following specification:

1. The circuit has one input, w, and one output, z.
2. All changes in the circuit occur on the positive edge of a clock signal.
3. The output z is equal to 1 if during two immediately preceding clock cycles the input w was equal to 1. Otherwise, the value of z is equal to 0.

Clock cycle:	t_0	t_1	t_2	t_3	t_4	t_5	t_6	t_7	t_8	t_9	t_{10}
w:	0	1	0	1	1	0	1	1	1	0	1
z:	0	0	0	0	0	1	0	0	1	1	0

Figure 8.2 Sequences of input and output signals.

State → A, B. C

T: Flip-flop

Truth Table

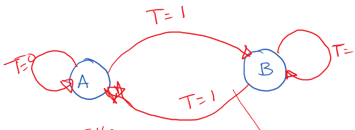
T	Q_{n+1}
0	Q_n (NC)
1	$\overline{Q_n}$

Excitation Table

	Present state Q_n	T	Next state Q_{n+1}	
A	0	0	0	A
A	0	1	1	B
B	1	0	1	B
B	1	1	0	A

state diagram

Two states
A → 0
B → 1



1. State diagram
2. state Table
3. Binary form of state Table
4. Simplify K-map
5. Implementation

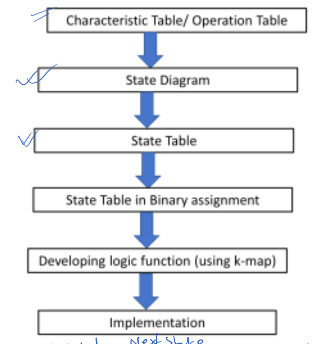
State Table

Present state Q_n	Next state Q_{n+1} T=0	T=1
A	A	B
B	B	A

z=0 w A = 0

State Diagram

Work Flow for designing a FSM



Characteristic Table/ Operation Table
State Diagram
State Table
State Table in Binary assignment
Developing logic function (using k-map)
Implementation

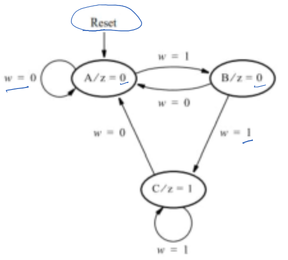
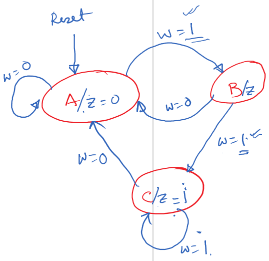


Figure 8.3 State diagram of a simple sequential circuit.



Moore design

Present state	Next state w= 0	w = 1	Output z
A	A	B	0
B	A	C	0
C	A	C	1

Present state y_2	y_1	Next State w=0 Y_2	Y_1	w=1 Y_2	Y_1	Output Z
0	0	0	0	0	1	0
0	1	0	0	1	0	0
1	0	0	0	1	0	1
1	1	d	d	d	d	d

not used state

don't care

For Y_1

$w \backslash y_2y_1$	00	01	11	10
0	0	0	d	0
1	1	0	d	0

$$Y_1 = w\bar{y}_2\bar{y}_1$$

D-Flip-Flops

2 → FF(s)

logic function

For Y_2

$w \backslash y_2y_1$	00	01	11	10
0	0	0	d	0
1	0	1	d	1

$$Y_2 = wy_1 + wy_2$$
$$= w(y_1 + y_2)$$

00 → A
01 → B
10 → C
11 →

For

For Z

$y_1 \backslash y_2$	0	1
0		1
1		d

$$Z = y_2$$

$$Y_1 = w\bar{y}_2\bar{y}_1$$

$$Y_2 = w(y_1 + y_2)$$

$$Z = y_2\bar{y}_1$$

State Table

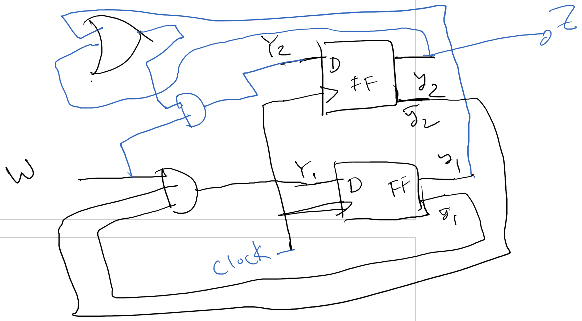
State Table: Present the information obtained from the state diagram in a tabular form called a state table.

The table indicates all transitions from each present state to the next state for different values of the input signal.

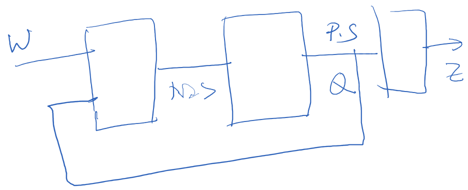
- Note that the output z is specified with respect to the present state, namely, the state that the circuit is in at present time.
- Note also that we did not include the Reset input; instead, we made an implicit assumption that the first state in the table is the starting state.

Present state	Next state $w = 0$	$w = 1$	Output z
A	A	B	0
B	A	C	0
C	A	C	1

Figure 8.4 State table for the sequential circuit in Figure 8.3.



More circuit



Implementation

	Present state y_2y_1	Next state $w = 0$ Y_2Y_1	$w = 1$ Y_2Y_1	Output z
A	00	00	01	0
B	01	00	10	0
C	10	00	10	1
	11	dd	dd	d

Figure 8.6 State-assigned table for the sequential circuit in Figure 8.4.

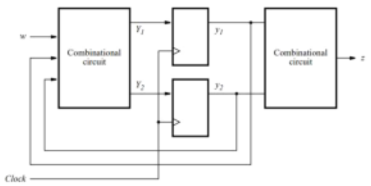


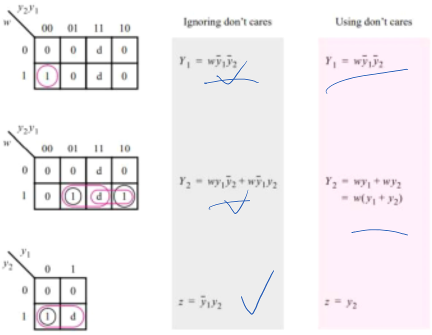
Figure 8.5 A general sequential circuit with input w, output z, and two state flip-flops.

Assign binary values for the states. Decide how many variables are required to represent the states. Don't care for unused states. For example, we need two variables for representing the states A, B, and C. Here, A=00, B=01, C=10, and the combination 11 is unused. The variables y2 and y1 are used for the present states and variables Y2 and Y1 are for the next states.

Implementation

Form k-maps for each variable Y1 and Y1 separately.
For k-maps, the variables are w, y2 and y1.
Simplify the logic functions for Y2 and Y1

We have simplified the functions for both “considering don’t care” and “ignoring don’t care” conditions.



$w \backslash y_2y_1$	00	01	11	10
0	0	0	d	0
1	1	0	d	0

$w \backslash y_2y_1$	00	01	11	10
0	0	0	d	0
1	0	1	d	1

$y_2 \backslash y_1$	0	1
0	0	0
1	1	d

Ignoring don’t cares	Using don’t cares
$Y_1 = w\bar{y}_1\bar{y}_2$ ✓	$Y_1 = w\bar{y}_1\bar{y}_2$
$Y_2 = wy_1\bar{y}_2 + w\bar{y}_1y_2$ ✓	$Y_2 = wy_1 + wy_2$ $= w(y_1 + y_2)$
$z = \bar{y}_1y_2$ ✓	$z = y_2$

Figure 8.7 Derivation of logic expressions for the sequential circuit in Figure 8.6.

Implementation

Use D Flip-Flop as D2=Y2 and D1=Y1

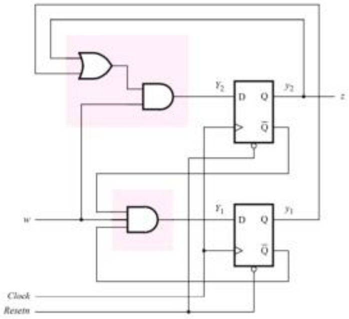


Figure 8.8 Final implementation of the sequential circuit in Figure 8.7.

Timing Diagram

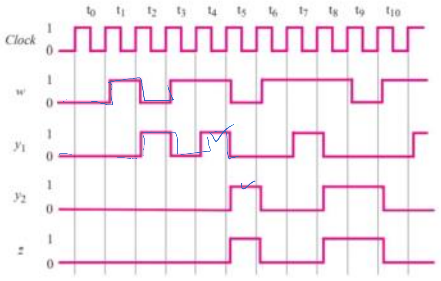


Figure 8.9 Timing diagram for the circuit in Figure 8.8.

$Y_1 =$

Example 1

Given the following logic circuit, clock signal, and input waveforms:

a) Derive the state-assigned table

b) Sketch the waveforms for Q1, Q2, Y1, and Y2 in the space provided.

Note: Assume zero delay for all gates and flip-flops.

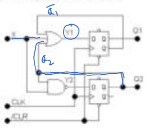
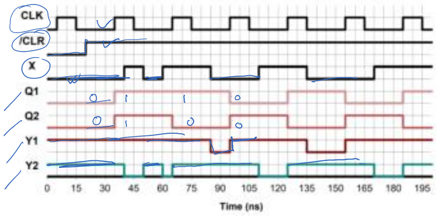


Figure 1



Q_2	Q_1	Y_1
0	0	1
0	1	0
1	0	1
1	1	1

$0 + \bar{0} + 0 = 1$

$Y_1 = X + \overline{Q_1} + Q_2$

$Y_2 = \overline{X \cdot Q_2} = \overline{0 \cdot 0} = 1$

N.S

P.S

$Q_2\,Q_1$	X=0	X=1
	$Y_2\,Y_1$	$Y_2\,Y_1$
0 0	1 1	1 1
0 1	1 0	1 1
1 0	1 1	0 1
1 1	1 1	0 1

$Y_1 = 0 + \bar{1} + 1 = 1$

$Y_2 = \overline{0 \cdot 1} = 1$

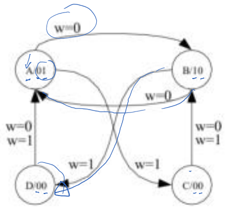
$Y_1 = 0 + \bar{1} + 0 = 0$

$Y_2 = \overline{0 \cdot 0} = 1$

Q_2	Q_1	Y_2
0	0	1
0	1	1
1	0	1
1	1	1

Example 2

4. The state diagram for a finite state machine (FSM) with one input *w* and two outputs *z2* and *z1* is given below



a) Does the above state diagram use a Moore or Mealy-type model to represent the FSM? Explain your answer.

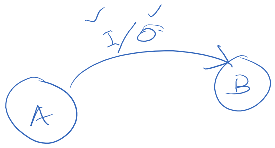
The state diagram represents Moore-type FSM, since outputs are completely defined by states and do not depend on inputs.

b) What is the minimum number of state variables required to represent the states? Explain your answer.

Two state variables are required, because $2^2 = 4$, where 4 is the number of states.

P.S	N.S $\omega=0$	$\omega=1$	O/P Z_2	Z_1
A	B	C	0	1
B	A	D	1	0
C	B	B	0	0
D	A	A	0	0

S



c) Using the state assignment: A=00, B=01, C=11, and D=10, develop the next state and output equations for implementing the FSM.

	$Q_2\,Q_1$	w=0 $D_2\,D_1$	w=1 $D_2\,D_1$	$z_2\,z_1$
A	0 0	0 1	1 1	0 1
B	0 1	0 0	1 0	1 0
C	1 1	0 1	0 1	0 0
D	1 0	0 0	0 0	0 0

D_2

$w \setminus Q_2Q_1$	00	01	11	10
0	0	0	0	0
1	1	1	0	0

$D_2 = \overline{Q_2}w$

D_1

$w \setminus Q_2Q_1$	00	01	11	10
0	1	0	1	0
1	1	0	1	0

$Y_1 = \overline{Q_2}\,\overline{Q_1} + Q_2Q_1$

z_2

$Q_2 \setminus Q_1$	0	1
0	0	1
1	0	0

$z_2 = \overline{Q_2}Q_1$

z_1

$Q_2 \setminus Q_1$	0	1
0	1	0
1	0	0

$z_1 = \overline{Q_2}\,\overline{Q_1}$