Solution to Midterm 2 Exam

Dec. 2. 2014

1. **Equivalent States** [*25 point*]

Reduce the following state transition table to a minimum number of states.
The starting state is a. The input is *X*. The output is *Z*.

state	*X* = 0	*X* = 1	Z
a	c	f	1
b	h	f	1
c	i	h	1
d	a	e	0
e	e	i	1
f	a	c	0
h	a	b	0
i	a	b	0
j	c	i	1

	b	c	d	e	f	h
b	c - h					
c	c − i f − h	h − i h − f				
d	×	×	×			
e	c − e f − i	h − e f − i	i − e h − i	×		
f	×	×	×	e - c	×	
h	×	×	×	e − b	×	c - b
i	×	×	×	e - b	×	c - b

a == j , b == c , f == h == i

Additionally, d, e, and j will never be reached because the starting state is a.

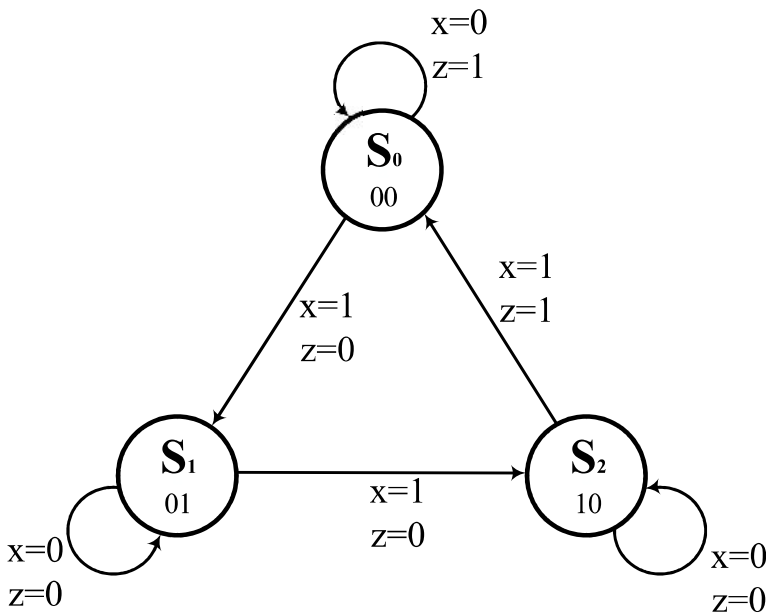## 2. Mealy State Machine Design [*25 point*]

Design a finite state machine that accepts an input binary sequence such as 001010011101… Its output is zero except when the number of 1's that have been seen is a multiple of three. For example
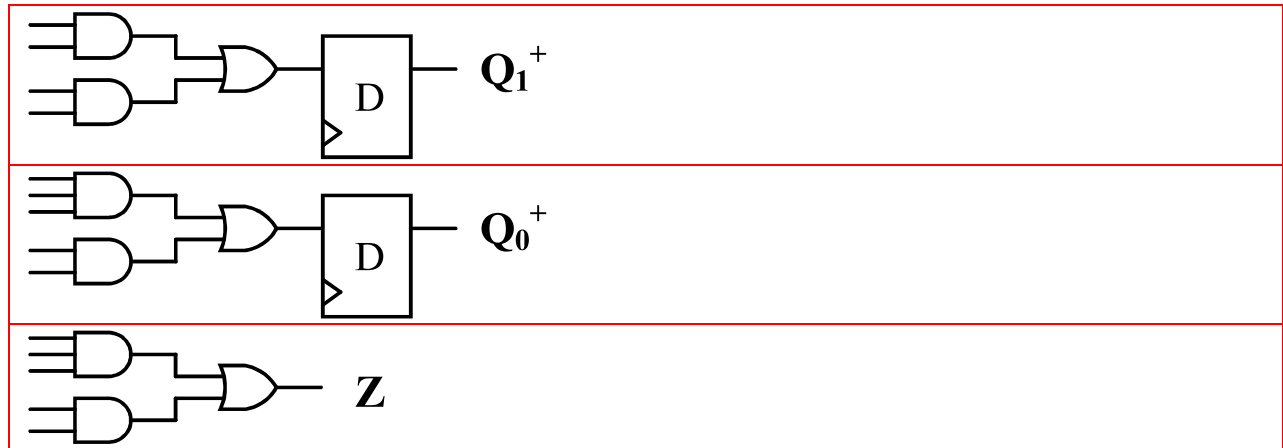
Input X :
Output Z :

(a) Draw a state transition diagram.
(b) Write a truth table for computing the input to D flip-fops for the next states.
(c) Write a truth table for computing the input to JK flip-fops for the next states.
(d) Write a truth table for computing the output.
(e) Implement the circuit with D flip-flops.
(f) Implement the circuit with JK flip-flops.

Q_1	Q_0	X	Q_1^+	Q_0^+	J_1	K_1	J_0	K_0	Z
0	0	0	0	0	0	X	0	X	1
0	0	1	0	1	0	X	1	X	0
0	1	0	0	1	0	X	X	0	0
0	1	1	1	0	1	X	X	1	0
1	0	0	1	0	X	0	0	X	0
1	0	1	0	0	X	1	0	X	1
1	1	0	X	X	X	X	X	X	X
1	1	1	X	X	X	X	X	X	X

Q_1^+		
Q_1Q_0 \ X	0	1
00	0	0
01	0	1
11	X	X
10	1	0

$$Q_1^+ = XQ_0 + \overline{X}Q_1$$

Q_0^+		
Q_1Q_0 \ X	0	1
00	0	1
01	1	0
11	X	X
10	0	0

$$Q_0^+ = X\overline{Q}_1\overline{Q}_0 + \overline{X}Q_0$$

Z		
Q_1Q_0 \ X	0	1
00	1	0
01	0	0
11	X	X
10	0	1

$$Z = \overline{X}\,\overline{Q}_1\overline{Q}_0 + XQ_1$$

Q_1^+

Q_0^+

Z

J_1		
Q_1Q_0 \ X	0	1
00	0	0
01	0	1
11	X	X
10	X	X

$$J_1 = XQ_0$$

K_1		
Q_1Q_0 \ X	0	1
00	X	1
01	X	X
11	X	X
10	0	1

$$K_1 = X$$

J_0		
Q_1Q_0 \ X	0	1
00	0	1
01	X	X
11	X	X
10	0	0

$$J_0 = X\overline{Q}_1$$

K_0		
Q_1Q_0 \ X	0	1
00	X	X
01	0	1
11	X	X
10	X	X

$$K_0 = X$$

3. Moore State Machine Design [*25 point*]

Solution The only sequences of input pairs which affect the output are of length two. Therefore, the previous and present inputs will determine the output, and the circuit must remember only the previous input pair. At first, it appears that three states are required, corresponding to the last input received being $X_1X_2 = 01, 10$ and (00 or 11). Note that it is unnecessary to use a separate state for 00 and 11 because neither input starts a sequence which leads to an output change. However, for each of these states the output could be either 0 or 1, so we will initially define six states as follows:

Previous Input (X_1X_2)	Output (Z)	State Designation
00 or 11	0	S_0
00 or 11	1	S_1
01	0	S_2
01	1	S_3
10	0	S_4
10	1	S_5

Using this state designation, we can then set up a state table (Table 14-7). The six-row table given here can be reduced to five rows, using the methods given in Unit 15.

TABLE 14-7

© Cengage Learning 2014

Present State	Z	Next State $X_1X_2 = 00$	01	11	10
S_0	0	S_0	S_2	S_0	S_4
S_1	1	S_1	S_3	S_1	S_5
S_2	0	S_0	S_2	S_0	S_4
S_3	1	S_1	S_3	S_0	S_5
S_4	0	S_0	S_3	S_1	S_4
S_5	1	S_1	S_2	S_1	S_5

The S_4 row of this table was derived as follows. If 00 is received, the input sequence has been 10, 00, so the output does not change, and we go to S_0 to remember that the last input received was 00. If 01 is received, the input sequence has been 10, 01, so the output must change to 1, and we go to S_3 to remember that the last input received was 01. If 11 is received, the input sequence has been 10, 11, so the output should become 1, and we go to S_1. If 10 is received, the input sequence has been 10, 10, so the output does not change, and we remain in S_4. Verify for yourself that the other rows in the table are correct. The state graph is shown in Figure 14-18.

FIGURE 14-18
State Graph for Example 3

© Cengage Learning 2014

4
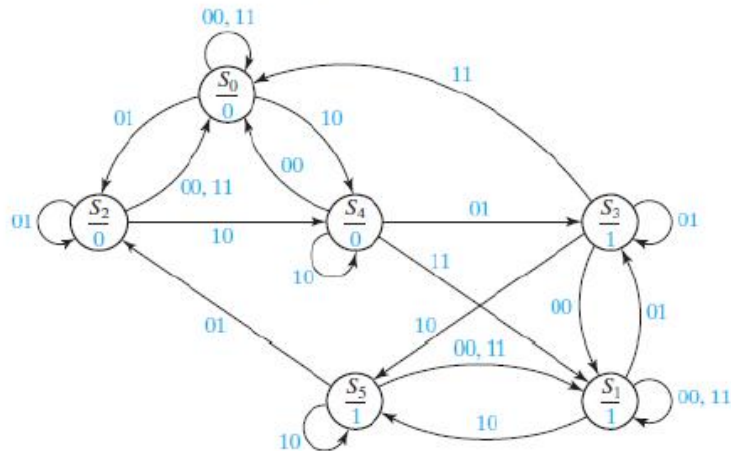
4. An Arithmetic Circuit [*25 point*]

The *Collatz* conjecture is a famous open problem in mathematics, proposed by Lothar Collatz in 1937. Consider the following iterative procedure. For any positive integer x,

- if x is odd, let $x = 3x + 1$;
- else let $x = x / 2$.

The conjecture is that, starting with any positive integer x, the procedure always terminates with $x = 1$. For instance, starting with $x = 5$, one follows the sequence 16, 8, 4, 2 and 1.

(a) Design a combinational circuit to implement a **single step** of the *Collatz* procedure, as follows. Let the bits $X = (X_0, \ldots, X_7)$ represent the initial value of x, least significant bit first. Let the bits $Y = (Y_0, \ldots, Y_7)$ represent the results, either $3X + 1$ or $X / 2$.

(b) Design a sequential circuit that **iterates** through the *Collatz* procedure. Design your circuit with inputs X, *start* and *clock*, and output Y. When *start* is set to 1, your circuit should read the initial value $X = (X_0, \ldots, X_7)$. Then, when *start* is set to 0, your circuit should iterate through the *Collatz* procedure – one iteration per clock cycle – indefinitely. It should **wrap around** 256: if the expected value of X is greater than 255, the value should be set to $X - 256$. If the value is 1, let it compute $3X + 1 = 4$ and keep going.

(c) Add a 8-bit counter with outputs $C_0, \ldots, C_7$ to your design. When *start* is set to 1, the counter should get cleared. Then it should count up for every iteration. Let it wrap around 256.
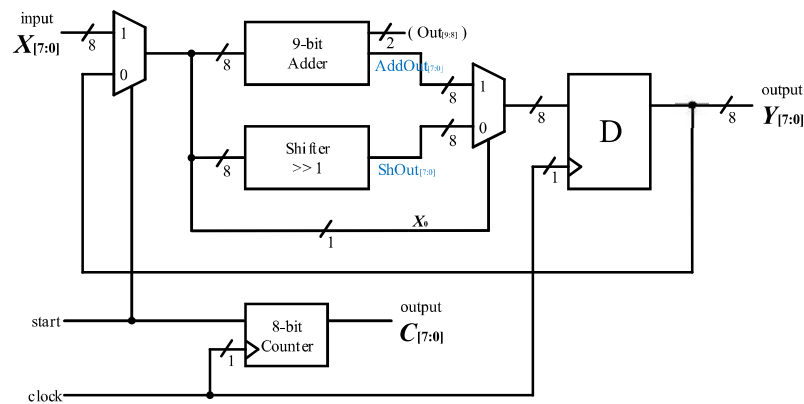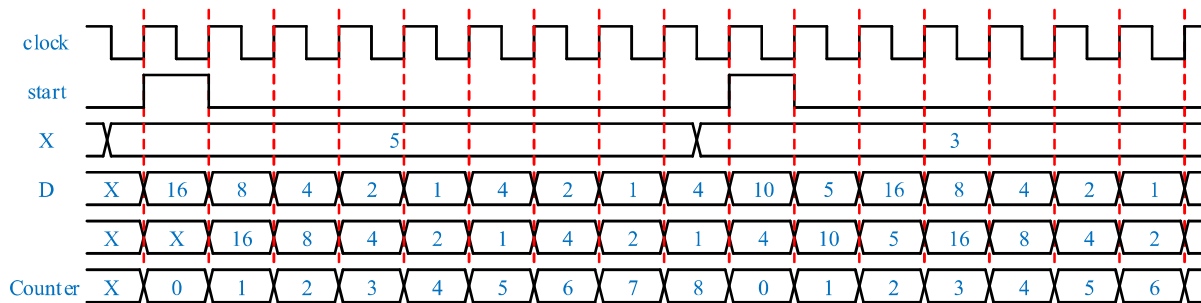
a. '*start*' / Feedback connection with Mux   (3pt)

b. Odd / Even check with Mux (LSB : x_0)   (2pt)

c. 8-bit Register with '*clock*' signal   (5pt)

d. Counter with '*start*' and '*clock*' signal   (5pt)

e. Combination Logic Circuit – $3X+1$ (Adder×2 or 9-bit Adder)   (5pt)

f. Combination Logic Circuit – $X/2$ (Shifter)   (5pt)

5