Problem

Design a Mealy sequential circuit (Figure 16-27) which investigates an input sequence X and will produce an output of $Z = 1$ for any input sequence ending in 0010 or 100.

Example:

$$X = 1\ 1\ 0\ 0\ 1\ 0\ 0\ 1\ 0\ 1\ 0\ 0\ 1\ 0\ 1$$
$$Z = 0\ 0\ 0\ 1\ 0\ 1\ 1\ 0\ 1\ 0\ 0\ 1\ 0\ 1\ 0$$

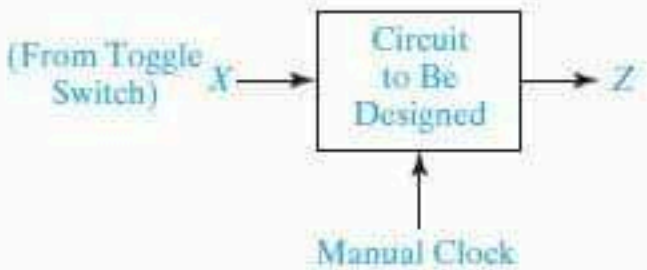
Notice that the circuit does not reset to the start state when an output of Z = 1 occurs. However, your circuit should have a start state and should be provided with a method for manually resetting the flip-flops to the start state. A minimum solution requires six states. Design your circuit using NAND gates, NOR gates, and three D flip-flops. Any solution which is minimal for your state assignment and uses 10 or fewer gates and inverters is acceptable. (Assign 000 to the start state.)

Test Procedure: First, check out your state table by starting in each state and making sure that the present output and next state are correct for each input. Then, starting in the proper initial state, determine the output sequence for each of the following input sequences:

(1) 0 0 1 1 0 1 0 0 1 0 1 0 1 0 0 0 1 0 0 1 0 0 1 0

(2) 1 1 0 0 1 1 0 0 1 0 1 0 1 0 0 1 0 1 0 1 0 0 1 0

FIGURE 16-27



First construct a state diagram that shows two paths that lead to an output of 1. One is for the sequence 0010 and the other is for the sequence 100. Let $S0$ be the start state.

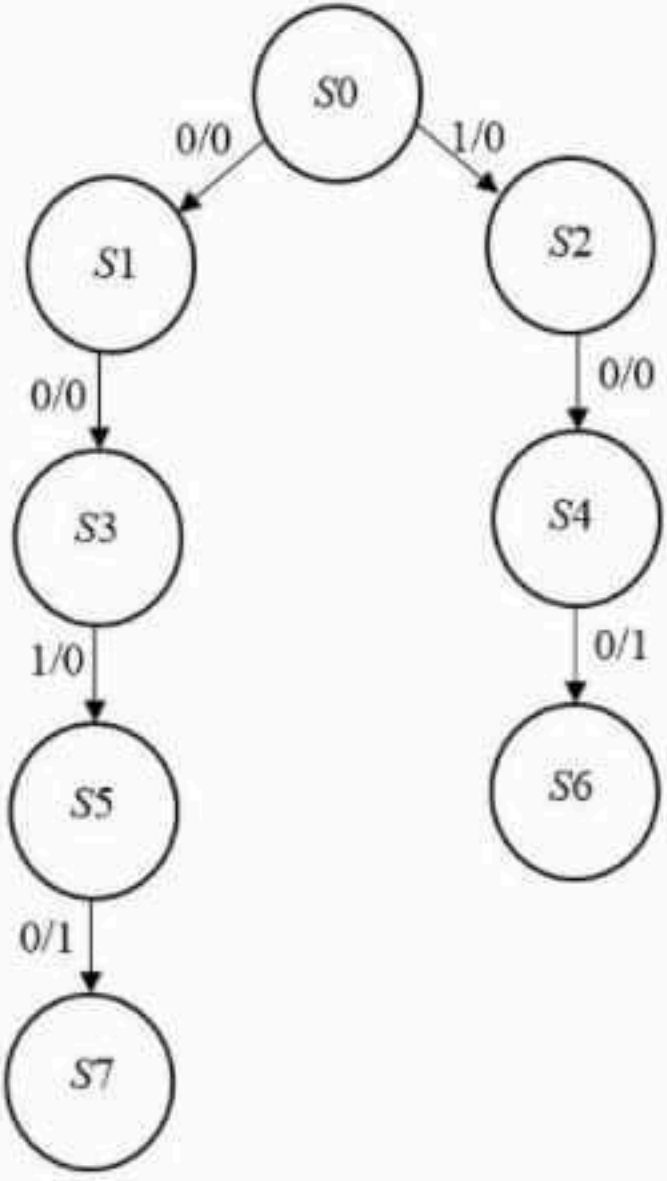


Figure 1

Now the rest of the state diagram needs to be determined. Note that the circuit does not reset to the start state when an output of 1 occurs. Therefore the interconnecting relationship between the two state sequences that leads to an output of 1 needs to be determined. First construct a table which lists all of the states and lists the possible next input not yet considered and the resulting sequence. Also list the possibility for any of the two sequences of interest from the resulting sequence. Also list the output and possible next states.

Present state	Next input	Resulting sequence	100 possible	0010 possible	Next state	Output
S1	1	01	Yes	No	*S2*	0
S2	1	11	Yes	No	*S2*	0
S3	0	000	No	Yes	*S3*	0
S4	1	101	Yes	No	*S2*	0
S5	1	0011	Yes	No	*S2*	0
S6	0	1000	No	Yes	*S3*	0
S6	1	1001	No	Yes	*S5*	0
S7	0	00100	Occurred	Yes	*S6*	1
S7	1	00101	Yes	No	*S2*	0

Table 1: Table that considers possible cases.

Now from Figure 1 and table 1 get the full state table shown as table 2.

Table 2: State table

State	Next state	Output
	$X = 0, X = 1$	$X = 0, X = 1$
S0	S1, S2	0, 0
S1	S3, S2	0, 0
S2	S4, S2	0, 0
S3	S3, S5	0, 0
S4	S6, S2	1, 0
S5	S7, S2	1, 0
S6	S3, S5	0, 0
S7	S6, S2	1, 0

Now verify the state table using the first given input sequence.

Table 3: Output and next state sequence for the first given input sequence

Current state	Input	Next state	Output
S0	0	S1	0
S1	0	S3	0
S3	1	S5	0
S5	1	S2	0
S2	0	S4	0
S4	1	S2	0
S2	0	S4	0
S4	0	S6	1
S6	1	S5	0
S5	0	S7	1
S7	1	S2	0
S2	0	S4	0
S4	1	S2	0

S4	1	S2	0
S2	0	S4	0
S4	0	S6	1
S6	0	S3	0
S3	1	S5	0
S5	0	S7	1
S7	0	S6	1
S6	1	S5	0
S5	0	S7	1
S7	0	S6	1
S6	1	S5	0
S5	0	S7	1

The output sequence is correct which means that the state table will correctly detect the desired input sequences. The same table can be made for the second given input sequence.

Now a state assignment needs to be chosen one where $S0$ is 000. Observe that $S5$ and $S2$ have the same next state so those should be adjacent. The same is true for $S4$ and $S1$ and $S1$ and $S3$. Therefore use the state assignment shown in as table 5.

Table 5: State assignment

$Q_2Q_3Q_1$	0	1
00	$S0$	$S3$
01	$S2$	
11	$S5$	
10	$S4$	$S1$

Now from the state table get a Karnaugh map for Q_1^*. Also for all of the unused state assignments let the values on the Karnaugh maps be 0. That makes it is possible to restart the circuit since the next state for those cases will be *S*0.

Table 7: Karnaugh map for Q_1^*

$Q_2Q_3 \setminus XQ_1$	00	01	11	10
00	1	1	0	0
01	0	0	0	0
11	0	0	0	0
10	1	1	0	0

Now from its Karnaugh map get a logic expression for Q_1^* and use De'Morgan's theorem to simplfy the logic expression so that it can be implemented with NAND gates, NOR gates, and inverters.

$$Q_1^* = X'Q_3'$$

$$\boxed{Q_1^* = (X + Q_3)'}$$

Now that the state table has been verified it should be minimized by combining equivalent states. Observe that states S7 and S4 have the same next states and outputs. This means that these two states can be combined into 1 state. The same is true for S3 and S6. Get the reduced state table shown as table 4.

Table 4: Reduced state table

State	Next state	Output
	$X = 0, X = 1$	$X = 0, X = 1$
S0	S1, S2	0, 0
S1	S3, S2	0, 0
S2	S4, S2	0, 0
S3	S3, S5	0, 0
S4	S3, S2	1, 0
S5	S4, S2	1, 0

Now plug in the state assignment to the state table.

Table 6: State table with state assignment plugged in

State $Q_1Q_2Q_3$	Next state $Q_1^+Q_2^+Q_3^+$	Output
	$X=0, X=1$	$X=0, X=1$
000	110, 001	0, 0
110	100, 001	0, 0
001	010, 001	0, 0
100	100, 011	0, 0
010	100, 001	1, 0
011	010, 001	1, 0

Now from the state table get a Karnaugh map for Q_2^+.

Table 8: Karnaugh map for Q_2^+

$Q_2Q_3 \setminus XQ_1$	00	01	11	10
00	1	0	1	0
01	1	0	0	0
11	1	0	0	0
10	0	0	0	0

Now from its Karnaugh map get a logic expression for Q_2^+ and use De'Morgan's theorem to simplify the logic expression so that it can be implemented with NAND gates, NOR gates, and inverters.

$$Q_2^+ = Q_2' X' Q_1' + Q_2 X' Q_1' + X Q_1 Q_2' Q_3'$$

$$= \overline{\left(Q_2' X' Q_1'\right)'} + \overline{\left(Q_2 X' Q_1'\right)'} + \overline{\left(X Q_1 Q_2' Q_3'\right)'}$$

$$\boxed{Q_2^+ = \overline{\left(Q_2' X' Q_1'\right)' \left(Q_2 X' Q_1'\right)' \left(X Q_1 Q_2' Q_3'\right)'}}$$

Comment

Now from the state table get a Karnaugh map for Q_3^+.

Table 9: Karnaugh map for Q_3^+

$Q_2Q_3 \setminus XQ_1$	00	01	11	10
00	0	0	1	1
01	0	0	0	1
11	0	0	0	1
10	0	0	1	1

Step 13 of 14

Now from its Karnaugh map get a logic expression for Z and use De Morgan's theorem to simplify the logic expression so that it can be implemented with NAND gates, NOR gates, and inverters.

$$Z = X'Q_1'Q_2$$

$$\boxed{Z = \overline{X + Q_1 + Q_2'}}$$

Comment

Step 14 of 14

The equations for the circuit are summarized as follows.

$$\boxed{Z = \overline{X + Q_1 + Q_2'}}$$

$$\boxed{Q_1^* = (X + Q_2)'}$$

$$\boxed{Q_2^* = \overline{\left(Q_2'XQ_1'\right)'\left(Q_2XQ_1'\right)'\left(XQ_1Q_2'Q_1'\right)'}}$$

$$\boxed{Q_1^* = \overline{\left(XQ_1'\right)'\left(XQ_1'\right)'}}$$

Observe from these equations that 2 NOR gates, 7 NAND gates, 1 inverter for the input X, and 3 D flip-flops are needed. The total gate count is 10 which meets the specification.

Now from its Karnaugh map get a logic expression for Q_3' and use De'Morgan's theorem to simplify the logic expression so that it can be implemented with NAND gates, NOR gates, and inverters.

$$Q_3' = XQ_3' + XQ_1'$$
$$= \overline{\left(XQ_3'\right)'} + \overline{\left(XQ_1'\right)'}$$

$$\boxed{Q_3' = \overline{\left(XQ_3'\right)'\left(XQ_1'\right)'}}$$

Comment

Now from the state table get a Karnaugh map for Z.

Table 10: Karnaugh map for Z

$Q_2Q_3 \setminus XQ_1$	00	01	11	10
00	0	0	0	0
01	0	0	0	0
11	1	0	0	0
10	1	0	0	0

Problem

Design a Mealy sequential circuit (Figure 16-27) which investigates an input sequence X and will produce an output of $Z = 1$ for any input sequence ending in 1101 or 011.

Example:

$$X = 0\ 0\ 1\ 1\ 0\ 1\ 1\ 0\ 1\ 0\ 1\ 1\ 0\ 1\ 0$$
$$Z = 0\ 0\ 0\ 1\ 0\ 1\ 1\ 0\ 1\ 0\ 0\ 1\ 0\ 1\ 0$$

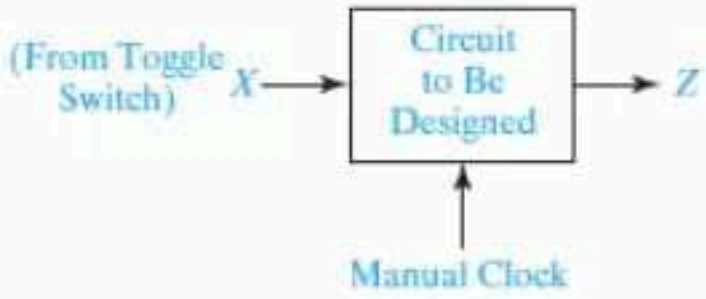
Notice that the circuit does not reset to the start state when an output of Z = 1 occurs. However, your circuit should have a start state and should be provided with a method for manually resetting the flip-flops to the start state. A minimum solution requires six states. Design your circuit using NAND gates, NOR gates, and three D flip-flops. Any solution which is minimal for your state assignment and uses nine or fewer gates and inverters is acceptable. (Assign 000 to the start state.)

Test Procedure: First, check out your state table by starting in each state and making sure that the present output and next state are correct for each input. Then, starting in the proper initial state, determine the output sequence for each of the following input sequences:

(1) 1 1 0 0 1 0 1 1 0 1 0 1 0 1 1 1 0 1 1 0 1 1 0 1

(2) 0 0 1 1 0 0 1 1 0 1 0 1 0 1 1 0 1 0 1 0 1 1 0 1

FIGURE 16-27



Now from the state table get a Karnaugh map for Z.

Table 10: Karnaugh map for Z

$Q_2Q_3 \setminus XQ_1$	00	01	11	10
00	0	0	0	0
01	0	0	0	0
11	1	0	0	0
10	1	0	0	0

Now from its Karnaugh map get a logic expression for Q_2^+ and use De'Morgan's theorem to simplfy the logic expression so that it can be implemented with NAND gates, NOR gates, and inverters.

$$Q_2^+ = \overline{X'Q_1Q_2'Q_3'} + \overline{XQ_1'Q_2'} + \overline{XQ_1'Q_3}$$

$$= \overline{\left(X'Q_1Q_2'Q_3'\right)'} + \overline{\left(XQ_1'Q_2'\right)'} + \overline{\left(XQ_1'Q_3\right)'}$$

$$\boxed{Q_2^+ = \overline{\left(X'Q_1Q_2'Q_3'\right)'\left(XQ_1'Q_2'\right)'\left(XQ_1'Q_3\right)'}}$$

Now from the state table get a Karnaugh map for Q_1^+. Also for all of the unused state assignments let the values on the Karnaugh maps be 0. That makes it is possible to restart the circuit since the next state for those cases will be S0.

Table 7: Karnaugh map for Q_1^+

$Q_2Q_3 \setminus XQ_1$	00	01	11	10
00	0	0	1	1
01	0	0	0	0
11	0	0	0	0
10	0	0	1	1

Now that the state table has been verified it should be minimized by combining equivalent states. Observe that states $S7$ and $S4$ have the same next states and outputs. This means that these two states can be combined into 1 state. The same is true for $S3$ and $S6$. Get the reduced state table shown below as table 4.

Table 4: Reduced state table

State	Next state	Output
	$X = 0, X = 1$	$X = 0, X = 1$
S0	S2, S1	0, 0
S1	S2, S3	0, 0
S2	S2, S4	0, 0
S3	S5, S3	0, 0
S4	S2, S3	0, 1
S5	S2, S4	0, 1

Now from figure 1 and table 1 get the full state table shown as table 2.

Table 2: State table

State	Next state	Output
	$X = 0, X = 1$	$X = 0, X = 1$
S0	S2, S1	0, 0
S1	S2, S3	0, 0
S2	S2, S4	0, 0
S3	S5, S3	0, 0
S4	S2, S6	0, 1
S5	S2, S7	0, 1
S6	S5, S3	0, 0
S7	S2, S6	0, 1

Now the rest of the state diagram needs to be determined. Note that the circuit does not reset to the start state when an output of 1 occurs. Therefore the interconnecting relationship between two state sequences that lead to an output of 1 needs to be determined. First construct a table which lists all of the states and lists the possible next input not yet considered and the resulting sequence. Also list the possibility for any of the two sequences of interest from the resulting sequence. Also list the output and possible next states.

Table 1: Table that considers possible cases

Present state	Next input	Resulting sequence	011 possible	1101 possible	Next state	Output
S1	0	10	Yes	No	S2	0
S2	0	00	Yes	No	S2	0
S3	1	111	No	Yes	S3	0
S4	0	010	Yes	No	S2	0
S5	0	1100	Yes	No	S2	0
S6	0	0110	No	Yes	S5	0
S6	1	0111	No	Yes	S3	0
S7	0	11010	Yes	No	S2	0
S7	1	11011	Occurred	Yes	S6	1

First construct a state diagram that shows two paths that lead to an output of 1. One is for the sequence 1101 and the other is for the sequence 011. Let S0 be the start state.

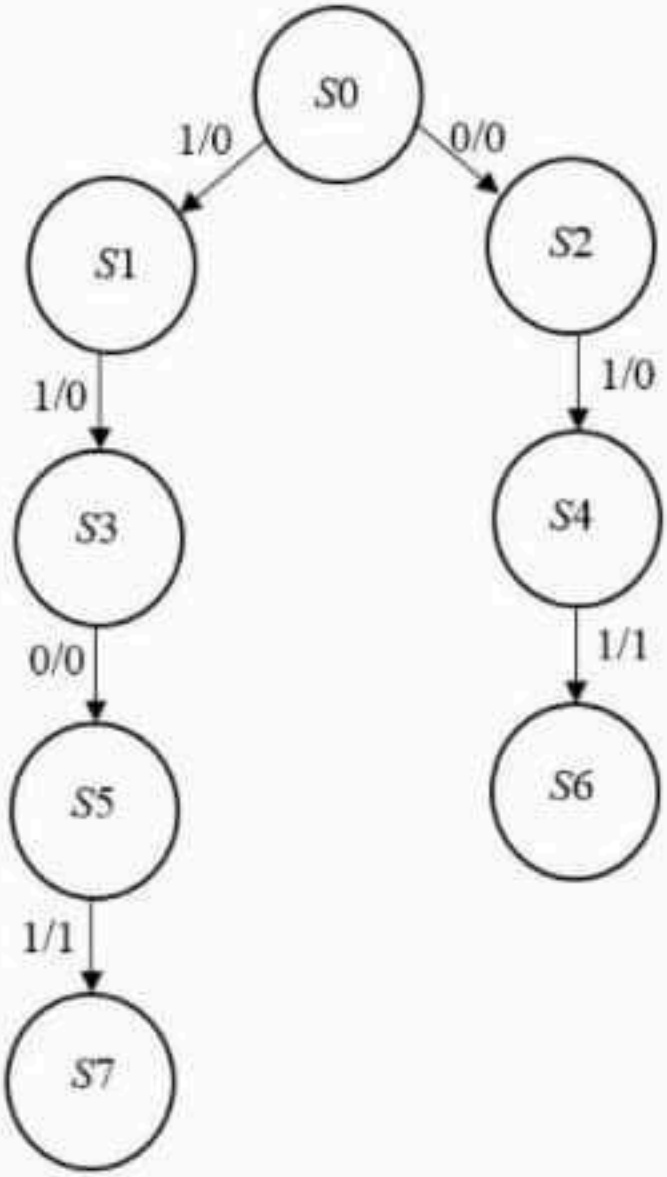


Figure 1: Initial state diagram which shows possible ways to get output of 1.

Now verify the state table using the first given input sequence.

Table 3: Output and next state sequence for the first given input sequence

Current state	Input	Next state	Output
S0	1	S1	0
S1	1	S3	0
S3	0	S5	0
S5	0	S2	0
S2	1	S4	0
S4	0	S2	0
S2	1	S4	0
S4	1	S6	1
S6	0	S5	0
S5	1	S7	1
S7	0	S2	0
S2	1	S4	0

S4	0	S2	0
S2	1	S4	0
S4	1	S6	1
S6	1	S3	0
S3	0	S5	0
S5	1	S7	1
S7	1	S6	1
S6	0	S5	0
S5	1	S7	1
S7	1	S6	1
S6	0	S5	0
S5	1	S7	1

The output sequence is correct which means that the state table will correctly detect the desired input sequences. The same table can be made for the second given input sequence to further verify the state table.

Now plug in the state assignment to the state table.

Table 6: State table with state assignment plugged in

State $Q_1Q_2Q_3$	Next state $Q_1^+Q_2^+Q_3^+$	Output
	$X = 0, X = 1$	$X = 0, X = 1$
000	001, 110	0, 0
110	001, 100	0, 0
001	001, 010	0, 0
100	011, 100	0, 0
010	001, 100	0, 1
011	001, 010	0, 1

Now a state assignment needs to be chosen one where S0 is 000. Observe that S5 and S2 have the same next state so those should be adjacent. The same is true for S4 and S1. Therefore use the state assignment shown in as table 5.

Table 5: State assignment

$Q_2Q_3 \setminus Q_1$	0	1
00	S0	S3
01	S2	
11	S5	
10	S4	S1

Now from its Karnaugh map get a logic expression for Q_1^+ and use De'Morgan's theorem to simplfy the logic expression so that it can be implemented with NAND gates, NOR gates, and inverters.

$$Q_1^+ = XQ_3'$$

$$\boxed{Q_1^+ = (X' + Q_3)'}$$

Now from the state table get a Karnaugh map for Q_2^+.

Table 8: Karnaugh map for Q_2^+

$Q_2Q_3 \setminus XQ_1$	00	01	11	10
00	0	1	0	1
01	0	0	0	1
11	0	0	0	1
10	0	0	0	0

Now from its Karnaugh map get a logic expression for Q_3^+ and use De'Morgan's theorem to simplfy the logic expression so that it can be implemented with NAND gates, NOR gates, and inverters.

$$Q_3^+ = X'Q_1' + X'Q_3'$$

$$= \overline{\left(X'Q_1'\right)'} + \overline{\left(X'Q_3'\right)'}$$

$$\boxed{Q_3^+ = \overline{\left(X'Q_1'\right)'\left(X'Q_3'\right)'}}$$

Now from the state table get a Karnaugh map for Q_3^*.

Table 9: Karnaugh map for Q_3^*

$Q_2Q_3 \setminus XQ_1$	00	01	11	10
00	1	1	0	0
01	1	0	0	0
11	1	0	0	0
10	1	1	0	0