

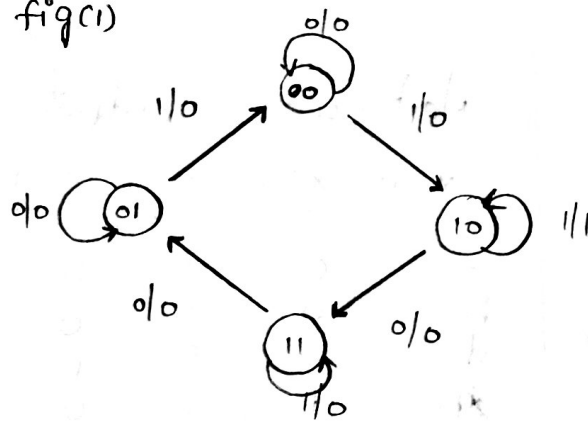
(1) Explain the following related to sequential circuits with suitable examples.

(i) State diagram.

(ii) State table.

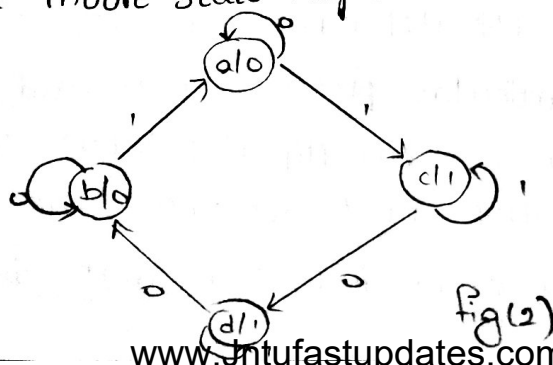
(iii) State assignment.

(A) state diagram:- A state diagram is defined as the diagrammatic representation of the performance of a sequential circuit. It is shown in fig(1)



In fig(1), the circles indicate the state of the sequential circuit. Each circle consists of binary number, which represented the specific state of the sequential circuit. The circles are connected by means of two direct lines i.e; one line connects the two circles, indicating the change in state whereas the other line connects the circle itself indicating that the next state is similar to the present state. Over the direct lines there are two binary digits separated by the symbol indicates '/'. The digit prior to the symbol indicates the input value, which results in transformation of state and the digit after the symbol indicates output value during the transition.

Example: The moore state diagram is shown in fig(2).



The direct lines have only one binary digit over them, which indicates the input resulting in the transformation of state whereas the output is indicated within the circle exactly below the present state. This is because the output state is independent of input and depends only on the present state.

(ii) State Stable:-

A stable table is a tabular representation of information contained in a state diagram. With the help of state table, it is easy to understand the implementation of a sequential circuit. An example of a state table is shown below.

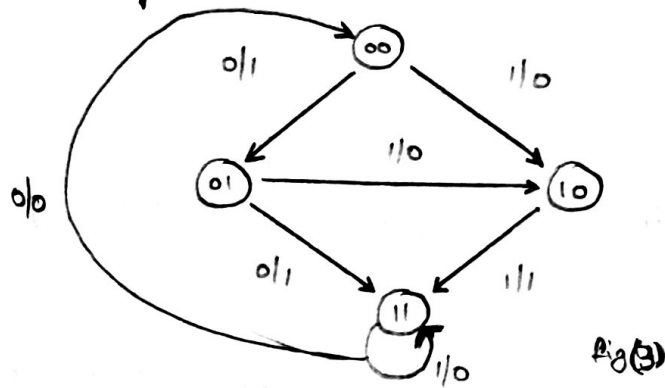
Present state	Next state		output	
	x=0	x=1	x=0	x=1
a	a	c	0	0
b	b	a	0	0
c	d	c	0	1
d	~~a~~ b	d	0	0

Table comprises of 3 parts. They are, present state, next state and the output. It basically provides relationship among these three parameters. The present state is the state of flipflop before the application of clock pulse, the next state indicates the state of flipflop after the clock pulse application and output state provides the values of output variables according to the present state. It may be noted that both next state and output have two sections corresponding to value of $x=0$ and $x=1$ respectively.

(iii) State Assignment:-

The performance of a sequential circuit is expressed in terms of four parameters, namely input, present state, the next state and the output. A specific flip-flop inputs are required to generate the next state at a particular present state and input. These flip-flop inputs are obtained from flip-flop input functions. In order to determine the flip-flop input functions, the state in the state diagram of binary must be indicated with binary values rather

alphabets. this process is termed as state assignment. The assignment of binary digits to the states must be in a manner that minimum logic gates are used to implement flip-flop input functions. An example of state assignment is shown in fig(3).



we can assign the binary values by following the set of rules for state assignment. Basically there are two rules for state assignment. They are,

Rule 1:- the assignment of states with some 'NEXT STATES' must be in such a manner that, they can be grounded into logically adjacent cells in a K-map.

example:-

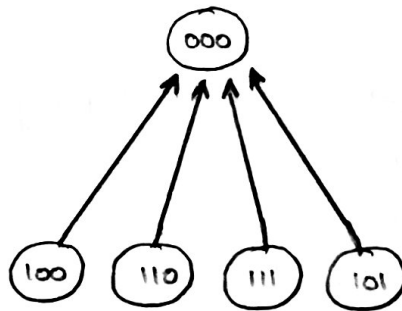


Fig (4).

In fig(4), there are four states whose next state is same i.e; (000). If we assign the binary values for these four states as 100, 110, 111, 101 then, they can be combined in K-map.

Rule 2:- The assignment of states, which are the 'NEXT STATES' of a single state should be in such a way that they can be grouped logically by using a K-map.

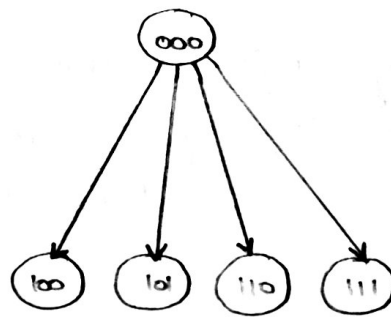
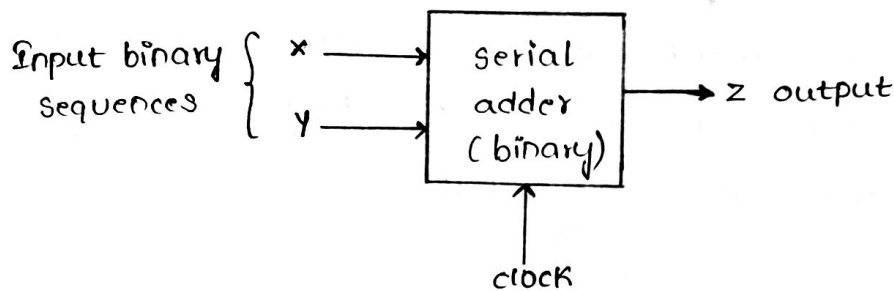


Fig (5).

4. Draw and explain the diagram of mealy-type FSM for serial adder.

fig(1) shows the model of serial adder that should be realized as a mealy network.



fig(1) structure of a binary serial adder

A serial binary consists of two inputs x, y and a single output z . The operand of each binary sequence are applied at x and y inputs respectively. The addition is performed bit by starting with a pair of least significant Bits (LSB's). For the synchronization purpose, the inputs at x and y are given before the triggering of clock pulse. An output z contains the sum result of two binary sequences in the form of time sequences with LSB Bit on the first position.

There exists four inputs combinations of x and y that are 00, 01, 10 and 11. The current output and next state is found only when the sequential circuit stores the previous output state. According to binary addition excluding the least significant pair of bits sum, the sum result of pair of bits of different order and position requires previously generated

carry bit. Let the value of carry bit be either 0 or 1, which should be stored for performing addition on other input bit pair. Hence, the present output is determined by current pair of input bits with previous carry bit and the next state is determined by considering A as the initial state.

consider a state diagram with each state as a node as shown in fig(2), four combinations are possible with two inputs a node should consider all combinations. Node A is sum of a pair of binary bits with no carry which results in a carry of either 0 or 1. For example, the sum of combinations 00, 01, and 10 yield a carry bit of 0 that remain in the initial state i.e., A whereas the addition of 11 combination yields a carry bit 1, hence the state is changed to B for the carry bit to be stored.

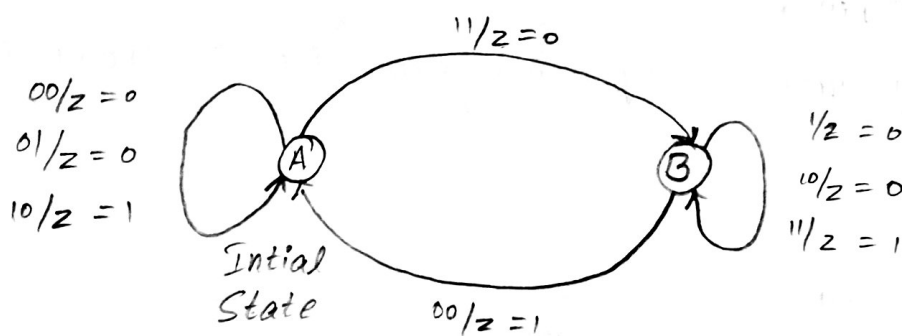


Fig:- State diagram of Serial Binary Adder

At node B, the sum bit is the addition of input combination of bits along with the previous carry. For combination 00, the sum bit is $0+1=1$, with carry bit = 0 which indicates state A, hence state is changed from B to A. for all other combinations i.e., 01, 10, 11, carry is generated and there is no change in the state. The state table of a mealy serial binary adder is shown in table.

Present state	Next state				Output (Z)			
	Input (xy)				Inputs (xy)			
	00	01	10	11	00	01	10	11
	A	A	A	B	0	1	1	0
A (Initial state)	A	A	A	B	0	1	1	0
B	A	B	B	B	1	0	0	1

If $A=0$ and $B=1$, the state assigned table is shown in table(2)

present state a	Next state (y)				output (z)			
	xy = 00 01 10 11				xy = 00 01 10 11			
0	0	0	0	1	0	1	1	0
1	0	1	1	1	1	0	0	1

Table(2): state - Assigned table.

on solving the state assigned table, the next-state and output equations are

$$y = xy + xa + ya$$

$$z = x \oplus y \oplus a$$

The corresponding circuit using above expressions is shown in fig(3)

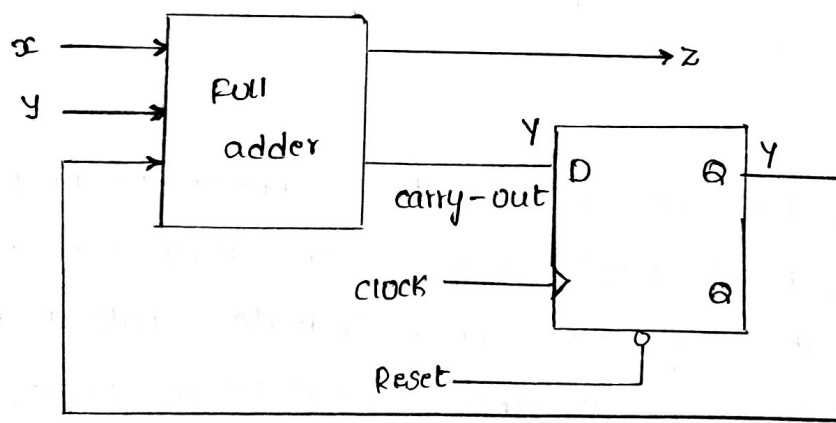


figure (3): circuit for the serial Adder Mealy FSM

(2) How to derive next state and output expressions and what is the flipflop chosen for Solving? 4

A. The next state and output Expressions are derived from the given data using karnaugh maps. The valuation of the circuit should be done based on the states it constrained. Additionally, valuation can be done by considering don't care condition where required otherwise, the analysis of resultant expressions will become more Complex.

The most commonly used flip-flop is D-flipflop, as the output is Same as its input.

For instance, consider a state assigned table of a Sequential circuit given below.

Present state $y_2 y_1$	Next state		output z
	$w=0$	$w=1$	
	$y_2 y_1$	$y_2 y_1$	
00	00	01	0
01	00	10	0
11	00	10	1
10	xx	xx	x

Table

Solving for y_1, y_2 and z using k-maps

for y_1

$w \backslash y_2 y_1$	00	01	11	10
0			x	
1	1		x	

with don't cares: $y_1 = w\bar{y}_1\bar{y}_2$

without don't cares: $y_1 = w\bar{y}_1\bar{y}_2$

for y_2

$w \backslash y_2 y_1$	00	01	11	10
0			x	
1		1	x	1

with don't cares: $y_2 = w y_1 + w y_2$
 $= w(y_1 + y_2)$

without don't cares: $y_2 = w y_1 y_2 + w y_1 \bar{y}_2$

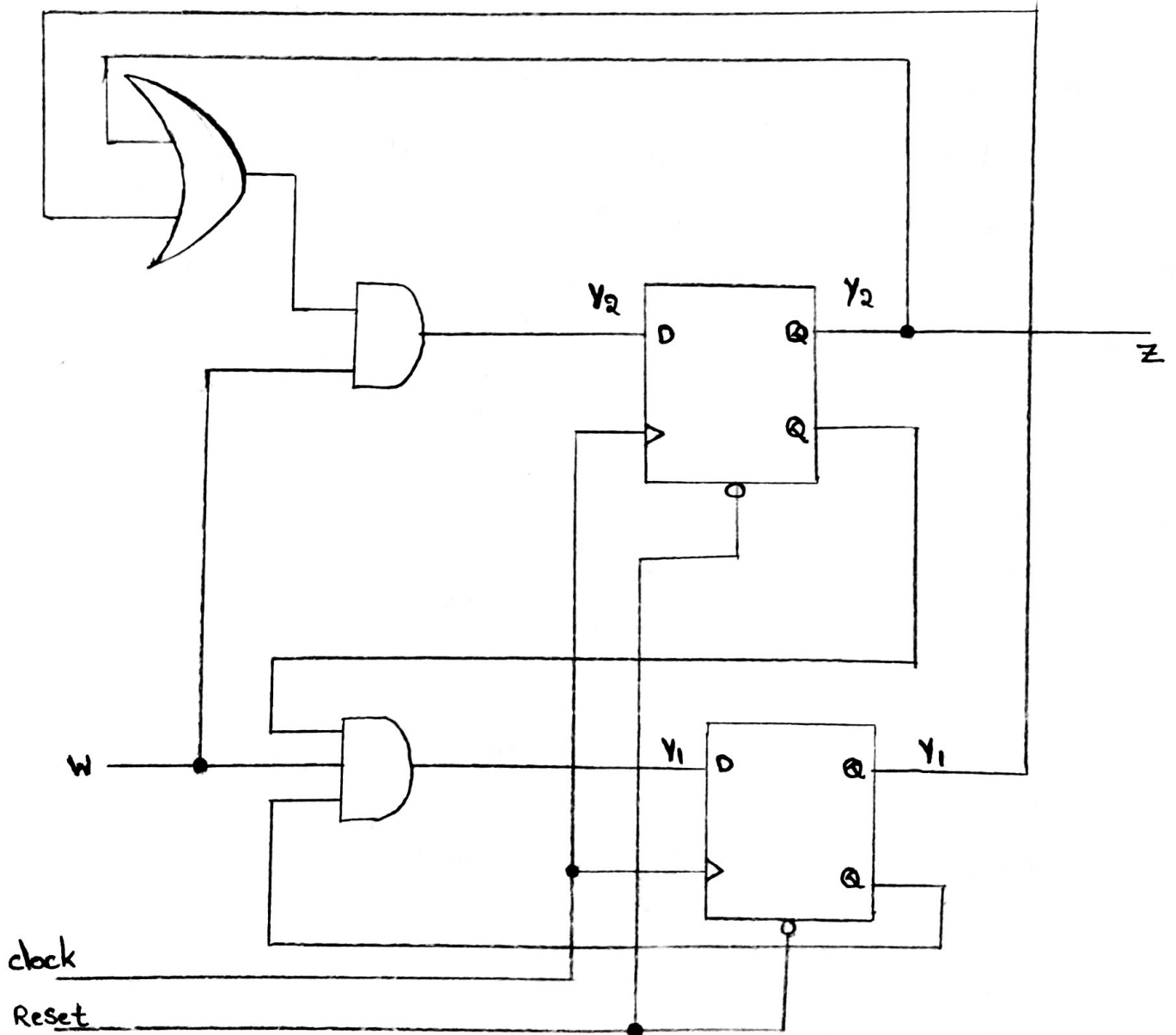
For z

y_2/y_1	0	1
0		
1	1	x

with don't cares : $z = y_2$

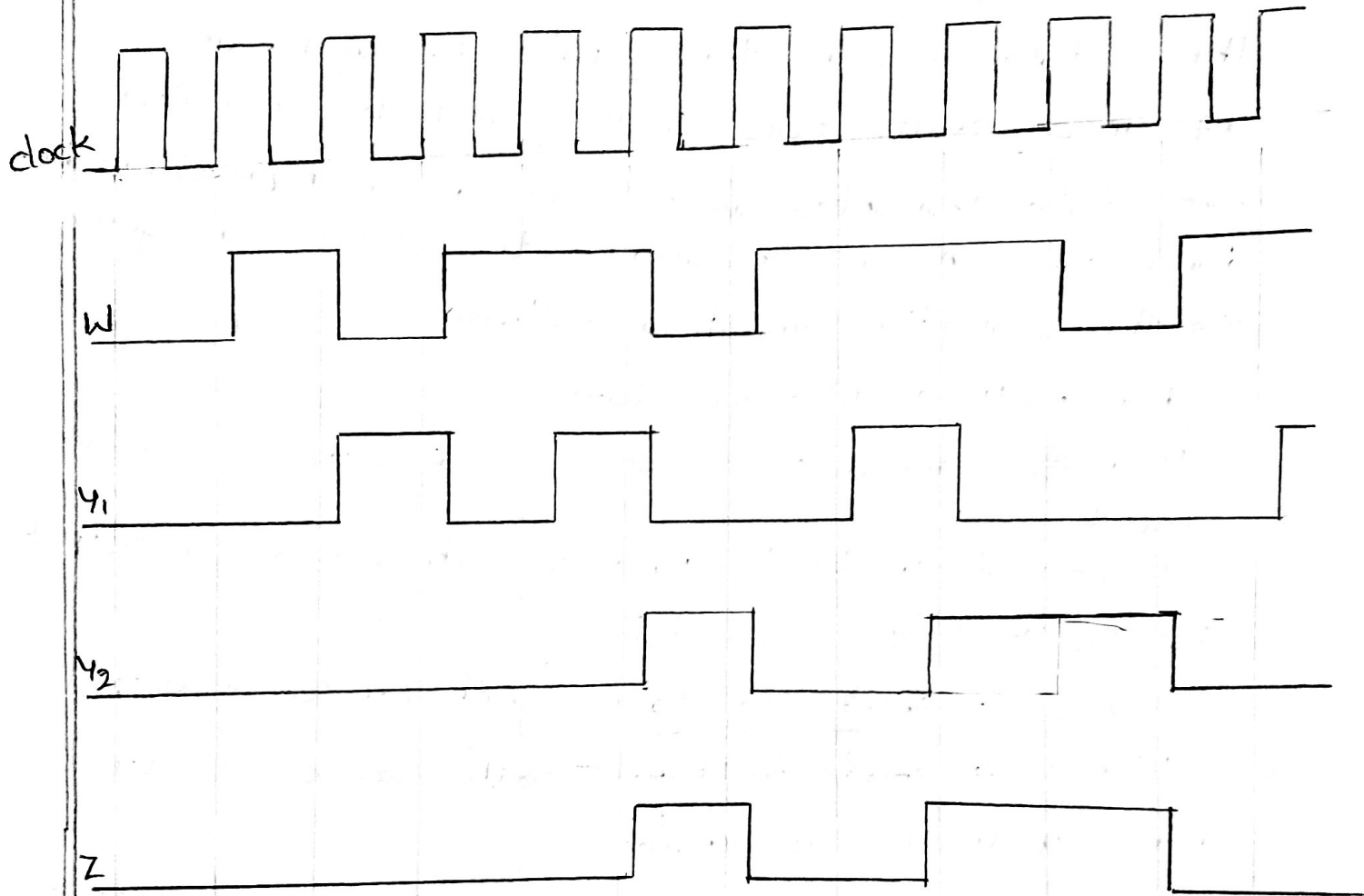
without don't cares : $z = y_1 y_2$

The circuit representation of above 'with don't cares' expressions is shown in fig(1)



Figure

Since, the positive edge triggered flipflops are used, the change in signals occur after positive edge of the clock. The delay in the signal depends on the propagation delays of the flipflops.



fig(2) : Timing Diagram

3. Explain one-hot Encoding technique with an Example?

A. one-hot Encoding is a state assignment technique, in which all the variables in each state are 0 except one variable, which holds the value 1, known as 'hot'. In this, number of flipflops to be assigned is

Equal to number of states

Example

consider the state diagram of a Sequential circuit as shown in fig.

From figure it is observed that, the circuit has three states. So, the three states A, B and C are assigned with 001, 010, 100, and the remaining five state Variables are not considered represented by don't cares. As a consequence, the one hot assignment for the Sequential circuit as follows,

For A (y_3, y_2, y_1) = 001 state;

If $x=0$, it remains in same state with zero as an output

If $x=1$, it shifts to state B with zero as an output

For B (y_3, y_2, y_1) = 010 state;

If $x=0$, it shifts to state A with Zero as an output

If $x=1$, it shifts to state C with Zero as an output

For C (y_3, y_2, y_1) = 100 state;

If $x=0$, it shifts to state A with Zero as an output

If $x=1$, it remains in same state with one as an output.

The one hot assignment for the Sequential circuit is shown in table.

6

Present state			Input	Next state			output
y_3	y_2	y_1	x	y_3	y_2	y_1	z
0	0	0	0	x	x	x	x
0	0	0	1	x	x	x	x
0	0	1	0	0	0	1	0
0	0	1	1	0	1	0	0
0	1	0	0	0	0	1	0
0	1	1	0	x	x	x	x
0	1	1	1	x	x	x	x
1	0	0	0	0	0	1	1
1	0	0	1	1	0	0	1
1	0	0	0	x	x	x	x
1	0	0	1	x	x	x	x
1	1	0	0	x	x	x	x
1	1	0	1	x	x	x	x
1	1	1	0	x	x	x	x
1	1	1	1	x	x	x	x

Table.

From above Table, the next state expressions can be obtained by using k-maps.

for y_3

		$y_1 x$			
$y_3 y_2$		00	01	11	10
00		x	x	0	0
01		0	1	x	x
11		x	x	x	x
10		0	1	x	x

↓
 $y_1 x$

$$\therefore y_3 = \overline{y_1} x$$

for y_2

$y_3 y_2$ \ $y_1 x$	00	01	11	10
00	x	x	1	0
01	0	0	x	x
11	x	x	x	x
10	0	0	x	x

$y_1 x$

$$\therefore y_2 = y_1 x$$

for y_1

$y_3 y_2$ \ $y_1 x$	00	01	11	10
00	x	x	0	1
01	1	0	x	x
11	x	x	x	x
10	1	0	x	x

$\bar{y}_1 x$

$$\therefore y_1 = \bar{x}$$

for z

$y_3 y_2$ \ $y_1 x$	00	01	11	10
00	x	x	0	0
01	0	0	x	x
11	x	x	x	x
10	0	1	x	x

y_3

$$\therefore z = y_3$$

The next outputs of the Sequential circuit are,

$$y_3 = \bar{y}_1 x$$

$$y_2 = y_1 x$$

$$y_1 = \bar{x}$$

Therefore to realize a Specified Sequential circuit three flipflops and two 2-input AND gates are required.