Sequence Detector FSM

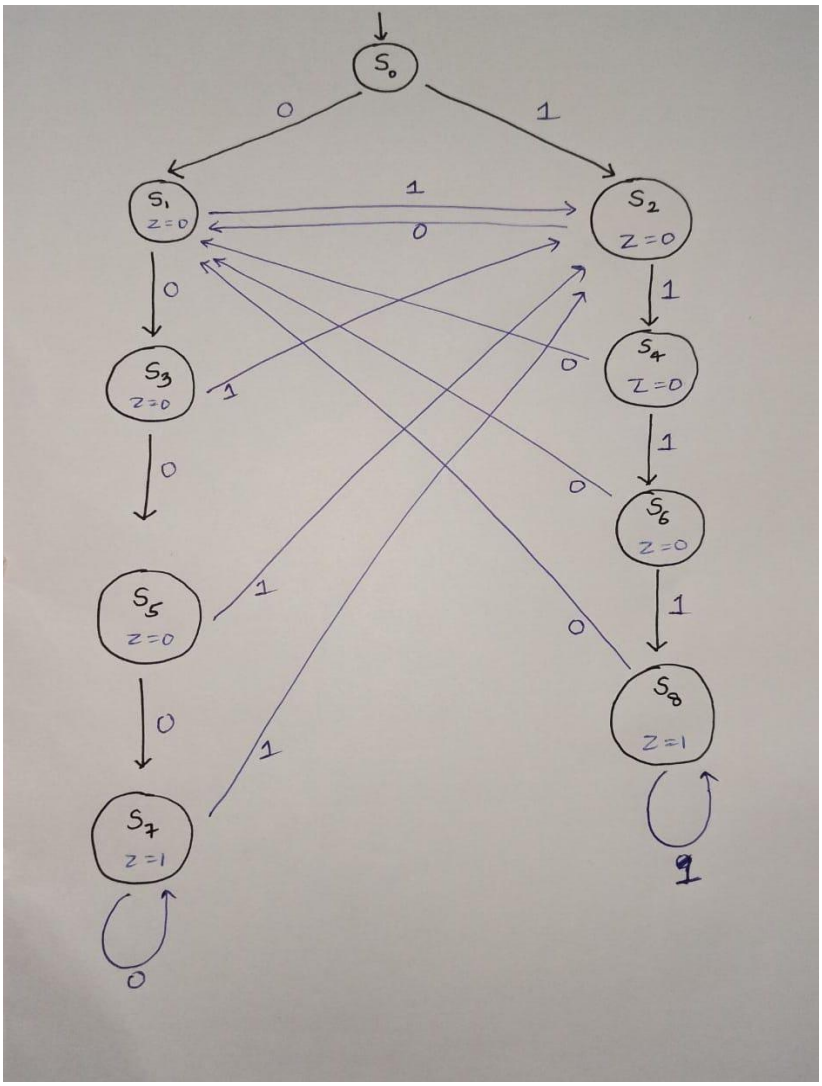
mailto:b20197@students.iitmandi.ac.in

Name- Gaurav Kumar

Mob- 8529143452

Question 1. Consider a finite state machine (FSM) that recognizes two specific sequences of applied input symbols, namely four consecutive 1s or four consecutive 0s. There is an input w and an output z whenever w = 1 or w = 0 for four consecutive clock pulses the value of z has to be 1; otherwise, z = 0. Overlapping sequences are allowed, so that if w = 1 for five consecutive clock pulses the output z will be equal to 1 after the fourth and fifth pulses. Figure 1 illustrates the required relationship between w and z.

State Diagram:



VHDL CODE:

D:/Sem 3/IC 210p/Lab/Lab_3/FSM.vhd (/fsm) - Default

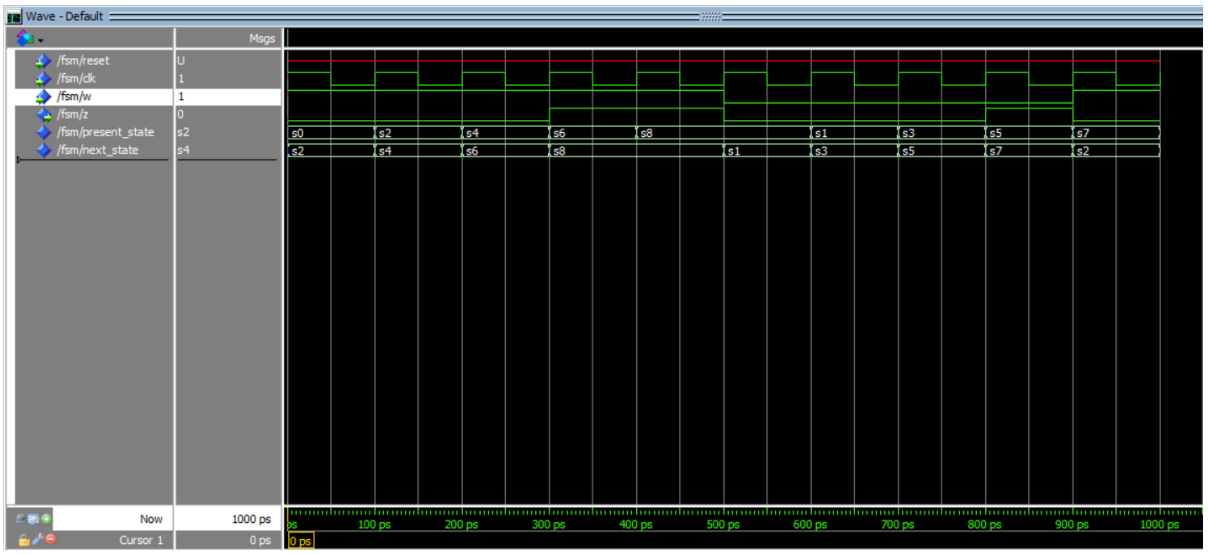
```
Ln#
 1  library ieee;
 2  use ieee.std_logic_1164.all;
 3  entity FSM is
 4  port(
 5    reset, clk, w : in std_logic;
 6    z : out std_logic );
 7  end FSM ;
 8
 9  architecture behavioral of FSM is
10
11  type state_available is (s0, s1, s2,s3,s4,s5,s6,s7,s8);
12  signal present_state, next_state : state_available := s0; -- -- Declaration of two signals of this type
13  begin
14  comb_proc : process (present_state, w)
15  begin
16  case present_state is
17  when s0 =>
18  if w = '0' then
19  next_state <= s1;
20  z <= '0';
21  else
22  next_state <= s2;
23  z <= '0';
24  end if;
25  when s1 =>
26  if w = '0' then
27  next_state <= s3;
28  z <= '0';
29  else
30  next_state <= s2;
31  z <= '0';
32  end if;
```

```
33    when s3 =>
34    if w = '0' then
35    next_state <= s5;
36    z <= '0';
37    else
38    next_state <= s2;
39    z <= '0';
40    end if;
41    when s5 =>
42    if w = '0' then
43    next_state <= s7;
44    z <= '1';
45    else
46    next_state <= s2;
47    z <= '0';
48    end if;
49    when s7 =>
50    if w = '0' then
51    next_state <= s7;
52    z <= '1';
53    else
54    next_state <= s2;
55    z <= '0';
56    end if;
```

```
57      when s2 =>
58      if w = '1' then
59      next_state <= s4;
60      z <= '0';
61      else
62      next_state <= s1;
63      z <= '0';
64      end if;
65      when s4 =>
66      if w = '1' then
67      next_state <= s6;
68      z <= '0';
69      else
70      next_state <= s1;
71      z <= '0';
72      end if;
73      when s6 =>
74      if w = '1' then
75      next_state <= s8;
76      z <= '1';
77      else
78      next_state <= s1;
79      z <= '0';
80      end if;
81      when s8 =>
82      if w = '1' then
83      next_state <= s8;
84      z <= '1';
85      else
86      next_state <= s1;
87      z <= '0';
88      end if;
89  end case;
90  end process;
91
92  clk_proc : process
93  begin
94  wait until (clk'event and clk = '1'); -- wait for rising edge of clk
95  if reset = '1' then
96  present_state <= s0;
97  else
98  present_state <= next_state;
99  end if;
100 end process;
101 end behavioral;
102
```

Output(Simulation):

In this, I have given input (w) as 1111100001 so, my output(z) is like 0 for first three time-period, and then 1 for two time-period as w is high for consecutive 4 or more time period. And then output goes to 0 again as input goes to 0 and then as input goes 0 for four consecutive time period, output again goes high. Then again input is again given 1, and that cause output goes to low as sequence changes.


Wave - Default
Msgs
/fsm/reset
/fsm/clk
/fsm/w
/fsm/z
/fsm/present_state
/fsm/next_state
U
1
1
0
s2
s4
s0 s2 s4 s6 s8 s1 s3 s5 s7
s2 s4 s6 s8 s1 s3 s5 s7 s2
Now
1000 ps
Cursor 1
0 ps
100 ps 200 ps 300 ps 400 ps 500 ps 600 ps 700 ps 800 ps 900 ps 1000 ps
0 ps

THE END