```
1        --  E:\PROJECTS\...\SERIAL_PARITY_GENERATOR_STATE_DIAGRAM.vhd
2        --  VHDL code created by Xilinx's StateCAD 9.2i
3        --  Fri Jan 18 19:11:57 2008
4
5        --  This VHDL code (for use with Xilinx XST) was generated using:
6        --  binary encoded state assignment with structured code format.
7        --  Minimization is enabled,  implied else is enabled,
8        --  and outputs are speed optimized.
9
10       LIBRARY ieee;
11       USE ieee.std_logic_1164.all;
12
13       ENTITY SERIAL_PARITY_GENERATOR_STATE_DIAGRAM IS
14          PORT (CLK,RESET,W: IN std_logic;
15             Q : OUT std_logic);
16       END;
17
18       ARCHITECTURE BEHAVIOR OF SERIAL_PARITY_GENERATOR_STATE_DIAGRAM IS
19          SIGNAL sreg : std_logic_vector (1 DOWNTO 0);
20          SIGNAL next_sreg : std_logic_vector (1 DOWNTO 0);
21          CONSTANT STATE0 : std_logic_vector (1 DOWNTO 0) :="00";
22          CONSTANT STATE1 : std_logic_vector (1 DOWNTO 0) :="01";
23          CONSTANT STATE2 : std_logic_vector (1 DOWNTO 0) :="10";
24          CONSTANT STATE3 : std_logic_vector (1 DOWNTO 0) :="11";
25
26       BEGIN
27          PROCESS (CLK, RESET, next_sreg)
28          BEGIN
29             IF ( RESET='1' ) THEN
30                sreg <= STATE0;
31             ELSIF CLK='1' AND CLK'event THEN
32                sreg <= next_sreg;
33             END IF;
34          END PROCESS;
35
36          PROCESS (sreg,W)
37          BEGIN
38             Q <= '0';
39
40             next_sreg<=STATE0;
41
42             CASE sreg IS
43                WHEN STATE0 =>
44                   Q<='0';
45                   IF ( W='1' ) THEN
46                      next_sreg<=STATE1;
47                   END IF;
48                   IF ( W='0' ) THEN
49                      next_sreg<=STATE0;
50                   END IF;
51                WHEN STATE1 =>
52                   Q<='1';
53                   IF ( W='0' ) THEN
54                      next_sreg<=STATE2;
55                   END IF;
56                   IF ( W='1' ) THEN
57                      next_sreg<=STATE1;
58                   END IF;
59                WHEN STATE2 =>
60                   Q<='1';
61                   IF ( W='1' ) THEN
```

```
62                      next_sreg<=STATE3;
63                   END IF;
64                   IF ( W='0' ) THEN
65                      next_sreg<=STATE2;
66                   END IF;
67                WHEN STATE3 =>
68                   Q<='0';
69                   IF ( W='0' ) THEN
70                      next_sreg<=STATE0;
71                   END IF;
72                   IF ( W='1' ) THEN
73                      next_sreg<=STATE3;
74                   END IF;
75                WHEN OTHERS =>
76             END CASE;
77          END PROCESS;
78       END BEHAVIOR;
79
```