

```
1  --
2  -- Work by : Chaari Mahdi and Daboussi Mariem , Engineering Students
3  --
4
5  LIBRARY ieee;
6  USE ieee.std_logic_1164.ALL;
7  use ieee.numeric_std.all;
8
9
10 ENTITY test_bench_flicker IS
11 END test_bench_flicker;
12
13 ARCHITECTURE behavior OF test_bench_flicker IS
14
15     -- Component Declaration for the Unit Under Test (UUT)
16
17     COMPONENT traffic_lights
18     PORT(
19         clk : IN  std_logic;
20         cnf : IN  std_logic;
21         rst : IN  std_logic;
22         switch_dip_main_timing : in std_logic_vector (7 downto 0); --
23         switch_dip_turning_left : in std_logic_vector (5 downto 0);
24         switch_dip_orange_light : in std_logic_vector (3 downto 0);
25         switch_dip_security_time : in std_logic_vector (1 downto 0);
26         switch_dip_pedestrian_time : in std_logic_vector (5 downto 0);
27         switch_dip_security_code : in std_logic_vector (7 downto 0);
28         bouton_okay : in std_logic;
29         Bouton_appel_pieton : inout std_logic_vector (3 downto 0);
30         lights_pieton : out std_logic_vector (3 downto 0);
31         Up_light : OUT  std_logic_vector(2 downto 0);
32         Up_left_light : OUT  std_logic_vector(2 downto 0);
33         Up_right_light : OUT  std_logic;
34         Down_light : OUT  std_logic_vector(2 downto 0);
35         Down_left_light : OUT  std_logic_vector(2 downto 0);
36         Down_right_light : OUT  std_logic;
37         Left_light : OUT  std_logic_vector(2 downto 0);
38         Left_left_light : OUT  std_logic_vector(2 downto 0);
39         Left_right_light : OUT  std_logic;
40         Right_light : OUT  std_logic_vector(2 downto 0);
41         Right_left_light : OUT  std_logic_vector(2 downto 0);
42         Right_right_light : OUT  std_logic
43     );
44     END COMPONENT;
45
46
47     --Inputs
48     signal clk : std_logic := '0';
49     signal cnf : std_logic := '0';
50     signal rst : std_logic := '0';
51     signal switch_dip_main_timing : std_logic_vector (7 downto 0) := "00000000";
52     signal switch_dip_turning_left : std_logic_vector (5 downto 0) := "000000";
53     signal switch_dip_orange_light : std_logic_vector (3 downto 0) := "0000";
54     signal switch_dip_security_time : std_logic_vector (1 downto 0) := "00";
55     signal switch_dip_pedestrian_time : std_logic_vector (5 downto 0) := "000000";
56     signal switch_dip_security_code : std_logic_vector (7 downto 0) := "00000000";
57     signal bouton_okay : std_logic := '0';
```

```
58     signal Bouton_appel_pieton: std_logic_vector(3 downto 0) := "0000";
59
60
61     --Outputs
62     signal lights_pieton : std_logic_vector(3 downto 0);
63     signal Up_light : std_logic_vector(2 downto 0);
64     signal Up_left_light : std_logic_vector(2 downto 0);
65     signal Up_right_light : std_logic;
66     signal Down_light : std_logic_vector(2 downto 0);
67     signal Down_left_light : std_logic_vector(2 downto 0);
68     signal Down_right_light : std_logic;
69     signal Left_light : std_logic_vector(2 downto 0);
70     signal Left_left_light : std_logic_vector(2 downto 0);
71     signal Left_right_light : std_logic;
72     signal Right_light : std_logic_vector(2 downto 0);
73     signal Right_left_light : std_logic_vector(2 downto 0);
74     signal Right_right_light : std_logic;
75
76     -- Clock period definitions<
77     constant clk_period : time := 20 ns;
78
79 BEGIN
80
81     -- Instantiate the Unit Under Test (UUT)
82     uut: traffic_lights PORT MAP (
83         clk => clk,
84         cnf => cnf,
85         rst => rst,
86         switch_dip_main_timing => switch_dip_main_timing, --
87         switch_dip_turning_left => switch_dip_turning_left,
88         switch_dip_orange_light => switch_dip_orange_light,
89         switch_dip_security_time => switch_dip_security_time,
90         switch_dip_pedestrian_time => switch_dip_pedestrian_time,
91         switch_dip_security_code => switch_dip_security_code,
92         bouton_okay => bouton_okay,
93         Bouton_appel_pieton => Bouton_appel_pieton,
94
95
96         lights_pieton => lights_pieton,
97         Up_light => Up_light,
98         Up_left_light => Up_left_light,
99         Up_right_light => Up_right_light,
100         Down_light => Down_light,
101         Down_left_light => Down_left_light,
102         Down_right_light => Down_right_light,
103         Left_light => Left_light,
104         Left_left_light => Left_left_light,
105         Left_right_light => Left_right_light,
106         Right_light => Right_light,
107         Right_left_light => Right_left_light,
108         Right_right_light => Right_right_light
109     );
110
111     -- Clock process definitions
112     clk_process :process
113     begin
114         clk <= '0';
```

```
115     wait for clk_period/2;
116     clk <= '1';
117     wait for clk_period/2;
118 end process;
119
120
121 -- Stimulus process
122 stim_proc: process
123 begin
124
125 --Simulation du fonctionnement "NIGHT" puis "DAY" sans appel piéton:
126 --    rst <= '1';
127 --    wait for 100 ns;
128 --    rst<= '0';
129
130 ----Simulation d'un appel piéton:
131 ----    rst <= '1';
132 ----    wait for 100 ns;
133 ----    rst<= '0';
134 ----    wait for 200 ns;
135 ----    bouton_appel_pieton<="1101";
136 ----    wait for 30 ns;
137 ----    bouton_appel_pieton <="0000";
138
139
140 -----Simulation de configuration du système:
141     rst <='1';
142     wait for 20 ns;
143     rst <='0';
144     wait for 200 ns;
145     rst <='1';
146     cnf<= '1';
147     wait for 25 ns;
148     switch_dip_security_code <= "10010110" ;
149     bouton_okay <='1';
150     wait for 5 ns;
151     bouton_okay <='0';
152     wait for 20 ns;
153     switch_dip_main_timing <= "00011001"; --25
154     bouton_okay <='1';
155     wait for 5 ns;
156     bouton_okay <='0';
157     wait for 20 ns;
158     switch_dip_turning_left <= "001111"; --15
159     bouton_okay <= '1';
160     wait for 5 ns;
161     bouton_okay <='0';
162     wait for 20 ns;
163     switch_dip_orange_light <= "1001"; --9
164     bouton_okay <='1';
165     wait for 5 ns;
166     bouton_okay <='0';
167     wait for 20 ns;
168     switch_dip_security_time <= "01"; --1
169     bouton_okay <='1';
170     wait for 5ns;
171     bouton_okay <='0';
```

```
172     wait for 20 ns;
173     switch_dip_pedestrian_time <= "001111"; --15
174     bouton_okay <='1';
175     wait for 5 ns;
176     bouton_okay <='0';
177     wait for 20 ns;
178     switch_dip_pedestrian_time <= "000000";
179     switch_dip_security_code <= "00000000";
180     switch_dip_main_timing <= "00000000";
181     switch_dip_turning_left <= "000000";
182     switch_dip_orange_light <= "0000";
183     switch_dip_security_time <= "00";
184     cnf<= '0';
185     wait for 20 ns;
186     rst <= '0';
187
188     wait for clk_period*10;
189
190
191     wait;
192 end process;
193
194 END;
195
```