

```
1  -----
2  -- Company:
3  -- Engineer:
4  --
5  -- Create Date:   15:58:01 05/20/2021
6  -- Design Name:
7  -- Module Name:   C:/Xilinx/14.7/VHDL_Design_Project/TB_SyncButton.vhd
8  -- Project Name:  VHDL_Design_Project
9  -- Target Device:
10 -- Tool versions:
11 -- Description:
12 --
13 -- VHDL Test Bench Created by ISE for module: SyncButton
14 --
15 -- Dependencies:
16 --
17 -- Revision:
18 -- Revision 0.01 - File Created
19 -- Additional Comments:
20 --
21 -- Notes:
22 -- This testbench has been automatically generated using types std_logic and
23 -- std_logic_vector for the ports of the unit under test.  Xilinx recommends
24 -- that these types always be used for the top-level I/O of a design in order
25 -- to guarantee that the testbench will bind correctly to the post-implementation
26 -- simulation model.
27 -----
28 LIBRARY ieee;
29 USE ieee.std_logic_1164.ALL;
30
31 -- Uncomment the following library declaration if using
32 -- arithmetic functions with Signed or Unsigned values
33 --USE ieee.numeric_std.ALL;
34
35 ENTITY TB_SyncButton IS
36 END TB_SyncButton;
37
38 ARCHITECTURE behavior OF TB_SyncButton IS
39
40     -- Component Declaration for the Unit Under Test (UUT)
41
42     COMPONENT SyncButton
43     PORT(
44         reset : IN  std_logic;
45         clock : IN  std_logic;
46         TrainButton : IN  std_logic; -- Train Button Input --
47         CarButton : IN  std_logic; -- Car Button Input --
48         PedButton : IN  std_logic; -- Pedestrian Button Input --
49         Counter : IN  std_logic_vector(4 downto 0); -- Timer --
50         MotorEnabler : OUT  STD_LOGIC; -- Enable And Disable Stepper Motor --
51         TrainAction : OUT  std_logic; -- Train Button Output --
52         CarWaitOutput : OUT  std_logic; -- Car Output When Train Is And Is Not
53         Passing Through --
54         PedWaitOutput : OUT  std_logic -- Pedestrian Output When Train Is And Is Not
55         Passing Through --
56     );
57     END COMPONENT;
```

```
56
57
58  --Inputs
59  signal reset : std_logic := '0';
60  signal clock : std_logic := '0';
61  signal TrainButton : std_logic := '0';
62  signal CarButton : std_logic := '0';
63  signal PedButton : std_logic := '0';
64  signal Counter : std_logic_vector(4 downto 0) := (others => '0');
65
66  --Outputs
67  signal TrainAction : std_logic;
68  signal CarWaitOutput : std_logic;
69  signal PedWaitOutput : std_logic;
70  signal MotorEnabler : std_logic;
71
72  -- Clock period definitions
73  constant clock_period : time := 10 ns;
74
75  BEGIN
76
77  -- Instantiate the Unit Under Test (UUT)
78  uut: SyncButton PORT MAP (
79      reset => reset,
80      clock => clock,
81      TrainButton => TrainButton,
82      CarButton => CarButton,
83      PedButton => PedButton,
84      Counter => Counter,
85      TrainAction => TrainAction,
86      CarWaitOutput => CarWaitOutput,
87      PedWaitOutput => PedWaitOutput,
88      MotorEnabler => MotorEnabler
89  );
90
91  -- Clock process definitions
92  clock_process : process
93  begin
94      clock <= '0';
95      wait for clock_period/2;
96      clock <= '1';
97      wait for clock_period/2;
98  end process;
99
100
101  -- Stimulus process
102  stim_proc: process
103  begin
104
105      reset <= '1';
106
107      -- hold reset state for 100 ns.
108      wait for 100 ns;
109      reset <= '0';
110
111      -- insert stimulus here --
112      -- Only Train Button Pressed --
```

```
113     wait for clock_period*10;
114     TrainButton <= '1';
115     CarButton <= '0';
116     PedButton <= '0';
117     Counter <= "00000";
118
119     wait for clock_period*10;
120     TrainButton <= '0';
121     CarButton <= '0';
122     PedButton <= '0';
123     Counter <= "11101";
124
125     wait for clock_period*10;
126     TrainButton <= '0';
127     CarButton <= '0';
128     PedButton <= '0';
129     Counter <= "00101";
130
131     wait for clock_period*10;
132     TrainButton <= '0';
133     CarButton <= '0';
134     PedButton <= '0';
135     Counter <= "11110";
136
137     wait for clock_period*10;
138     TrainButton <= '0';
139     CarButton <= '0';
140     PedButton <= '0';
141     Counter <= "11110";
142
143     -- Car And Pedestrian Button Pressed When Train Passing --
144     wait for clock_period*10;
145     TrainButton <= '1';
146     CarButton <= '0';
147     PedButton <= '0';
148     Counter <= "00000";
149
150     wait for clock_period*10;
151     TrainButton <= '0';
152     CarButton <= '1';
153     PedButton <= '0';
154     Counter <= "11101";
155
156     wait for clock_period*10;
157     TrainButton <= '0';
158     CarButton <= '0';
159     PedButton <= '0';
160     Counter <= "11001";
161
162     wait for clock_period*10;
163     TrainButton <= '0';
164     CarButton <= '0';
165     PedButton <= '1';
166     Counter <= "10101";
167
168     wait for clock_period*10;
169     TrainButton <= '0';
```

```
170     CarButton <= '0';
171     PedButton <= '0';
172     Counter <= "00101";
173
174     wait for clock_period*10;
175     TrainButton <= '0';
176     CarButton <= '0';
177     PedButton <= '0';
178     Counter <= "11110";
179
180     -- Car Button Pressed After Train Passed Through --
181     wait for clock_period*10;
182     TrainButton <= '0';
183     CarButton <= '1';
184     PedButton <= '0';
185     Counter <= "11000";
186
187     wait for clock_period*10;
188     TrainButton <= '0';
189     CarButton <= '0';
190     PedButton <= '0';
191     Counter <= "10000";
192
193     wait for clock_period*10;
194     TrainButton <= '0';
195     CarButton <= '0';
196     PedButton <= '0';
197     Counter <= "00101";
198
199     wait for clock_period*10;
200     TrainButton <= '0';
201     CarButton <= '0';
202     PedButton <= '0';
203     Counter <= "11000";
204
205     -- Pedestrian Button Pressed After Train Passed --
206     wait for clock_period*10;
207     TrainButton <= '1';
208     CarButton <= '0';
209     PedButton <= '0';
210     Counter <= "11110";
211
212     wait for clock_period*10;
213     TrainButton <= '0';
214     CarButton <= '0';
215     PedButton <= '1';
216     Counter <= "11100";
217
218     wait for clock_period*10;
219     TrainButton <= '0';
220     CarButton <= '0';
221     PedButton <= '0';
222     Counter <= "00100";
223
224     wait for clock_period*10;
225     TrainButton <= '0';
226     CarButton <= '0';
```

```
227     PedButton <= '0';
228     Counter <= "00101";
229
230     wait for clock_period*10;
231     TrainButton <= '0';
232     CarButton <= '0';
233     PedButton <= '0';
234     Counter <= "11110";
235
236     -- Pedestrian Button Pressed When Gate Is Open --
237     wait for clock_period*10;
238     TrainButton <= '0';
239     CarButton <= '0';
240     PedButton <= '1';
241     Counter <= "00000";
242
243     wait for clock_period*10;
244     TrainButton <= '0';
245     CarButton <= '0';
246     PedButton <= '0';
247     Counter <= "11101";
248
249     wait for clock_period*10;
250     TrainButton <= '0';
251     CarButton <= '0';
252     PedButton <= '0';
253     Counter <= "00101";
254
255     wait for clock_period*10;
256     TrainButton <= '0';
257     CarButton <= '0';
258     PedButton <= '0';
259     Counter <= "10100";
260
261     -- Train Button Pressed When Pedestrian Is Passing --
262     wait for clock_period*10;
263     TrainButton <= '0';
264     CarButton <= '0';
265     PedButton <= '1';
266     Counter <= "00000";
267
268     wait for clock_period*10;
269     TrainButton <= '0';
270     CarButton <= '0';
271     PedButton <= '0';
272     Counter <= "00100";
273
274     wait for clock_period*10;
275     TrainButton <= '1';
276     CarButton <= '0';
277     PedButton <= '0';
278     Counter <= "00100";
279     wait;
280 end process;
281
282 END;
```