

Assignment 1 Extended

By: Abdelrahman Maher Hassan

Question 1

Codes:

```
1  module dff(clk, rst, d, q, en);
2      parameter USE_EN = 1;
3      input  clk, rst, d, en;
4      output reg q;
5
6      always @(posedge clk) begin
7          if (rst)
8              q <= 0;
9          else
10             if(USE_EN)
11                 if (en)
12                     q <= d;
13                 else
14                     q <= q;
15             else
16                 q <= d;
17         end
18     endmodule
19
20
```

Test Bench 1:

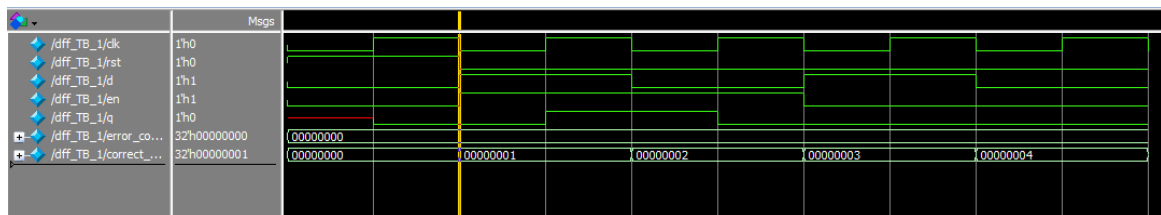
```
1  module dff_TB_1();
2
3      parameter USE_EN = 1;
4
5      reg clk, rst, d, en;
6      wire q;
7
8      dff #(.USE_EN(USE_EN)) D_FlipFlop ( .clk(clk),
9                                           .rst(rst),
10                                          .d(d),
11                                          .q(q),
12                                          .en(en) );
13
14      integer error_count, correct_count;
15
16      initial begin
17          clk = 0;
18          forever
19              #1 clk = ~clk;
20      end
21
22      initial begin
23          error_count = 0;
24          correct_count = 0;
25          en = 0;
26          d = 0;
27          check_reset();
28
29          en = 1;
30          d = 1;
31          check_result(d);
32
33          d = 0;
34          check_result(d);
35
36          en = 0;
37          d = 1;
38          check_result(0);
39
40          d = 0;
41          check_result(0);
42      end
43  endmodule
```

```

39
40     d = 0;
41     check_result(0);
42
43     $display("%t: The test is ended with %d correct tests and %d error tests.", $time, correct_count, error_count);
44
45     $stop;
46 end
47
48 task check_reset ();
49     rst = 1;
50     @(negedge clk);
51
52     if (q != 0)
53     begin
54         error_count++;
55         $display("%t: Error in the reset", $time);
56     end
57
58     else
59         correct_count++;
60     rst = 0;
61 endtask
62
63 task check_result (integer expected_Output);
64     @(negedge clk);
65     if (expected_Output != q)
66     begin
67         error_count++;
68         $display("%t: Error in the output = %d, and the expected is = %d", $time, q, expected_Output);
69     end
70
71     else
72         correct_count++;
73 endtask
74 endmodule

```

Simulation 1:



Test Bench 2:

```

1 module dff_TB_2();
2
3     parameter USE_EN = 0;
4
5     reg clk, rst, d, en;
6     wire q;
7
8     dff #(.USE_EN(USE_EN)) D_FlipFlop ( .clk(clk),
9                                           .rst(rst),
10                                          .d(d),
11                                          .q(q),
12                                          .en(en) );
13
14     integer error_count, correct_count;
15
16     initial begin
17         clk = 0;
18         forever
19             #1 clk = ~clk;
20     end
21
22     initial begin
23         error_count = 0;
24         correct_count = 0;
25         en = 0;
26         d = 0;
27         check_reset();
28
29         en = 1;
30         d = 1;
31         check_result(d);
32
33         d = 0;
34         check_result(d);
35
36         en = 0;
37         check_result(d);
38
39         d = 1;
40         check_result(d);
41

```

```

39         d = 1;
40         check_result(d);
41
42         $display("%t: The test is ended with %d correct tests and %d error tests.", $time, correct_count, error_count);
43
44         $stop;
45     end
46
47     task check_reset ();
48         rst = 1;
49         @(negedge clk);
50
51         if (q != 0)
52             begin
53                 error_count++;
54                 $display("%t: Error in the reset", $time);
55             end
56         else
57             correct_count++;
58         rst = 0;
59     endtask
60
61
62     task check_result (integer expected_Output);
63         @(negedge clk);
64         if (expected_Output != q)
65             begin
66                 error_count++;
67                 $display("%t: Error in the output = %d, and the expected is = %d", $time, q, expected_Output);
68             end
69         else
70             correct_count++;
71     endtask
72 endmodule
73

```

Simulation 2:

	Msgs										
/dff_TB_2/dk	No Data-										
/dff_TB_2/rst	No Data-										
/dff_TB_2/d	No Data-										
/dff_TB_2/en	No Data-										
/dff_TB_2/q	No Data-										
/dff_TB_2/error_co...	No Data-	00000000									
/dff_TB_2/correct_...	No Data-	00000000		00000001		00000002		00000003		00000004	