```verilog
1   /*------------------------------------------------------------------------
2   Name:    Lamin Jammeh
3   CLass:   CE6325 Fall_2024
4   Project: 1
5   Project Description: Teestbench for Project1
6   **creeate a clock signal
7   **Initialize all the input signals
8   **apply some Sample Data_in as [6325] in 10 time unit intervals
9   **apply reset to test the reset signal
10  **apply anothe set of sample Data_in as [2024] in 10 time unit intervals
11  **monitor the signals and end the test
12
13  -------------------------------------------------------------------------*/
14
15  module FIR_Filter_Project1_tb;
16
17  parameter order = 15;
18  parameter word_size_in = 8;
19  parameter word_size_out = 2 * word_size_in + 2;
20
21  //declare ports for the design
22  wire    [word_size_out -1:0]    Data_out;
23  reg     [word_size_in -1:0]     Data_in;
24  reg                             clock, reset;
25
26  //declare the unit under test UUT
27  FIR_Filter_Project1 #(order, word_size_in, word_size_out) UUT(.Data_out(Data_out),
28                                                                 .Data_in(Data_in),
29                                                                 .clock(clock),
30                                                                 .reset(reset)
31                                                                 );
32
33  // Instantiate the clock signal
34  initial
35     begin
36        clock = 0;
37        forever #5 clock = ~clock;
38     end
39
40  //Instantiate the diiferent test scenarios to validate the design
41  //*******Scenario 1 initialize all input signals
42  initial
43     begin
44        reset = 1;
45        Data_in =0;
46
47        #10    reset = 0;   //wait for 10 timing units for the inital signals to go through
48
49  //Scenario 2 apply some Data_in samples and observe the outputs use [6 3  2  5] @ 100 time
    unit intervals
50        #150  Data_in = 8'd6;          //make sure time is long enough for Data_in to mkae it
    to the last filter COefficient
51        #150  Data_in = 8'd3;
52        #150  Data_in = 8'd2;
53        #150  Data_in = 8'd5;
54
55  //Scenario 3 test the reset signal to validate the behavior
56        #10    reset = 1;         //Sample registers should be cleared
57        #10    reset = 0;         //Sample register will accept Data_in
58
59  //Scenario 4 apple more samples to make sure the design works after reset
60        #150  Data_in = 8'd2;
61        #150  Data_in = 8'd0;
62        #150  Data_in = 8'd2;
63        #150  Data_in = 8'd4;
64
65  //stop the test
66        #150  $stop;
67     end
```

```verilog
68
69     // Monitor output
70     initial
71        begin
72           $monitor("Time: %4t\t | Data_in: %0d\t | Data_out: %3d\t | reset: %b", $time,
     Data_in, Data_out, reset); //use %10 as padding for time and data_out
73        end
74
75     endmodule
76
77
78
```