

```
1  `timescale 1ns/1ps
2
3  module tb_Multi_8b();
4
5      // Entradas
6      reg clk;
7      reg rst;
8      reg start;
9      reg [7:0] A;
10     reg [7:0] B;
11
12     // Saídas
13     wire [15:0] Result;
14     wire fimOperacao;
15     wire Z;
16     wire OV;
17
18     // Instanciação do módulo a ser testado
19     Multi_8b uut (
20         .clk(clk),
21         .rst(rst),
22         .start(start),
23         .A(A),
24         .B(B),
25         .Result(Result),
26         .fimOperacao(fimOperacao),
27         .Z(Z),
28         .OV(OV)
29     );
30
31     // Clock: alterna a cada 5 ns
32     always #5 clk = ~clk;
33
34     initial begin
35         // Inicialização
36         clk = 0;
37         rst = 1;
38         start = 0;
39         A = 8'd0;
40         B = 8'd0;
41
42         // Espera pelo reset
43         #10;
44         rst = 0;
45
46         // Teste 1: 0 * 0
47         A = 8'd0;
48         B = 8'd0;
49         start = 1;
50         #10;
51         start = 0;
52         wait(fimOperacao);
53         #10;
54
55         // Teste 2: 5 * 10 = 50
56         A = 8'd5;
57         B = 8'd10;
58         start = 1;
59         #10;
60         start = 0;
61         wait(fimOperacao);
62         #10;
63
64         // Teste 3: 255 * 255 = 65025
65         A = 8'd255;
66         B = 8'd255;
67         start = 1;
68         #10;
69         start = 0;
70         wait(fimOperacao);
71         #10;
72
73         // Teste 4: 128 * 2 = 256
74         A = 8'd128;
75         B = 8'd2;
76         start = 1;
77         #10;
78         start = 0;
79         wait(fimOperacao);
80         #10;
81
```

```
82      // Teste 5: 1 * 200
83      A = 8'd1;
84      B = 8'd200;
85      start = 1;
86      #10;
87      start = 0;
88      wait(fimOperacao);
89      #10;
90
91      // Finaliza simulação
92      $display("Testes finalizados.");
93      $stop;
94  end
95
96  endmodule
97
```