

```
module AddSub8_testbench();
```

```
    reg input1;
    reg [7:0] input_bus1;
    reg [7:0] input_bus2;
    wire overflow;
    wire [7:0] output_bus;
```

```
    AddSub8    UUT    (.A(input_bus1), .B(input_bus2), .sub(input1), .S(output_bus),
.ovfl(overflow));
```

```
// below is the "stimuli," the values for the inputs
// be sure to select a range of inputs that will fully exercise your design
```

```
    initial
    begin
```

```
        //----- Current Time:  0ns
        input_bus1 = 8'b00000001;
        input_bus2 = 8'b00000001;
        input1 = 1'b0;
        //Simple test, 1+1.
        #100; //This advances time by 100 units (ns in this case)
        // ----- Current Time:  100ns
        input_bus1 = 8'b01111111;
        input_bus2 = 8'b00000001;
        input1 = 1'b0;
        //Overflow test.
        #100; // ----- Current Time:  200ns
        input_bus1 = 8'b00000100;
        input_bus2 = 8'b00000001;
        input1 = 1'b1;
        //Simple subtraction test, 4 - 1.
        #100; // ----- Current Time:  300ns
        input_bus1 = 8'b11111111;
        input_bus2 = 8'b00000100;
        input1 = 1'b0;
        //A harder test, -1 + 4.
        #100; // ----- Current Time:  400ns
        input_bus1 = 8'b01111111;
        input_bus2 = 8'b10000001;
        input1 = 1'b0;
        //127 + -127. Zero and positive + negative.
        #100; // ----- Current Time:  500ns
        input_bus1 = 8'b11111000;
        input_bus2 = 8'b11110000;
```

```

input1 = 1'b1;
//A complicated test. -8 - (-16).
#100; // ----- Current Time: 600ns
input_bus1 = 8'b10000001;
input_bus2 = 8'b11111110;
input1 = 1'b0;
//Negative overflow, -127 + -2.
#100; // ----- Current Time: 700ns
input_bus1 = 8'b11111111;
input_bus2 = 8'b11111111;
input1 = 1'b0;
//Negative simple test, -1 + -1.
#100; // ----- Current Time: 800ns
input_bus1 = 8'b10000000;
input_bus2 = 8'b10000000;
input1 = 1'b0;
//Lowest negative overflow, -128 + -128.
#100; // ----- Current Time: 900ns
input_bus1 = 8'b01111111;
input_bus2 = 8'b01111111;
input1 = 1'b0;
//Highest positive overflow, 127 + 127.
#100; // ----- Current Time: 1000ns
input_bus1 = 8'b10000001;
input_bus2 = 8'b10000001;
input1 = 1'b0;
//Second lowest negative overflow, -127 + -127.
#100; // ----- Current Time: 1100ns
input_bus1 = 8'b01111111;
input_bus2 = 8'b10000000;
input1 = 1'b0;
//Highest possible positive overflow with subtraction, 127 - (-128).
end
endmodule

```