Prob 1

1.1

**Verilog**

module cp\_4bit (

input [3:0] A ,

input [3:0] B ,

output A\_lt\_B,

output A\_eq\_B,

output A\_gt\_B

);

wire X3; // X3 = A[3] xnor B[3]

wire X2; // X2 = A[2] xnor B[2]

wire X1; // X1 = A[1] xnor B[1]

wire X0; // X0 = A[0] xnor B[0]

wire outl1; // outl1 = (~A[3]) & B[3]

wire outl2; // outl2 = X3 & (~A[2]) & B[2]

wire outl3; // outl3 = X3 & X2 & (~A[1]) & B[1]

wire outl4; // outl4 = X3 & X2 & X1 & (~A[0]) & B[0]

wire outg1; // outg1 = A[3] & (~B[3])

wire outg2; // outg2 = X3 & A[2] & (~B[2])

wire outg3; // outg3 = X3 & X2 & A[1] & (~B[1])

wire outg4; // outg4 = X3 & X2 & X1 & A[0] & (~B[0])

// X3 = A[3] xnor B[3]

xnor x0(X3, A[3], B[3]);

// X2 = A[2] xnor B[2]

xnor x1(X2, A[2], B[2]);

// X1 = A[1] xnor B[1]

xnor x2(X1, A[1], B[1]);

// X0 = A[0] xnor B[0]

xnor x3(X0, A[0], B[0]);

// outl1 = (~A[3]) & B[3]

and a0(outl1, ~A[3], B[3]);

// outl2 = X3 & (~A[2]) & B[2]

and a1(outl2, X3, ~A[2], B[2]);

// outl3 = X3 & X2 & (~A[1]) & B[1]

and a3(outl3, X3, X2, ~A[1], B[1]);

// outl4 = X3 & X2 & X1 & (~A[0]) & B[0]

and a4(outl4, X3, X2, X1, ~A[0], B[0]);

// A\_lt\_B = outl1 | outl2 | outl3 | outl4

or o0(A\_lt\_B, outl1, outl2, outl3, outl4);

// A\_eq\_B = X3 & X2 & X1 & X0

and a5(A\_eq\_B, X3, X2, X1, X0);

// outg1 = A[3] & (~B[3])

and a6(outg1, A[3], ~B[3]);

// outg2 = X3 & A[2] & (~B[2])

and a7(outg2, X3, A[2], ~B[2]);

// outg3 = X3 & X2 & A[1] & (~B[1])

and a8(outg3, X3, X2, A[1], ~B[1]);

// outg4 = X3 & X2 & X1 & A[0] & (~B[0])

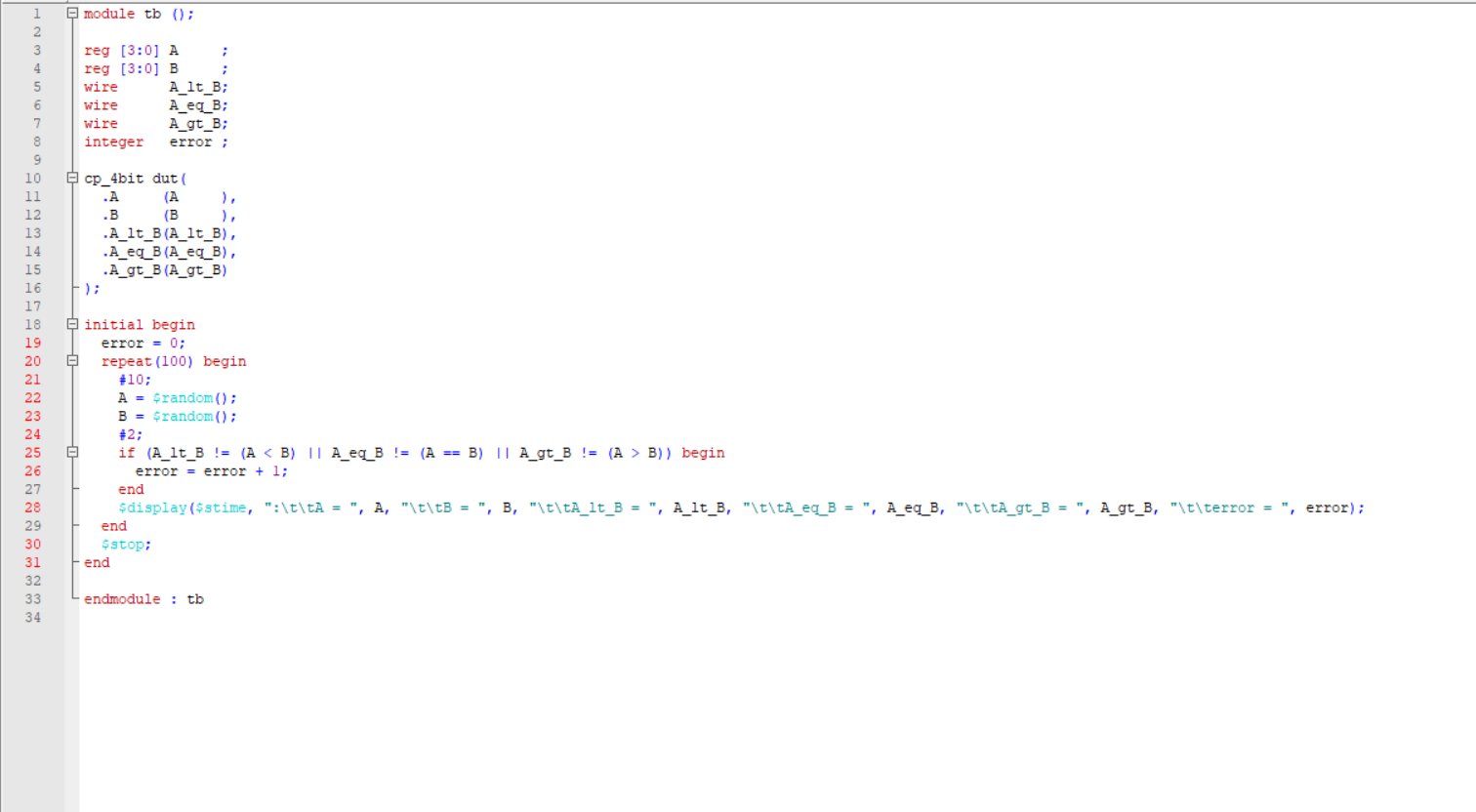
and a9(outg4, X3, X2, X1, A[0], ~B[0]);

// A\_gt\_B = outg1 | outg2 | outg3 | outg4

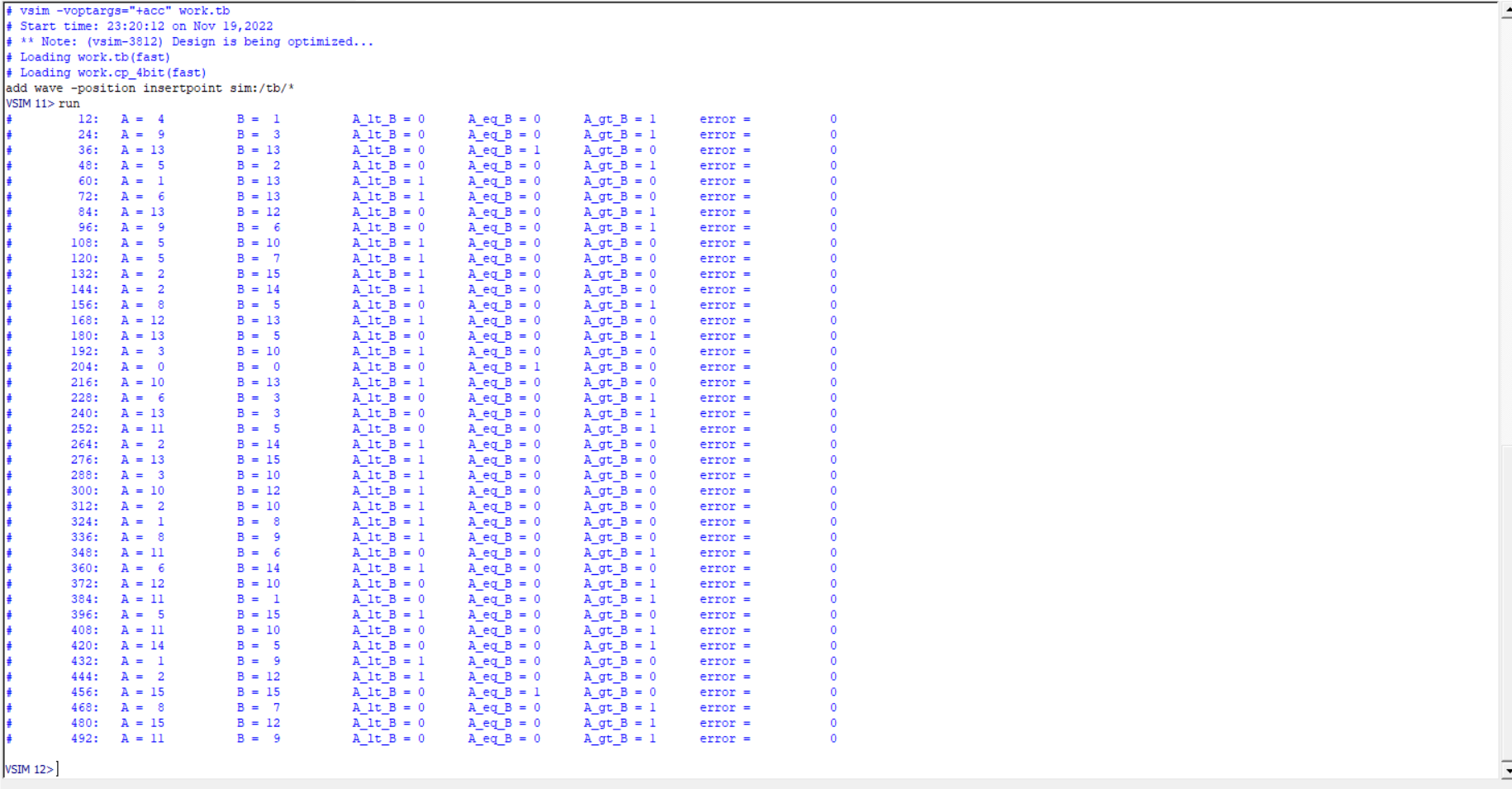
or o1(A\_gt\_B, outg1, outg2, outg3, outg4);

endmodule : cp\_4bit

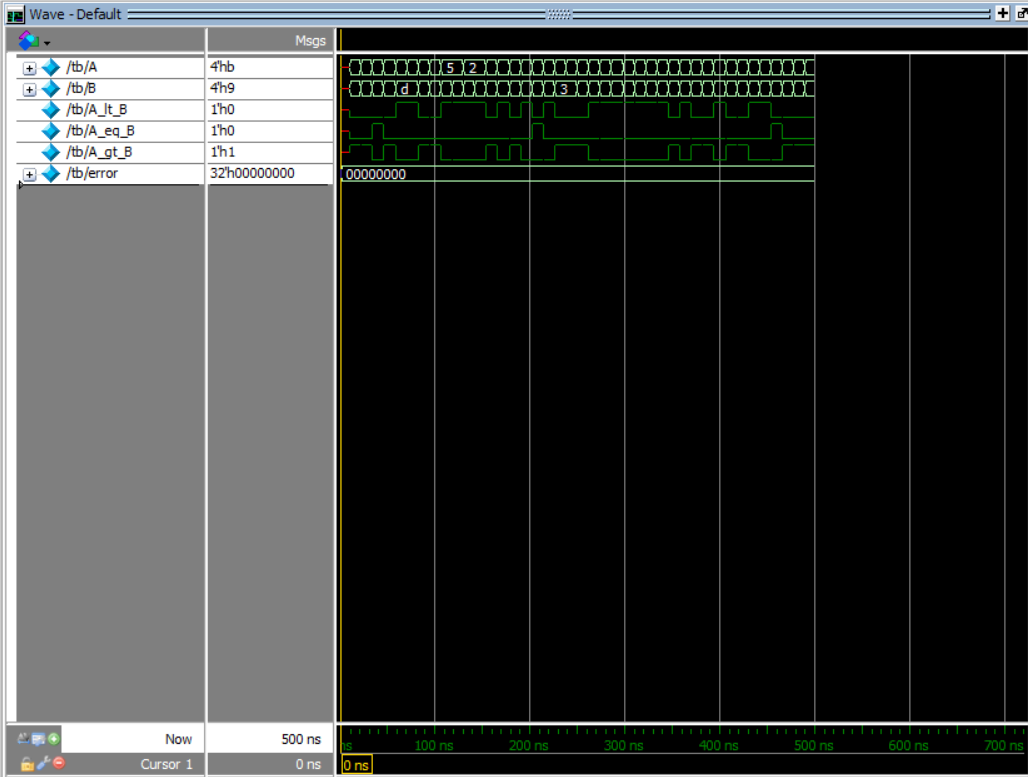
**Testbench**



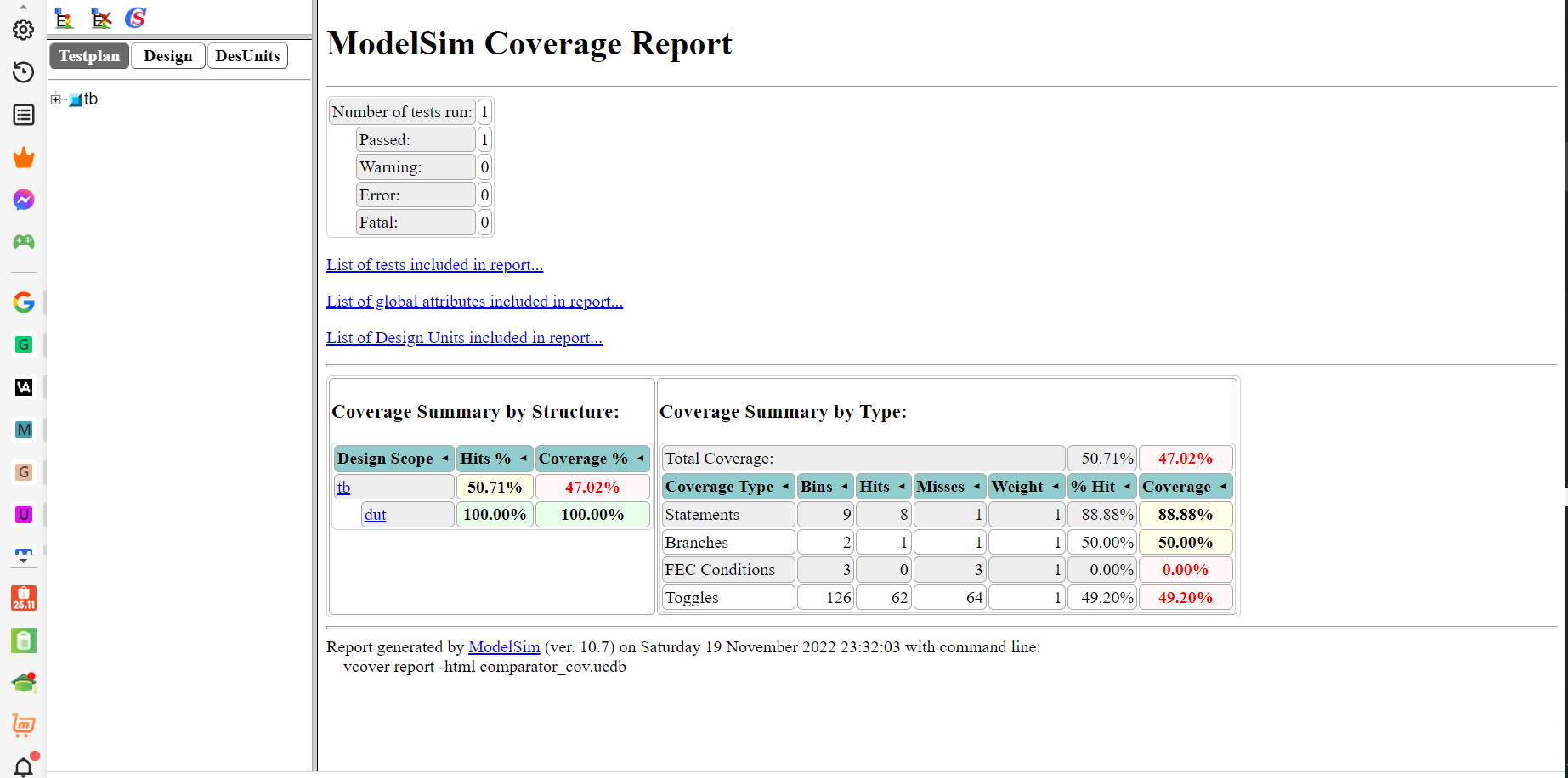
**Script**



**Wave**



**Report**

**s**