VERILOG

Examples

```
module halfadder(a,b,sum,carry);
input a,b;
output sum, carry;

assign sum = a^b;    // sum bit
assign carry = (a&b) ; //carry bit
endmodule

module main;
reg  a, b;
wire sum, carry;

halfadder add(a,b,sum,carry);

initial begin
   $monitor($time, "a=%b   b=%b   sum=%b   cout=%b", a, b, sum, carry);
   a=0; b=0;
   #10  b=1;
   #10 a=1;
   #10 b=0;
   #10 $finish;
   end

endmodule
```

```
tigger 198 % a.out
                0a=0   b=0   sum=0   cout=0
               10a=0   b=1   sum=1   cout=0
               20a=1   b=1   sum=0   cout=1
               30a=1   b=0   sum=1   cout=0
```

```
module halfAdd(
    output      sum, cout,
    input       a,b);

xor #2 (sum, a,b);
and #2 (cout, a, b);
endmodule
```

```
module tBench;

   wire  su, co, a, b;

   halfAdd      adder1(su, co, a, b);
   testAdd      tb(a,b,su,co);

endmodule
```

```
module testAdd(a,b,sum,cout);
    input   sum,cout;
    output a,b;
    reg    a,b;

initial begin
    $monitor($time, "a=%b   b=%b   sum=%b   cout=%b", a, b, sum, cout);
    a=0; b=0;
    #10  b=1;
    #10 a=1;
    #10 b=0;
    #10 $finish;
    end

endmodule
```

```
tigger 200 % a.out
                0a=0   b=0   sum=x   cout=x
                2a=0   b=0   sum=0   cout=0
               10a=0   b=1   sum=0   cout=0
               12a=0   b=1   sum=1   cout=0
               20a=1   b=1   sum=1   cout=0
               22a=1   b=1   sum=0   cout=1
               30a=1   b=0   sum=0   cout=1
               32a=1   b=0   sum=1   cout=0
```

```
module top(
  input       clk,
  input       rst_n,
  input       enable,
  input  [9:0] data_rx_1,
  input  [9:0] data_rx_2,
  output [9:0] data_tx_2
);

subcomponent subcomponent_instance_name (
 .clk(clk), .rst_n(rst_n), .data_rx(data_rx_1), .data_tx(data_tx) );

endmodule
```

```
module subcomponent(
   input         clk,
   input         rst_n,
   input  [9:0] data_rx,
   output [9:0] data_tx
);
```

```
module top(   ...);


subcomponent
subcomponent_instance_name (
  clk, rst_n, data_rx_1, data_tx );

endmodule
```