

# Homework 4, Lab Portion: Intro to Verilog

## module\_s.sv

```
module module_d(
    input logic a, b,
    output logic y1, y2, y3
);

    assign y1 = a & b;
    assign y2 = a | b;
    assign y3 = a ^ b;
endmodule
```

## module\_d.sv

```
module module_d(
    input logic a, b,
    output logic y1, y2, y3
);

    assign y1 = a & b;
    assign y2 = a | b;
    assign y3 = a ^ b;

endmodule
```

## module\_b.sv

```
module module_b(
    input logic a, b,
    output logic y1, y2, y3
);

    always_comb begin
        y1 = a & b;
        y2 = a | b;
        y3 = a ^ b;
    end

```

```
endmodule
```

## testbench.sv

```
'timescale 1ns/10ps

module testbench;
    logic a, b;
    logic y1_s, y2_s, y3_s;
    logic y1_d, y2_d, y3_d;
    logic y1_b, y2_b, y3_b;

    module_s
        structural_version(.a(a), .b(b), .y1(y1_s), .y2(y2_s), .y3(y3_s));
    module_d
        dataflow_version(.a(a), .b(b), .y1(y1_d), .y2(y2_d), .y3(y3_d));
    module_b
        behavioral_version(.a(a), .b(b), .y1(y1_b), .y2(y2_b), .y3(y3_b));

    initial begin
        $display("Time\ta\tb\ty1_s\ty2_s\ty3_s\ty1_d\ty2_d\ty3_d\ty1_b\ty2_b\ty3_b")
        $monitor("%0t\t%b\t%b\t%b\t%b\t%b\t%b\t%b\t%b\t%b\t%b",
                 $time, a, b, y1_s, y2_s, y3_s, y1_d, y2_d, y3_d,
                 y1_b, y2_b, y3_b);
        a = 0;
        b = 0;
        #10;
        a = 0;
        b = 1;
        #10;
        a = 1;
        b = 0;
        #10;
        a = 1;
        b = 1;
        #10;
    end
end
```