6.3.2021

BILKENT UNIVERSITY

COMPUTER ENGINEERING DEPT.

CS223 LAB 3 SECTION 4

LAB REPORT

ZÜBEYİR BODUR

21702382

Spring 2021

1. Circuit Schematics

a) 8:1 MUX

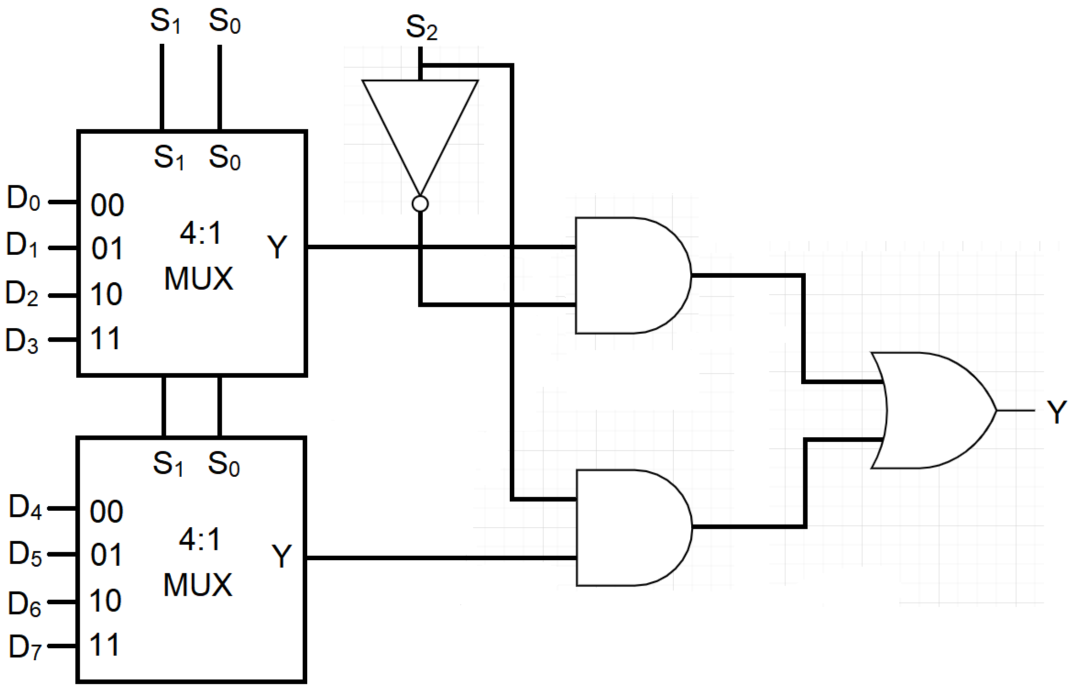

Figure 1: Circuit Schematic for 8:1 MUX

b) F(A, B, C, D)

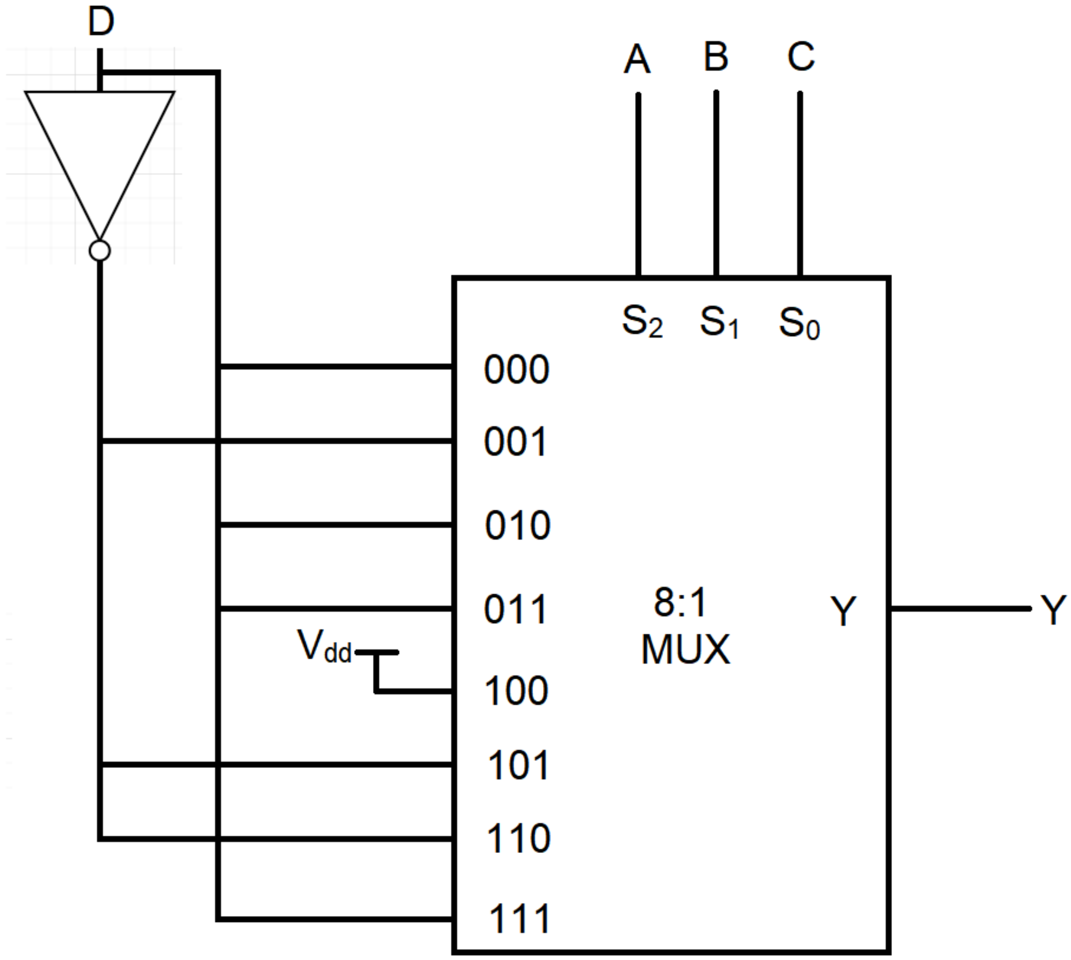

Figure 2: Circuit Schematic for F(A, B, C, D) = $\sum(1, 2, 5, 7, 8, 9, 10, 12, 15)$

2. SystemVerilog Modules & Testbenches

a) Behavorial - 2 : 4 Decoder

```
module dec2_4(
    output logic [3:0] Y,
    input logic  [1:0] D
    );
    always_comb begin
        case(D)
            2'b00: Y = 4'b0001;
            2'b01: Y = 4'b0010;
            2'b10: Y = 4'b0100;
            2'b11: Y = 4'b1000;
            default: Y = 4'b0000;
        endcase
    end
endmodule
```

```
module dec_test();
    logic [3:0] my_Y;
    logic [1:0] D;
    integer i;
    dec2_4 uut( .Y(my_Y),
                .D(D));
    initial begin
        D = 2'b0;
        for(i = 0; i < 4; i = i + 1) begin
            #10 D = D + 1;
        end
    end
endmodule
```

b) Behavorial - 4 : 1 MUX

```
module mux4_1(
    output logic Y,
    input logic [3:0] D,
    input logic [1:0] S
    );
    always_comb begin
        case(S)
            2'b00: Y = D[0];
            2'b01: Y = D[1];
            2'b10: Y = D[2];
            2'b11: Y = D[3];
            default: Y = 1'bZ;
        endcase
    end
endmodule
```

```
module mux4_1_test();
    logic [3:0] D;
    logic [1:0] S;
    logic Y;
    integer i, j;
    mux4_1 uut( .Y(Y),
              .D(D),
              .S(S));
    initial begin
        D = 4'b0; S = 2'b0;
        for(i = 0; i < 16; i = i + 1) begin
            for (j = 0; j < 4; j = j + 1) begin
                #10 S = S + 1;
            end
            #10 D = D + 1;
        end
    end
endmodule
```

c) 8 : 1 MUX

```
module mux8_1(
    output logic Y,
    input logic [7:0] D,
    input logic [2:0] S
    );
    logic lo, hi;
    mux4_1 muxlo(lo, D[3:0], S[1:0]);
    mux4_1 muxhi(hi, D[7:4], S[1:0]);
    //assign Y = S[2] ? hi : lo;
    logic s_not_and_lo, s_and_hi, s_not;
    inv sinv(s_not, S[2]);
    and2 and2lo(s_not_and_lo, s_not, lo);
    and2 and2hi(s_and_hi, S[2], hi);
    or2 out(Y, s_not_and_lo, s_and_hi);
endmodule
```

```
module mux8_1_test();
    logic [7:0] D;
    logic [2:0] S;
    logic Y;
    integer i, j;
    mux8_1 uut( .Y(Y),
               .D(D),
               .S(S));
    initial begin
        D = 8'b0; S = 3'b0;
        for(i = 0; i < 256; i = i + 1) begin
            for (j = 0; j < 8; j = j + 1) begin
                #10 S = S + 1;
            end
            #10 D = D + 1;
        end
    end
endmodule
```

d) $F(A, B, C, D) = \sum(1, 2, 5, 7, 8, 9, 10, 12, 15)$

```
module f(
    output logic y,
    input logic a, b, c, d
    );
    logic [7:0] inputs;
    logic [2:0] select;
    assign inputs = {d, ~d, ~d, 1'b1, d, d, ~d, d};
    assign select = {a, b, c};
    mux8_1 out(y, inputs, select);
endmodule
```

```
module f_test();
    logic [3:0] in;
    logic y;
    integer i;
    f uut(  .y(y),
            .a(in[3]),
            .b(in[2]),
            .c(in[1]),
            .d(in[0]));
    initial begin
        in = 4'b0;
        for (i = 0; i < 16; i = i + 1) begin
            # 10 in = in + 1;
        end
    end
endmodule
```