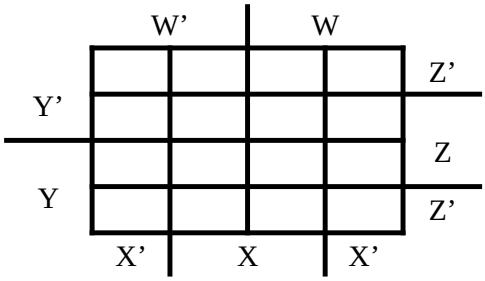
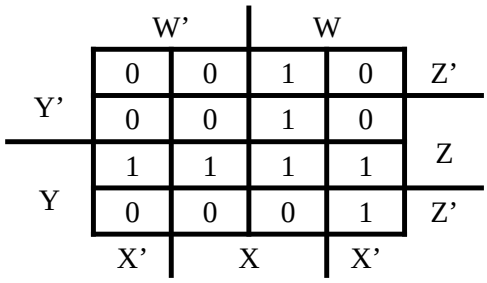
1) For a function F(W,X,Y,Z) that has the following truth table, fill in the equivalent Karnaugh map for it, below.

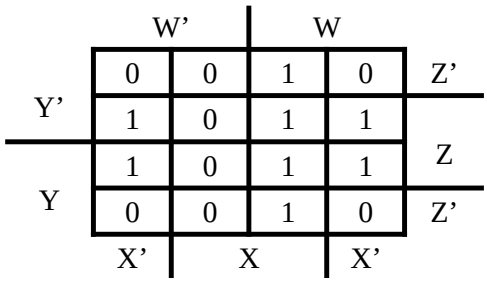
W	X	Y	Z	F(W,X,Y,Z)
0	0	0	0	0
0	0	0	1	0
0	0	1	0	0
0	0	1	1	0
0	1	0	0	0
0	1	0	1	1
0	1	1	0	0
0	1	1	1	1
1	0	0	0	0
1	0	0	1	1
1	0	1	0	1
1	0	1	1	0
1	1	0	0	1
1	1	0	1	0
1	1	1	0	1
1	1	1	1	0



2) For the following Karnaugh map, draw a **solid circle** around each essential prime implicant for the ON set of the function. Draw a **dotted circle** around each non-essential prime implicant.



3) For the Boolean function F(W,X,Y,Z) represented by the following Karnaugh map, find a **minimal sum-of-products** expression.



F(W,X,Y,Z) = _____

4) For the Karnaugh map, below, for F(W,X,Y,Z), we don't care about some states because their input values are not possible. Create a **minimal sum-of-products** expression for the ON set of F(W,X,Y,Z).

		W'		W	
Y'		0	0	0	d
		d	0	d	d
Y		0	d	0	1
		d	1	d	d
		X'	X		X'

Z'	
	Z
Z'	

F(W,X,Y,Z) = _____

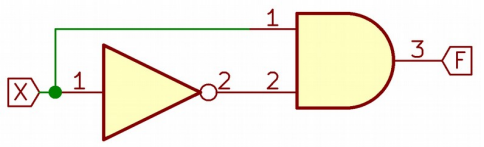
5) For the Karnaugh map, below, for F(W,X,Y,Z), we don't care about some states because their input values are not possible. Use the OFF set of the function to create a **minimal sum-of-products** expression for the **complement**, F'(W,X,Y,Z).

		W'		W	
Y'		d	0	1	1
		d	0	d	1
Y		0	d	d	0
		d	0	d	1
		X'	X		X'

Z'	
	Z
Z'	

F'(W,X,Y,Z) = _____

Use the following circuit for questions 6 and 7.



6) What kind of hazard exists? (i.e., Static-0 or Static-1?)

7) Describe X when the glitch occurs.

8) The complement of a Boolean function is expressed as a sum of products, F'(X,Y,Z) = X·Y'·Z + X·Y·Z' + Y·Z. Use DeMorgan's law to transform it into a **product of sums** expression for F(X,Y,Z).

Use the Karnaugh map, below, for questions 9, 10, and 11. Assume a minimal sum-of-products expression is used to realize the Boolean function F(W,X,Y,Z) with gates.

		W'		W	
		0	0	1	1
Y'		1	1	1	1
		0	1	0	0
Y		0	1	0	0
		X'	X		X'

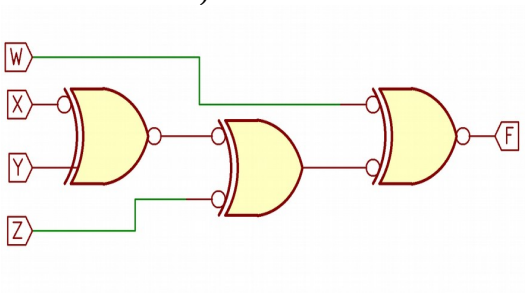
Z'	
	Z
Z'	

9) What kind of logic hazards exist? (i.e., static -1 or static-0)

10) Describe W, X, Y, and Z when the glitch occurs.

11) Write a complete sum-of-products expression for F that is glitch-free. Do so by adding a single term to the minimal sum-of-product expression.

12) For the Boolean function realized by the circuit, below, complete the truth table. (Reminder: You should not need to evaluate each case if you remember how a network of XOR and XNOR gates can be reduced.)

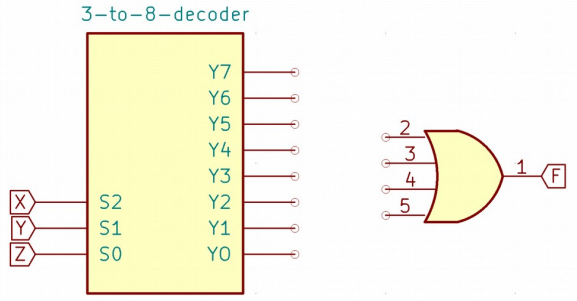


W	X	Y	Z	F
0	0	0	0	
0	0	0	1	
0	0	1	0	
0	0	1	1	
0	1	0	0	
0	1	0	1	
0	1	1	0	
0	1	1	1	
1	0	0	0	
1	0	0	1	
1	0	1	0	
1	0	1	1	
1	1	0	0	
1	1	0	1	
1	1	1	0	
1	1	1	1	

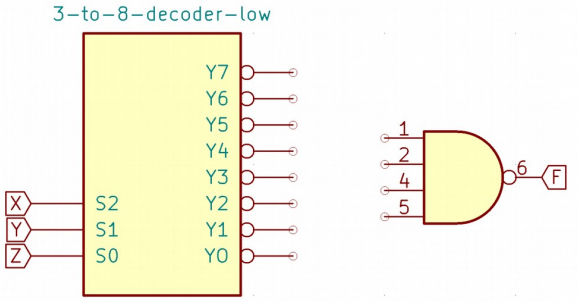
Refer to the truth table, below, for questions 13, 14, 15, and 16.

X	Y	Z	F(X,Y,Z)
0	0	0	0
0	0	1	0
0	1	0	1
0	1	1	1
1	0	0	0
1	0	1	1
1	1	0	0
1	1	1	1

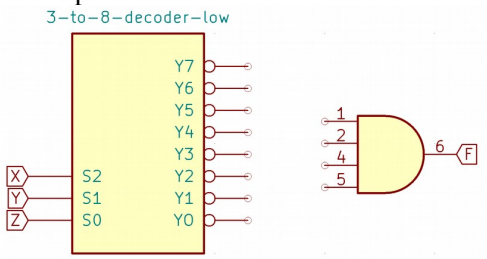
13) Wire a simple 3-to-8 decoder, with active-high outputs, to a 4-input OR gate to implement the Boolean function.



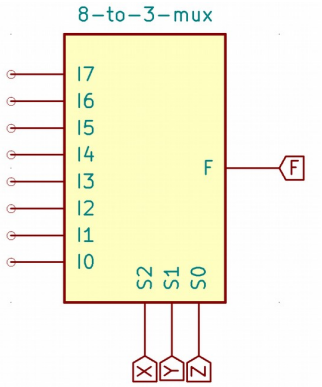
14) Wire a 3-to-8 decoder, with active-low outputs, to a 4-input NAND gate to implement the Boolean function.



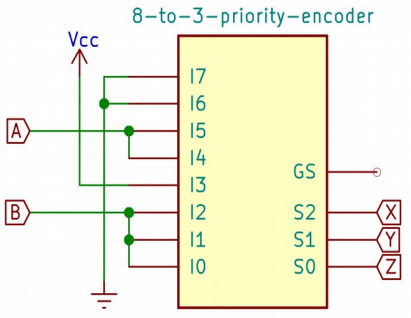
15) Wire a 3-to-8 decoder, with active-low outputs, to a 4-input AND gate to implement the Boolean function.



16) Wire the inputs of a simple 8-to-3 multiplexer, with active-high inputs, to either 1 (logic high, or V_{DD}) or 0 (Gnd) to implement the Boolean function.



17) Write a Boolean expression for the X output of an 8-to-3 priority encoder with active-high inputs and outputs. Express your answer in terms of A or B or their negations.



X=_____

Here are two example Verilog modules:

```
module fun(p,q,r,s);
    input wire p,q;
    output wire r,s;
    assign {r,s} = {p & q, p ^ q};
endmodule

module fun2fa(a,b,x);
    input wire [1:0] a, b;
    output wire [2:0] x;
    wire c1,c2,c3;
    fun u0(.p(a[0]), .q(b[0]),
           .r(c1), .s(x[0]));
    fun u1(.p(a[1]), .q(c1),
           .r(c2), .s(s1));
    fun u2(.p(b[1]), .q(s1),
           .r(c3), .s(x[1]));
    assign x[2] = c2 | c3;
endmodule
```

18) Write a Boolean expression for x[0] in terms of a[1:0] and b[1:0]. Make it as simple as possible.

19) Write a Verilog module called 'SoP' that implements the Boolean function **F = X'·Y·Z + X'·Y'·Z' + X·Y'·Z** List the ports in any order. Be sure to use proper Verilog syntax for any AND, OR, or NOT operators.

20) Write a Verilog module named ‘snooze’ with 1-bit input wires named ‘a’ and ‘b’, and a 1-bit output wire named ‘out’. ‘out’ should be ‘1’ if and only if ‘a’ is high, and ‘b’ is low.

21) For signals defined in a Verilog module as ‘`wire [68:0]X;`’ and ‘`wire [23:1]Y;`’, state the size of the following vector: `{X[8:7],Y[12:11],X[9:6],X[10:2]}`

22) For the following Verilog module,

```
module decode(x,a);
       input wire [2:0]a;
       output wire [7:0]x;
       wire [7:0] map [7:0];
       assign map[0] = 8'hf1;
       assign map[1] = 8'he2;
       assign map[2] = 8'hd3;
       assign map[3] = 8'hc4;
       assign map[4] = 8'hb5;
       assign map[5] = 8'ha6;
       assign map[6] = 8'h97;
       assign map[7] = 8'h88;
       assign x = map[a];
endmodule
```

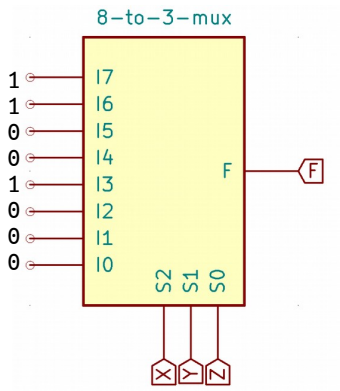
What is the value of x[3:0] when a == 3’b110 ?
Express your answer as a Verilog binary literal.
Yes, we just want the lower 4 bits of x.
In binary. You can do it!

23) For the following Verilog module

```
module priority42(in,s);
    input wire [3:0] in;
    output wire [2:0] s;
    assign s = in[3] == 1 ? 3'b111 :
               in[2] == 1 ? 3'b110 :
               in[1] == 1 ? 3'b101 :
               in[0] == 1 ? 3'b100 :
               3'b000;
endmodule
```

What is the value of s when in == 4'b1010 ?

24) For the circuit below, write a <u>minimal</u> sum-of-products expression for F(X,Y,Z).



25) For the following Verilog code, what is the value of f when x=1, y=1, z=0?

```
module mux8to1(in,s,out);
    input wire [7:0] in;
    input wire [2:0] s;
    output wire out;
    assign out = (s== 3'b111) ? in[7] :
                 (s== 3'b110) ? in[6] :
                 (s== 3'b101) ? in[5] :
                 (s== 3'b100) ? in[4] :
                 (s== 3'b011) ? in[3] :
                 (s== 3'b010) ? in[2] :
                 (s== 3'b001) ? in[1] :
                                in[0];
endmodule

module multi(x,y,z,f);
  input x,y,z;
  output [2:0]f;
  mux8to1 u2(8'b10010101,{x,y,z},f[2]);
  mux8to1 u1(8'b01010011,{x,y,z},f[1]);
  mux8to1 u0(8'b01100101,{x,y,z},f[0]);
endmodule
```