Solutions - Homework 1

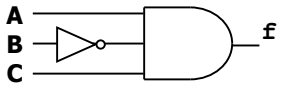
(Due date: January 22nd @ 5:30 pm)
Presentation and clarity are very important!

PROBLEM 1 (20 PTS)

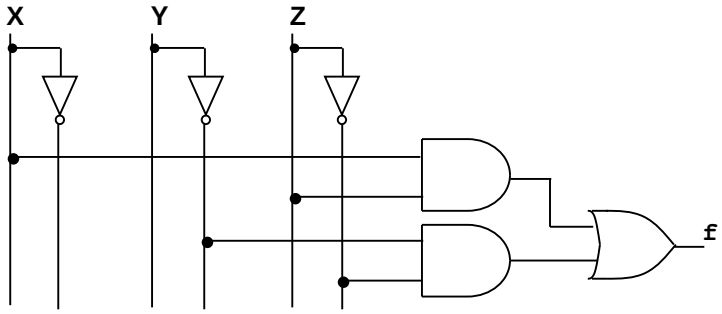
a) Simplify the following functions using ONLY Boolean Algebra Theorems. For each resulting simplified function, sketch the logic circuit using AND, OR, XOR, and NOT gates. (12 pts)

- ✓ $F = \overline{A(B + \bar{C}) + \bar{A}}$
- ✓ $F(X, Y, Z) = \prod(M_2, M_4, M_6, M_7)$
- ✓ $F = (Z + \bar{Y})(\bar{Z} + X)(\bar{Y} + X)$
- ✓ $F = \overline{(\bar{X} + \bar{Y})Z + \bar{X}YZ}$

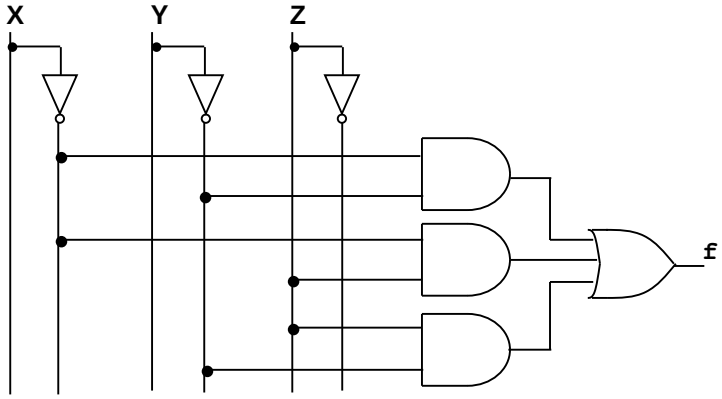
✓ $F = \overline{A(B + \bar{C}) + \bar{A}} = \overline{A(B + \bar{C})}.A = (\bar{A} + \overline{B + \bar{C}}).A = (\overline{B + \bar{C}}).A = A\bar{B}C$



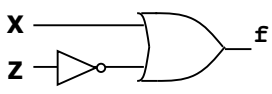
✓ $F = (Z + \bar{Y})(\bar{Z} + X)(\bar{Y} + X) = (Z + \bar{Y})(\bar{Z} + X)$ (Consensus Theorem)
$(Z + \bar{Y})(\bar{Z} + X) = ZX + \bar{Z}\bar{Y} + X\bar{Y} = ZX + \bar{Z}\bar{Y}$ (Consensus Theorem)



✓ $F(X, Y, Z) = \prod(M_2, M_4, M_6, M_7) = \sum(m_0, m_1, m_3, m_5) = \bar{X}\bar{Y}\bar{Z} + \bar{X}\bar{Y}Z + \bar{X}YZ + X\bar{Y}Z = \bar{X}\bar{Y} + \bar{X}YZ + X\bar{Y}Z$
$F(X, Y, Z) = \bar{X}\bar{Y} + \bar{X}YZ + X\bar{Y}Z = \bar{X}(\bar{Y} + YZ) + X\bar{Y}Z = \bar{X}(\bar{Y} + Z) + X\bar{Y}Z = \bar{X}\bar{Y} + \bar{X}Z + X\bar{Y}Z$
$F(X, Y, Z) = \bar{X}\bar{Y} + \bar{X}Z + X\bar{Y}Z = \bar{X}\bar{Y} + Z(\bar{X} + X\bar{Y}) = \bar{X}\bar{Y} + Z(\bar{X} + \bar{Y}) = \bar{X}\bar{Y} + Z\bar{X} + Z\bar{Y}$



✓ $F = \overline{(\bar{X} + \bar{Y})Z + \bar{X}YZ} = \overline{(\bar{X} + \bar{Y})Z}.\overline{\bar{X}YZ} = (X + Y + \bar{Z})(X + \bar{Y} + \bar{Z}) = (A + Y)(A + \bar{Y}), A = X + \bar{Z}$
$F = (A + Y)(A + \bar{Y}) = A = X + \bar{Z}$



b) For the following Truth table with two outputs: (8 pts)
- Provide the Boolean functions using the Canonical Sum of Products (SOP), and Product of Sums (POS).
- Express the Boolean functions using the minterms and maxterms representations.
- Sketch the logic circuits as Canonical Sum of Products and Product of Sums.

x	y	z	f_1	f_2
0	0	0	0	0
0	0	1	1	0
0	1	0	1	1
0	1	1	1	1
1	0	0	1	0
1	0	1	0	1
1	1	0	1	1
1	1	1	0	1

Sum of Products

$f_1 = \bar{X}\bar{Y}Z + \bar{X}Y\bar{Z} + \bar{X}YZ + X\bar{Y}\bar{Z} + XY\bar{Z}$

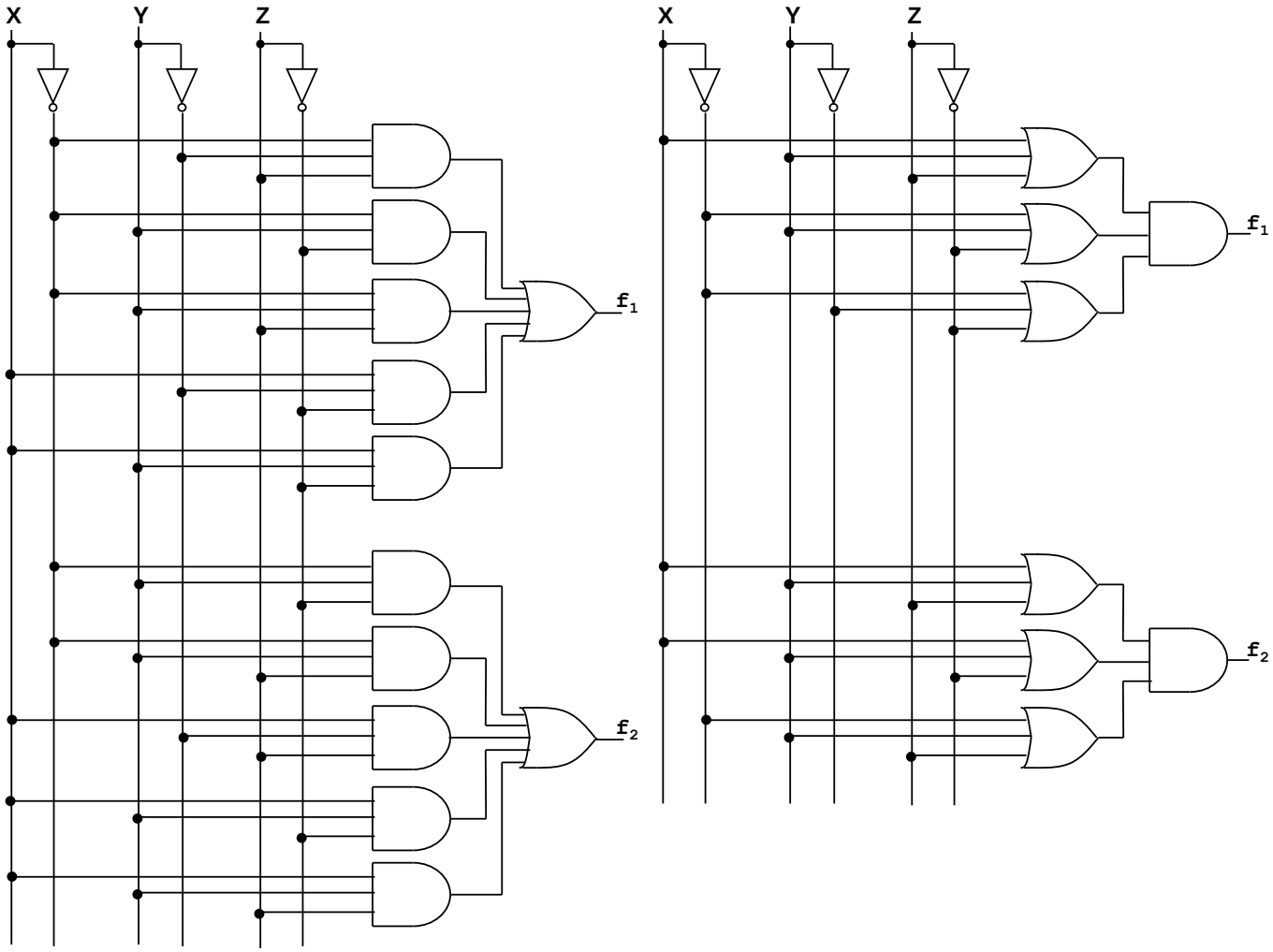
$f_2 = \bar{X}Y\bar{Z} + \bar{X}YZ + X\bar{Y}Z + XY\bar{Z} + XYZ$

Product of Sums

$f_1 = (X + Y + Z)(\bar{X} + Y + \bar{Z})(\bar{X} + \bar{Y} + \bar{Z})$

$f_2 = (X + Y + Z)(X + Y + \bar{Z})(\bar{X} + Y + Z)$

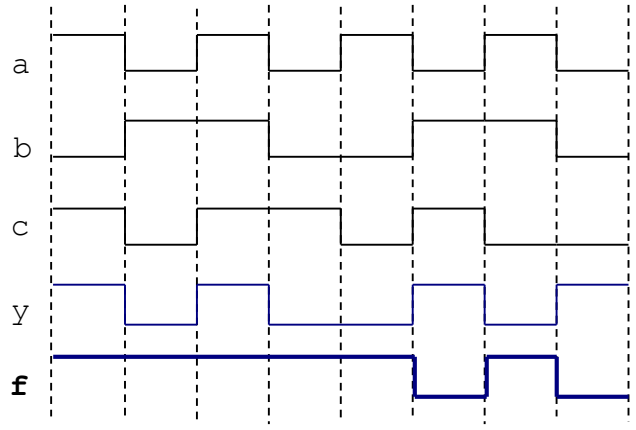
Minterms and maxterms: $f_1 = \sum(m_1, m_2, m_3, m_4, m_6) = \prod(M_0, M_5, M_7)$.
$f_2 = \sum(m_2, m_3, m_5, m_6, m_7) = \prod(M_0, M_1, M_4)$.



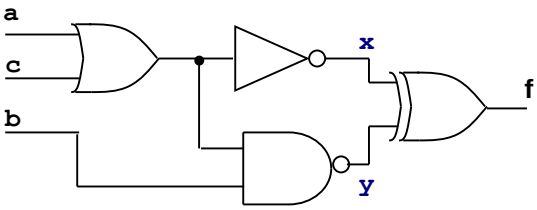
PROBLEM 2 (20 PTS)

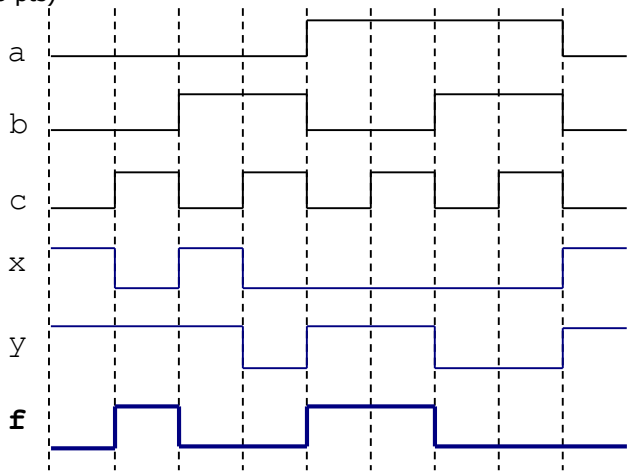
a) Complete the timing diagram of the logic circuit whose VHDL description is shown below: (5 pts)

```
library ieee;
use ieee.std_logic_1164.all;

entity circ is
  port ( a, b, c: in std_logic;
         f: out std_logic);
end circ;

architecture st of circ is
  signal x, y: std_logic;

begin
  x <= a nor b;
  y <= x xor c;
  f <= y nand (not a);
end st;
```



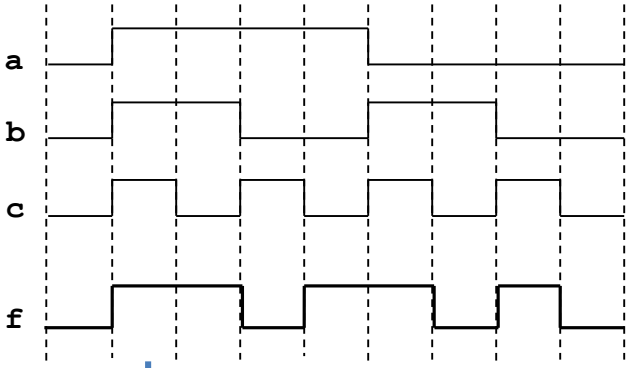
b) Complete the timing diagram of the following circuit: (5 pts)





c) The following is the timing diagram of a logic circuit with 3 inputs. Sketch the logic circuit that generates this waveform. Then, complete the VHDL code. (10 pts)

```
library ieee;
use ieee.std_logic_1164.all;

entity circ is
 port ( a, b, c: in std_logic;
        f: out std_logic);
end circ;

architecture st of circ is
  signal x,y,z: std_logic;
begin
  x <= a and not(c);
  y <= not(a) and c;
  z <= b and c;
  f <= x or y or z;
end st;
```



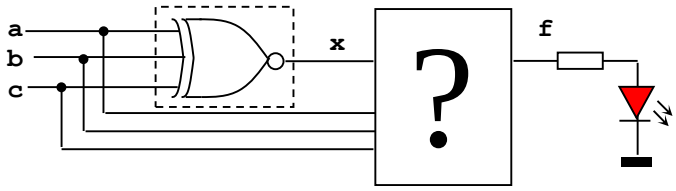
a	b	c	f
0	0	0	0
0	0	1	1
0	1	0	0
0	1	1	1
1	0	0	1
1	0	1	0
1	1	0	1
1	1	1	1

c \ ab	00	01	11	10
0	0	0	1	1
1	1	1	1	0

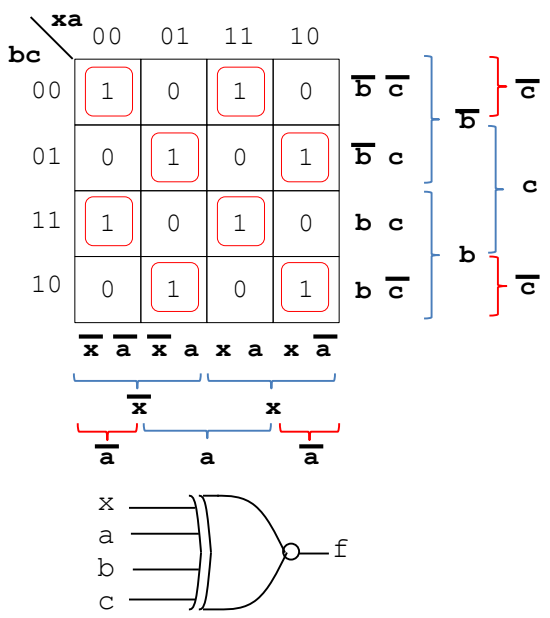
$$f = \bar{a}c + a\bar{c} + bc$$

PROBLEM 3 (15 PTS)

Design a circuit (simplify your circuit) that verifies the logical operation of a 3-input XNOR gate. f = '1' (LED ON) if the XNOR gate does NOT work properly. Assumption: when the XNOR gate is not working, it generates 1's instead of 0's and vice versa.



x	a	b	c	f
0	0	0	0	1
0	0	0	1	0
0	0	1	0	0
0	0	1	1	1
0	1	0	0	0
0	1	0	1	1
0	1	1	0	1
0	1	1	1	0
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	0
1	1	1	1	1



$$f = \bar{x}\bar{a}\bar{b}\bar{c} + \bar{x}\bar{a}bc + \bar{x}a\bar{b}c + \bar{x}ab\bar{c} + xa\bar{b}\bar{c} + xabc + x\bar{a}\bar{b}c + x\bar{a}b\bar{c}$$

$$f = \bar{x}\bar{a}\left(\overline{b \oplus c}\right) + \bar{x}a(b \oplus c) + xa\left(\overline{b \oplus c}\right) + x\bar{a}(b \oplus c)$$

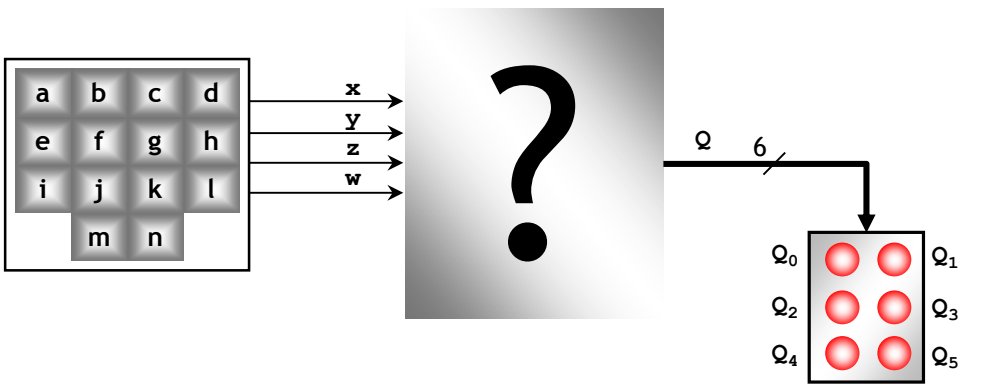
$$f = (\overline{x \oplus a})\left(\overline{b \oplus c}\right) + (x \oplus a)(b \oplus c)$$

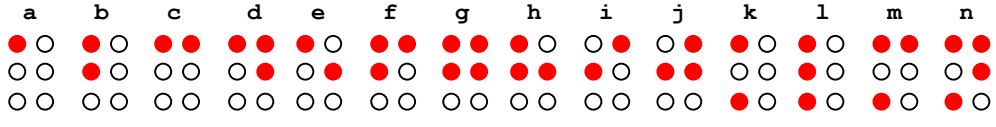
$$f = \overline{(x \oplus a) \oplus (b \oplus c)} = \overline{(x \oplus a \oplus b \oplus c)}$$

PROBLEM 4 (30 PTS)

A 16-letter keypad produces a 4-bit code as shown in the table. We want to design a logic circuit that converts those 4-bit codes to Braille code, where the 6 dots are represented by LEDs. A raised (or embossed) dot is represented by an LED ON (logic value of '1'). A missing dot is represented by a LED off (logic value of '0').

- ✓ Complete the truth table for each output (Q_0-Q_5).
- ✓ Provide the simplified expression for each output (Q_0-Q_5). Use Karnaugh maps for Q_5-Q_2 and the Quine-McCluskey algorithm for Q_1-Q_0. Note it is safe to assume that the codes `1110` and `1111` will not be produced by the keypad.





x	y	z	w	Letter
0	0	0	0	a
0	0	0	1	b
0	0	1	0	c
0	0	1	1	d
0	1	0	0	e
0	1	0	1	f
0	1	1	0	g
0	1	1	1	h
1	0	0	0	i
1	0	0	1	j
1	0	1	0	k
1	0	1	1	l
1	1	0	0	m
1	1	0	1	n
1	1	1	0	
1	1	1	1	

x	y	z	w	Q_5	Q_4	Q_3	Q_2	Q_1	Q_0	Letter
0	0	0	0	0	0	0	0	0	1	a
0	0	0	1	0	0	0	1	0	1	b
0	0	1	0	0	0	0	0	1	1	c
0	0	1	1	0	0	1	0	1	1	d
0	1	0	0	0	0	1	0	0	1	e
0	1	0	1	0	0	0	1	1	1	f
0	1	1	0	0	0	1	1	1	1	g
0	1	1	1	0	0	1	1	0	1	h
1	0	0	0	0	0	0	1	1	0	i
1	0	0	1	0	0	1	1	1	0	j
1	0	1	0	0	1	0	0	0	1	k
1	0	1	1	0	1	0	1	0	1	l
1	1	0	0	0	1	0	0	1	1	m
1	1	0	1	0	1	1	0	1	1	n
1	1	1	0	X	X	X	X	X	X	
1	1	1	1	X	X	X	X	X	X	

Q_4

zw \ xy	00	01	11	10
00	0	0	1	0
01	0	0	1	0
11	0	0	X	1
10	0	0	X	1

Q_3

zw \ xy	00	01	11	10
00	0	1	0	0
01	0	0	1	1
11	1	1	X	0
10	0	1	X	0

Q_2

zw \ xy	00	01	11	10
00	0	0	0	1
01	1	1	0	1
11	0	1	X	1
10	0	1	X	0

$Q_5 = 0$

$Q_4 = xz + xy$

$Q_3 = \bar{x}y\bar{w} + \bar{x}zw + x\bar{z}w$

$Q_2 = yz + \bar{x}\bar{z}w + x\bar{y}\bar{z} + x\bar{y}w$

$Q_1 = \bar{x}\bar{y}z + y\bar{z}w + \bar{x}z\bar{w} + x\bar{z}$

$Q_0 = \bar{x} + y + z$

- $Q_1 = \sum m(2,3,5,6,8,9,12,13) + \sum d(14,15)$.

Number of ones	4-literal implicants	3-literal implicants	2-literal implicants	1-literal implicants
1	m_2 = 0010 ✓ m_8 = 1000 ✓	**$m_{2,3}$ = 001-** **$m_{2,6}$ = 0-10** $m_{8,9}$ = 100- ✓ $m_{8,12}$ = 1-00 ✓	**$m_{8,9,12,13}$ = 1-0-** ~~$m_{8,12,9,13}$ = 1-0-~~ **$m_{12,14,13,15}$ = 11--** ~~$m_{12,13,14,15}$ = 11--~~	We can't combine any further, so we stop here
2	m_3 = 0011 ✓ m_5 = 0101 ✓ m_6 = 0110 ✓ m_9 = 1001 ✓ m_{12}= 1100 ✓	**$m_{5,13}$ = -101** **$m_{6,14}$ = -110** $m_{9,13}$ = 1-01 ✓ $m_{12,13}$ = 110- ✓ $m_{12,14}$ = 11-0 ✓		
3	m_{13}= 1101 ✓ m_{14}= 1110 ✓	$m_{13,15}$ = 11-1 ✓ $m_{14,15}$ = 111- ✓		
4	m_{15}= 1111 ✓			

Prime Implicants		Minterms 2	3	5	6	8	9	12	13
$m_{2,3}$	$\bar{x}\bar{y}z$	X	**X**						
$m_{2,6}$	$\bar{x}z\bar{w}$	X			**X**				
$m_{5,13}$	$y\bar{z}w$			**X**					X
$m_{6,14}$	$yz\bar{w}$				X				
$m_{8,9,12,13}$	$x\bar{z}$					**X**	**X**	**X**	X
$m_{12,14,13,15}$	xy							X	X

$\rightarrow Q_1 = \bar{x}\bar{y}z + y\bar{z}w + x\bar{z} + \bar{x}z\bar{w}$

- $Q_0 = \sum m(0,1,2,3,4,5,6,7,10,11,12,13) + \sum d(14,15)$: Too many minterms. We better optimize $\overline{Q_0} = \sum m(8,9) + \sum d(14,15)$:

Number of ones	4-literal implicants	3-literal implicants	2-literal implicants
1	m_8 = 1000 ✓	**$m_{8,9}$ = 100-**	We can't combine any further, so we stop here
2	m_9 = 1001 ✓		
3	m_{14}= 1110 ✓	**$m_{14,15}$ = 111-**	
4	m_{15}= 1111 ✓		

Prime Implicants		Minterms 8	9
$m_{8,9}$	$x\bar{y}\bar{z}$	**X**	**X**
$m_{14,15}$	xyz		

$\rightarrow \overline{Q_0} = x\bar{y}\bar{z}, \quad \Rightarrow \quad Q_0 = \overline{x\bar{y}\bar{z}} = \bar{x} + y + z$

PROBLEM 5 (15 PTS)

The following die has a sensor on each side. Whenever a side rests on a surface, the sensor on that side outputs a logic '1' (which is transmitted wirelessly to a controller); otherwise, it outputs a '0'. The sensors are named S_i, where i is the number printed on the respective side.

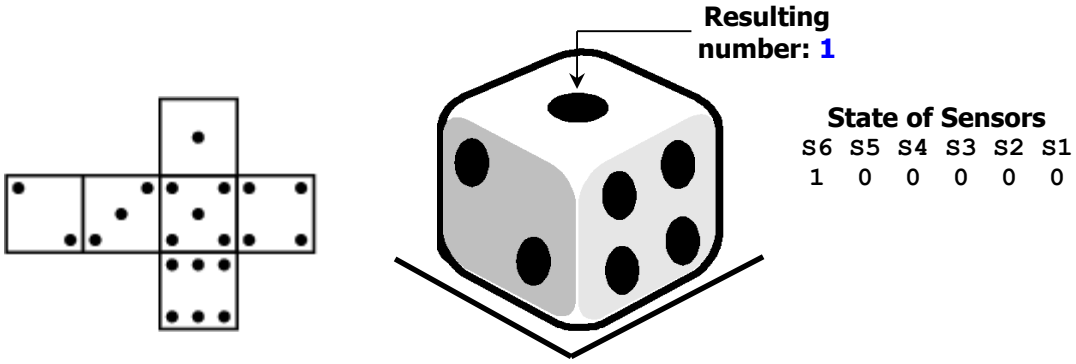
We want to design a circuit that reads the state of the 6 sensors and outputs the value of the upper surface. That decimal value is represented using the binary number system (see table below). A LED ON is represented by the logic value of '1', and a LED OFF is represented by a logic value of '0'.

For example, in the figure below, the resting side has six dots while the upper surface has one. The resulting decimal number is then '**1**', which is represented by 001 on the LEDs. The state of the sensors is S6=1, S5=0, S4=0, S3=0, S2=0, S1=0.

Under normal circumstances, we expect only one sensor activated at a time. However, due to a variety of problems, we might have the following cases:

- Two or more sensors produce a '1' at the same time: In this case, the state of the LEDs must be 000.
- No sensor produces a '1': In this case, the state of the LEDs must be 000.

✓ Using the state of the sensors as inputs, provide the Boolean expression for each LED: L_2, L_1, L_0. Use the canonical Sum of Products for each function.



L_2	L_1	L_0	Number
0	0	0	–
0	0	1	1
0	1	0	2
0	1	1	3
1	0	0	4
1	0	1	5
1	1	0	6
1	1	1	–

S1	S2	S3	S4	S5	S6	L_2	L_1	L_0	Number
0	0	0	0	0	0	0	0	0	–
0	0	0	0	0	1	0	0	1	1
0	0	0	0	1	0	0	1	0	2
0	0	0	1	0	0	0	1	1	3
0	0	1	0	0	0	1	0	0	4
0	1	0	0	0	0	1	0	1	5
1	0	0	0	0	0	1	1	0	6
					...	0	0	0	–

$$L_2 = \overline{S1}\,\overline{S2}S3\overline{S4}\,\overline{S5}\,\overline{S6} + \overline{S1}S2\overline{S3}S4\,\overline{S5}\,\overline{S6} + S1\overline{S2}\,\overline{S3}\,\overline{S4}\,\overline{S5}\,\overline{S6}$$

$$L_1 = \overline{S1}\,\overline{S2}\,\overline{S3}\,\overline{S4}S5\overline{S6} + \overline{S1}\,\overline{S2}\,\overline{S3}S4\overline{S5}\,\overline{S6} + S1\overline{S2}\,\overline{S3}\,\overline{S4}\,\overline{S5}\,\overline{S6}$$

$$L_0 = \overline{S1}\,\overline{S2}\,\overline{S3}\,\overline{S4}\,\overline{S5}S6 + \overline{S1}S2\overline{S3}S4\overline{S5}\,\overline{S6} + \overline{S1}S2\overline{S3}\,\overline{S4}\,\overline{S5}\,\overline{S6}$$