CSI 2111 (Fall 2004)
Assignment # 2 Solution

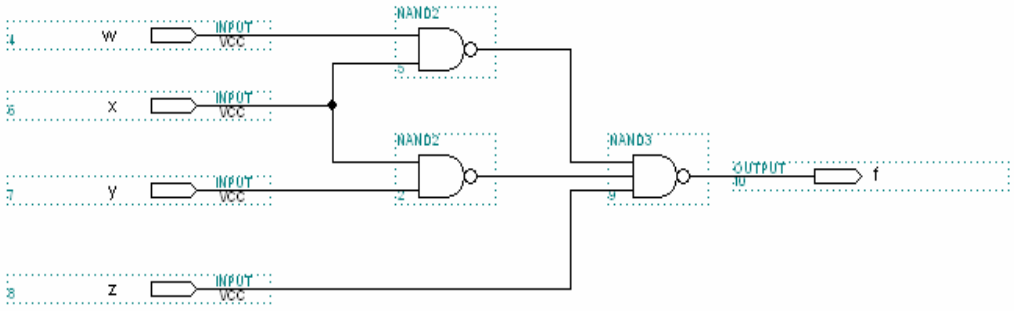
Q1. For this question, use Max+plus II in *Functional SNF Extractor* mode. Use a different directory for each question (same name as your VHDL file / graph file).

Consider the following POS form for *f*: $f(w, x, y, z) = (x + z') \cdot (w + y + z')$

a) Realize *f* **graphically** using only NAND gates of suitable sizes (no NOT gates). Call it *a2q1a*. (15)

Solution 1 – a

f(w, x, y, z) = (x + z') . (w + y + z')
= wx + xy + xz' + wz' + yz' + z'
= wx + xy + z'
= ((wx)'. (xy)'. z)'
= (w ↑ x) ↑ (x ↑ y) ↑ z




```
%       z y x w   f %

   0.0> 0 0 0 0 = 1
 100.0> 1 0 0 0 = 0
 200.0> 0 1 0 0 = 1
 300.0> 1 1 0 0 = 0
 400.0> 0 0 1 0 = 1
 500.0> 1 0 1 0 = 0
 600.0> 0 1 1 0 = 1
 700.0> 1 1 1 0 = 1
 800.0> 0 0 0 1 = 1
 900.0> 1 0 0 1 = 0
1000.0> 0 1 0 1 = 1
1100.0> 1 1 0 1 = 0
1200.0> 0 0 1 1 = 1
1300.0> 1 0 1 1 = 1
1400.0> 0 1 1 1 = 1
1500.0> 1 1 1 1 = 1
1600.0> X X X X = X
;
```

b) Realize *f* again in **VHDL**, and use only NOR gates this time. Call it *a2q1b.* (15)

Solution 1 – b

f(w, x, y, z) = (x + z') . (w + y + z')
= ((x + z')' + (w + y + z')')'
= ((x + z')' + ((w + y)'' + z')')'
= (x ↓ z') ↓ ((w ↓ y)' ↓ z')
= (x ↓ (z ↓ z)) ↓ (((w ↓ y) ↓ (w ↓ y)) ↓ (z ↓ z))

```
-- Implementation of f(w,x,y,z)=(x+z')(w+y+z') using NOR gates.

LIBRARY ieee;
USE ieee.std_logic_1164.all;

ENTITY a2q1b IS
       PORT ( w, x, y, z  : IN  STD_LOGIC;
              f           : OUT STD_LOGIC );
END a2q1b;

ARCHITECTURE nor_only OF a2q1b IS
SIGNAL s1,s2 : STD_LOGIC;
BEGIN
    s1 <= z NOR z;
    s2 <= w NOR y;
    f  <=  (x NOR s1) NOR ((s2 NOR s2) NOR s1);
END nor_only;
```

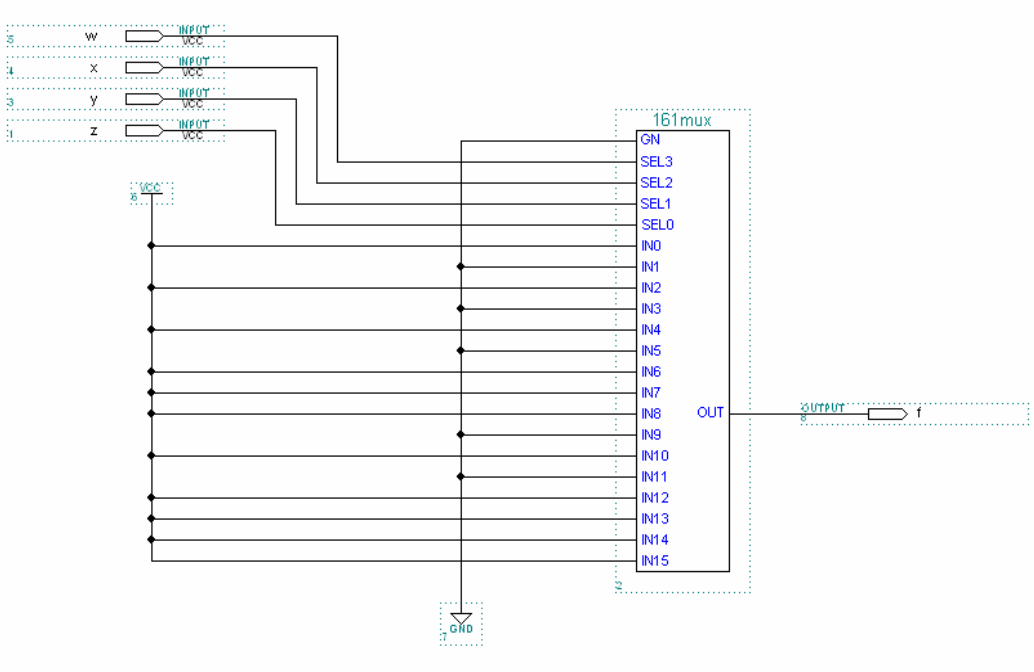
```
%       z y x w   f %

   0.0> 0 0 0 0 = 1
 100.0> 1 0 0 0 = 0
 200.0> 0 1 0 0 = 1
 300.0> 1 1 0 0 = 0
 400.0> 0 0 1 0 = 1
 500.0> 1 0 1 0 = 0
 600.0> 0 1 1 0 = 1
 700.0> 1 1 1 0 = 1
 800.0> 0 0 0 1 = 1
 900.0> 1 0 0 1 = 0
1000.0> 0 1 0 1 = 1
1100.0> 1 1 0 1 = 0
1200.0> 0 0 1 1 = 1
1300.0> 1 0 1 1 = 1
1400.0> 0 1 1 1 = 1
1500.0> 1 1 1 1 = 1
1600.0> X X X X = X
;
```

c) Realize *f* again **graphically,** but use a 16 to 1 multiplexer (i.e., *161mux* symbol) this time. Call it *a2q1c.* (15)

Solution 1 – c

f(w, x, y, z) = Σ m(0, 2, 4, 6, 7, 8, 10, 12, 13, 14, 15)




```
%        z y x w    f %

    0.0> 0 0 0 0 = 1
  100.0> 1 0 0 0 = 0
  200.0> 0 1 0 0 = 1
  300.0> 1 1 0 0 = 0
  400.0> 0 0 1 0 = 1
  500.0> 1 0 1 0 = 0
  600.0> 0 1 1 0 = 1
  700.0> 1 1 1 0 = 1
  800.0> 0 0 0 1 = 1
  900.0> 1 0 0 1 = 0
 1000.0> 0 1 0 1 = 1
 1100.0> 1 1 0 1 = 0
 1200.0> 0 0 1 1 = 1
 1300.0> 1 0 1 1 = 1
 1400.0> 0 1 1 1 = 1
 1500.0> 1 1 1 1 = 1
 1600.0> X X X X = X
;
```

Compare the results of the three cases for all the combinations of *w, x, y, z* (0 to 15). They should be similar. You can compare the signals (*waveforms*) or the truth tables generated by the tool (for one or the other of the diagrams).

The truth tables for the three cases are same !!

Q2. Consider the functions *P(x,y,z)*, which generates the parity bit *P* according to the even parity scheme, and *C(x,y,z,P)*, which checks the parity according to the even parity scheme. The function *P* takes the value 0 if and only if the number of 1's in the sequence *xyz* is even, otherwise it takes the value 1. The function *C* takes the value 0 if and only if the number of 1's in the sequence *xyzP* is even (correct), otherwise it takes the value 1 (error detected).

a) Give the canonical SOP form of the functions *P* and *C*. (5)

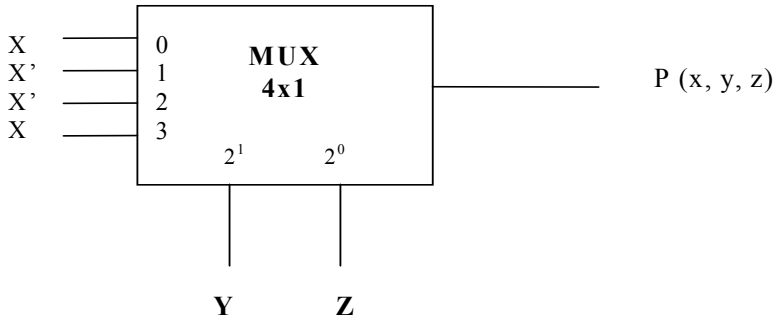
Solution 2 – a

P (x, y, z) = Σ m(1, 2, 4, 7)
C (x, y, z, P) = Σ m(1, 2, 4, 7, 8, 11, 13, 14)

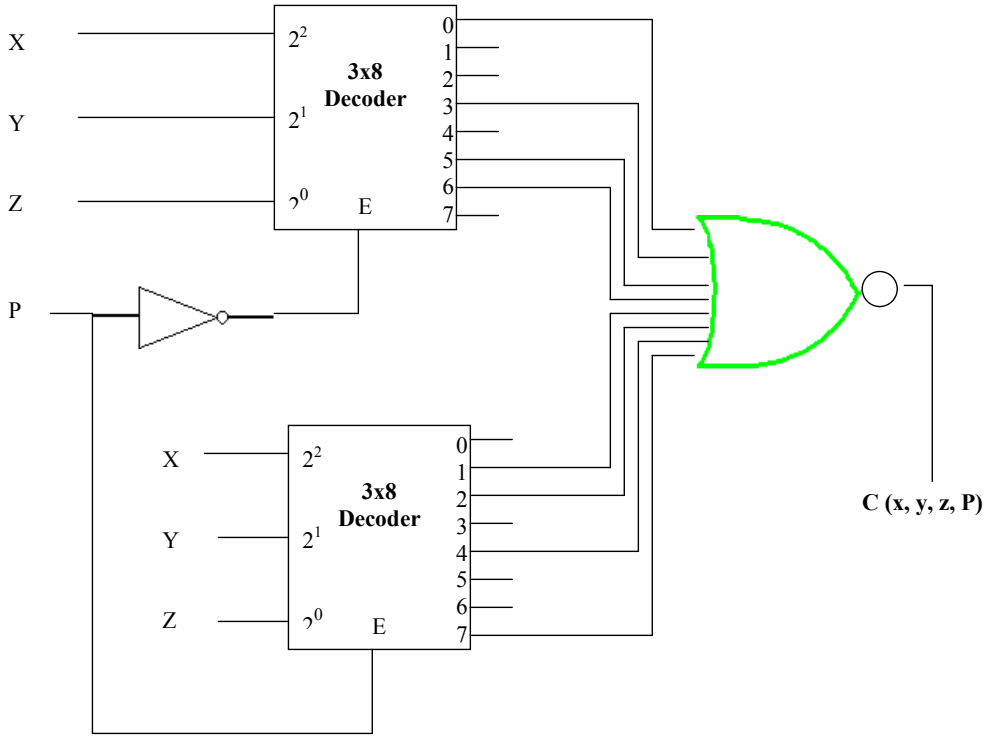
b) Realize *P(x,y,z)* on paper with a 4 to 1 MUX, with *y* and *z* as command inputs. Assume you also have all variables in complemented form. Identify the inputs/outputs of the MUX. (15)

Solution 2 – b

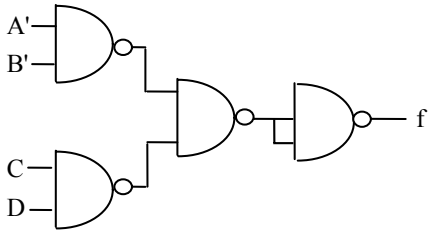
	I_0	I_1	I_2	I_3
x’	0	1	2	3
x	4	5	6	7



c) Realize *C(x,y,z,P)* on paper with two 8 to 1 decoders with *Enable* input, one NOT gate, and one NOR gate (of suitable size) only. Use *x,y,z* as the control inputs. Identify the inputs/outputs of the decoders. (15)



Q3. a) Among the following functions, which one is implemented using this circuit? Why? (5)



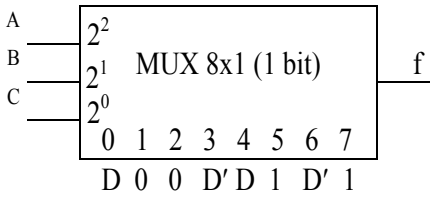
(1) f(A,B,C,D) = Σm(4,5,6,7,8,9,10) +X(12,13,14,15)
(2) f(A,B,C,D) = Σm(4,5,6,8,9,10,14)
(3) f(A,B,C,D) = Σm(0,1,2,3,6,7,14,15)
(4) f(A,B,C,D) = Σm(0,1,2,3,7,11,15)
(5) f(A,B,C,D) = Σm(4,5,6,8,9,10,12,13,14)+X(7,11)

Solution 3 – a

f(A, B, C, D) = ((A' . B')' . (C . D)')''
= (A' . B')' . (C . D)'
= (A + B) . (C' + D')
= AC' + BC' + AD' + BD'
= Σm (4, 5, 6, 8, 9, 10, 12, 13, 14)

The correct answer is : **(5) f(A,B,C,D) = Σm(4,5,6,8,9,10,12,13,14)+X(7,11)**

b) Among the following functions, which one is implemented using the following circuit? Why? (5)



(1) f(A,B,C,D) = Σm(2,5,9,10,11,12,13,15)
(2) f(A,B,C,D) = Σm(0,7,8,10,11,13,14,15)
(3) f(A,B,C,D) = Σm(3,4,9,10,11,12,13,15)
(4) f(A,B,C,D) = Σm(3,5,6,7,8,12,13,15)
(5) f(A,B,C,D) = Σm(1,6,9,10,11,12,14,15)

Solution 3 – b

	I_0	I_1	I_2	I_3	I_4	I_5	I_6	I_7
D’	0	2	4	(6)	8	(10)	(12)	(14)
D	(1)	3	5	7	(9)	(11)	13	(15)

The correct answer is : **(5) f(A,B,C,D) = Σm(1,6,9,10,11,12,14,15)**

Q4. a) At least, how many 8 to 1 MUX are necessary to build a 4096 to 1 MUX? How many external control inputs will this large MUX have? (5)

To build a 4096 to 1 MUX, we need a total of **585** (8x1) MUX. We need a 4-layer circuit – with 512 MUX in the first layer, 64 MUX in the second layer, 8 MUX in the second layer and a single MUX in the final layer.
The large MUX would have **12** control inputs.

b) At least, how many 4 x 16 decoders with *Enable* input are necessary to build a 8 x 256 decoder with *Enable* input? (5)

We need **17** (4x16) decoders to build a a (8x256) decoder. One decoder would control the other 16 decoders.