Assignment # 2 Solutions - CSI 2111

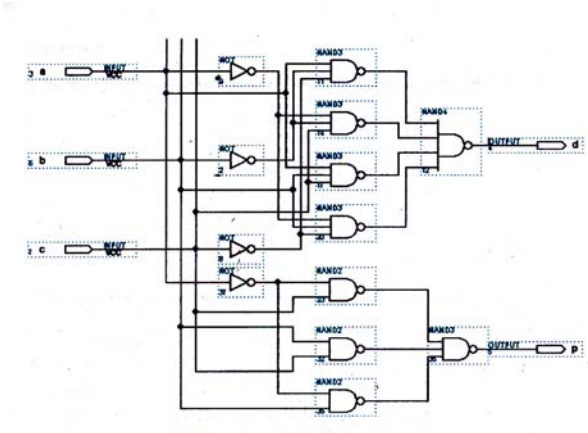
Q1. We need to design a full subtractor which computes $a - b - c$, where c is the borrow from the next less significant digit that produces a difference, d, and a borrow from the next more significant bit, p.

a) Give the truth table for the full subtractor. (5)

a	b	c	p	d
0	0	0	0	0
0	0	1	1	1
0	1	0	1	1
0	1	1	1	0
1	0	0	0	1
1	0	1	0	0
1	1	0	0	0
1	1	1	1	1

b) Implement the circuit using only NAND gates and inverters. (5)

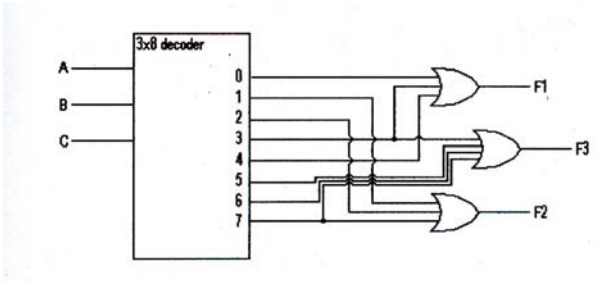
d = [(ab'c')'(a'bc')'(a'b'c)'(abc)']'and
p = [(a'b)'(a'c)'(bc)']'



(Figure courtesy of Dominique Bruneau and Martin Charrette)

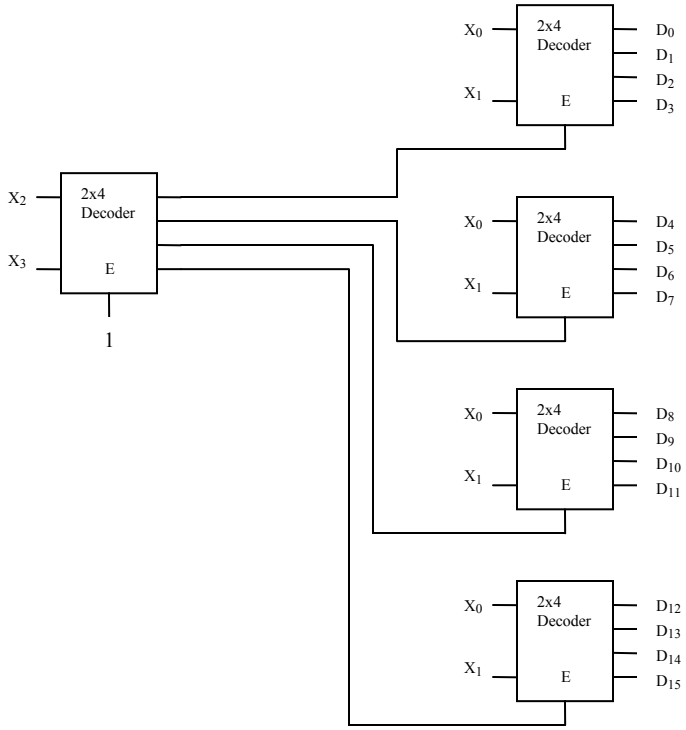
Q2. a) Implement, with a decoder and external OR gates, the combinational circuit specified by the following three Boolean functions: (5)

$f_1(A, B, C) = \Sigma m(0,3,4)$
$f_2(A, B, C) = \Sigma m(1,2,7)$
$f_3(A, B, C) = \Pi M(0,1,2,4)$



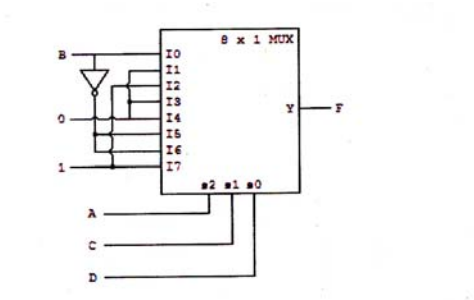
(Figure courtesy of Dominique Bruneau and Martin Charrette)

b) Design a 4-to-16 line decoder with *Enable* input using five 2-to-4 line decoders with *Enable* inputs. (5)



Q3. a) Implement the following Boolean function with an 8-to-1 line multiplexer and a single inverter with variable B as an input. (5)

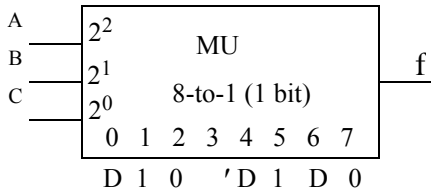
$$f(A, B, C, D) = \Sigma m(2, 4, 6, 9, 10, 11, 15)$$



(Figure courtesy of Dominique Bruneau and Martin Charrette)

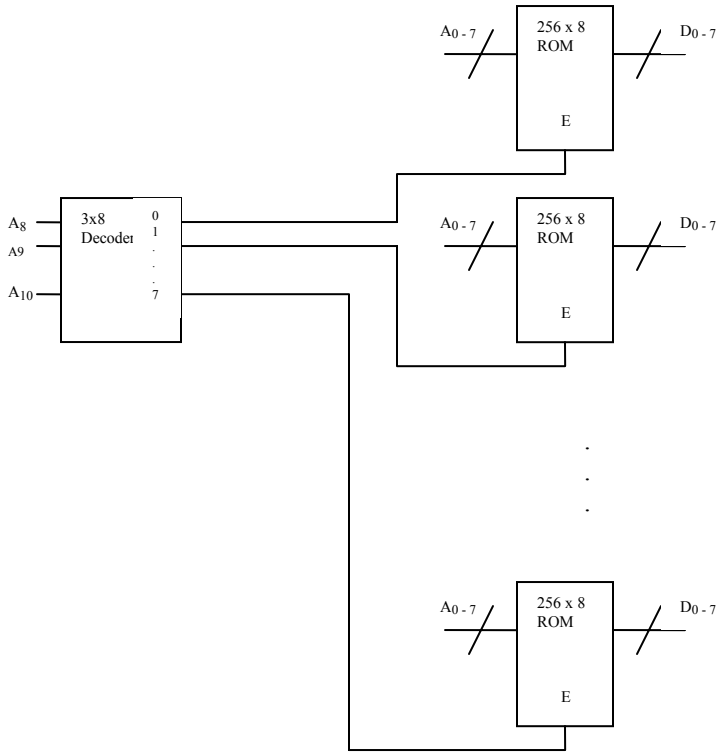
b) Give the canonical sum of product expression for the function which is implemented using the following circuit.

(5)



$f(A,B,C,D) = \Sigma m(1,2,3,6,9,10,11,13)$

Q4. a) Given a 256 x 8 ROM chip with *Enable* input, show the external connections necessary to construct a 2K x 8 ROM with eight chips and a decoder. (5)



b) Specify the size of a ROM (number of words and number of bits per word) necessary to implement a binary multiplier that multiplies two 8-bit numbers. (5)

We need 16 bits to represent the result, and since there are two operands (each of 8-bits), the RAM table size is 2^{16} x 16 = 64K x 16.

Q5. Consider the function ***f*** (w, x, y, z) = Σm (1, 3, 7, 11, 13, 14, 15).

a) Implement *f* in VHDL, using AND, OR, NOT gates. Call it *circuit5a*. (5)

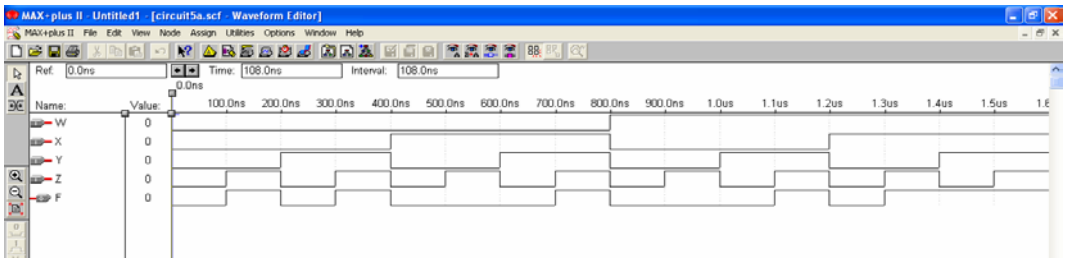
b) Implement *f* in VHDL, but use **only** NAND gates (no NOT gates!). Call it *circuit5b*. (5)

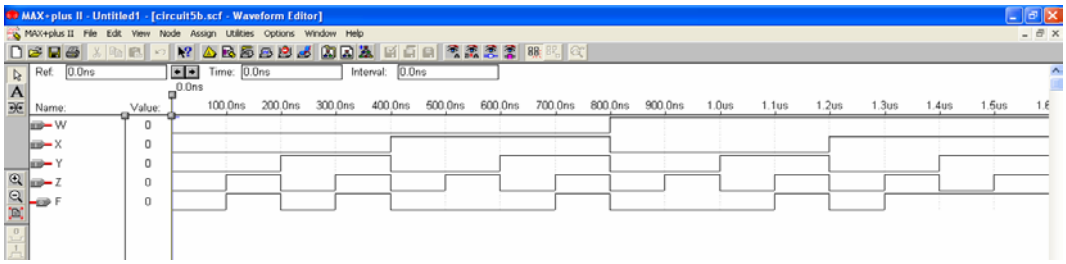
c) Implement *f* again in VHDL, but use **only** NOR gates this time. Call it *circuit5c*. (5)

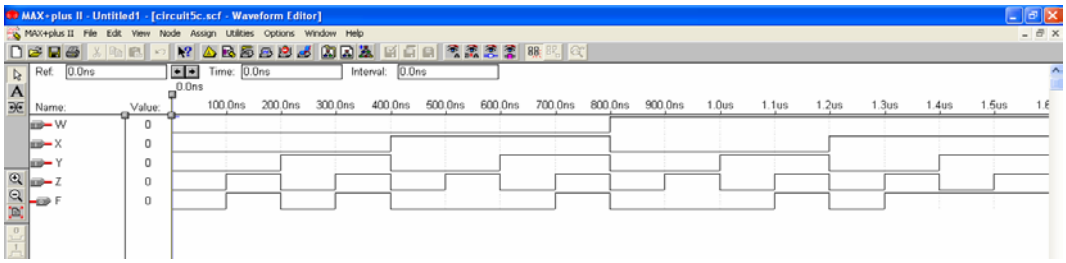
d) Compare the simulation results of the three implementations in VHDL for all combinations of the inputs *w, x, y, z*. Please include the timing diagram (0 ns to 1600 ns) and the truth tables for 1a), 1b), and 1c) and explain the differences, if any. (5)

For this question, use Max+plus II in the *Functional SNF Extractor* mode. This mode eliminates the signal transmission delays, which makes checking the equivalence between two functions easier.

Solution:
Part (a) See attached file "circuit5a.vhd".
Part (b) See attached file "circuit5b.vhd".
Part (c) See attached file "circuit5c.vhd".
Part (d) The simulation results from "circuit5a", "circuit5a", "circuit5a" should be exactly same, because they implement the same function. See attached files : "circuit5a.scf", "circuit5b.scf", "circuit5a.scf".







```
-- Question 5 (a)
--  Implements f(w,x,y,z)= m(1,3,7,13,14,15) using AND,OR,NOT
--
--                f = (y.z) + (w'.x'.z) + (w.x)(y + z)
--

LIBRARY ieee;
USE ieee.std_logic_1164.all;

ENTITY circuit5a IS
      PORT ( W, X, Y, Z : IN  STD_LOGIC;
                F            : OUT STD_LOGIC);
END circuit5a;


ARCHITECTURE andornot OF circuit5a IS
SIGNAL S1, S2, S3 : STD_LOGIC;
BEGIN
      S1 <= (Y AND Z);
      S2 <= ((NOT W) AND (NOT X) AND Z);
      S3 <= (W AND X AND (Y OR Z));
      F <= (S1 OR S2 OR S3);
END andornot;

-- Question 5 (b)
--  Implements f(w,x,y,z)= m(1,3,7,13,14,15) using NAND only
--
--                f = ((y.z)'.(w'.x'.z)'.(w.x.y)'.(w.x.z)')'
--

LIBRARY ieee;
USE ieee.std_logic_1164.all;

ENTITY circuit5b IS
      PORT ( W, X, Y, Z : IN  STD_LOGIC;
                F            : OUT STD_LOGIC);
END circuit5b;


ARCHITECTURE nandonly OF circuit5b IS
SIGNAL A, B, C, D : STD_LOGIC;
BEGIN
-- F = (A.B.C.D)' = (((A.B)')'.((C.D)')')'
      F <= ((A NAND B) NAND '1') NAND ((C NAND D) NAND '1');
-- A = (Y.Z)'
      A <= (Y NAND Z);
-- B = (W'.X'.Z)'
      B <= ( (((W NAND W) NAND (X NAND X)) NAND '1') NAND Z );
-- C = (W.X.Y)'
      C <= ( ((W NAND X) NAND '1') NAND Y );
-- D = (W.X.Z)'
      D <= ( ((W NAND X) NAND '1') NAND Z );
END nandonly;


-- Question 5 (C)
--  Implements f(w,x,y,z)= m(1,3,7,13,14,15) using NOR only
--
--                f = (y+z).(x+z).(w+z).(w+x'+y).(w'+x+y)          [ POS
form ]
```

```
--                  = ((y+z)'+(x+z)'+(w+z)'+(w+x'+y)'+(w'+x+y)')'

LIBRARY ieee;
USE ieee.std_logic_1164.all;

ENTITY circuit5c IS
      PORT ( W, X, Y, Z : IN  STD_LOGIC;
                F            : OUT STD_LOGIC);
END circuit5c;


ARCHITECTURE noronly OF circuit5c IS
SIGNAL A, B, C, D, E : STD_LOGIC;
SIGNAL P, Q              : STD_LOGIC;
BEGIN
-- A = (Y+Z)'
      A <= (Y NOR Z);
-- B = (X+Z)'
      B <= (X NOR Z);
-- C = (W+Z)'
      C <= (W NOR Z);
-- D = (W+X'+Y)' = ( ((W+X')')' + Y )'
      D <= ( ((W NOR (X NOR X)) NOR '0') NOR Y );
-- E = (W'+X+Y)' = ( ((W'+X)')' + Y )'
      E <= ( (((W NOR W) NOR X) NOR '0') NOR Y );

-- F = (A+B+C+D+E)' = ( ((P + Q)')' + E )'  where P = (A+B) and Q=(C+D)
      F <= ( ((P NOR Q) NOR '0') NOR E );
-- P = A+B = ((A+B)')'
      P <= (A NOR B) NOR '0';
-- Q = C+D = ((A+B)')'
      Q <= (C NOR D) NOR '0';

END noronly;
```