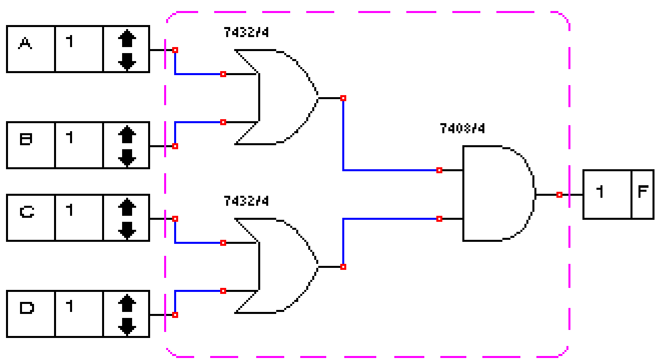
CSCI 220: Computer Architecture-I
Instructor: Pranava K. Jha

Design of Combinational Circuits

Q. 1. Four large tanks at a chemical plant contain different liquids being heated. Liquid-level sensors are being used to detect whenever the level in tank *A* or tank *B* rises above a predetermined level. Temperature sensors in tanks *C* and *D* detect when the temperature in either of these tanks drops below a prescribed temperature limit. Assume that the liquid-level sensor outputs *A* and *B* are LOW when the level is satisfactory and HIGH when the level is too high. Also, the temperature-sensor outputs *C* and *D* are LOW when the temperature is satisfactory and HIGH when the temperature is too low. Design a logic circuit that will detect whenever the level in tank *A* or tank *B* is too high at the same time that the temperature in either tank *C* or tank *D* is too low.

It is not difficult to see that circuit output is given by $(A + B)(C + D)$.



Q. 2. Consider a four-input function $F(A, B, C, D)$ that outputs a 1 whenever an odd number of its inputs are 1.

- Construct the truth table
- Generate the K-map. Can you simplify the function?
- Present an implementation of the function using two-input XOR gates.

Truth Table

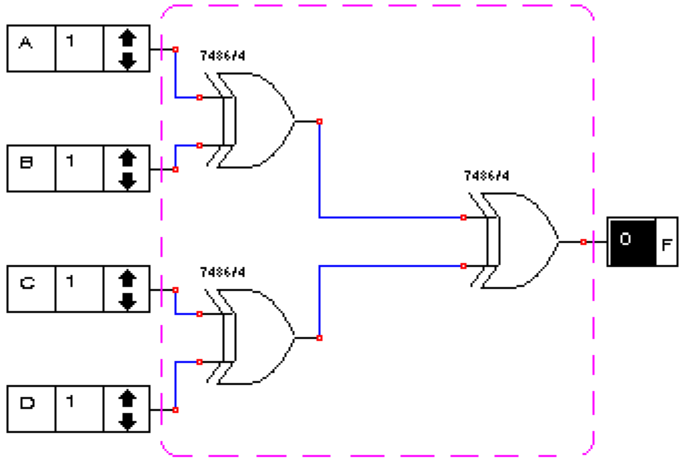
A	*B*	*C*	*D*	*F*
0	0	0	0	0
0	0	0	1	1
0	0	1	0	1
0	0	1	1	0
0	1	0	0	1
0	1	0	1	0
0	1	1	0	0
0	1	1	1	1
1	0	0	0	1
1	0	0	1	0
1	0	1	0	0
1	0	1	1	1
1	1	0	0	0
1	1	0	1	1
1	1	1	0	1
1	1	1	1	0

Karnaugh map is immediate.

	C'D'	*C'D*	*CD*	*CD'*
A'B'	0	1	0	1
A'B	1	0	1	0
AB	0	1	0	1
AB'	1	0	1	0

It is clear from the map that simplification is not possible. Further, it is not difficult to show that

$$F(A, B, C, D) = A \oplus B \oplus C \oplus D.$$


A
1
B
1
C
1
D
1
7486/4
7486/4
7486/4
0
F

Q. 3. Construct a truth table for converting a four-bit binary string *ABCD* into a four-bit binary string *WXYZ* in *Gray code*. Use the scheme of building a sequence in Gray code that was given in the class. Further, obtain simplified expressions for *W*, *X*, *Y* and *Z* in terms of *A*, *B*, *C* and *D*.

Binary				Gray code			
A	*B*	*C*	*D*	*W*	*X*	*Y*	*Z*
0	0	0	0	0	0	0	0
0	0	0	1	0	0	0	1
0	0	1	0	0	0	1	1
0	0	1	1	0	0	1	0
0	1	0	0	0	1	1	0
0	1	0	1	0	1	1	1
0	1	1	0	0	1	0	1
0	1	1	1	0	1	0	0
1	0	0	0	1	1	0	0
1	0	0	1	1	1	0	1
1	0	1	0	1	1	1	1
1	0	1	1	1	1	1	0
1	1	0	0	1	0	1	0
1	1	0	1	1	0	1	1
1	1	1	0	1	0	0	1
1	1	1	1	1	0	0	0

It is clear that $W = A$. For *X*, *Y* and *Z*, build Karnaugh maps to derive simplified expressions.

	C'D'	*C'D*	*CD*	*CD'*
A'B'	0	0	0	0
A'B	1	1	1	1
AB	0	0	0	0
AB'	1	1	1	1

K-map for *X*

	C'D'	*C'D*	*CD*	*CD'*
A'B'	0	0	1	1
A'B	1	1	0	0
AB	1	1	0	0
AB'	0	0	1	1

K-map for *Y*

	C'D'	*C'D*	*CD*	*CD'*
A'B'	0	1	0	1
A'B	0	1	0	1
AB	0	1	0	1
AB'	0	1	0	1

K-map for *Z*

Simplified expressions:

$$W = A$$

$$X = A'B + AB' = A \oplus B$$

$$Y = B'C + BC' = B \oplus C$$

$$Z = C'D + CD' = C \oplus D$$

Q. 4. A four-variable logic function α(*A*, *B*, *C*, *D*) is a 1 if any three or all four variables are equal to 1, and is a 0 otherwise. This is called the *majority function* of four variables. Obtain a minimum sum-of-products expression for α(*A*, *B*, *C*, *D*).

Inputs				Output
A	*B*	*C*	*D*	α
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	0
0	1	1	0	0
0	1	1	1	1
1	0	0	0	0
1	0	0	1	0
1	0	1	0	0
1	0	1	1	1
1	1	0	0	0
1	1	0	1	1
1	1	1	0	1
1	1	1	1	1

Use a Karnaugh map to derive the following expression in minimum sum-of-products form:

$$\alpha(A, B, C, D) = ABC + ABD + ACD + BCD.$$

Here is the majority function of six variables *A*, *B*, *C*, *D*, *E* and *F*:

ABCD + *ABCE* + *ABCF* + *ABDE* + *ABDF* + *ABEF* + *ACDE* + *ACDF* + *ACEF* +
ADEF + *BCDE* + *BCDF* + *BCEF* +
BDEF + *CDEF*

Q. 5. Design a minimal two-level combinational network that detects the presence of any of the six illegal code groups in the 8421 code by providing a logic-1 output.

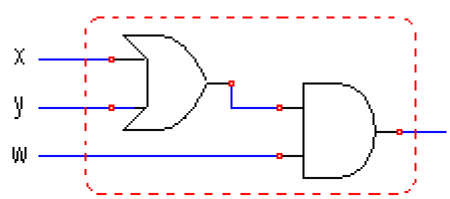
The truth table is immediate.

8	4	2	1	
w	x	y	z	f
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	0
0	1	1	0	0
0	1	1	1	0
1	0	0	0	0
1	0	0	1	0
1	0	1	0	1
1	0	1	1	1
1	1	0	0	1
1	1	0	1	1
1	1	1	0	1
1	1	1	1	1

The Karnaugh map follows.

	$y'z'$	$y'z$	yz	yz'
$w'x'$	0	0	0	0
$w'x$	0	0	0	0
wx	1	1	1	1
wx'	0	0	1	1

It is clear that $f(w, x, y, z) = wx + wy = w(x + y)$. A two-level implementation follows.



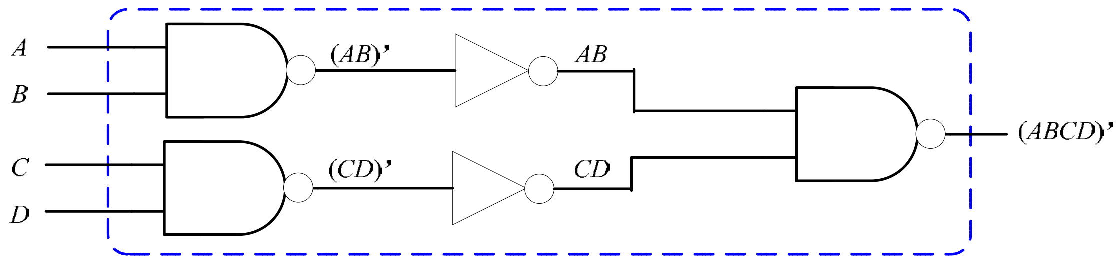
Q. 6. A 4-input NAND gate is defined by the following function:

$$f(A, B, C, D) = (ABCD)' = A' + B' + C' + D'$$

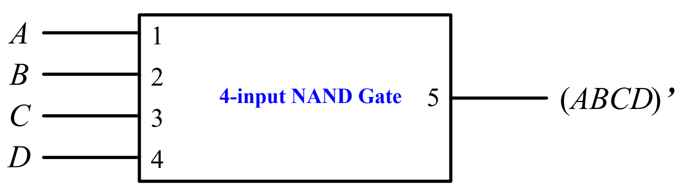
a. Design a 4-input NAND gate using 2-input NAND gates and NOT gates. (Use as few gates as possible.)
b. Design an 8-input NAND gate using two 4-input NAND gates and necessary 2-input NAND gates and NOT gates.

Note: IC-7420 houses dual 4-input NAND gates.

Here is an implementation of a 4-input NAND gate.



In order to build an 8-input NAND gate, let us first represent a 4-input NAND gate by means of the following block diagram.



An implementation of an 8-input NAND gate follows.