

CSE 140 Homework Four

February 25, 2007

*Only **Problem Set Two** will be graded. Turn in only solutions to **Problem Set Two** which will be due on Feb. 27 2007 at 1:00pm*

1 Problem Set One

- textbook 4.2(b)
- textbook 4.4(b)
- Using the map method, determine the prime implicants of 4.4(b).
- textbook 4.5(a)(d)
- textbook 4.6(a)(d)
- textbook 4.7(a)
- textbook 4.9(a)
- textbook 4.17(a)

2 Problem Set Two

1 (Karnaugh Maps & Don't cares) This question concerns the conversion between two decimal codes: **Excess-3** and **2421**, both of which are presented in the following table.

Decimal	Excess-3				2421			
	a	b	c	d	w	x	y	z
0	0	0	1	1	0	0	0	0
1	0	1	0	0	0	0	0	1
2	0	1	0	1	0	0	1	0
3	0	1	1	0	0	0	1	1
4	0	1	1	1	0	1	0	0
5	1	0	0	0	1	0	1	1
6	1	0	0	1	1	1	0	0
7	1	0	1	0	1	1	0	1
8	1	0	1	1	1	1	1	0
9	1	1	0	0	1	1	1	1

(Part A) This part concerns conversion from Excess-3 code to 2421 code. Please use Karnaugh Maps to find the **minimal Prime Implicant** Expression for the *2421* digit **x**.

(Part B) This part concerns conversion from 2421 code to Excess-3 code. Please use Karnaugh Maps to find the **minimal Prime Implicate** Expression for the *Excess-3* digit **b**.

For both of these parts, first fill in the entries for the Karnaugh map. (**Do not forget to fill in the missing row/column designators also.**) Then use the filled in Karnaugh map to derive the minimal expressions. *Hint: you can use don't-care conditions to simplify both cases.*

x =

ab\cd	00			
00				

b =

wx\yz	00			
00				

x =

b =

2 (Tabulation Method)

(Part A) State whether the following statements are True (T) or False (F), and give your reason.

- (1) A 0-subcube in G_0 can always be merged with a 0-subcube in G_1 to generate a 1-subcube.
- (2) A logic function exists such that a subcube in G_i and another subcube in G_{i+2} can be combined into a larger subcube.
- (3) No two subcubes that belong to the same group, G_i , can be combined into a larger subcube.

(Part B) The Quine McCluskey method is applied on the minterms of a 4-variable logic function. You know that the function has 8 minterms and the minimum distance between any two minterms is two. You also know that in the list of 0-subcubes, there is a single element in the G_0 group. How many elements are there in the other 0-subcube groups, namely, G_1 , G_2 , G_3 , and G_4 for this logic function?

(Part C) The Quine McCluskey method is applied on the minterms of a 4-variable function. You know that the function has 8 minterms. The Quine McCluskey algorithm terminates with four 1-subcube Prime Implicants (PIs). You also know that in the list of 0-subcubes, there is a single element in G_0 . How many elements are there in the other 0-subcube groups, namely, G_1 , G_2 , G_3 , and G_4 ?

3 You are given a 13-variable AND gate whose inputs are $x_0x_1x_2x_3x_4x_5x_6x_7x_8x_9x_{10}x_{11}x_{12}$. The arrival times of the inputs of this gate differ. The latest arriving signal is x_4 . The signals $x_6x_7x_8$ arrive before x_4 but after the remaining signals.

(Part A) Decompose the given 13-variable AND gate into 3-variable AND gates. Find an input configuration to this decomposed AND gate such that the overall delay of the system is minimum.

(Part B) How many levels in general do you need in order to decompose a m -variable AND gate into n -variable AND gates?

4 As you must have noticed from our discussion of minterms and maxterms and also from our discussion of Karnaugh maps to derive minimal sum-of-products and product-of-sums representations, every Boolean function can be expressed by just using AND, OR and NOT gates. However, in terms of gate count, the total number of gates (not counting the NOT gates) that are used in the implementation may not be minimal when compared to nonstandard representations. The use of the functions such as XOR and XNOR to generate nonstandard implementations may reduce the total gate count.

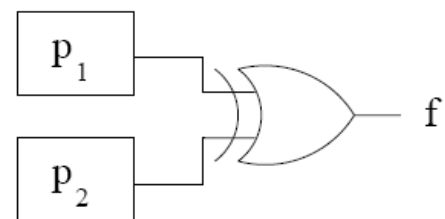
(Part A) Given the following Karnaugh map of the function, f , please derive the minimal sum-of-products representation.

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

$f =$

(Part B) For the same Karnaugh map given in (PART A), decompose the function into 2 parts, p_1 and p_2 , that are in sum-of-products form and compose the resulting parts by an XOR gate to obtain the original function as depicted in the following graph. Please identify the decomposition that requires the minimum total number of AND and OR gates.

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



Karnaugh map for p_1 :

ab \ cd	00	01	11	10
00				
01				
11				
10				

Karnaugh map for p_2 :

ab \ cd	00	01	11	10
00				
01				
11				
10				

$$f = (\quad) \oplus (\quad)$$