

CSE 140 Homework Three

October 25, 2016

*Only **Problem Set Part B** will be graded. Turn in only **Problem Set Part B** which will be due on November 2, 2016 (Wednesday) at 3:00pm.*

*Submit homework via gradescope <gradescope.com> as a **PDF**. Other formats may or may not be viewed and evaluated accurately.*

1 Problem Set Part A

All questions in this part are from Roth&Kinney, 7th Edition.

- 5.5, 5.8, 5.10, 5.11, 5.12, 5.13, 5.20, 5.28, 5.30, 5.32, 5.33, 5.34, 5.42
- 6.6, 6.9, 6.11, 6.16, 6.17, 6.24

2 Problem Set Part B

1 (Karnaugh Map Equivalence)

In this problem we explore the equivalence of K-Maps under some basic transformations. In particular we see what happens when the variables are re-positioned and when the variable values are shifted. To be a useful K-Map, Hamming adjacency of adjacent squares must be maintained; hence, transforms must preserve **Hamming adjacency** for variable value assignment.

For clarification, here is a variable re-positioning:

$x \backslash y$	0	1
0	1	1
1		

$y \backslash x$	0	1
0	1	
1	1	

And here is a variable value shifting:

$x \backslash y$	0	1
0	1	
1		1

$x \backslash y$	1	0
0		1
1	1	

(Part A) Given the following K-Maps, complete the partially filled in transformed K-Map.

$wx \backslash yz$	00	01	11	10
00		1		
01		1	1	
11				
10	1	1		

=

$xw \backslash zy$	00	01	11	10
00				
01	1			1
11				
10				1

$wx \backslash yz$	00	01	11	10
00				1
01	1	1		1
11	1	1		1
10				1

=

$wy \backslash xz$	11	01	00	10
10	1			1
11		1		
01				
00	1			1

(Part B-D) For the next parts we give you the original and the fully transformed K-Map; however, we leave out the complete definition of the transformation. Please fill in the name and/or value assignments which make the two K-Maps equal. If no such name/value assignment exists, cross off the transformed K-Map and write “INVALID” next to it.

(Part B) Here we just leave out the *variable name* definitions.

$wx \backslash yz$	00	01	11	10
00				
01		1	1	
11		1	1	1
10				

=

$\backslash$	00	10	11	01
01				
11	1	1	1	1
10			1	
00				

$wx \backslash yz$	00	01	11	10
00	1			
01	1	1	1	1
11	1			1
10				1

=

$\backslash$	01	00	10	11
11		1	1	
10			1	
00		1		
01	1	1	1	1

(Part C) Here we just leave out the *value* definitions. Remember that **hamming adjacency must be preserved** for the variable value assignment.

$wx \backslash yz$	00	01	11	10
00		1		1
01		1		1
11	1	1	1	1
10		1	1	

=

$zw \backslash xy$				
	1	1		
			1	1
	1	1	1	1
	1			1

$wx \backslash yz$	00	01	11	10
00		1		
01		1		
11	1		1	1
10		1	1	

=

$wx \backslash yz$				
		1	1	1
	1			
	1		1	
	1			

(Part D) Now both the *variable name* and most of the *value* definitions are hidden. Remember that **Hamming adjacency must be preserved** for the variable value assignment.

$wx \backslash yz$	00	01	11	10
00	1	1		
01	1	1		
11				
10				1

=

$\backslash$	00			
	1	1		
				1
00	1	1		

$wx \backslash yz$	00	01	11	10
00			1	
01	1	1	1	1
11		1		
10		1		1

=

$\backslash$			10	
01	1		1	
				1
	1	1		1
	1	1		

2. Tabulation with Subcubes

During a particularly boring 140 lecture one day, you scribble something about the Tabulation step of the Quine-McCluskey algorithm in your notes and promptly forget about it. Days later during the process of studying your notes for the midterm, you chance upon your doodles and go to your professor's office hours to see if he can figure out what your daydreams were all about.

You can make out four-variable Karnaugh maps doodled all over your notes, and having a feeling that's where it all began (because the last subject you actually remember from that lecture was Karnaugh maps), you and your professor set out to decipher the musings you wrote down that day.

The number **16** appears everywhere in your notes, and instead of just 1's and 0's in your Karnaugh maps, they have everything from **0 to 4**. Curious. Your professor quickly points out that the numbers you've written in the Karnaugh maps equal the number of 1's in the corresponding minterm; for example, the minterm 1010 would have the number 2 because there are two 1's.

For example, the Karnaugh map

$yz \backslash wx$	00	01	11	10
00	0	1		
01		2	3	
11			4	
10				

has **4** prime implicants: $w'y'z'$, $w'xy'$, $xy'z$, and wxz ; and the sum of the minterms is **10**, not 16.

(Part A) In the top right corner of your notes, you've written a large Karnaugh map with **8 individual minterms** circled. Next to the map is a sum of all of the numbers you've circled, and that sum adds up to 16.

Minterms are Column I implicants from the Tabulation method; you notice that all of the minterms you've circled are also **prime implicants**, so no Column II implicants exist. On the Karnaugh map below, circle the correct minterms and fill in the number of 1's in each minterm in the Karnaugh map. Which single logic gate implements this function?

$yz \backslash wx$	00	01	11	10
00				
01				
11				
10				

The _____ gate implements this function.

(Part B) Now that you've solved the mystery of one corner of your notes, it's time to tackle another part of your notes. The bottom half of your notes has a jumble of Karnaugh maps, each with a different configuration of exactly **4 prime implicants of 2 minterms each** (that is, Column II implicants from the Tabulation method). Your professor points out a curious property of the implicants you've circled: the numbers for the minterms within the circles still **add up to 16** in each of the Karnaugh maps. You also notice that the implicants **cannot form consensus implicants**, although in some configurations the implicants **may overlap** on certain minterms.

Three classes of configurations in particular catch your eye; one configuration where **8** minterms are circled, another where only **6** minterms are circled, and yet another where only **5** minterms are circled.

In the blank Karnaugh maps below, please circle the correct subcubes for each configuration and fill in the number of 1's for each minterm.

5 minterms					6 minterms					8 minterms				
$yz \backslash wx$	00	01	11	10	$yz \backslash wx$	00	01	11	10	$yz \backslash wx$	00	01	11	10
00					00					00				
01					01					01				
11					11					11				
10					10					10				

(Part C) You're starting to get the hang of deciphering your notes, but before you can start celebrating, your professor turns your notes over. On the back is yet another Karnaugh map, with **2 prime implicants of 4 minterms each** (in other words, Column III implicants). Once again, the circled minterms **add up to 16**, but this time, you're perplexed as to why you only wrote down one Karnaugh map. Surely there can't be only one way to choose implicants?

After running through some calculations, you realize that there is only one class of choices: the implicants are entirely **separate**; moreover, these two implicants **cannot form consensus implicants**. Please circle the two implicants in the Karnaugh map below, and fill in the Karnaugh map with the number of 1's in each of the circled minterms. Which single logic gate implements this function, and what are its inputs?

$yz \backslash wx$	00	01	11	10
00				
01				
11				
10				

The _____ gate implements this function.
Inputs for the gate:

(Part D) Finally done with your page of notes, you let out a sigh of relief. Your professor, however, keeps on scribbling, and after muttering “Aha!” under his breath, tells you that you should explore the consequences of overlapped implicants on the sums.

At that moment, two of your fellow students walk into office hours, and after getting introduced to your sum-of-16 Karnaugh maps, they can’t agree on what the lowest possible sum of two 4-minterm implicants will be. Student A says that the lowest sum is **8**, but student B says the lowest sum is **7**.

Who is correct, and how are the implicants arranged in the correct answer? What other sums are possible? Fill in the Karnaugh map below with your arrangement of **2 4-minterm implicants** with the smallest sum, write in which student is correct, and then find **all other possible sums** for two 4 minterm implicants.

$yz \backslash wx$	00	01	11	10	Student <u> </u> is correct.
00					
01					
11					
10					

Other possible sums are...

3 (Quine McCluskey)

As you recall from lecture, the Quine McCluskey algorithm involves two main steps:

1. Use the subcube table to identify the *prime implicants*.
2. Use the prime implicant table to reduce the prime implicants to the cyclic core (or even better to the null table).

In this problem, we ask you to explore these steps and extensions to these methods. For all parts, assume the input functions have **no don't cares**. Also please keep in mind that the prime implicant tables refer to boolean functions of arbitrary number of literals.

(Part A) As a first step, we have provided the following prime implicant tables with capital Roman letters defining prime implicants and lower case Greek letters defining minterms. Each table describes a potential *starting state*, i.e. no rows or columns have yet been crossed out. For each table, please say if the table is a *valid starting state*; if it is invalid, explicitly specify which aspects make it invalid.

	α	β	γ	δ	ϵ
A	X	X			
B		X	X		
C			X	X	
D				X	X

Valid?

Contradictions:

	α	β	γ	δ	ϵ	ζ
A	X	X				X
B			X	X	X	
C			X			X

Valid?

Contradictions:

	α	β	γ	δ	ϵ	ζ	ι	κ
A	X	X	X	X				
B				X	X	X	X	
C		X		X	X			X

Valid?

Contradictions:

	α	β	γ
A	X	X	
B		X	X
C	X		X

Valid?

Contradictions:

	α	β	γ	δ	ϵ	ζ
A	X	X				
B		X	X			
C			X	X		
D				X	X	
E					X	X
F	X					X

Valid?

Contradictions:

	α	β	γ	δ	ϵ
A	X	X	X	X	
B		X	X	X	X

Valid?

Contradictions:

(Part B) While studying with a study group, Joan, a student known for good ideas, proposes a new row dominance rule. This rule can be applied to the prime implicant table at any time during its reduction. She calls it the “row union domination rule”, and gives the following steps:

1. Two rows A and B can be replaced by a new row $A \cup B$ which covers all minterms covered by either A or B .
2. If this row dominates a row C , all minterms covered by C are covered by A or B , and C can thus be crossed out.
3. If a union row $A \cup B$ exists in the final cover, both A and B must participate in the final cover.

She claims this will *always* produce *correct* final covers. As an example she provides the following:

	α	β	γ	δ
A	X	X		
B			X	X
C		X	X	

	α	β	γ	δ
$A \cup B$	X	X	X	X
C		X	X	

$A \cup B$ dominates C , so the C implicant can be removed, leading to $A \cup B$, and hence A and B together providing a complete cover. This is a correct final cover since all minterms show up in either A or B . (In this case, it also happens to be a minimal cover, a point that will be explored further in (Part C).)

Do you agree that correct complete covers will **always** occur, regardless of the choice of rows to union? Please use valid examples from (Part A) to justify your result.

(Part C) Now we consider the optimality of the set of correct solutions produced using this rule. In Joan’s example, choosing A and B to union yields an optimal solution; however, applying union to A and C would yield a correct but suboptimal solution (since all 3 implicants will thus be selected for the final cover). Thus the optimality of results depends on the choice of rows you union. Do you believe all tables with more than 1 essential implicant have at least one pair of rows upon which the union row dominance rule can be applied and yield an optimal solution? Use valid tables from (Part A) to justify your result.

(Part D) Never wanting to be outdone, the Professor and TAs start conspiring to show their vast knowledge of the Quine McCluskey algorithm. They provide this rule they call *intersection column dominance* which again can presumably be invoked at any stage of the Quine McCluskey PI reduction process:

1. If two minterms α and β are jointly covered by *at least one implicant*, they can be replaced by a new column $\alpha \cap \beta$. The $\alpha \cap \beta$ has an 'x' only in the rows which have an 'x' in both the α **AND** β columns.
2. If a column γ dominates a non-empty $\alpha \cap \beta$, then γ can be crossed off.

Using Joan's earlier example, they demonstrate this rule thusly:

	α	β	γ	δ
A	X	X		
B			X	X
C		X	X	

	α	$\beta \cap \gamma$	δ
A	X		
B			X
C		X	

$\beta \cap \gamma$, α , and δ are all essential so A, B, and C are all in the final cover. This is a *correct* final cover since all minterms show up in one of these implicants. (On the other hand, this is suboptimal, but this will be explored in (Part E).)

Given the failed optimality, the professor is concerned this rule might not always produce *correct* final covers. Do you believe a correct final cover will always be produced, or are there pairs of columns with non-empty intersections which will produce incorrect covers when the intersection column dominance rule is applied? Please use examples from the valid prime implicant tables from (Part A) to justify your beliefs.

(Part E) As the earlier example demonstrates, there are pairs of columns that will produce correct but suboptimal covers. Do tables exist with pairs of columns for which application of the intersection column dominance rule will yield an optimal final cover? Please use examples from the valid prime implicant tables from (Part A) to justify your beliefs.