Lab 07 Specification – Exploring Logisim and simplification techniques using logical expressions.
Due (via your git repo) no later than 8 a.m., Friday, 9th November 2018.
50 points
1 week Individual Lab

Lab Goals

- Practice Boolean Algebra Simplification
- Practice Circuit Development using Logisim
- Practice Karnaugh Map Simplification

Suggestions for Success

- Take a look at the suggestions for successfully completing the lab assignment, which is available at:
`https://www.cs.allegheny.edu/sites/amohan/resources/suggestions.pdf`

Learning Assignment

To do well on this assignment, you should also read

- Principles of Computer Hardware by Alan Clements: Chapter 02 - 2.1 to 2.3; 2.5;
- Karnaugh Map Documentation is accessible via the following link:

  `https://drive.google.com/file/d/1cFkUehvCIady_1GcaNbNVZALckW97Rrn/view?usp=sharing`

Assignment Details

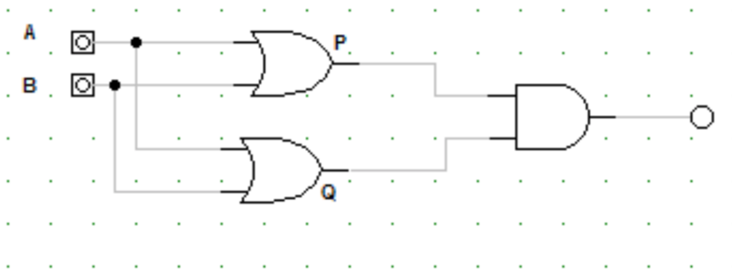
In this lab assignment, you will work on simplifying boolean expressions using Boolean Algebra [axioms, laws, identity, and negation] techniques and Karnaugh Maps. Additionally, you will learn how to use the Logisim tool to design and develop simulated digital circuits.

Part 1: Digital Circuit Development)

The basic element of a Digital Computer is a circuit. The basic elements of a circuit are gates and flip-flops. Gates are combinational logic elements, because their output depends on only the current inputs. Flip-flops are sequential logic elements, because their output depends on the current inputs and the past history. We discussed in class several fundamental gates such as; AND, OR, NOT, NAND, NOR, and XOR gates. It is important to realize how to design digital circuits by using one or more of these fundamental gates.

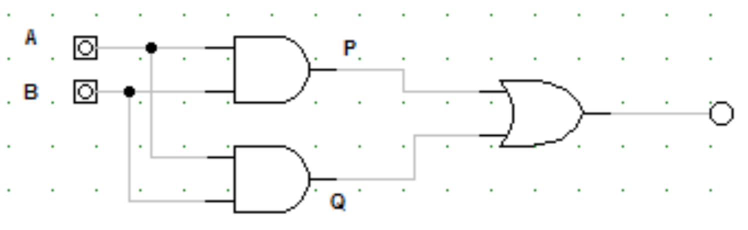
Write down the logic expressions of the circuits below, and fill out the truth tables with your expressions. Then, design the digital circuits using Logisim and verify your truth table with the results you collected from executing the circuit. Attach the screenshot of your final digital circuit that you had designed using Logisim. An example of the screenshot is listed below.

1. **(A)**



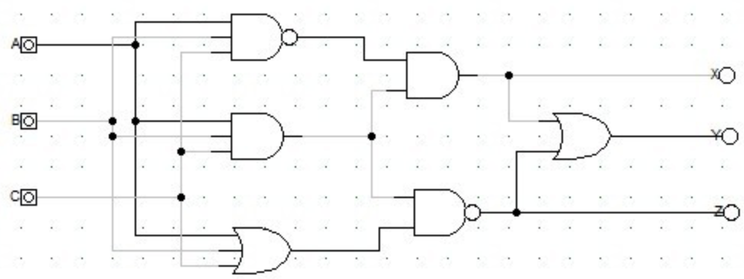
Inputs		Intermediate Values		Output
A	B	P	Q	
0	0			
0	1			
1	0			
1	1			

2. **(B)**



Inputs		Intermediate Values		Output
A	B	P	Q	
0	0			
0	1			
1	0			
1	1			

3. **(C)**



Inputs			Output		
A	B	C	X	Y	Z
0	0	0			
0	0	1			
0	1	0			
0	1	1			
1	0	0			
1	0	1			
1	1	1			

Part 2: Digital Circuit Design and Development)

Design and Develop a circuit that consists of four inputs A, B, C, D and two outputs X, Y. The output of the circuit is represented as:

```
Output = Floor (Input/4)
```

For example:

1. If input is 2 (in decimal) then output is Floor (2/4) = 0
2. If input is 4 (in decimal) then output is Floor (4/4) = 1
3. If input is 7 (in decimal) then output is Floor (7/4) = 1
4. If input is 10 (in decimal) then output is Floor (10/4) = 2
5. If input is 13 (in decimal) then output is Floor (13/4) = 3
6. If input is 15 (in decimal) then output is Floor (15/4) = 3

   To recap the circuit can generate one of the 4 distinct outputs [0,1,2,3], but it can be repeated for some inputs.

   An important part of solving this problem is that you should first draw the truth table, and then design your circuit based on analyzing the data represented in your truth table.

Part 3: Boolean Expression Simplification Using Boolean Algebra)

Simplify the following boolean expression using strictly boolean algebra approach. Do not use Karnaugh Map simplification.

1. A.$\overline{B}$.$\overline{C}$ + $\overline{A}$.$\overline{B}$.$\overline{C}$ + $\overline{A}$.B.$\overline{C}$ + $\overline{A}$.$\overline{B}$.C

   Note: The expected output after simplification is: $\overline{B}$.$\overline{C}$ + $\overline{A}$.$\overline{C}$ + $\overline{A}$.$\overline{B}$

2. A.B.C + $\overline{A}$.B.C + A.$\overline{B}$.C + A.$\overline{B}$.$\overline{C}$ + $\overline{A}$.B.$\overline{C}$ + $\overline{A}$.$\overline{B}$.$\overline{C}$

   Note: The expected output after simplification is: B.C + A.$\overline{B}$ + $\overline{A}$.$\overline{C}$

Part 4: Boolean Expression Simplification Using KMaps)

Simplify the following boolean expression using strictly Karnuagh Maps. Present clearly your KMap tables and color/shade your boxes to show the different groups that you had created.

1. $(\overline{A} \cdot \overline{B} \cdot \overline{C}) + (\overline{B} \cdot C \cdot \overline{D}) + (\overline{A} \cdot B \cdot C \cdot \overline{D}) + (A \cdot \overline{B} \cdot \overline{C})$
2. $F(A, B, C, D) = \sum(0, 1, 2, 4, 5, 6, 8, 9, 12, 13, 14)$
3. $(\overline{A} \cdot C) + (\overline{A} \cdot B) + (A \cdot \overline{B} \cdot C) + (B \cdot C)$

Submission Details

1. You should produce your solution file to all the parts, in the form of one PDF file. Other file formats such as JPEG, PNG, DOCX, submissions would not be accepted. Your solution file should preferably be in typed format. If you are unable to type your solution file (for example KMap tables), you may draw clear hand-written images and have them scanned and attached to the PDF file. If the quality of images are not clearly in a readable format, this would show negative effect to your lab grade.
2. Your lab assignment solution file must be uploaded to your repository by the due date and time.
3. Questions about the lab? Bring them to class on Tuesday morning!

Before you turn in this assignment, you must ensure that the course instructor has read access to your git repository that is named according to the convention `cs200F2018-<your user name>`.

Grading Rubric

1. If you complete Part 1 fully, as per the requirement outlined above, you will receive a total of 15 points. There will be 5 points awarded for each individual task in Part 1.
2. If you complete Part 2 fully, as per the requirement outlined above, you will receive a total of 10 points.
3. If you complete Part 3 fully, as per the requirement outlined above, you will receive a total of 10 points. There will be 5 points awarded for each individual task in Part 3.
4. If you complete Part 4 fully, as per the requirement outlined above, you will receive a total of 15 points. There will be 5 points awarded for each individual task in Part 4.

5. Failure to upload the lab assignment solution file to your git repo, will lead you to receive ”0” points given for the lab submission. In this case, there is no solid base to grade the work.
6. There will be a partial credit (30% of the points allocated for the part) given if the solution had presented all the intermediate steps correctly but unable to produce the final expected result.
7. It is highly recommended to validate if the correct version of the submission is being submitted before due date and make sure to follow the honor code policy described in the course syllabus. If it is a late submission, then it is the student’s responsibility to let the professor know about it after the final submission had been made to GitHub. In this way, an updated version of student’s submission will be used for grading. If the student did not communicate about the late submission, then automatically, the most updated version before the submission deadline will be used for grading purposes. If the student had not made any submission, then in this case, there is 0 points given to the student, automatically for the lab work.
8. If you needed any clarification on your lab grade, talk to the Professor. The lab grade may be changed if deemed appropriate.