Department of Computer Science
Bldg 315
900 Fifth Street
Nanaimo, BC
V9R 5S5
Canada

CSCI 355
Fall Term
Digital Logic and Computer Organization

Lab #4
Combinational Logic and basics of Verilog HDL

Date Due: Saturday, October 12, 11:59 pm **Total Marks: 32**

General Instructions

- **This lab assignment is individual work. You may discuss questions and problems with anyone, but the work you hand in for this assignment must be your own work. Put your name, student number and instructor's name on the cover page of your report. Failure to follow these conventions will result in a deduction of grades for you. Do not submit folders, zip documents, even if you think it will help.**
- **Assignments must be submitted to VIULearn.**
- **VIULearn will not let you submit work after the assignment deadline. It is advisable to hand in each answer that you are happy with as you go. You can always revise and resubmit as many times as you like before the deadline; only your most recent submission will be graded.**
- **Partial credit will be given as appropriate, so hand in all attempts.**

1. Objectives

i. Familiarization with Karnaugh Maps
ii. Designing a minimum sum-of-products function with Karnaugh map
iii. Designing a minimum product-of-sums function with Karnaugh map
iv. Familiarization with the basics of Verilog HDL
v. Understanding waveforms generated with input/output signals

2. Health and Safety

Any laboratory environment may contain conditions that are potentially hazardous to a person's health if not handled appropriately. Watch for posted information in and around the laboratories, and on the class website.
https://adm.viu.ca/sites/default/files/viu_safety_design_for_facilities_standard_draft.pdf

3. Background

3.1 Karnaugh Maps

Karnaugh Maps (k-maps) is the geometric representation of a logic expression that facilitates logic minimization. It is an alternative to the truth table form for representing a function. It allows for easy discovery of groups of minterms than can be combined (i.e., apply the combining property). Refer to our lecture slides (Week 4) to revise k-maps.

3.2 Three-Way Light Control

The truth table for a three-way light control which we discussed in our lecture class, where each one of three switches can turn a central light on or off, regardless of the state of the other two switches, is given in Table 3-1.

Table 3-1: Three-Way Light Control

x_1	x_2	x_3	f
0	0	0	0
0	0	1	1
0	1	0	1
0	1	1	0
1	0	0	1
1	0	1	0
1	1	0	0
1	1	1	1

4. Pre-Lab Exercises

You have until Friday to complete and submit the pre-lab exercises. You can discuss the pre-lab exercises during the in-lab session as well.

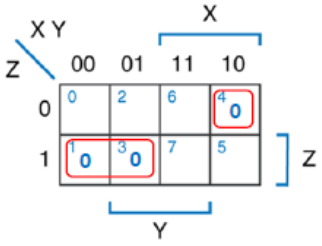
4.1 Using Karnaugh maps, find minimal sum-of-products expressions for the following logic functions. Indicate the distinguished 1-cells in each one that you do.

a. $F = X'YZ' + XZ' + XY + XZ + Y'Z$

b. $F = XY' + WY' + XZ' + WZ' + Y'Z + W'X$

4.2 In the same way that a Karnaugh map can be used to find a minimum sum-of-products, it can be used to find a minimum product-of-sums. In this case, you put only the zeroes into the Karnaugh map and follow the same rules, but then each prime implicant represents a maxterm. This approach is often useful if there are more 1s than 0s in the truth table. For example:

$F = \prod_{X,Y,Z}(1,3,4)$



$F = (X+Z') \cdot (X'+Y+Z)$

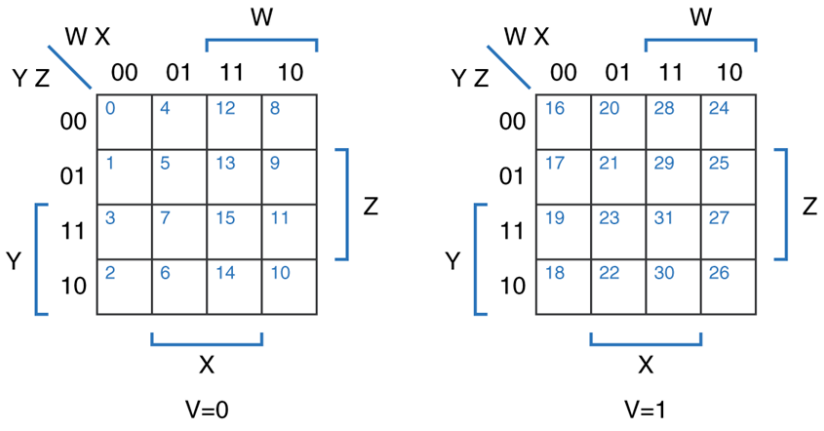
Using Karnaugh maps, find minimal product-of-sums expressions for the following logic functions. Indicate the distinguished 0-cells in each one that you do.

a. $F = \Pi_{W,X,Y,Z}(0,1,8,9,10,12,14)$

b. $F = \Pi_{W,X,Y,Z}(0,15)$

4.3 A five-variable Karnaugh map can be drawn for a 5-variable function as shown below. In each map, cells that occupy the same relative position in the V=0 and V=1 sub-maps are considered to be adjacent. Find a minimal sum of products expression for the following logic function using a 5-variable map.

$$F = \sum_{V,W,X,Y,Z}(0,7,8,9,12,13,15,16,22,23,30,31)$$



4.4 Sometimes the specification of a combinational circuit is such that its output does not matter for certain input combinations, called "don't-cares". An example would be a prime number detector for a BCD input (4-bit number but only decimals 0 to 9 are used). In this case, put "d" in the Karnaugh map for these don't-care cases (i.e., outputs related to decimal inputs 10 to 15), and if including them allows creating larger minterms, then include them, otherwise do not. And of course, never include any minterms that are all "ds".

a. Using Karnaugh maps, find a minimal sum-of-products expression for a BCD evenly divisible-by-3 detector NOT using don't-cares, represented by
   $F = \Sigma_{W,X,Y,Z}\, m(0,3,6,9)$
b. Using Karnaugh maps, find a minimal sum-of-products expression for a BCD evenly divisible-by-3 detector using don't-cares, represented by
   $F = \Sigma_{W,X,Y,Z}\, m(0,3,6,9) + \Sigma D(10,11,12,13,14,15)$.

4.4 Review lecture slides (Week 3) for Verilog HDL.

5. Equipment Required

Cub machines running Linux

6. Lab Procedure

6.1 Three-Way Light Control using structural Verilog

i. Write the structural (i.e., using gate primitives) Verilog HDL code for a three-way light control (see Section 3.2) with the module interface as shown below, implemented as a sum-of-products. Simulate in Icarus Verilog with testbench to verify operation.

```
Module threeWayLight
(
        input wire x1,
        input wire x2,
        input wire x3,
        output wire light
);
```

ii. You should include a Verilog main code, testbench, and picture of the timing diagram in your lab report.

iii. Fill out the truth table with the result from the timing diagram and comment if the waveform is as expected.

REQUIRED: Demonstrate the waveform to me to prove that your code is working. Make sure that I record that you successfully demonstrated the Three-Way Light Control circuit functioning.

6.2 Three-Way Light Control using Behavioural Verilog

i. Write the continuous behavioural assignment (i.e., using “assign”) Verilog HDL code for the same three-way light control with the module interface as shown in 7.1.

ii. You should include a Verilog main code, testbench, and a picture of the timing diagram in your lab report.

iii. Fill out the truth table with the result from the timing diagram and comment if the waveform is as expected.

REQUIRED: Demonstrate the waveform to me to prove that your code is working. Make sure that I record that you successfully demonstrated the Three-Way Light Control circuit functioning.

6.3 Discuss pre-lab tasks with your partner and demonstrate that you understood the solutions.