Your Name: _____   netid:_____   Group #:

Name: _____   netid:_____

Name: _____   netid:_____

Name: _____   netid:_____

ECE 120 Worksheet 5: From Problem Statement to Digital Circuit

Before you begin today's discussion, be sure that you are familiar with terminology, such as literals, minterms, maxterms, canonical forms, implicants, prime implicants, and K-maps. Also be sure that you know how to find canonical SOP and POS forms for Boolean functions, and that you know how to use K-maps. To check these skills, you can make up a truth table at random, find a Boolean expression for the function, then check your result by writing a truth table for your expression.

Use the area heuristic—number of literals plus the number of operators, not including complemented literals—to find minimal expressions in today's problems. Choosing a minimal number of prime implicants from a K-map will give you minimal solutions in this sense.

Truth table for Problem 1

A	B	C	D	V
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	1
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

K-map for Problem 1

V

CD \ AB	00	01	11	10
00	0	0	1	0
01	0	0	1	1
11	0	1	1	1
10	0	0	1	1

1. From Problem Statement to Digital Circuit

You may have noticed that the carry out of a one-bit adder is a majority function of the two addends and the carry in. In other words, the carry out is a 1 whenever two or more of the inputs are 1s.

Let's design a majority voting unit to calculate the outcome of a vote by a committee of four members, *A*, *B*, *C*, and *D*. For each member, a value of 1 means a yes vote, and a value of 0 means a no vote. Let's use the same convention for the vote result *V(A,B,C,D)*.

Sadly, a committee size of four can result in a tied vote, so let's say that, in such cases, the vote *V* goes with *A*'s vote.

a. Can you write down *V* as a function of *A*, *B*, *C*, and *D* by just thinking the problem through? It's ok if your answer is, "No," but think about the question.

    AB + AC + AD + BCD

b. Fill in the truth table on the previous page, then look at the truth table to answer these questions:

    i. How many product terms appear in the canonical SOP form of *V*?   **8**
        (Note: you do **not** need to write down the canonical SOP form.)

    ii. How many sum terms appear in the canonical POS form of *V*?   **8**
        (Note: Again, you do **not** need to write down the canonical POS form.)

c. Copy the bits from the truth table to the K-map. Be careful with ordering. Then use the K-map to find a minimal SOP expression for *V*.

    AB + AC + AD + BCD

d. Draw the circuit using AND and OR gates. (You should only need five gates.)
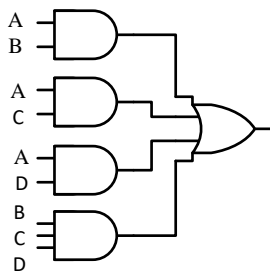
2. A Simple Thermal Controller

Prof. Lumetta needs some help with the water heater in his home. The design is as follows. The unit has a control dial that allows the user to set the desired temperature using a two-bit unsigned value C_1C_0. A temperature sensor reads the current water temperature, also as a two-bit unsigned value T_1T_0. A digital circuit, which you must design, then decides whether to turn on a heating element or not. The heating element is controlled by logic signal H, which should be set as follows:

$$H = \begin{cases} 0 & \text{if } T \geq C \\ 1 & \text{if } T < C \end{cases}$$

a. Fill in the truth table below for function H.

C_1	C_0	T_1	T_0	H
0	0	0	0	0
0	0	0	1	0
0	0	1	0	0
0	0	1	1	0
0	1	0	0	1
0	1	0	1	0
0	1	1	0	0
0	1	1	1	0
1	0	0	0	1
1	0	0	1	1
1	0	1	0	0
1	0	1	1	0
1	1	0	0	1
1	1	0	1	1
1	1	1	0	1
1	1	1	1	0
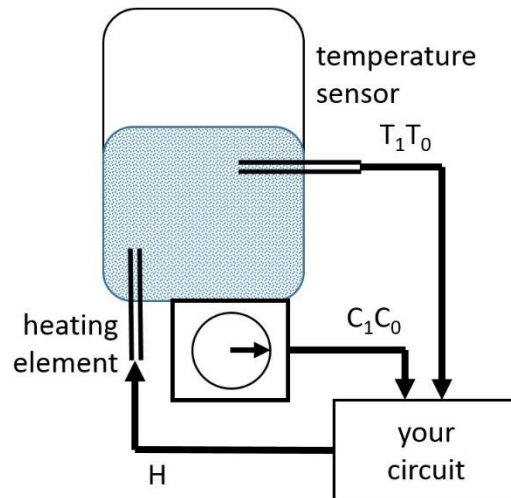
H C_1C_0

T_1T_0	00	01	11	10
00	0	1	1	1
01	0	0	1	1
11	0	0	0	0
10	0	0	1	0

b. Copy the bits from the truth table to the K-map above. Be careful with ordering.

c. Use the K-map to find a minimal SOP expression for H.

$C_1 T_1' + C_0 T_1' T_0' + C_1 C_0 T_0'$

3. Two-bit Thermal Control

Prof. Lumetta has issues. Apparently, the thermal control circuit that you designed for Part 2 today doesn't quite solve the problem. His water heater radiates too much heat. So we need to use a two-bit unsigned output H_1H_0 and keep the heat on at a low level even when the temperature reaches the desired level.

Remember that the inputs provided by the system are a control dial that allows the user to set the desired temperature using a two-bit unsigned value C_1C_0, and a temperature sensor that reads the current water temperature, also as a two-bit unsigned value T_1T_0.

Your function H should then be given by:

$$H = \begin{cases} \min(3, C - T + 1) & \text{if } T \leq C \\ 0 & \text{if } T > C \end{cases}$$

The use of the minimum function ensures that we do not try to set the heat to 4 when C is 3 and T is 0.

Fill in the K-maps below, then derive minimal expressions for H_1 and H_0. *Hints: H_1 should be quite easy. For H_0, consider both SOP and POS expressions, and use the better of the two.*

H_1

T_1T_0 \ C_1C_0	00	01	11	10
00	0	1	1	1
01	0	0	1	1
11	0	0	0	0
10	0	0	1	0

H_0

T_1T_0 \ C_1C_0	00	01	11	10
00	1	0	1	1
01	0	1	1	0
11	0	0	1	0
10	0	0	0	1