

Exp 4: Combinational Circuit Design

Engineering 357 - Digital Design Lab

Fall 2015

Farnam Adelkhani

Date Experiment Performed: 9-29-2015

Date Report Submitted: 10-1-2015

Exp. 4 Combinational Circuit Design

Abstract and Objective:

The general objective and purpose of this lab assignment is to gain further familiarity with the process of conceiving, planning and laying out a solution regarding the creating and formulation of a digital circuit design. We must begin by fully understanding the problem at hand, after the problem is thoroughly understood then we can identify the inputs and outputs of the circuit we are designing. Once the inputs and outputs are identified we can use the known relationships to represent the data in a truth table or Boolean equation.

The two methods of translating a combinational logic problem into a model are first; the method for a small circuit, constructing a truth table in which all the possible input combinations are listed. The second method is to translate the problem statement into mathematical models to express input-output relationships directly in Boolean algebra form. Sometimes the second method makes for an easier and more natural interpretation of the problem.

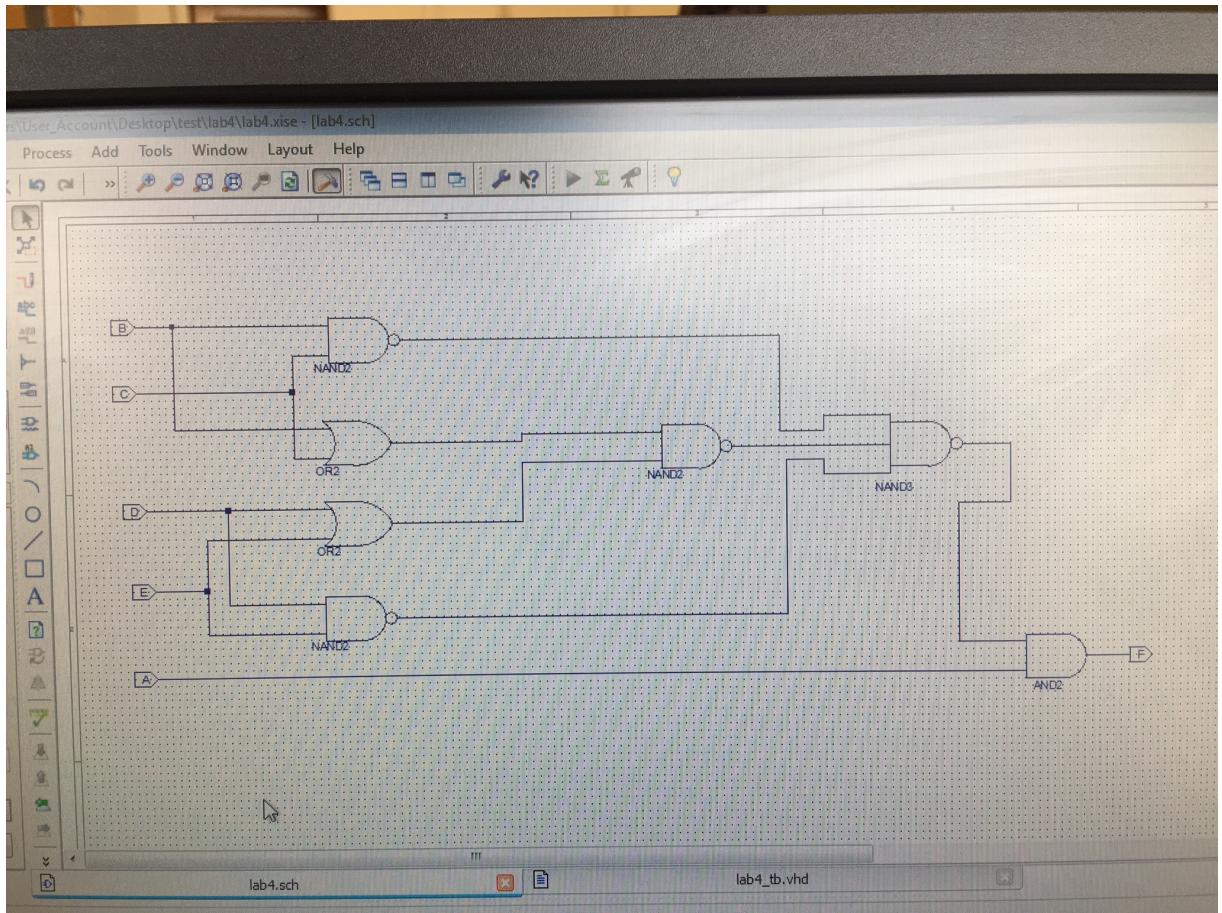
Components Used:

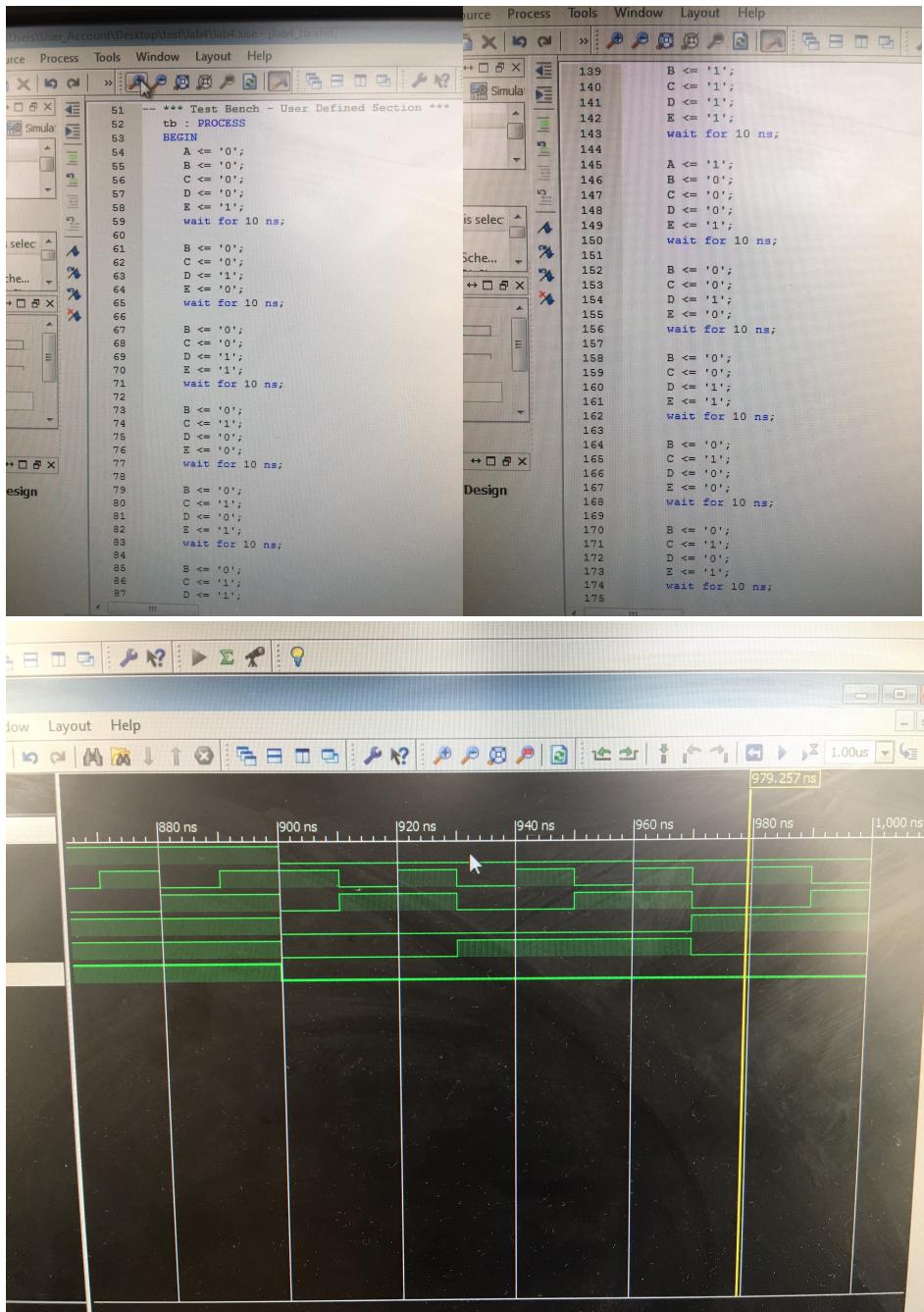
- ⊕ Xilinx software on PC
- ⊕ The following gates/integrated circuits:
 - 74LS00
 - 74LS04
 - 74LS08
 - 74LS32
 - 74LS86A

Procedures and result:

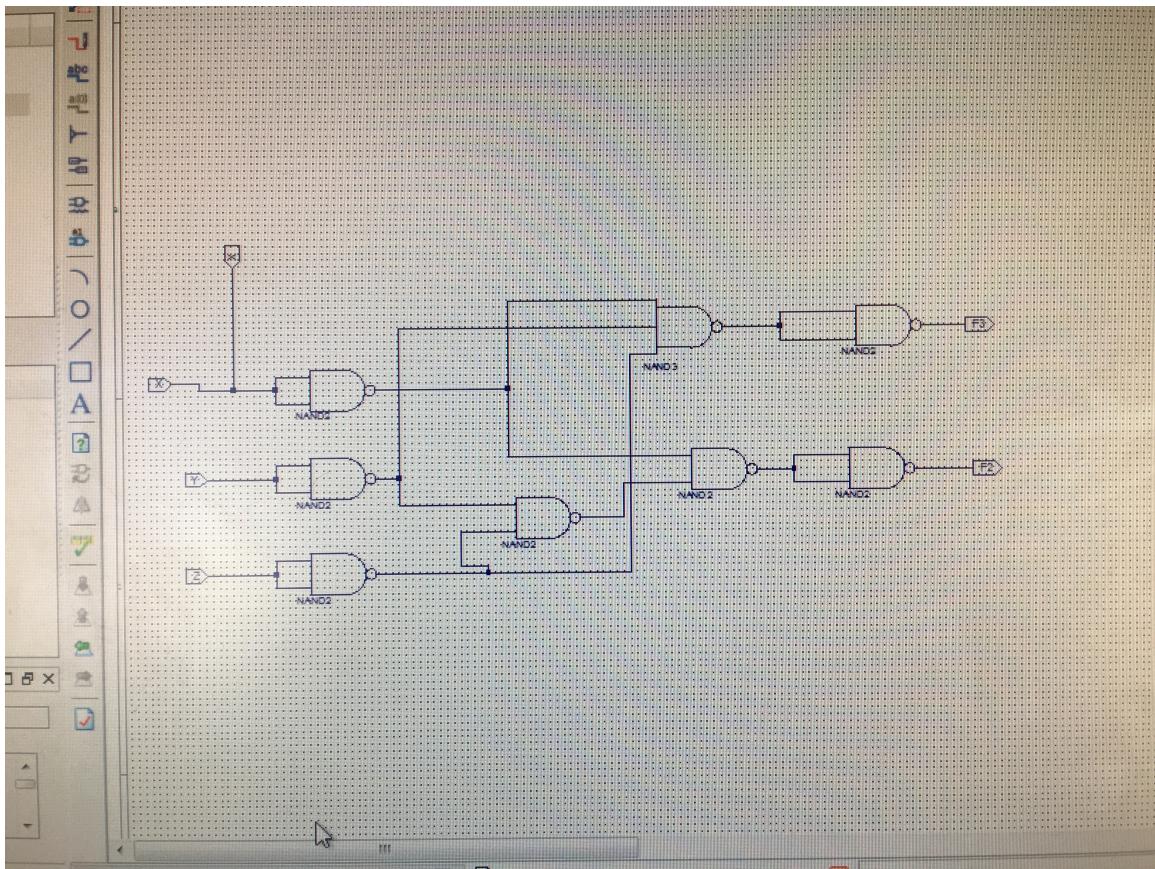
1. This experiment was started at home by completing the pre-lab assignment which included 4 questions. The goal of each question was essentially to come up with the expressions that satisfy the implied problem given to us. In problem #1 we are given a four-bit binary number representation and its grey code. We are asked to design a circuit to convert the binary values into grey code. It is also advised to the experimenter do so using XOR gates. Ultimately, the logic diagram included 3 XOR sharing the 4 inputs and outputting G_0, G_1, G_2 and G_3 .

2. The next problem statement wants the experimenter to create a three-way light control. We are to design a switching circuit that can turn a light on and off from three different switches. For this problem a truth table was setup and filled out in accordance with the given rules. We are to find the output given the information we need to determine its value at each given combination of inputs. This problem was solved by creating a k-map from the input data and using that map to come up with the simplified expression that will give us our logic diagram.
3. In problem #3 the designer is asked to create a voting machine for 5 users. One of the users however, is the chairman and she/he has veto power. In other words, the motion is only passed if two of the voters B-E agree with the chairman (A). The statement used to find our logic diagram in this problem was $F = A\{BC + (B + C)(D + E) + DE\}$, which was obtained from a k-map.





4. The last problem statement given to us is #4 which asks to create 3 different logic diagrams base on the data given in the table. We then are construct our logic diagram in Xilinx and map the outputs. We confirm that the desired outputs are being conveyed for each of the logic circuits.



Post lab and Conclusion

This was a practical and useful lab in circuit design where we learn to implement circuits that can be used to perform useful tasks, such as turning a light on via the flip of anyone of 3 light switches. Another useful implementation would be to constructing a voting machine, to be used in a town meeting or perhaps a game show. The technique of coming up with a truth table or k-map for a stated problem or given equation is the first step to completing this task. After you have a simplified equation and therefor you know the cheapest route to building the desired circuit with the minimal amount of components necessary, then work in Xilinx can begin. You can construct your logic circuit and test it in Xilinx in order to save more money in future troubleshooting costs and hardware failure expenses that may arise from poor circuitry. A well designed and implemented circuit is one that is cost effective and efficient for the task it has been designed for. It is interesting to have reached a level of engineering where we can implement systems such as the sprinklers that we grew up watching work.

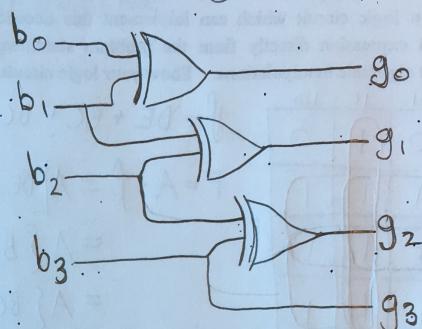
Combinational Circuit Design

$$\begin{aligned}g_3 &= b_3 \\g_2 &= b_3 \oplus b_2 \\g_1 &= b_2 \oplus b_1 \\g_0 &= b_1 \oplus b_0\end{aligned}$$

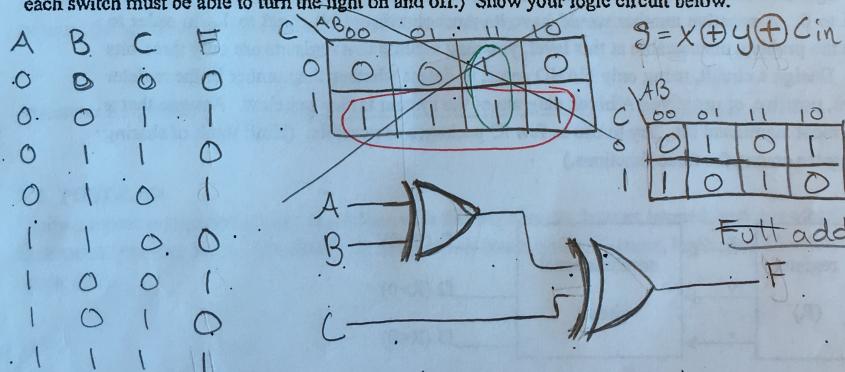
4-5

$\oplus = \text{XOR Operation}$

binary code b3 b2 b1 b0	Gray code g3 g2 g1 g0
0 0 0 0	0 0 0 0
0 0 0 1	0 0 0 1
0 0 1 0	0 0 1 1
0 0 1 1	0 0 1 0
0 1 0 0	0 1 1 0
0 1 0 1	0 1 1 1
0 1 1 0	0 1 0 1
0 1 1 1	0 1 0 0
1 0 0 0	1 1 0 0
1 0 0 1	1 1 0 1
1 0 1 0	1 1 1 1
1 0 1 1	1 1 1 0
1 1 0 0	1 0 1 0
1 1 0 1	1 0 1 1
1 1 1 0	1 0 0 1
1 1 1 1	1 0 0 0



2. Three-way light control. Design a switching circuit which can turn a light on or off from three different switches. You may assume that only one switch will be acted on at any given time. (Hint: Start with a truth table in which the light is off when all three switches are at logic-0; then determine what the outputs should be for other input combinations knowing that each switch must be able to turn the light on and off.) Show your logic circuit below.



→ should see the paper exclusive OR

3. Voting machine. A committee of five members, A - E, each controls a toggle switch. When a vote is to be taken, each member simply sets his/her toggle switch according to his/her wish: **up** (logic-1) for yes and **down** (logic-0) for no. No abstentions are permitted. A motion passes when the chairperson and two or more other members vote yes, in which case the decision light is automatically turned on. Note that the chairperson (the one who controls

4-6

So the motion is only passed when at least 2 (two) of B-E agree in conjunction w/ A (veto)

Experiment 4

switch A) has veto power. That is, a motion is defeated as long as the chairperson votes no.

Design a logic circuit which can implement this decision making task. Hint: obtain a Boolean expression directly from the problem statement and then simplify it by using Boolean algebraic manipulations. Show your logic circuit below.

DE	00	01	11	10
BC	00	00	1	0
	01	01	1	1
	11	11	1	1
	10	01	1	1

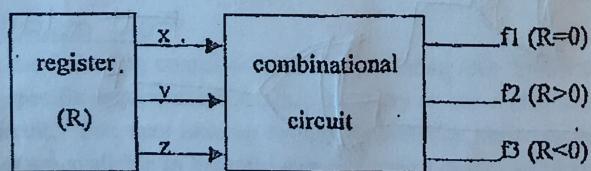
$$f = DE + BC + DC + CE + BD + BE$$

$$F = A \cdot f = A [DE + BC + DC + CE + BD + BE]$$

$$= A \{ B(C + D + E) + C(D + E) + DE \}$$

$$= A \{ BC + (B + C)(D + E) + DE \}$$

4. Status flags. Sometimes it is necessary for a digital computer to know whether a number stored in a register is zero, positive non-zero, or negative. A positive number would have its most significant bit equal to 0 and the remaining bits not all 0s. A zero would have all bits equal to 0. A negative number would have its most significant bit equal to 1. In order to make the problem manageable at this level, you may assume that registers are only three bits long. Design a circuit, using only NAND gates, to detect whether the number in the register is zero, positive, or negative. A block diagram of the system is shown below. Assume that x is the most significant bit. Try to use as few IC packages as possible. (Hint: think of sharing logic gates among the three functions.)



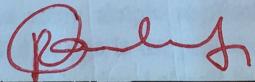
Show your circuit to generate the status flags, f1, f2 and f3, here:

See page attached @ end of report

V. LABORATORY WORK

Construct all four circuits you designed in the PRE-LAB Section, one at a time, and verify the operations of these circuits using the digital trainer. You must demonstrate the operations of one or more (as specified by your lab instructor) of these circuits to your instructor.

Instructor's signature: _____



Date: _____

09/29/15

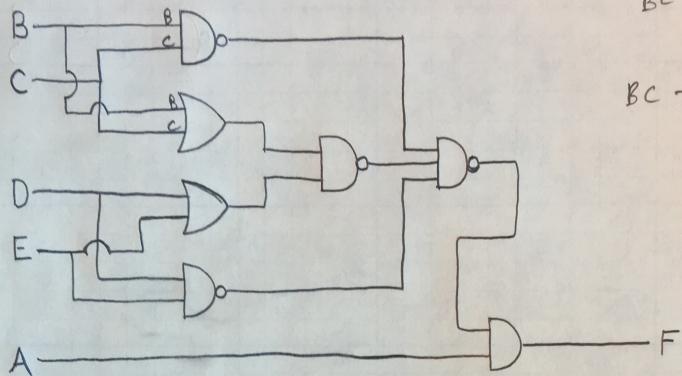
VI. POST-LAB

Write a report summarizing your experience with this experiment, lessons learned, and any other comments you may have. Also attach all design work (truth tables, K-maps, logic simplification steps, etc.).

Problems 3 & 4
Exp #4 - Pre-Lab

Faruam Adelkhan

Problem #3



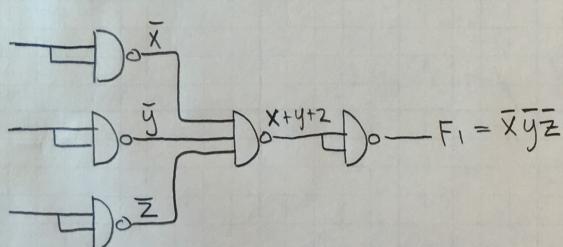
$$\overline{BC} \quad \overline{DE} \quad (\overline{B+C})(\overline{D+E})$$

$$BC + DE + (\overline{B+C})(\overline{D+E}) =$$

Problem #4

x	y	z	$F_1(R=0)$	$F_2(R>0)$	$F_3(R<0)$
0	0	0	1	1	0
0	0	1	0	0	0
0	1	0	0	0	0
0	1	1	0	0	0
1	0	0	0	0	1
1	0	1	0	0	0
1	1	0	0	0	1
1	1	1	0	0	1

$$F_1(R=0) = \overline{x}\overline{y}\overline{z} = \overline{x+y+z} \quad (\text{DeMorgan's theorem})$$



$F_3(R<0)$

$$\overline{F}_3 = X$$

$$F_2(R>0) = \overline{x} \cdot (\overline{y} + \overline{z}) = \overline{x} + \overline{y}\overline{z} \quad \leftarrow \text{DeMorgan's}$$

