CS 240 Lab 2 Assignment

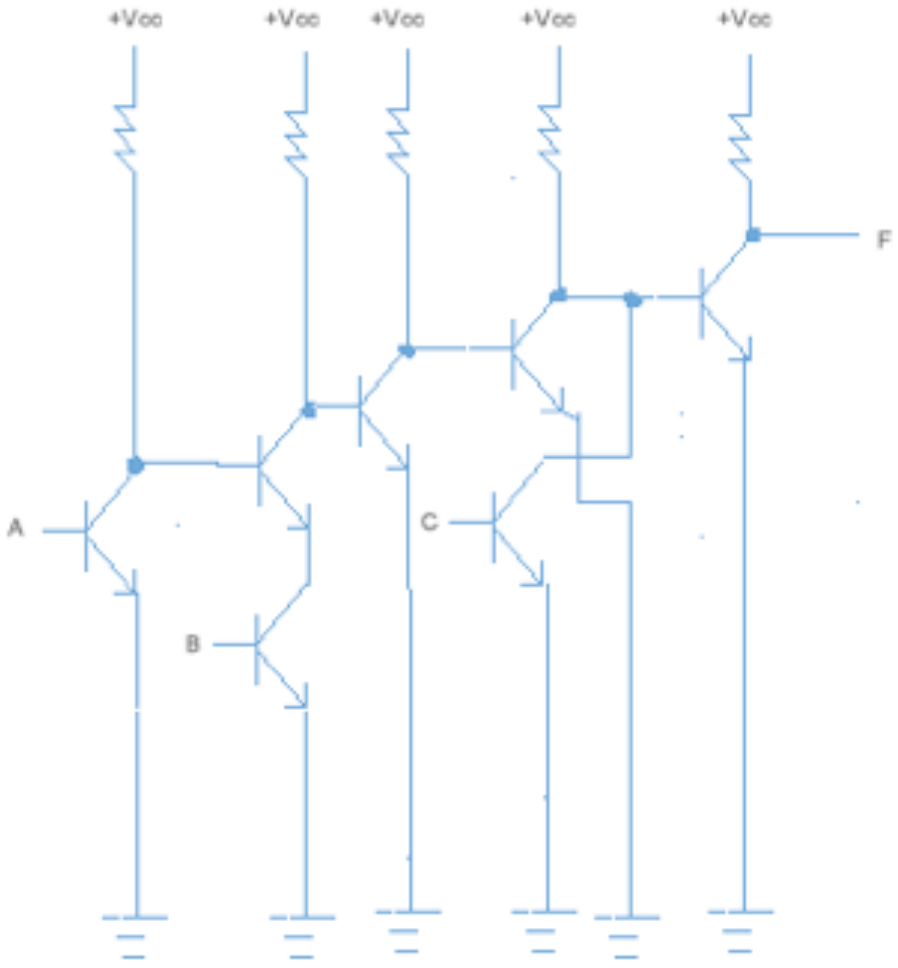
Each student should make their own copy of the following exercises, and complete on their own.

Start in lab if there is time remaining after the previous exercises are completed.

Print and hand in a hardcopy at the beginning of lab on Thursday.

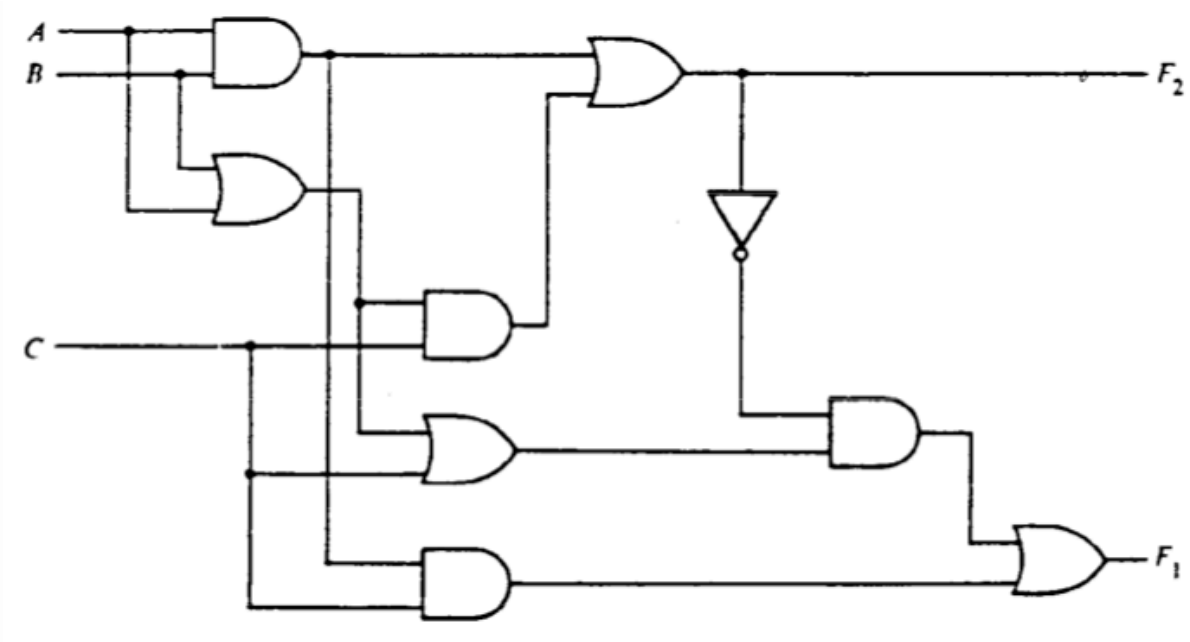
You are allowed to get help from tutors or instructors.

1. Write a boolean function for F produced by the following transistor circuit (HINT: you should try to identify subcircuits for basic gates you have seen, and work your way from left to right):



Re-draw the circuit using basic gate symbols: (okay to either sketch on paper or build in LogicWorks and paste image here):

2a. Create the boolean expressions for output functions F1 and F2 via a direct translation of the circuit (do not simplify the expressions in any way):



2b. Write truth tables for each of F1 and F2 and then use the truth tables to write a sum-of-products form for each of the functions.

3. Draw unsimplified circuits to implement the following Boolean expressions. Remember that the apostrophe ′ notation indicates logical negation of the preceding term. The apostrophe binds tightly. For example, AB′ means (A)(B′).
NOTE: you can either draw on paper or use LogicWorks to build these circuits, then paste the images here.
You may use multiple-input AND, OR, NOR, or NAND gates where they are useful (instead of using only 2-input gates)

a. (A+B)(A+B′)

b. ABC+A′B+ABC′

c. A′B′+A′BC′+(A+C′)′

4. Write a Boolean expression and draw a circuit in sum-of-products form for the output Y of this truth table. Use multiple-input AND and OR gates where useful (instead of using only 2-input gates). Do not simplify!

A	B	C		Y
0	0	0		1
0	0	1		0
0	1	0		1
0	1	1		0
1	0	0		1
1	0	1		1
1	1	0		0
1	1	1		1

5. Draw two circuits that implement two-input XOR , i.e. AB' + A'B
(In both circuits, use the smallest number of gates you can):

a. Using only 2-input NAND gates.

b. Using only 2-input NOR gates.

6. Using the laws of Boolean algebra , find the simplest equivalent of the Boolean expressions below using only products, sums, and negation (*i.e.*, no XOR). Show your derivation step by step. Label each step with the name of the rule you apply.

a. (A+B)(A+B′)

b. ABC+A′B+ABC′

c. A′B′+A′BC′+(A+C′)′

d. This one is a challenge problem.

The simplified form is A’ + B’C’ + B’D’ + BD. Show all the steps, and if you can’t quite work it out, get as far as you can.

A′B′C′D′+AB′C′+AB′CD′+ABD+A′B′CD′+BC′D+A′