COMPUTER SYSTEMS I — LAB 2
A 4-bit incrementor

For this week, we will perform a slightly different kind of arithmetic operation. This one will later be used, in Lab 3, to build a fundamentally different kind of circuit. Until then, though, this circuit has a simple definition ...

1 An incrementor

Specifically, you will design and implement a *4-bit binary incrementor*. There are 16 possible integers that can be formed with 4 bits, from 0 to 15 (or, in binary, from 0000 to 1111). Given that the incrementor should add 1 to each of these possible input values, we can make a table that shows each possible input value and its corresponding output value:

(original) input value	**(incremented) output value**
0	1
1	2
2	3
3	4
4	5
5	6
6	7
7	8
8	9
9	10
10	11
11	12
12	13
13	14
14	15
15	0

Notice that when the input is at its maximum possible value (15), the output "wraps around" to the minimum possible value (0).

2 A suggested approach

Here is a set of suggested steps that may help you both solve this problem and implement your solution as a circuit:

1. **Determine the output functions:** The output of your circuit should be a 4-bit number. Therefore, you have four output bits, each of whose value is determined by some Boolean logic function composed of the four inputs. Write out a truth table for the four input bits and their corresponding output bits, and then use the table to write the four output functions in disjunctive normal form.

2. **Simplify the output functions:** The output functions determined in the previous step are helpful, but likely more complex than necessary. Use Karnaugh maps and/or Boolean algebraic transformations to simplify those functions.

3. **Draw the circuit:** Draw a circuit that computes the four simplified output functions that you developed in the previous step. You may have to draw the circuit a few times to come up with a clean layout that is easy to read.

4. **Build the circuit:** Use the switches, LEDs, chips, and wires to implement the circuit you've drawn. Be sure to leave time for debugging!

3 Finishing up / Demonstrating your work

Every week, there are some things you'll have to do whether or not you've completed the assignment.

1. **Demonstrating your work:** If you've finished the lab, then you need to show it to the professor or the TA. *You won't get credit unless we've seen it work!* Be sure to show both the half-adder and the full-adder. If you do not complete the lab by the end of the lab session (a likely a common occurrence), then you can show your work later in the week, before the next lab.

2. **Saving your work for another day:** First, be sure to label your ETS-7000; *Unlabelled work may be dismantled!* You need not put it away, since we lack the shelving to have everyone put theirs into storage. However, someone may reasonably move your ETS-7000 during the week. If you move someone else's, please do so carefully.

3. **Cleaning up**: When you're done, put away everything and clean up your area. That is, clean up the little bits of wire and insulation, put away the tools, etc. Leave a clean workspace, or else this lab will be a disaster before long, making it a difficult place in which to get work done.

This assignment is due on Friday, September 23, at the start of the next lab session