COMPUTER SCIENCE 20, SPRING 2015
Module #8 (Logic and Computers)

Author: Charles Zhang
Reviewer: Corinne Curcie
Last modified: February 12, 2015

Executive Summary

1. Logic gates. The logic gate is a physical device performing a logic operation. List of common logic gates:

Gate	Operation
AND	$a \wedge b$
OR	$a \vee b$
NOT	$\neg a$
NAND	$\neg(a \wedge b)$
NOR	$\neg(a \vee b)$
XOR	$a \oplus b$
XNOR	$\neg(a \oplus b)$

   The gates NAND and NOR are special because they are functionally complete: each one of them is capable of representing all possible propositional formulas. This means that every logic circuit can be built using only NAND gates or using only NOR gates.

2. Binary Arithmetic. There is a close relationship between arithmetic operations (addition, subtraction,...) and logic operations (AND, OR, ...) when the numbers being manipulated are represented in binary. Computers perform arithmetic using logic circuits that are essentially physical implementations of propositional formulas.

Check-in question

1. The result of the addition $110110_2 + 10111_2$ is: (Note: the sub-script of 2 signifies that the numbers are in base 2, also known as binary. The answers below are in decimal.)

   (a) 89
   (b) 83
   (c) 77
   (d) 65
   (e) 71

In-Class Problems

1. Perform the following operations in binary:

   (a) $11011_2 + 11010_2$

   (b) $10101_2 - 1010_2$

   (c) check your answers by performing the same operations in decimal.

2. A light bulb is connected to three light switches so that:

   - When any one switch is in the on position (and the other two are in the off position), the light bulb is lit.
   - When any switch is flipped from on to off or vice versa, the light bulb goes out if it was lit or lights up if it was off.

   Construct a circuit that could connect the three switches to the bulb in this way.

3. Construct the following gates:

   (a) Using only NOT and OR gates construct an AND gate.

   (b) Using only AND, NOT, and OR gates construct an NXOR gate. What if you can only use NOT and OR gates?

4. Design a half-subtractor circuit: for inputs a, b (each a single bit), give formulas for the difference $a - b$ and the "borrow bit" c indicating whether a bit must be borrowed from the next digit.

5. **Challenge.** Construct a two-by-two bit multiplier: Design a circuit which takes in as input a_1, a_2, b_1 and b_2, and outputs a binary representation of their product. How many bits does the output need to be?