CS130 Lab Exercise #4

Purpose

This exercise exposes students to conditions in which data values – the resulting output in this case – exceed the size (i. e. the number of available bits in the Accumulator or in a given register) For this purpose, an anemic (by today's standards at least) 4-bit processor (the Intel 4004) will be emulated.

Procedure

1. Peruse and follow the steps in the companion document entitled: "Background for CS130 Lab Exercise #4 – Intel 4004.PDF". This document is posted on "Canvas" in the lab exercise section for Week #4.
2. Use – as required – the document "Instruction Format - Intel 4004.PDF" and "Intel 4004 Instructions Set.PDF", both posted on "Canvas" in the lab exercise section for Week #4.
3. Refer to the "Background" document of Step #1 above. In "Code Example #1" in the document on Page 3, what word-pair – other than one containing the word "fetch" – is used to describe the operation effected in the first line of code?
4. In "Code Example #1" in the document on Page 3, what is to be accomplished by this program? (Specify by referring to the numeric values in decimal, not hexadecimal.)
5. In "Code Example #1", into which register or registers is the output being placed?
6. In "Code Example #1", will the resulting numeric value fit into one register?
7. Modify the code used in "Code Example #1" to place the result into Register R5 and clear out any values that are stored in Registers R0 and R1.
8. Create and test a program for the 4004 processor to multiply the unsigned values `3` and `4`.
9. Create and test a program for the 4004 processor to multiply the unsigned values`13` and `13`. (I will test your program with other values – values that do not produce a product greater than 255.)
10. Create and test a program for the 4004 processor to multiply the signed values `-2` and `+3`. (I will test your program with other values – values that do not produce a product greater in magnitude than `-128`.)

Submit your report with a format something close to the following:

Example of a possible submittal format (with jocularity inserted to break the tedium!):

1. I read it and I cannot express my joy in having done so!
2. I read both but they are definitely not of Pulitzer quality!
3. 2-word name of the operation.
4. Mathematically, here is what is being done: (faux example) $A = \pi R^2$ where R = 39.37
5. The result resides in Register xyz
6. That number is too big to fit and needs to go on a diet!
7. Here's my gnu assembly language code: `. . . .`
8. Here's my assembly language code: `. . . .`
9. Here's my assembly language code: `. . . .`
10. Here's my assembly language code: `. . . .`

E-mail to RSturlaCS130@GMail.com using the 'usual and proper' naming convention.