CS 3843 Computer Organization, Fall 2013 Assignment 2 Solution

Due Friday, September 27, 2013

This assignment is due at the beginning of class on the due date. There will be a 10 percent penalty for late assignments.

Solve the following problems and hand them in at the beginning of class on the due date. You may use a basic calculator, but you must show how you got your answer.

Write out the solutions neatly. Problems should be in order. Your **name** and the assignment number should be in the upper right corner of the first sheet you hand in. Stack the pages neatly and put a single stable in the upper left corner. Make sure the staple does not obscure any of your writing.

1. (14 points)
   a. (5 points) Assume that *x*, *y*, and *z* are unsigned 12-bit integers. Suppose x = 1925, *y* = 3012, and *z* = *x* + *y*. What is the value of *z*?

      Sol: $z = (x+y) \bmod 2^{12} = 4937 \bmod 4096 = 841$

   b. (5 points) Assume that *x*, *y*, and *z* are unsigned 12-bit integers. Suppose *x* = 1925, *y* = 3012, and *z* = *x* - *y*. What is the value of *z*?

      Sol: $z = (x-y) \bmod 2^{12} = (-1087) \bmod 4096 = 4096-1087 = 3009$

   c. (4 points) Assume that *x*, *y*, and *z* are two's complement 12-bit integers. Suppose *x* = 1925, *y* = 1302, and *z* = *x* + *y*. What is the value of *z*?

      Sol: z = *x* + *y* = 1925 + 1302 = 3227 > 2047

      Let *N** = 3227 representing -*N* where *N* = 2^{12} - *N** = 4096 - 3227 = 869

      Therefore z = -*N* = -869

2. (10 points) Show how to use shifting, adding and subtracting to efficiently multiply *x* by 1273.

   Sol: 1273 = 100 1111 1001 = $2^{10} + 2^8 - 2^3 + 1$
   Therefore $1273 \times x = (x << 10) + (x << 8) - (x << 3) + x$

3. (20 points) Logical Operators in C and Bit-level Operators

   Suppose that x and y have byte values 0x3F and 0x75, respectively. Fill in the following table indicating the byte value of the different C expressions:

Expression	Value	Expression	Value
x & y	0x35	*x && y*	0x01
x | y	0x7F	*x || y*	0x01
~x |~ y	0xCA	*!x || !y*	0x00
x & ! y	0x00	*x && ~y*	0x01

4. (30 points) Shift operations (left shift and right shift - logical and arithmetic) on single-byte quantities. Each of the answers should be 8 binary digits or 2 hexadecimal digits.

   Answer the following questions for x = 0x54 (10 points)

   a) what is its binary representation?

   b) what is x << 3 in binary representation and in hexadecimal representation?

   c) what is x >> 2 (logical) in binary representation and in hexadecimal representation?

   d) what is x >> 2 (arithmetic) in binary representation and in hexadecimal representation?

   Repeat for x = 0x98 (10 points) and x= 0x45 (10 points).

   Sol:

a) 0x54= 0101 0100 b) 1010 0000 = 0xA0 c) 0001 0101 = 0x15 d) 0001 0101 = 0x15	a) 0x98= 1001 1000 b) 1100 0000 = 0xC0 c) 0010 0110= 0x26 d) 1110 0110 = 0xE6	a) 0x45= 0100 0101 b) 0010 1000 = 0x28 c) 0001 0001 = 0x11 d) 0001 0001 = 0x11

5. (30 points) Suppose we truncate a 5-bit value (represented by hex digits 0 through 1F) to a 4-bit value (represented as hex digit 0 through F). Fill in the table below showing the effect of this truncation for some cases, in terms of the unsigned and two's-complemented interpretations of those bit patterns.

Hex		Unsigned		Two's Complement	
Original (hex)	Truncated	Original (decimal)	Truncated	Original (decimal)	Truncated
0	0	0	0	0	0
3	3	3	3	2	2
11	1	11	11	11	-5
17	7	17	1	-15	1
1F	F	31	15	-1	-1