CISC 260 Machine Organization and Assembly Language

Practice Midterm Exam

This is an open-note exam. You are allowed to use notes. You are NOT allowed to use electronic devices except standard calculators.

1. [25 points] Data representations and arithmetic

a. Convert 33_{ten} into a 8-bit two's complement binary number.

0010 0001

b. What decimal number does the following two's complement 8-bit binary number represent?

1100 1010 **-54**

c. Is there an overflow for an 8-bit machine when subtracting a two's complement integer x from a two's complement integer y as given below? Show your work.

x = 1000 1011 and y = 0111 0100

-x=0111 0101

y-x=11101001

so overflow occurs

d. Show the negation of the following integer in two's complement.

X = $1101\ 0110\ 0111\ 0101_{two}$

0010 1001 1000 1011

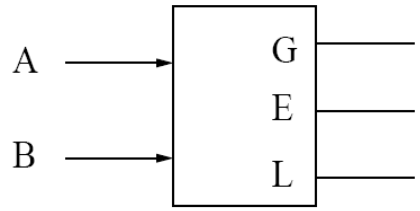
e. In multiplying the following two integers A and B, how many times the (properly shifted) multiplicand is added to the (intermediate) product C = A x B if the multiplication is implemented using the shift-add algorithm?

A = 1010 0101
B = 0110 1001

4

2. [20 points] Boolean Logic and Gates

A comparator circuit has two 1 bit inputs A and B and three 1 bit outputs G (greater), E (Equal) and L (less than)



G = 1, if A > B
0, otherwise

E = 1, if A = B
0, otherwise

L = 1, if A < B
0, otherwise

a. Fill out the truth table

A	B	G	E	L
0	0	0	1	0
0	1	0	0	1
1	0	1	0	0
1	1	0	1	0

b. Write the Boolean expression in canonical form corresponding to the above truth table

G= A&~B

E=(A&B)|(~A&~B)

L=~A&B

c. Implement the circuit by using AND, OR and NOT gates. Draw the wiring diagram.

3. [25 points] ARM Instruction set

   a. If register r4 has a value 0x f000 000c, what is the value in r0 as the result of running the following ARM assembly language program ?

```
        CMP r4, #0
        BLE L1
        MOV r5, #1
        B L2
   L1:  MOV r5, #2
   L2:  MOV r0, r5
```

   Write the value in decimal: **r0 =** **2**

   b. For the following ARM assembly code,

```
Address          code
-----------------------------------------------------------
0x0000 1000      Main: MOV r4, #5
0x0000 1004            BL FOO
0x0000 1008            SWI 0x11
0x0000 100C       FOO: MOV r5, #1
0x0000 1010        L1: CMP r4, #0
0x0000 1014            BLE L2
0x0000 1018            MUL r6, r5, r4
0x0000 101C            MOV r5, r6
0x0000 1020            SUB r4, r4, #1
0x0000 1024            B L1
0x0000 1028       L2:  MOV r0, r5
0x0000 102C            MOV pc, r14
```

   i. When the program halts, what are the values in the following registers?
   r0 = **120**
   r14 = **0x0000 1008**
   r15 = **0x0000 1008**

   ii. How many time has the instruction "MUL r6, r5, r4" been executed?

   5

   iii. What does the program compute?

   factorial

4. [30 points] ARM Assembly programming

The following is a C function that takes an integer n > 0 and returns 1 + ... + n.

```
int sum_to (int n) {
     if (n<=1) return 1;
     else
          return n + sum_to(n-1);
}
```

a) You are asked to translate the program into ARM assembly code. You may assume that n is in r0, and write the returned value in r1.

b) If n = 5, how many activation frames are pushed onto the stack during the execution of the above program.