16.482 / 16.561: Computer Architecture and Design

Spring 2014

Homework #1

Due **Friday, 2/7/13**

Notes:

- While typed submissions are preferred, handwritten submissions are acceptable.
- Any handwritten solutions that are scanned and submitted electronically must be clearly legible and combined into a single file—simply sending a picture of each scanned page is not an acceptable form of submission.
- This assignment is worth a total of 100 points.

1. (30 points) For each part of the following question, assume the following initial state. Note that your answers to each part should use the values below—your answer to part (a), for example, should not affect your answer to part (b).

   - `$t0 = 0x00000007, $t1 = 0x00000009, $t2 = 0x00100000`
   - `mem[0x100000] = 0x02732009, mem[0x100004] = 0x8899AABB`

   For each sequence of instructions below, list **all** changed registers or memory locations and their new values. When listing memory values, list the entire word—for example, if a byte is written to 0x00100000, show the values at addresses 0x00100000-0x00100003.

a.
```
addi    $t4, $t0, 4
add     $t5, $t4, $t4
sub     $t6, $t5, $t0
```

b.
```
or      $t5, $t0, $t1
andi    $t6, $t5, 0xFFF3
sll     $t7, $t5, 6
```

c.
```
lui     $s0, 0x0010
lw      $s2, 0($s0)
addi    $s3, $s2, -6
sw      $s3, 4($s0)
```

d.
```
lhu     $t4, 2($t2)
lh      $t5, 4($t2)
sll     $t6, $t4, 16
or      $t7, $t4, $t5
sw      $t7, 0($t2)
```

e.
```
slt     $s0, $t1, $t0
beq     $s0, $zero, L
add     $t0, $t0, $t1
L: add  $t3, $t0, $t0
```

2. (30 points) For the examples below, show how binary multiplication would proceed using (i) the pencil and paper method (the most basic method discussed in class) and (ii) Booth's Algorithm. **Note that, in some cases, your result for the pencil and paper method may not be correct.** For Booth's Algorithm, for each step, show what operations are performed in that step, and what the state of the product/multiplier register is at the end of each step. Assume each operand uses four bits.

a. 4 x 5
b. (-2) x 7
c. (-8) x (-3)

3. (20 points) Convert each of the following decimal values into single-precision IEEE floating-point format. Show all steps, including how you calculate the fraction and biased exponent stored in the number. *(Note: I encourage you to convert each result into hexadecimal, which will help ensure that your assignments are graded and returned relatively quickly!)*

a. 4.125
b. -75
c. 0.34375
d. -141.75
e. 16.561 (determine the closest approximation you can)

4. (20 points) Convert each of the following IEEE single-precision floating-point values into decimal values. Show all steps of your work.

f. 0x43020000
g. 0xc0f80000
h. 0x3eaaaaab
i. 0xc17e0000
j. 0xdeadbeef