Department of Electrical and Computer Engineering
University of Wisconsin – Madison

ECE 552 Introduction to Computer Architecture

In-Class Exercise (09/11)

Write all answers within the boxes provided for each question. Any response outside the designated boxes will be disregarded. Answers may be either handwritten or typed. If handwritten, ensure that your writing is clear and legible.

1. Assume you fetch one instruction from memory, shown in binary below. The most-significant bit (bit 31) is leftmost, and the least-significant bit (bit 0) is rightmost. Write down the instruction in RV32I assembly syntax (i.e., opcode, register operands).
   `00000000011000101111010000110011`

   and x8, x5, x6

   0000000 00110 00101 111 01000 0110011

2. Consider the following RV32I assembly program.

```
L0: addi $s0, $s0, 0
L1: addi $s1, $s1, 1
L2: addi $s2, $s2, 2
L3: addi $s3, $s3, 3
L4: beq  $s4, $zero, 29
```

   Fill in the blank assuming that the binary encoding of instruction L4 is as follows:

   11111110000010100000111011100001

   00000 10100 11101

3. Consider the following RV32I assembly program. Assume that there is a bug in your machine that leads to both **bits 10 and 24** of the instruction word both being **stuck at 1** for every instruction. Rewrite this program to reflect the instructions that would actually execute on your processor due to this bug.

```
L0: addi $s1, $s2, 1
L1: add  $s4, $s2, $t5
L2: ori  $s1, $s1, 34
L3: slli $s0, $t5, 8
L4: lw   $s4, 0($s0)
L5: bne  $s4, $zero, L2
L6: jal  $s4, L0
```

010 000 01001 001 0011
010 000 01001 001 0011

10 000 10100 011 0011
10 000 11100 011 0011

01 110 01001 001 0011
01 110 01001 001 0011

10 001 00100 001 0011
10 001 01100 001 0011

0 010 10100 000 0011
0 010 11100 000 0011

0 001 00011 110 0011
0 001 01011 110 0011

00000000000000000000 10100 110 1111
00000000000000000000 11100 110 1111

L0:	addi $s1, $s2, 33
L1:	add $t3, $s2, $t5
L2:	ari $s1, $s1, 1609
L3:	slli $a2, $1, 20
L4:	lw $t3, 16($s0)
L5:	lw $s4, $a6, L2
L6:	jal $t3, L0

Dont know the addy of L2

Dont know the addy of L0

We could assume the bits are mostly 1 and wont be affected