Guttu Sai Abhishek, roll number: 180050036

Computer Architecture Theory + Lab   (CS 305/341)

Assignment 4: MIPS ISA   Due Date: 22/09/20
(Theory Assignment 2)

<u>note:</u> for pseudo instructions when I write it's native instructions or native instructions's machine code, I wrote it while separating them with comma(,). The order of execution native instructions/machine code is from left to right while separated by that comma(s).

1. What are the MIPS instructions or instruction sequences corresponding to each of the following pseudoinstructions?

      subi, li, mov, la, beqz, , ble,   bleu, seq
      seq stands for "set if equal to"
   Figure these out yourself, then use the SPIM simulator to verify your answer. Note that there may be multiple answers to each of the above.
   Ans:
   subi $t1 $t2 val -> add $at $0 $0 ,lui $at val[31:16], ori $at $at val[15:0], sub $t1 $t2 $at

   li $t1 val -> if all the 16 least significant bits of val are zero then lui $t1 val[31:16]
            else if all the 16 most significant bits of val are zero then ori $t1 val[15:0]
            else lui $at val[31:16], ori $t1 $at val[15:0]
       (OR) lui $at val[31:16], ori $t1 $at val[15:0]

   mov $t1 $t2 -> addu $t1 $0 $t2

   la $t1 label -> similar to internal logic of li $t1 val where val is 32 bit address of label
             (OR) lui $at addr(label)[31:16], ori $t1 $at addr(label)[15:0]

   beqz $t1 label -> beq $t1 $0 label

   ble $t1 $t2 label -> slt $at $t2 $t1, beq $at $0 label

   bleu $t1 $t2 label ->sltu $at $t2 $t1, beq $at $0 label

   seq $t1 $t2 $t3 -> beq $t2 $t3 label1
                ori $t1 $0 0
                beq $0 $0 label2
                 label1: ori $t1 $0 1
                 label2: <next instruction to seq>
             <u>note:</u> label1, label2 numeric values specifically are 12, 8(number of bytes to
   jump in the user text segment) and are 3,2(number of instructions to jump inside the instruction
   encoding)

2. What is the machine code corresponding to each of the following instructions/pseudoinstructions?

(Answer should be in hex).

```
        sub   $t0, $t7, $s5
        andi  $5, $s5, 89
        sll   $s4, $s4, 3
        bge   $s4, $t1,  300
        lb    $s0, 100($t1)
```

Figure these out yourself, then use the SPIM simulator to verify your answer.

Ans: The order(serial number 1 to 5) in which I am writing solution below is in the same order of the instructions in question above.

1) 0x01f54022

2)0x32a50059

3)0x0014a0c0

4)0x0289082a, 0x1020004b

5)0x81300064

the below part is for serial number 4:

explanation 1:

(slt $1 $s4 $t1, beq $t1 $0 300 are the native instructions of bge $s4 $t1 300) Now for the second native instruction I have used 75(=300/4) while writing machine code. This is because if 300 appear(in the end of beq instruction) in user text segment in qtspim then 300/4 will be used in encoding of that instruction(beq) in qtspim. ie., qtspim takes 300, when it appears in user text segment, as number of bytes difference from PC to target address and qtspim takes that 300 and convert it to 300/4 = 75(number of instruction from PC to target instruction) while encoding in machine code.

explanation 2:

For <beq $rs $rt imm> the encoding is

<opcode_for_beq><rs_ register_number><rt_register_number><imm/4's_value>. (note the order of rs, rt)

This reasoning is based on my observations.(explanation 1,2 are not exlusive ones. They have to combined to explain my answer)

3. Study the following program carefully, then answer the questions below.

```
        .data
arr:    .space 100

        .text
        .globl main
main:   li      $t0, 0
        li      $t1, 0
        li      $t4, 0
        li      $t5, 4
        li      $s0, 1
        li      $s1, 1
```

```
        li      $s3, 6
        sw      $s1, arr($t1)
go:     addi    $t1, 4
        sw      $s1, arr($t1)
        addi    $t1, 4
        sw      $s1, arr($t1)
here:   addi    $t1, 4
        lw      $t6, arr($t4)
        lw      $t7, arr($t5)
L1:     add     $t6, $t6, $t7
        sw      $t6, arr($t1)
        addi    $t4, 4
        addi    $t5, 4
        addi    $t0, 1
        bne     $t0, $s0, here
L2:     addi    $s0, $s0, 1
        addi    $t4, 4
        addi    $t5, 4
        li      $t0, 0
        bne     $s0, $s3, go
        j       $ra
```

- The machine code corresponding to the instruction at label L1 is

  _____0x01cf7020_____   (put answer in hex) .

- The number of times the instruction at label L1 is executed is _____15_____ .

- The number of times the instruction at label L2 is executed is _____5___ .

- Upon program termination, the content of array, arr is

  Ans: 1,1,1,2,1,1,3,3,1,1,4,6,4,1,1,5,10,10,5,1,1,6,15,20,15 from index 0 to 24 (in decimal) (considering as word)

  ___ ___ ___ ___ ___ ___ ___ ___ ___ ___ ___ ___ ___ ___ ___

  ___ ___ ___ ___ ___ ___ ___ ___ ___ ___ ___ ___ ___ ___ ___

- The content of register t4 is _____80(in decimal)_____ .

- The content of register t6 is _____6(in decimal)_____ .

Figure these out yourself, then use the SPIM simulator to verify your answer