Name: Ningyuan Zhang

Section: K

Instructor: Joseph Zambreno

Date: 9/10/2016

Home Work #2 Solutions

1).

(a).

Register file will be slower.

More registers must be saved into stack at context switch

The number of bits in a register field will be increased. It’s not allowed in MIPS.

(b).

Because MIPS includes add immediate, and immediates can be both positive or negative, subtract immediate can be redundant.

(c).

ABI is the application binary interface.

ISA is the interface between the hardware and software.

ABI consists of ISA and the OS interface. $zero is part of ISA as well as part of ABI, and it is a hardware feature. For $sp, which is the stack pointer, is part of the ABI, but not part of ISA, and it is a compiler feature. Different register can be used as the stack pointer with different compiler.

$at is used by assembler in certain code sequences.

2.

(a).

Loop: sll $t1, $s3, 2 # temp reg t1 = i \*4

add $t1, $t1, $s6 # t1 = address of save[i]

lw $t0, 0($t1) # temp reg t0 = save[i]

bne $t0, $s5, Exit # goto Exit if save[i] != k

add $s3, $s3, 1 # loop body: i = i + 1

j Loop # repeat the loop

9 instructions are executed.

(b).

All of them are I format.

Instr Opcode Rs Rt Addr/immed

addi 8 0 $v0 0

lw 35 $a0 $v1 0

sw 43 $a1 $v1 0

addi 8 $a0 $a0 4

addi 8 $a1 $a1 4

beq 4 0 $v1 loop

3.

(a).

dsrl $s0 $s1 # shift two registers right one position

(b).

swap: xor $0, $0, $1

xor $1, $0, $1

xor $0, $0, $1

(c).

I format

Loop: slt $t3, $0, $t2

beq $t3, $0, Exit

sub $t2, $t2, 1

j Loop

Exit: