Computer Architecture Lab Report Week 4

Full name: On Quang Tung

Student ID: 20226096

Assignment 1

- TH1: Khởi tạo $s1 = $s2 = 100(0x00000064)

```
#Laboratory Exercise 4, Home Assignment 1
.text

li $s1 100
li $s2 100

start:
li $t0,0 #No Overflow is default status
addu $s3,$s1,$s2 # s3 = s1 + s2
xor $t1,$s1,$s2 #Test if $s1 and $s2 have the same sign
bltz $t1,EXIT #If not, exit
slt $t2,$s3,$s1
bltz $s1,NEGATIVE #Test if $s1 and $s2 is negative?
beq $t2,$zero,EXIT #s1 and $s2 are positive
 # if $s3 > $s1 then the result is not overflow
j OVERFLOW
NEGATIVE:
bne $t2,$zero,EXIT #s1 and $s2 are negative
 # if $s3 < $s1 then the result is not overflow
OVERFLOW:
li $t0,1 #the result is overflow
EXIT:
```

➔ Chương trình chạy đúng kết quả mong đợi: $t0 = 0 (không OVERFLOW)

- TH2: Khởi tạo $s1 = $s2 = -100(0xffffff9c)

```
#Laboratory Exercise 4, Home Assignment 1
.text

li $s1 -100
li $s2 -100

start:
li $t0,0 #No Overflow is default status
addu $s3,$s1,$s2 # s3 = s1 + s2
xor $t1,$s1,$s2 #Test if $s1 and $s2 have the same sign
bltz $t1,EXIT #If not, exit
slt $t2,$s3,$s1
bltz $s1,NEGATIVE #Test if $s1 and $s2 is negative?
beq $t2,$zero,EXIT #s1 and $s2 are positive
 # if $s3 > $s1 then the result is not overflow
j OVERFLOW
NEGATIVE:
bne $t2,$zero,EXIT #s1 and $s2 are negative
 # if $s3 < $s1 then the result is not overflow
OVERFLOW:
li $t0,1 #the result is overflow
EXIT:
```

➔ Chương trình chạy đúng kết quả mong đợi: $t0 = 0 (không OVERFLOW)

- TH3: Khởi tạo $s1 = 2147483647(0x7fffffff), $s2 = 1(0x00000001)

```
#Laboratory Exercise 4, Home Assignment 1
.text

li $s1 2147483647
li $s2 1

start:
li $t0,0 #No Overflow is default status
addu $s3,$s1,$s2 # s3 = s1 + s2
xor $t1,$s1,$s2 #Test if $s1 and $s2 have the same sign
bltz $t1,EXIT #If not, exit
slt $t2,$s3,$s1
bltz $s1,NEGATIVE #Test if $s1 and $s2 is negative?
beq $t2,$zero,EXIT #s1 and $s2 are positive
 # if $s3 > $s1 then the result is not overflow
j OVERFLOW
NEGATIVE:
bne $t2,$zero,EXIT #s1 and $s2 are negative
 # if $s3 < $s1 then the result is not overflow
OVERFLOW:
li $t0,1 #the result is overflow
EXIT:
```

➔ Chương trình chạy đúng kết quả mong đợi: $t0 = 1 (OVERFLOW)

- TH4: Khởi tạo $s1 = -2147483648(0x80000000), $s2 = -1(0xffffffff)

```
#Laboratory Exercise 4, Home Assignment 1
.text

li $s1 -2147483648
li $s2 -1

start:
li $t0,0 #No Overflow is default status
addu $s3,$s1,$s2 # s3 = s1 + s2
xor $t1,$s1,$s2 #Test if $s1 and $s2 have the same sign
bltz $t1,EXIT #If not, exit
slt $t2,$s3,$s1
bltz $s1,NEGATIVE #Test if $s1 and $s2 is negative?
beq $t2,$zero,EXIT #s1 and $s2 are positive
 # if $s3 > $s1 then the result is not overflow
j OVERFLOW
NEGATIVE:
bne $t2,$zero,EXIT #s1 and $s2 are negative
 # if $s3 < $s1 then the result is not overflow
OVERFLOW:
li $t0,1 #the result is overflow
EXIT:
```

➔ Chương trình chạy đúng kết quả mong đợi: $t0 = 1 (OVERFLOW)

- TH5: Khởi tạo $s1 =100(0x00000064), $s2 = -100(0xffffff9c)

```
#Laboratory Exercise 4, Home Assignment 1
.text

li $s1 100
li $s2 100

start:
li $t0,0 #No Overflow is default status
addu $s3,$s1,$s2 # s3 = s1 + s2
xor $t1,$s1,$s2 #Test if $s1 and $s2 have the same sign
bltz $t1,EXIT #If not, exit
slt $t2,$s3,$s1
bltz $s1,NEGATIVE #Test if $s1 and $s2 is negative?
beq $t2,$zero,EXIT #s1 and $s2 are positive
 # if $s3 > $s1 then the result is not overflow
j OVERFLOW
NEGATIVE:
bne $t2,$zero,EXIT #s1 and $s2 are negative
 # if $s3 < $s1 then the result is not overflow
OVERFLOW:
li $t0,1 #the result is overflow
EXIT:
```

➔ Chương trình chạy đúng kết quả mong đợi: $t0 = 0 (không OVERFLOW)

Assignment 2

```
.text
 li $s0, 0x12345678
 andi $t0, $s0, 0xff000000 #mask to extract MSB
 srl $t0, $t0, 24 #shift right 24 bits
 andi $t1, $s0, 0xffffff00 #Clear LSB
 ori $t2, $s0, 0x000000ff #LSB of $s0 turns into bits of 1
 andi $t3, $s0, 0
```

- Trích xuất bits MSB của $s0: Kết quả lưu ở $t0
- Clear LSB của $s0: Kết quả lưu ở $t1
- Biến LSB của $s0 thành dãy bit 1: Kết quả lưu ở $t2
- Clear $s0: Kết quả lưu ở $t3

Assignment 3

a) abs $s0, $s1

```
sra $at, $s1, 0x0000001f
xor $s0, $at, $s1
subu $s0, $s0, $at
```

b) move $s0, $s1

```
addu $s0, $0, $s1
```

c) not $s0, $s1

```
nor $s0, $s1, $0
```

d) ble $s1, $s2, label

```
slt $at, $ s2, $s1
beq $at, $0, label
```

Assignment 4

```
.text
li $s1, 0x7fffffff
li $s2, 1

START:
li $t0, 0 #default status: No OVERFLOW
addu $s3, $s1, $s2 # $s3 = $s1 + $s2

xor $t1, $s1, $s2 #Test if $s1 and $s2 have the same sign
bltz $t1, EXIT #If not, exit

xor $t2, $s1, $s3 #Test if $s1 and $s3 have the same sign
bgtz $t2, EXIT #If yes, EXIT

OVERFLOW:
li $t0,1 #the result is overflow

EXIT:
```

- TH1: Khởi tạo $s1 = 2147483647(0x7fffffff), $s2 = 1(0x00000001)

➔ Kết quả khớp với mong đợi

- TH2: Khởi tạo $s1 = -2147483648(0x80000000), $s2 = -1(0xffffffff)

➔ Kết quả khớp với mong đợi

- TH3: Khởi tạo $s1 =100(0x00000064), $s2 = -100(0xffffff9c)

➔ Kết quả khớp với mong đợi

Assignment 5

```
.text
li $s0, 6 # a = 6
li $s1, 2 # b = 8
andi $t1, $s1, 0xffffffff # tmp = 8
li $t2, 1 # value 1
andi $s2, $s0, 0xffffffff # res = $s0

loop:
beq  $t1, $t2, endloop # endloop if tmp = 1
srl $t1, $t1, 1 # temp/=2
sll $s2, $s2, 1 # res*=2
j loop
endloop:
```

- TH1: Khởi tạo $s0 =6, $s2 = 8, với kết quả mong đợi $s2 = 48(0x00000030)

➔ Kết quả khớp với mong đợi

- TH2: Khởi tạo $s0 =6, $s2 = 2, với kết quả mong đợi $s2 = 12(0x0000000c)

➔ Kết quả khớp với mong đợi