**­­­­Computer Architecture Lab Report Week 4**

**Full name: Nguyễn Hồng Phúc**

**Student ID: 20225659**

Assignment 1

-TH1: $s1=0x3; $s2=0x6

.text

addi $s1,$zero,3

addi $s2,$zero,6

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

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:

=> $t0 =0, tràn số không xảy ra

-TH2: $s1=0x7ffffffff; $s2=0x1

.text

addi $s1,$zero,2147483647

addi $s2,$zero,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

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:

=> $t0 =1, tràn số xảy ra

-TH3: $s1=0x80000000; $s2=0x10

.text

addi $s1,$zero,-2147483648

addi $s2,$zero,10

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

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:

=> $t0=0, không có hiện tượng tràn số với 2 số trái dấu nhau

-TH4: $s1=0xfffffc18; $s2=0xfffffffe

.text

addi $s1,$zero,-1000

addi $s2,$zero,-2

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

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:

=> $t1=0, không xảy ra tràn số

-TH5: $s1=0xfffffffe; $s2=0x7ffffffe

.text

addi $s1,$zero,-2147483648

addi $s2,$zero,-2

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

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:

=>$t0=1, có hiện tượng tràn số xảy ra với hai số âm quá lớn

Assignment 2

.text

li $s0, 0x12345678 #khoi tao $s0

andi $s1,$s0,0xff000000

sra $s2,$s1,24 #Extract MSB of $s0

andi $s3,$s0,0xffffff00 #Clear LSB of $s0

andi $s4,$s0,0x000000ff #Set LSB of $s0

xor $s5,$s0,$s0 #Clear $s0

Assignment 3

a. sra $at, $s1, 0x1f

xor $s0, $s1, $s1

subu $s0, $s0, $at

b. addu $s0, $zero, $s1

c. nor $s0, $s1

d. slt $at, $s2, $s1

beq $at, $zero, label

Assignment 4

**TH1: $s1,$s2 cùng dấu nhưng không xảy ra hiện tượng tràn số**

.text

start:

addi $s1,$zero,1000

addi $s2,$zero,2000

li $t0,0 #Ket qua $t0 = 0 neu khong tran so

addu $s3, $s1, $s2 # s3 = s1 + s2

xor $t1, $s1, $s2 #Kiem tra xem $s1 va $s2 co cung dau khong

bltz $t1, EXIT #Neu $t1 < 0, exit

xor $t2, $s3, $s1 #Kiem tra xem $s1 va $s3 co cung dau khong

bgtz $t2, EXIT #Neu $t2 > 0, exit

j OVERFLOW

OVERFLOW:

li $t0,1 #Neu tran so, ket qua $t0 = 1

EXIT:

=> $t0=0, không xảy ra hiện tượng tràn số

**TH2: 2 số dương, xảy ra hiệnt tượng tràn số**

.text

start:

addi $s1,$zero,1000000000

addi $s2,$zero,2000000000

li $t0,0 #Ket qua $t0 = 0 neu khong tran so

addu $s3, $s1, $s2 # s3 = s1 + s2

xor $t1, $s1, $s2 #Kiem tra xem $s1 va $s2 co cung dau khong

bltz $t1, EXIT #Neu $t1 < 0, exit

xor $t2, $s3, $s1 #Kiem tra xem $s1 va $s3 co cung dau khong

bgtz $t2, EXIT #Neu $t2 > 0, exit

j OVERFLOW

OVERFLOW:

li $t0,1 #Neu tran so, ket qua $t0 = 1

EXIT:

**TH3: 2 số âm, xảy ra hiện tượng tràn số**

.text

start:

addi $s1,$zero,-1000000000

addi $s2,$zero,-2000000000

li $t0,0 #Ket qua $t0 = 0 neu khong tran so

addu $s3, $s1, $s2 # s3 = s1 + s2

xor $t1, $s1, $s2 #Kiem tra xem $s1 va $s2 co cung dau khong

bltz $t1, EXIT #Neu $t1 < 0, exit

xor $t2, $s3, $s1 #Kiem tra xem $s1 va $s3 co cung dau khong

bgtz $t2, EXIT #Neu $t2 > 0, exit

j OVERFLOW

OVERFLOW:

li $t0,1 #Neu tran so, ket qua $t0 = 1

EXIT:

Assignment 5

.text

addi $s0, $zero, 30 #Dua so bi nhan vao thanh ghi $s0

addi $s1, $zero, 2048

addi $t0, $zero, 1 #Cai dat thanh $t0 co gia tri 1

loop:

beq $s1, $t0, exit #Neu $s1 chi con gia tri la 1 thi ket thuc vong lap

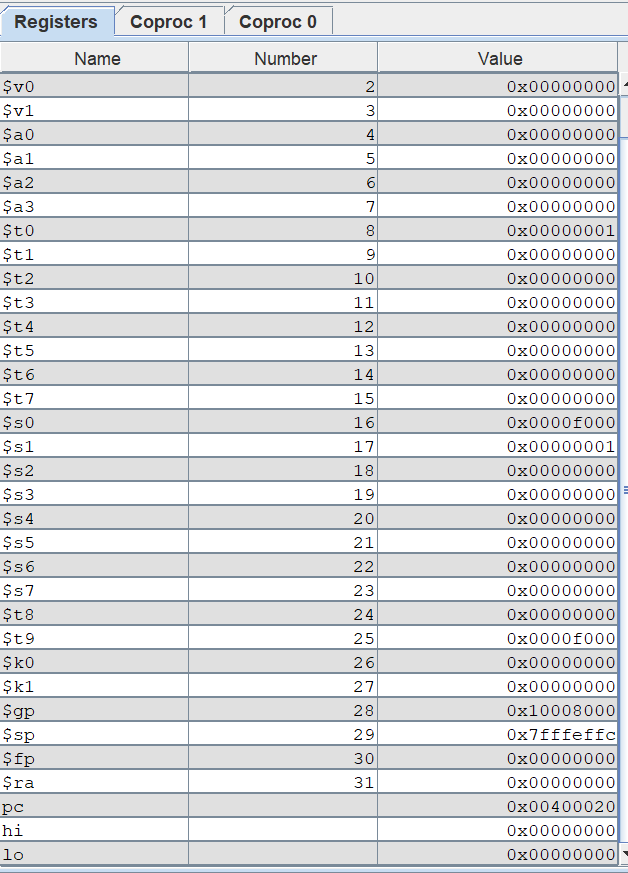
sll $s0, $s0, 1 #Tang gia tri thanh ghi $s0 len 2 lan

srl $s1, $s1, 1 #Giam gia tri thanh ghi $s1 di 2 lan

j loop #Lap lai

exit:

add $s5, $zero, $s0 #Luu ket qua vao thanh ghi $s5



Kết quả cho thấy đúng với lý thuyết.