Báo cáo thực hành KTMT tuần 4

Họ và tên: Đỗ Gia Huy

MSSV: 20215060

Assignment 1

TH1: $s1=0x9 ,$s2=0x8

.text

start:

```
        addi        $s1,$zero,9

        addi        $s2,$zero,8

        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$s3>$s1then 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 tràn số

TH2: $s1=0x7fffffff ,$s2=0x11

.text

start:

  addi $s1,$zero,2147483647

  addi $s2,$zero,17

  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$s3>$s1then 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

TH3: $s1=-5 ,$s2=6

.text

start:

  addi $s1,$zero,-5

  addi $s2,$zero,6

  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$s3>$s1then 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 trản số

TH4: $s1=-5 ,$s2=-6

.text

start:

```
        addi        $s1,$zero,-5

        addi        $s2,$zero,-6

        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$s3>$s1then 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 tràn số

TH5: $s1=0x80000001,$s2=-6

.text

start:

      addi $s1,$zero,-2147483647

      addi $s2,$zero,-6

      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$s3>$s1then 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

Assignment 2

Trích xuất 8 bit đầu (Kết quả ở thanh $t0)

Xoá 8 bit cuối (Kết quả ở thanh $t1)

Biến 8 bit cuối thành 1 (Kết quả ở thanh $t2)

Xoá sạch $s0

Code:

.text

```
li      $s0, 0x12345678
andi    $t0,$s0,0xff000000
srl     $t0,$t0,24
andi    $t1,$s0,0xffffff00
ori     $t2,$s0,0x000000ff
andi    $s0,$s0, 0
```

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

text

start:

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

      li $s1, 0x7fffffff

      li $s2, 6

      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

Code (Xét trường hợp $s0 = 11 va $s1 = 2048 = 2^11:

.text

```
        addi  $s0, $zero, 11      #Dua so bi nhan vao thanh ghi $s0

        addi  $s1, $zero, 2048    #Dua so nhan vao thanh ghi $s1

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

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

        sla   $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   $t9, $zero, $s0     #Luu ket qua vao thanh ghi $t9
```

⇨ Kết quả sau khi chạy code này:

$v1	3	0x00000000
$a0	4	0x00000000
$a1	5	0x00000000
$a2	6	0x00000000
$a3	7	0x00000000
$t0	8	0x00000001
$t1	9	0x00000000
$t2	10	0x00000000
$t3	11	0x00000000
$t4	12	0x00000000
$t5	13	0x00000000
$t6	14	0x00000000
$t7	15	0x00000000
$s0	16	0x00005800
$s1	17	0x00000001
$s2	18	0x00000000
$s3	19	0x00000000
$s4	20	0x00000000
$s5	21	0x00000000
$s6	22	0x00000000
$s7	23	0x00000000
$t8	24	0x00000000
$t9	25	0x00005800
$k0	26	0x00000000
$k1	27	0x00000000
$gp	28	0x10008000
$sp	29	0x7fffeffc
$fp	30	0x00000000
$ra	31	0x00000000
pc		0x00400020

⇨ **Đúng với lý thuyết**