Báo cáo thực hành Kiến trúc máy tính

Assigment 1:

week4_a1.asm

```
1  #Laboratory Exercise 4, Home Assignment 1
2  .text
3  start:
4          addi   $s1,$zero, -12      #s1 = -4
5          addi  $s2,$zero, 200       #s2 = 10
6
7          li       $t0,0             #No Overflow is default status
8          addu  $s3,$s1,$s2          # s3 = s1 + s2
9          xor     $t1,$s1,$s2        #Test if $s1 and $s2 have the same sign
10         bltz    $t1,EXIT           #If not, exit
11         slt     $t2,$s3,$s1
12         bltz    $s1,NEGATIVE       #Test if $s1 and $s2 is negative?
13         beq     $t2,$zero,EXIT     #s1 and $s2 are positive
14                                    # if $s3 > $s1 then the result is not overflow
15         j OVERFLOW
16 NEGATIVE:
17         bne     $t2,$zero,EXIT     #s1 and $s2 are negative
18                                    # if $s3 < $s1 then the result is not overflow
19 OVERFLOW:
20         li       $t0,1             #the result is overflow
21 EXIT:
22
```

- **TH1: Hai số trái dấu:**

$s1	17	-12
$s2	18	200
$t0	8	0

- **TH2: Hai số dương không overflow:**

$s1	17	1048575
$s2	18	15798840
$t0	8	0

- **TH3: Hai số dương có overflow:**

$s1	17	0x7fffffff
$s2	18	0x0000000f

$t0	8	0x00000001

- **TH4: Hai số âm không overflow:**

$s1	17	-34945
$s2	18	-43649
$s3	19	-78594
$t0	8	0

- **TH5: Hai số âm có overflow:**

$s1	17	0xffffffff
$s2	18	0x80000000
$s3	19	0x7fffffff
$t0	8	0x00000001

Assigment 2:

```
#Laboratory Exercise 4, Home Assignment 1
.text
start:
        li      $s0, 0x0f001234
        andi    $t0, $s0, 0xff00000     # extra MSB
        andi    $t1, $s0, 0xffffff00    # Clear LSB
        or      $t2, $s0, 0x000000ff    #Set LSB
        xor     $t3,$s0,$s0             #Clear $s0
```

$s0	16	0x0f001234
$t0	8	0x0f000000
$t1	9	0x0f001200
$t2	10	0x0f0012ff
$t3	11	0x00000000

Assigment 3:

a. abs $s0,$s1

$s0 <= | $s1

week4_a3.asm*

```
1  .text
2  start:
3          li      $s1, -20                # s1 = -20
4          bgtz    $s1,  positive          # if(s1>0) then positive
5          sub     $s0, $zero, $s1         #s0 = 0 - s1=20
6          j end
7  positive:
8          add $s0, $zero, $s1             #s0=0 + s1=20
9  end:
```

Số âm

$s0	16	20
$s1	17	-20

Edit Execute

week4_a3.asm

```
1  .text
2  start:
3          li      $s1, 40                 # s1 = -40
4          bgtz    $s1,  positive          # if(s1>0) then positive
5          sub     $s0, $zero, $s1         #s0 = 0 - s1
6          j end
7  positive:
8          add $s0, $zero, $s1             #s0=0 + s1
9  end:
```

Registers Coproc 1 Coproc 0

Name	Number	Value
$zero	0	0
$at	1	0
$v0	2	0
$v1	3	0
$a0	4	0
$a1	5	0
$a2	6	0
$a3	7	0
$t0	8	0
$t1	9	0
$t2	10	0
$t3	11	0
$t4	12	0
$t5	13	0
$t6	14	0
$t7	15	0
$s0	16	40
$s1	17	40
$s2	18	0
$s3	19	0
$s4	20	0
$s5	21	0
$s6	22	0
$s7	23	0
$t8	24	0
$t9	25	0
$k0	26	0
$k1	27	0
$gp	28	268468224
$sp	29	2147479548
$fp	30	0

b. move $s0,$s1

$s0 <= $s1

```
.text
        li      $s1, -20
        add     $s0, $zero,$s1      #move $s0, s1;
```

$s0	16	-20
$s1	17	-20

c. not $s0, $s1

$s0 <= bit invert ($s1)

week4_a3b.asm

```
.text
        li      $s1, 0xffffffff
        li      $t1, 1
        sub     $s0, $zero,$s1      #s0 =  0- s1

        sub     $s0,$s0, $t1        #s0 = s1 - 1
```

$s0	16	0x00000000
$s1	17	0xffffffff

d. ble $s1,$s2,label

week4_a3b.asm | test.asm

```
1  .text
2        li    $s1, 4          #s1 = 4
3        li    $s2, 5          #s2 = 5
4        slt   $t1, $s1, $s2   # s1 < s2  ? t1 = 1: t1 =0
5        bne   $t1, $zero, L   #  s1 ! = 0 -> L if(  s1 < s2) -> L
6        beq   $s1, $s2, L     # s1 = s2 -> L     if(s1 =s2) -> L
7        j end
8  L:
9        add   $t0, $s1, $s2
10 end:
```

$t0	8	0x00000009
$t1	9	0x00000001
$t2	10	0x00000000
$t3	11	0x00000000
$t4	12	0x00000000
$t5	13	0x00000000
$t6	14	0x00000000
$t7	15	0x00000000
$s0	16	0x00000000
$s1	17	0x00000004
$s2	18	0x00000005

Assigment 4:

```
1  #Laboratory Exercise 4, Home Assignment 1
2  .text
3  start:
4          addi   $s1,$zero, 0xffffffff       #s1 = -4
5          addi  $s2,$zero,  0x80000000   #s2 = 10
6
7          li       $t0,0                 #No Overflow is default status
8          addu  $s3,$s1,$s2              # s3 = s1 + s2
9          xor    $t1,$s1,$s2             #Test if $s1 and $s2 have the same sign
10         bltz    $t1,EXIT               #If not, exit
11
12
13
14         xor $t1, $s3, $s2              #check s1, s3 have sign
15         bgtz $t1, EXIT                 # neu cung dau => exit
16
17 OVERFLOW:
18         li       $t0,1                 #the result is overflow
19 EXIT:
20
```

Kết quả giống như bài 1

Assigment 5:

```
.text
        li $s0,5                # s0=3
        li $s1,16               # s1=16
        li $t0, 0               # count = 0
LOOP:
        beq    $s1,1,End        # s1 = 1 => end
        srl     $s1,$s1,1       #  s1/2;
        addi   $t0,$t0,1        # count++
        j LOOP
End:
        sllv  $s2,$s0,$t0
```

$zero	0	0
$at	1	1
$v0	2	0
$v1	3	0
$a0	4	0
$a1	5	0
$a2	6	0
$a3	7	0
$t0	8	4
$t1	9	0
$t2	10	0
$t3	11	0
$t4	12	0
$t5	13	0
$t6	14	0
$t7	15	0
$s0	16	5
$s1	17	1
$s2	18	80