

ĐẠI HỌC BÁCH KHOA HÀ NỘI
TRƯỜNG CÔNG NGHỆ THÔNG TIN VÀ TRUYỀN THÔNG



BÁO CÁO

Bài tập thực hành tuần 4

Học phần: Thực hành kiến trúc máy tính

Giảng viên hướng dẫn: Lê Bá Vui

Sinh viên thực hiện: Phạm Huy Cảnh - 20194490

Mã lớp: 130938

Hà Nội, tháng 4 năm 2022

1. Assignment 1:

Trường hợp 1: Cộng hai số khác dấu

```
1 #Laboratory Exercise 4, Home Assignment 1
2
3 .text
4     li $s1, -123
5     li $s2, 123
6
7 start:
8     li $t0, 0           # No Overflow is default status
9     addu $s3, $s1, $s2  # s3 = s1 + s2
10    xor $t1, $s1, $s2   # Test if $s1 and $s2 have the same sign
11    bltz $t1, EXIT      # If not, exit
12    slt $t2, $s3, $s1   # Test if $s1 and $s2 is negative?
13    bltz $s1, NEGATIVE  # s1 and $s2 are positive
14    beq $t2, $zero, EXIT # if $s3 > $s1 then the result is not overflow
15    # if $s3 > $s1 then the result is not overflow
16    } OVERFLOW
17
18 NEGATIVE:
19     bne $t2, $zero, EXIT # s1 and $s2 are negative
20     # if $s3 < $s1 then the result is not overflow
21
22 OVERFLOW:
23     li $t0, 1           # the result is overflow
24
25 EXIT:
26
27
28
29
30
```

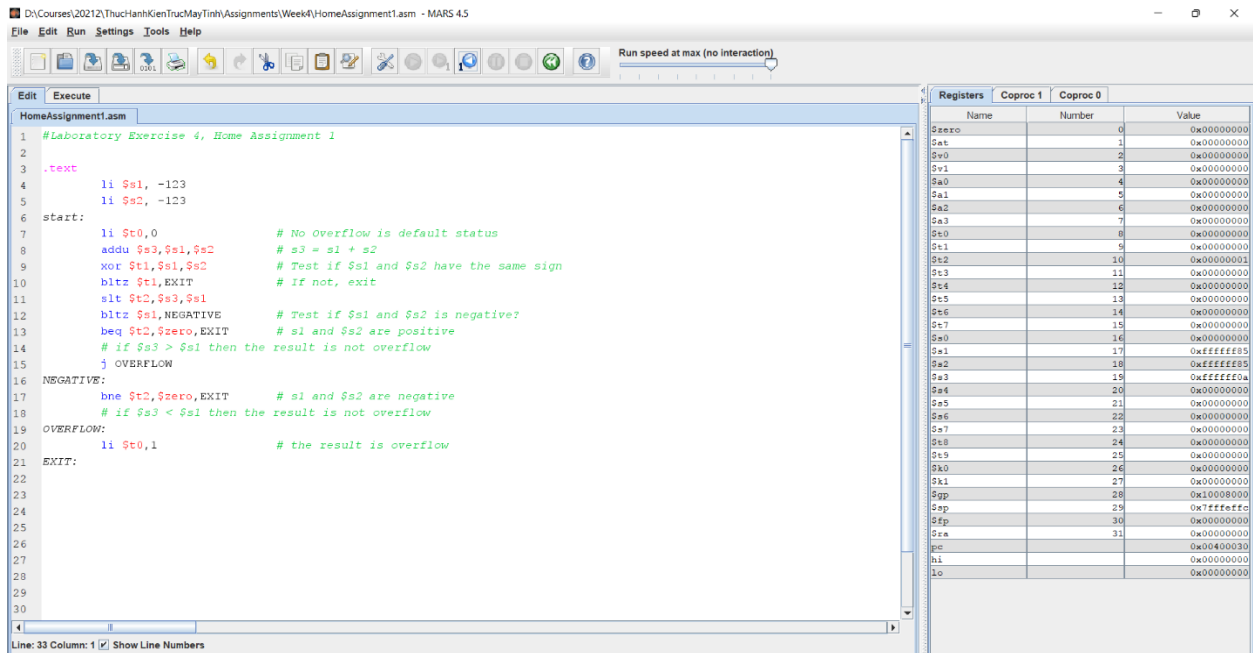
Registers	Coproc 1	Coproc 0	
Name	Number		Value
\$zero	0		0x00000000
\$at	1		0x00000000
\$v0	2		0x00000000
\$v1	3		0x00000000
\$a0	4		0x00000000
\$a1	5		0x00000000
\$a2	6		0x00000000
\$a3	7		0x00000000
\$t0	8		0x00000000
\$t1	9		0xffffffff
\$t2	10		0x00000000
\$t3	11		0x00000000
\$t4	12		0x00000000
\$t5	13		0x00000000
\$t6	14		0x00000000
\$t7	15		0x00000000
\$a0	16		0x00000000
\$a1	17		0xffffffff
\$a2	18		0x0000007b
\$a3	19		0x00000000
\$a4	20		0x00000000
\$a5	21		0x00000000
\$a6	22		0x00000000
\$a7	23		0x00000000
\$t8	24		0x00000000
\$t9	25		0x00000000
\$k0	26		0x00000000
\$k1	27		0x00000000
\$gp	28		0x10008000
\$sp	29		0x7ffffeff
\$fp	30		0x00000000
\$ra	31		0x00000000
\$pc			0x00400030
\$hi			0x00000000
\$lo			0x00000000

Trường hợp 2: Cộng hai số dương

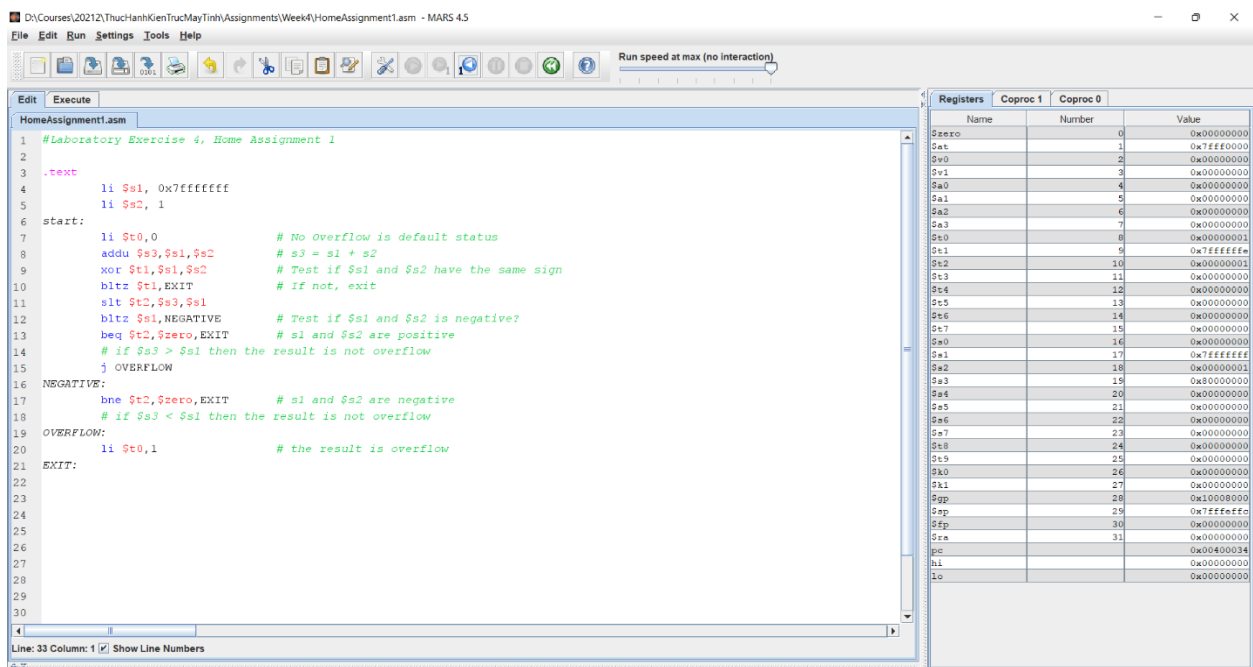
```
1 #Laboratory Exercise 4, Home Assignment 1
2
3 .text
4     li $s1, 123
5     li $s2, 123
6
7 start:
8     li $t0, 0           # No Overflow is default status
9     addu $s3, $s1, $s2  # s3 = s1 + s2
10    xor $t1, $s1, $s2   # Test if $s1 and $s2 have the same sign
11    bltz $t1, EXIT      # If not, exit
12    slt $t2, $s3, $s1   # Test if $s1 and $s2 is negative?
13    bltz $s1, NEGATIVE  # s1 and $s2 are positive
14    beq $t2, $zero, EXIT # if $s3 > $s1 then the result is not overflow
15    # if $s3 > $s1 then the result is not overflow
16    } OVERFLOW
17
18 NEGATIVE:
19     bne $t2, $zero, EXIT # s1 and $s2 are negative
20     # if $s3 < $s1 then the result is not overflow
21
22 OVERFLOW:
23     li $t0, 1           # the result is overflow
24
25 EXIT:
26
27
28
29
30
```

Registers	Coproc 1	Coproc 0	
Name	Number		Value
\$zero	0		0x00000000
\$at	1		0x00000000
\$v0	2		0x00000000
\$v1	3		0x00000000
\$a0	4		0x00000000
\$a1	5		0x00000000
\$a2	6		0x00000000
\$a3	7		0x00000000
\$t0	8		0x00000000
\$t1	9		0x00000000
\$t2	10		0x00000000
\$t3	11		0x00000000
\$t4	12		0x00000000
\$t5	13		0x00000000
\$t6	14		0x00000000
\$t7	15		0x00000000
\$a0	16		0x00000000
\$a1	17		0x0000007b
\$a2	18		0x0000007b
\$a3	19		0x000000f6
\$a4	20		0x00000000
\$a5	21		0x00000000
\$a6	22		0x00000000
\$a7	23		0x00000000
\$t8	24		0x00000000
\$t9	25		0x00000000
\$k0	26		0x00000000
\$k1	27		0x00000000
\$gp	28		0x10008000
\$sp	29		0x7ffffeff
\$fp	30		0x00000000
\$ra	31		0x00000000
\$pc			0x00400030
\$hi			0x00000000
\$lo			0x00000000

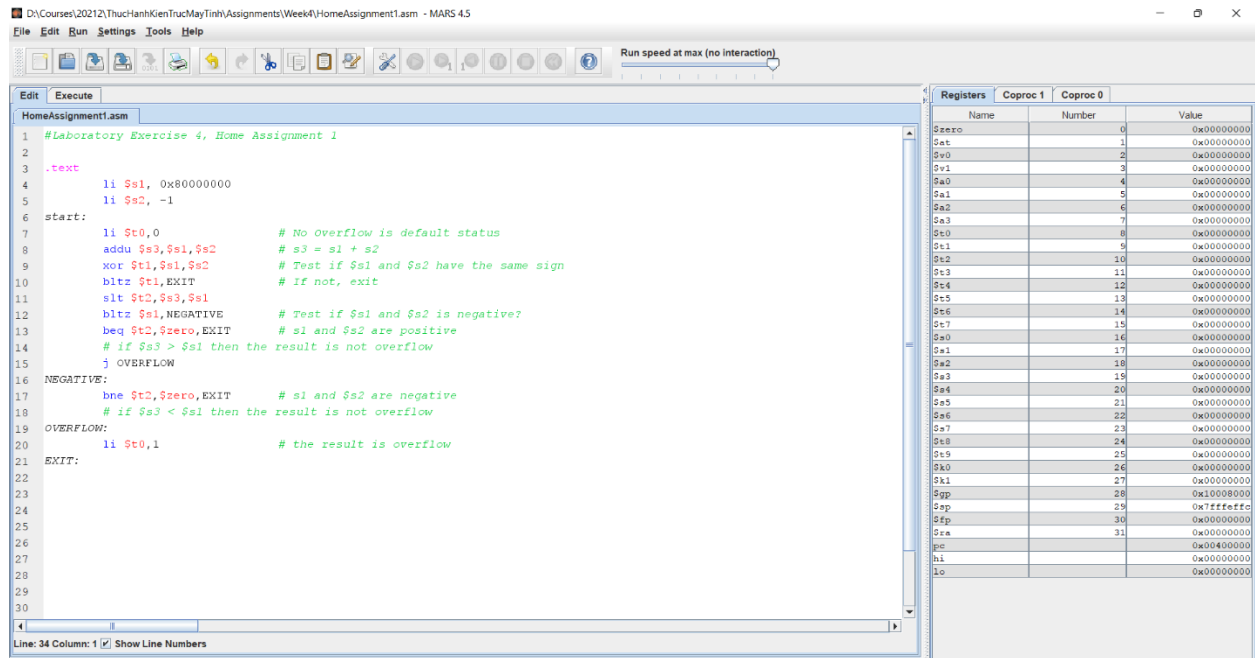
Trường hợp 3: Cộng hai số âm



Trường hợp 4: Cộng hai số dương – tràn số



Trường hợp 5: Cộng hai số âm – tràn số



2. Assignment 2:

```

.text

li $s0, 0x12345678

andi $t1, $s0, 0xff000000

andi $t2, $s0, 0xffffffff00

or $t3, $s0, 0x000000ff

xor $s0, $s0, $s0

```

Khi chạy lệnh: `li $s0, 0x12345678`

The screenshot shows the MARS MIPS simulator interface. The assembly code in the editor is as follows:

```
1 #Laboratory Exercise 4, Assignment 2
2
3 .text
4 li $s0, 0x12345678
5 andi $t1, $s0, 0xff000000 # Extract MSB of $s0
6 andi $t2, $s0, 0xfffffff0 # Clear LSB of $s0
7 or $t3, $s0, 0x000000ff # Set LSB of $s0 (bits 7 to 0 are set to 1)
8 xor $s0, $s0, $s0 # Clear $s0 ($s0=0, must use logical instructions)
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
26
27
28
29
30
```

The register window on the right shows the state of the MIPS registers. The register \$s0 is highlighted in green and contains the value 0x12345678.

Name	Number	Value
\$zero	0	0x00000000
\$at	1	0x12340000
\$v0	2	0x00000000
\$v1	3	0x00000000
\$a0	4	0x00000000
\$a1	5	0x00000000
\$a2	6	0x00000000
\$a3	7	0x00000000
\$t0	8	0x00000000
\$t1	9	0x00000000
\$t2	10	0x00000000
\$t3	11	0x00000000
\$t4	12	0x00000000
\$t5	13	0x00000000
\$t6	14	0x00000000
\$t7	15	0x00000000
\$s0	16	0x12345678
\$s1	17	0x00000000
\$s2	18	0x00000000
\$s3	19	0x00000000
\$s4	20	0x00000000
\$s5	21	0x00000000
\$s6	22	0x00000000
\$s7	23	0x00000000
\$s8	24	0x00000000
\$s9	25	0x00000000
\$t0	26	0x00000000
\$t1	27	0x00000000
\$gp	28	0x10000000
\$gp	29	0x7ffffc00
\$fp	30	0x00000000
\$ra	31	0x00000000
\$hi		0x00000000
\$lo		0x00000000

Khi chạy lệnh: `andi $t1, $s0, 0xff000000`

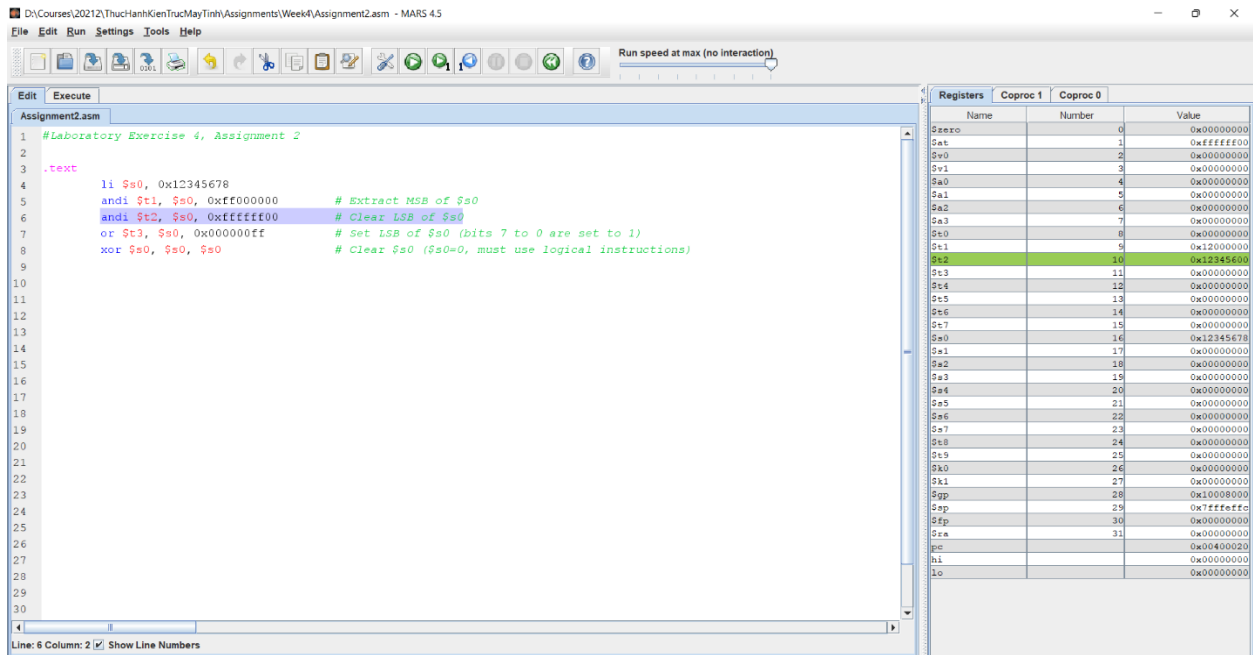
The screenshot shows the MARS MIPS simulator interface. The assembly code in the editor is as follows:

```
1 #Laboratory Exercise 4, Assignment 2
2
3 .text
4 li $s0, 0x12345678
5 andi $t1, $s0, 0xff000000 # Extract MSB of $s0
6 andi $t2, $s0, 0xfffffff0 # Clear LSB of $s0
7 or $t3, $s0, 0x000000ff # Set LSB of $s0 (bits 7 to 0 are set to 1)
8 xor $s0, $s0, $s0 # Clear $s0 ($s0=0, must use logical instructions)
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
26
27
28
29
30
```

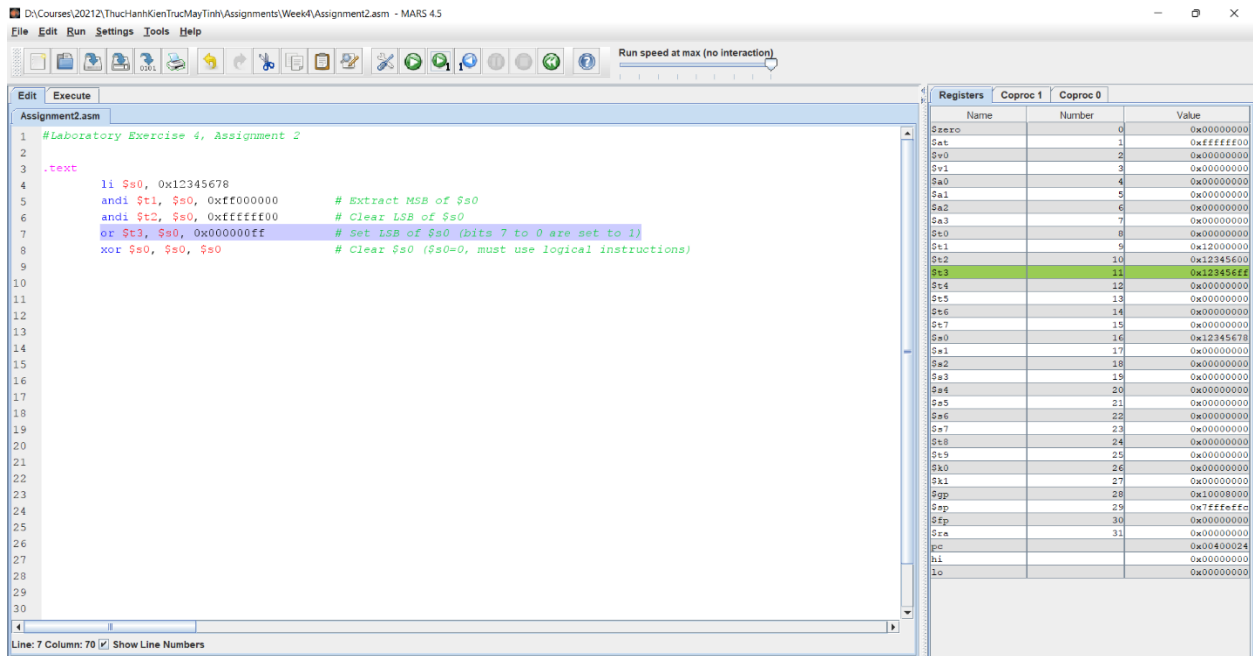
The register window on the right shows the state of the MIPS registers. The register \$t1 is highlighted in green and contains the value 0x12000000.

Name	Number	Value
\$zero	0	0x00000000
\$at	1	0xff000000
\$v0	2	0x00000000
\$v1	3	0x00000000
\$a0	4	0x00000000
\$a1	5	0x00000000
\$a2	6	0x00000000
\$a3	7	0x00000000
\$t0	8	0x00000000
\$t1	9	0x12000000
\$t2	10	0x00000000
\$t3	11	0x00000000
\$t4	12	0x00000000
\$t5	13	0x00000000
\$t6	14	0x00000000
\$t7	15	0x00000000
\$s0	16	0x12345678
\$s1	17	0x00000000
\$s2	18	0x00000000
\$s3	19	0x00000000
\$s4	20	0x00000000
\$s5	21	0x00000000
\$s6	22	0x00000000
\$s7	23	0x00000000
\$s8	24	0x00000000
\$s9	25	0x00000000
\$t0	26	0x00000000
\$t1	27	0x00000000
\$gp	28	0x10000000
\$gp	29	0x7ffffc00
\$fp	30	0x00000000
\$ra	31	0x00000000
\$hi		0x00000000
\$lo		0x00000000

Khi chạy lệnh: `andi $t2, $s0, 0xfffffffff0`



Khi chạy lệnh: `or $t3, $s0, 0x000000ff`



Khi chạy lệnh: `xor $s0, $s0, $s0`

D:\Courses\2021\2\ThucHanhKienTrucMayTinh\Assignments\Week4\Assignment2.asm - MARS 4.5

File Edit Run Settings Tools Help

Run speed at max (no interaction)

Edit Execute

Assignment2.asm

```
1 #Laboratory Exercise 4, Assignment 2
2
3 .text
4 li $s0, 0x12345678
5 andi $t1, $s0, 0xffff0000 # Extract MSB of $s0
6 andi $t2, $s0, 0xfffffff0 # Clear LSB of $s0
7 or $t3, $s0, 0x000000ff # Set LSB of $s0 (bits 7 to 0 are set to 1)
8 xor $s0, $s0, $s0 # Clear $s0 ($s0=0, must use logical instructions)
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
26
27
28
29
30
```

Registers Coproc 1 Coproc 0

Name	Number	Value
\$zero	0	0x00000000
\$at	1	0xffffffff
\$v0	2	0x00000000
\$v1	3	0x00000000
\$a0	4	0x00000000
\$a1	5	0x00000000
\$a2	6	0x00000000
\$a3	7	0x00000000
\$t0	8	0x00000000
\$t1	9	0x12000000
\$t2	10	0x12345600
\$t3	11	0x123456ff
\$t4	12	0x00000000
\$t5	13	0x00000000
\$t6	14	0x00000000
\$t7	15	0x00000000
\$s0	16	0x00000000
\$s1	17	0x00000000
\$s2	18	0x00000000
\$s3	19	0x00000000
\$s4	20	0x00000000
\$s5	21	0x00000000
\$s6	22	0x00000000
\$s7	23	0x00000000
\$s8	24	0x00000000
\$t9	25	0x00000000
\$t0	26	0x00000000
\$t1	27	0x00000000
\$gp	28	0x10000000
\$gp	29	0xffffffff
\$fp	30	0x00000000
\$ra	31	0x00000000
\$pc		0x00400028
\$hi		0x00000000
\$lo		0x00000000

Line: 8 Column: 2 Show Line Numbers

3. Assignment 3:

```
.text

li $s0, 0
li $s1, -5
li $s2, 5

# a. abs $s0, $s1
sra $t0, $s1, 31
xor $s0, $t0, $s1
subu $s0, $s0, $t0

# b. move $s0, $s1
addu $s0, $zero, $s1

# c. not $s0, $s1
nor $s0, $s1, $zero

# d. ble $s1, $s2, label
slt $t0, $s2, $s1
beq $t0, $zero, label

label:
```


D:\Courses\20212\ThucHanhKienTrucMayTinh\Assignments\Week4\Assignment3.asm - MARS 4.5

File Edit Run Settings Tools Help

Run speed at max (no interaction)

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	-1
\$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	-5
\$s2	18	5
\$s3	19	0
\$s4	20	0
\$s5	21	0
\$s6	22	0
\$s7	23	0
\$s8	24	0
\$s9	25	0
\$k0	26	0
\$k1	27	0
\$gp	28	268469224
\$sp	29	2147479548
\$fp	30	0
\$ra	31	0
pc		4194332
hi		0
lo		0

```

1 # Laboratory Exercise 4, Assignment 3
2 # Author: Pham Huy Canh
3 .text
4     li $s0, 0
5     li $s1, -5
6     li $s2, 5
7
8     # a. abs $s0, $s1
9     sra $t0, $s1, 31      # Dịch phải 31 bit của s1 (Mục đích để biến các bit giống với giá trị của bit dấu)
10    xor $s0, $t0, $s1
11    subu $s0, $s0, $t0
12
13    # b. move $s0, $s1
14    addu $s0, $zero, $s1
15
16    # c. not $s0, $s1
17    nor $s0, $s1, $zero
18
19    # d. ble $s1, $s2, label
20    slt $t0, $s2, $s1
21    beq $t0, $zero, label
22
23 label:
24
25
26
27
28
29
30

```

Line: 8 Column: 1 Show Line Numbers

Mars Messages Run I/O

Clear

D:\Courses\20212\ThucHanhKienTrucMayTinh\Assignments\Week4\Assignment3.asm - MARS 4.5

File Edit Run Settings Tools Help

Run speed at max (no interaction)

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	-1
\$t1	9	0
\$t2	10	0
\$t3	11	0
\$t4	12	0
\$t5	13	0
\$t6	14	0
\$t7	15	0
\$s0	16	4
\$s1	17	-5
\$s2	18	5
\$s3	19	0
\$s4	20	0
\$s5	21	0
\$s6	22	0
\$s7	23	0
\$s8	24	0
\$s9	25	0
\$k0	26	0
\$k1	27	0
\$gp	28	268469224
\$sp	29	2147479548
\$fp	30	0
\$ra	31	0
pc		4194336
hi		0
lo		0

```

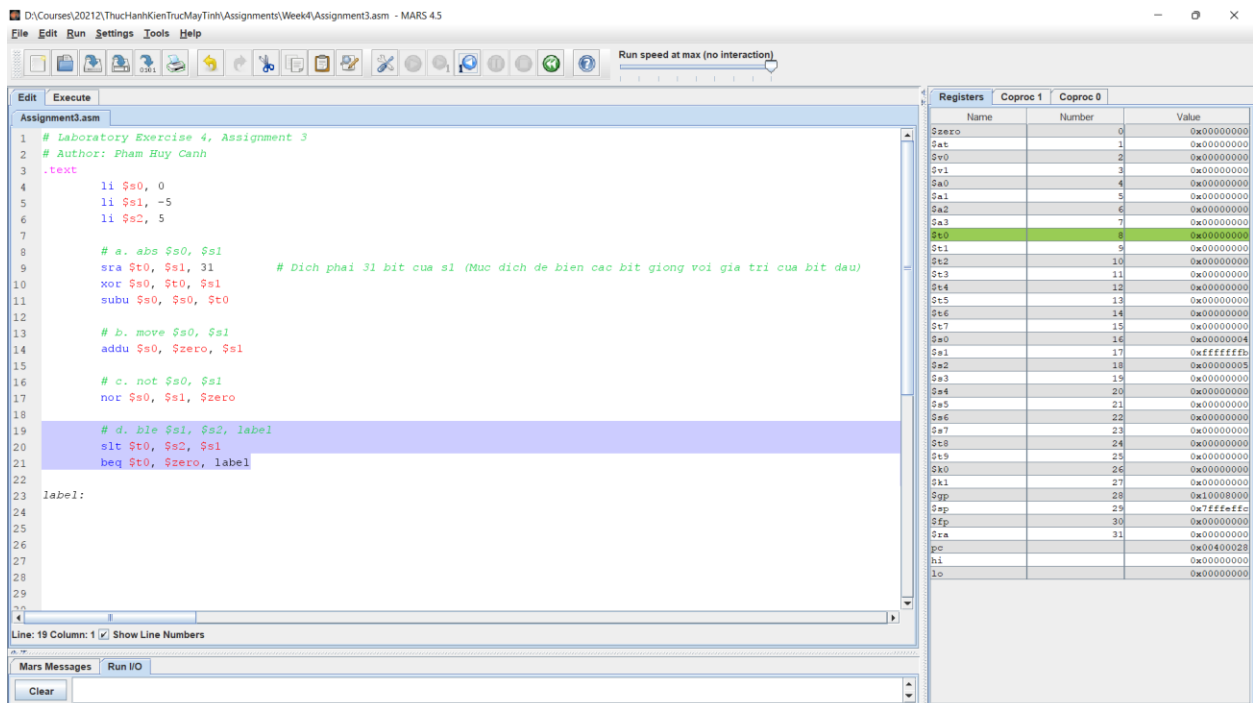
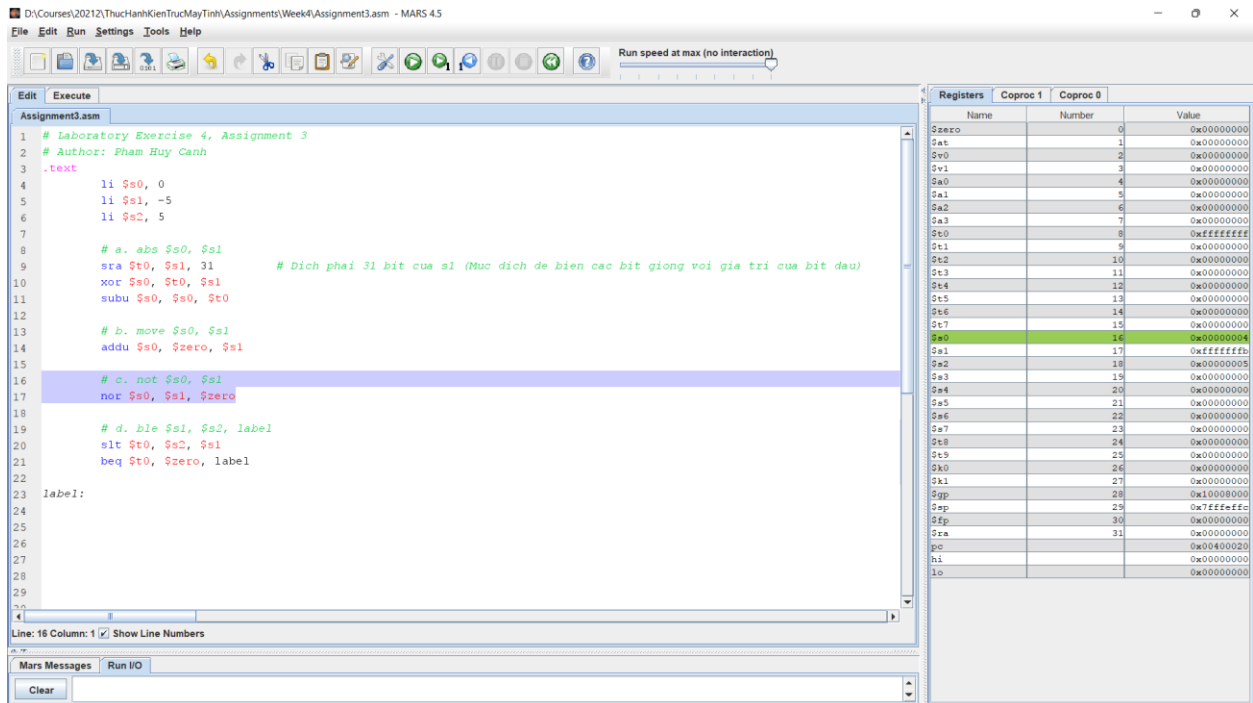
1 # Laboratory Exercise 4, Assignment 3
2 # Author: Pham Huy Canh
3 .text
4     li $s0, 0
5     li $s1, -5
6     li $s2, 5
7
8     # a. abs $s0, $s1
9     sra $t0, $s1, 31      # Dịch phải 31 bit của s1 (Mục đích để biến các bit giống với giá trị của bit dấu)
10    xor $s0, $t0, $s1
11    subu $s0, $s0, $t0
12
13    # b. move $s0, $s1
14    addu $s0, $zero, $s1
15
16    # c. not $s0, $s1
17    nor $s0, $s1, $zero
18
19    # d. ble $s1, $s2, label
20    slt $t0, $s2, $s1
21    beq $t0, $zero, label
22
23 label:
24
25
26
27
28
29
30

```

Line: 13 Column: 1 Show Line Numbers

Mars Messages Run I/O

Clear



4. Assignment 4:

```
.text
```

```
li $s0, -2001
```

```
li $s1, -123
```

```
li $t0, 0
```

```
xor $t1, $s0, $s1
```

```
blez $t1, Exit
```

```
addu $t2, $s0, $s1
```

```
xor $t1, $s1, $t2
```

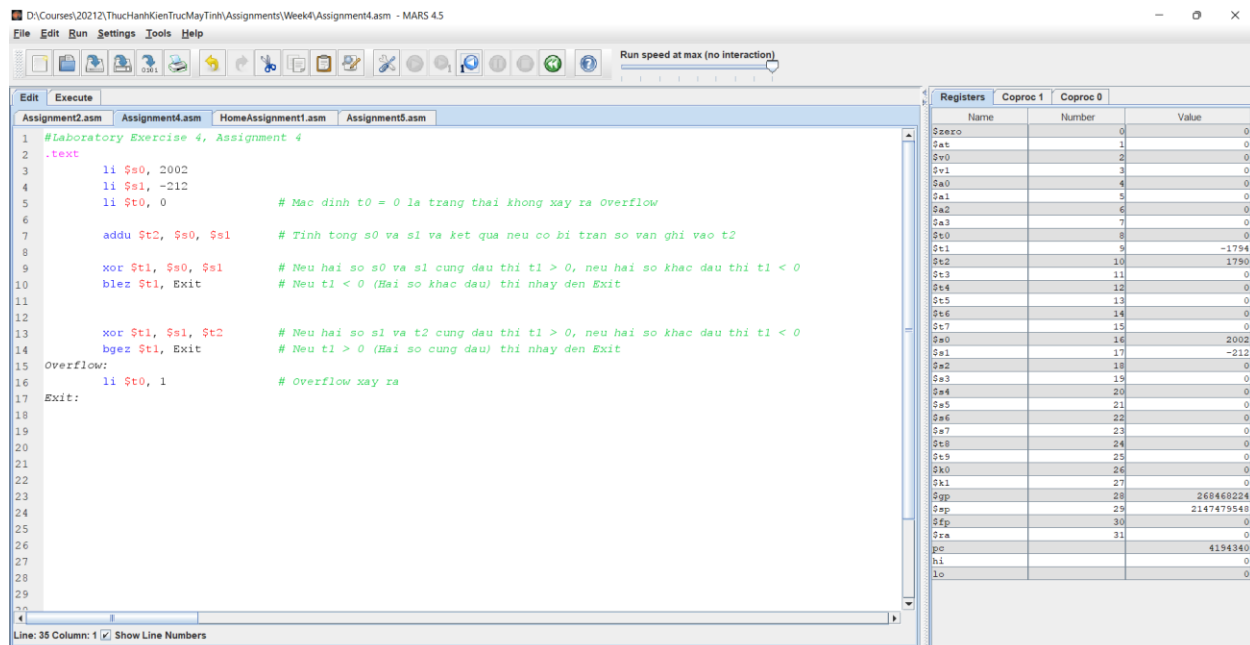
```
bgez $t1, Exit
```

Overflow:

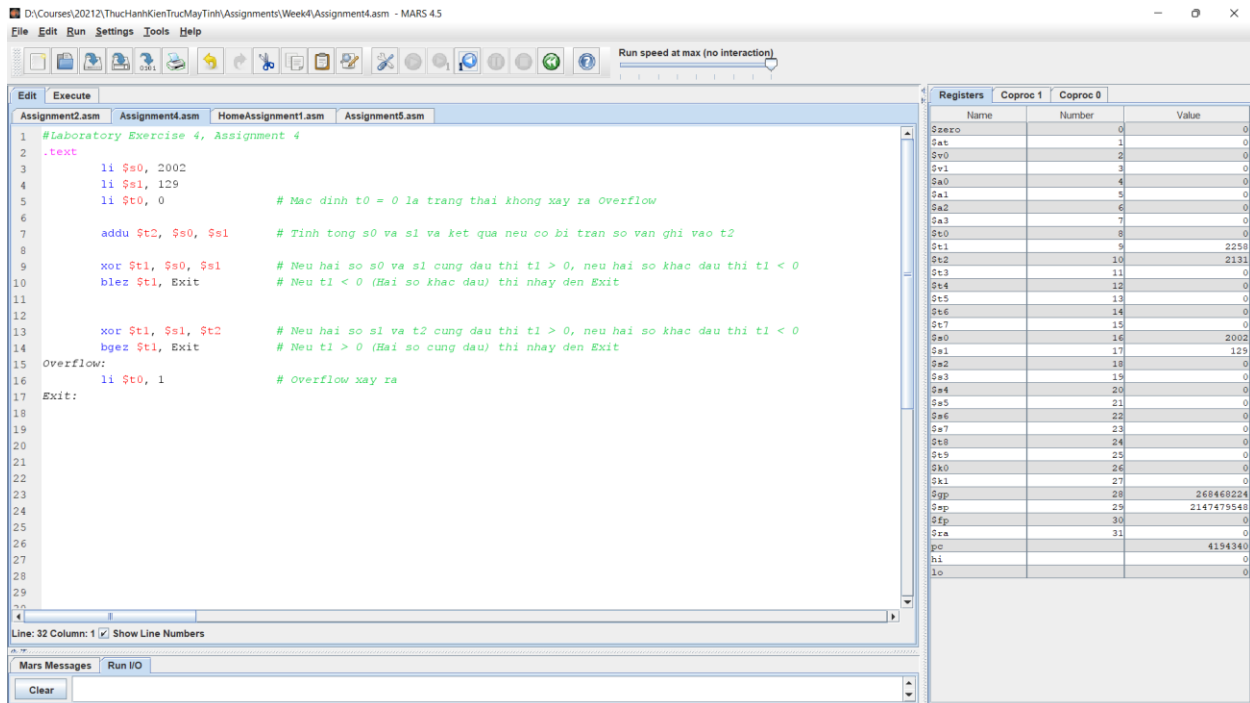
```
li $t0, 1
```

Exit:

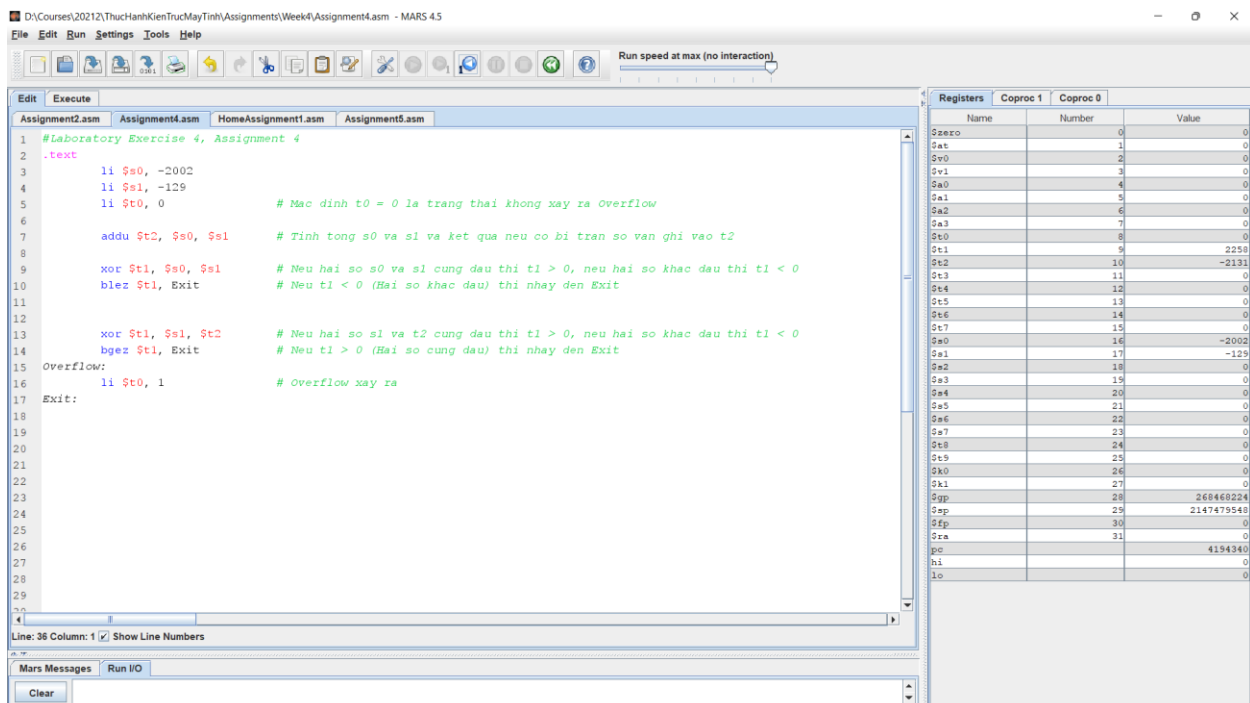
Trường hợp 1: Cộng hai số khác dấu



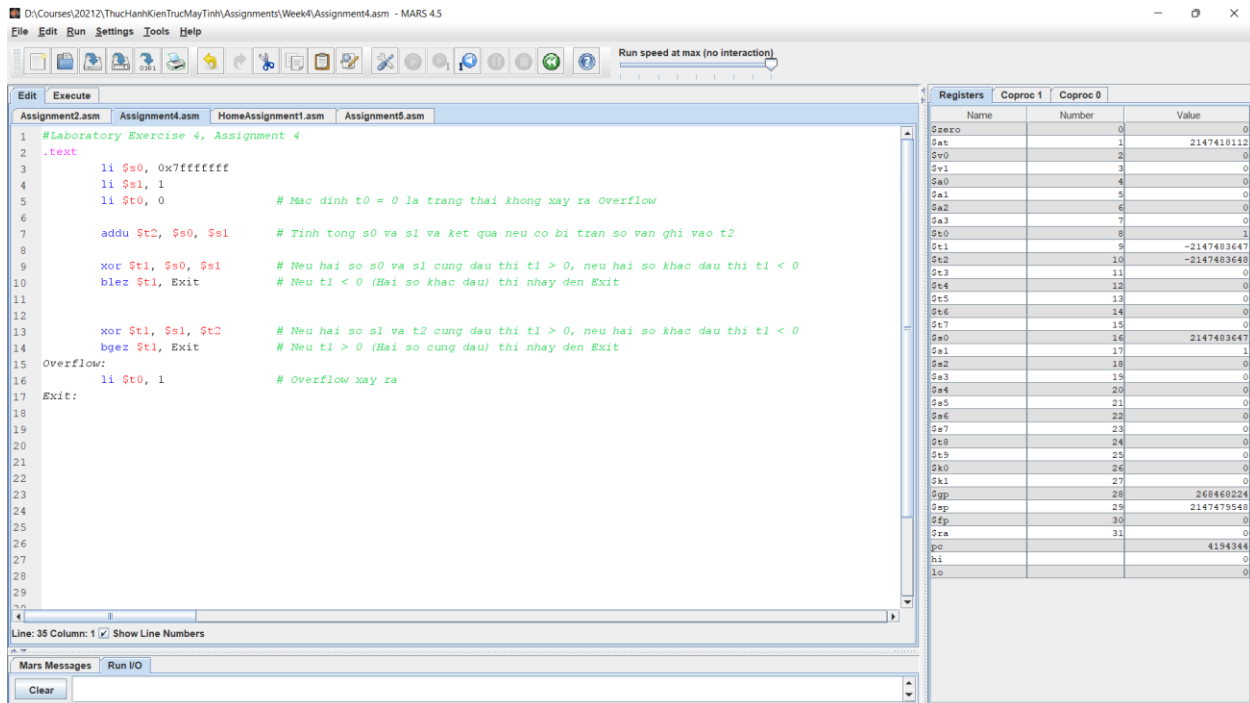
Trường hợp 2: Cộng hai số dương



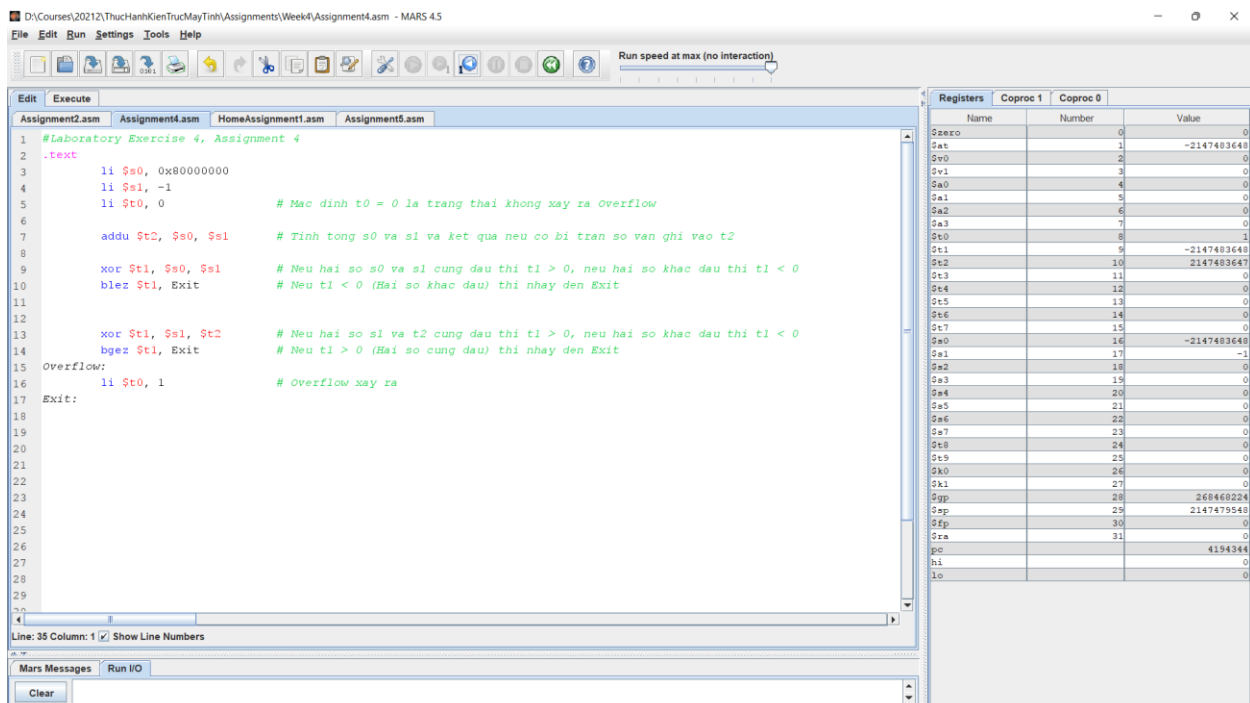
Trường hợp 3: Cộng hai số âm



Trường hợp 4: Cộng hai số dương – tràn số



Trường hợp 5: Cộng hai số âm – tràn số



5. Assignment 5:

```
.text

li $s0, 10

li $s1, 16

li $s2, 0

move $t1, $s1

loop:

    beq $t1, 1, multiple
    srl $t1, $t1, 1
    addi $s2, $s2, 1
    j loop

multiple:

    sllv $t0, $s0, $s2
```

D:\Courses\20212\ThucHanhKienTrucMayTinh\Assignments\Week4\Assignment5.asm - MARS 4.5

File Edit Run Settings Tools Help

Run speed at max (no interaction)

Edit Execute

Assignment3.asm Assignment2.asm Assignment4.asm Assignment5.asm

```
1 #Laboratory Exercise 4, Assignment 5
2 .text
3     li $s0, 10
4     li $s1, 16
5     li $s2, 0           # i = 0
6     move $t1, $s1
7 loop:
8     beq $t1, 1, multiple # Kiem tra t1 = 1 thi nhay den multiple
9     srl $t1, $t1, 1      # Dich phai t1 sang 1 bit (t1 = t1 / 2)
10    addi $s2, $s2, 1     # i = i + 1
11    j loop
12 multiple:
13    sllv $t0, $s0, $s2   # Dich trai s0 sang s2 bit (t0 = s0 * 2^s2)
14
15
16
17
18
19
20
21
22
23
24
25
26
27
28
29
30
```

Line: 36 Column: 2 Show Line Numbers

Registers	Coproc 1	Coproc 0
Name	Number	Value
\$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	160
\$t1	9	1
\$t2	10	0
\$t3	11	0
\$t4	12	0
\$t5	13	0
\$t6	14	0
\$t7	15	0
\$n0	16	0
\$n1	17	16
\$n2	18	4
\$n3	19	0
\$n4	20	0
\$n5	21	0
\$n6	22	0
\$n7	23	0
\$n8	24	0
\$n9	25	0
\$n10	26	0
\$n11	27	0
\$gp	28	268468224
\$fp	29	2147479548
\$sp	30	0
\$re	31	0
\$hi		4194344
\$lo		0

Consolutions

1. What is the difference between SLLV and SLL instructions?

- Lệnh `sll $s1, $s2, imm`: Dịch trái `$s2` số bit được quy định ở phần immediate, sau đó lưu kết quả vào `$s1`.

- Lệnh `sllv $s1, $s2, $s3`: Dịch trái `$s2` số bit được quy định bởi 5 bit trật tự thấp (low-order) của `$s3`, mang giá trị từ 0-31 và lưu kết quả vào `$s1`.

2. What is the difference between SRLV and SRL instructions?

- Lệnh `srl $s1, $s2, imm`: Dịch phải `$s2` số bit được quy định ở phần intermediate, sau đó lưu kết quả vào `$s1`.

- Lệnh `srlv $s1, $s2, $s3`: Dịch phải `$s2` số bit được quy định bởi 5 bit trật tự thấp (low-order) của `$s3`, mang giá trị từ 0-31 và lưu kết quả vào `$s1`.