

Report Lab 10

Nguyễn Khánh Nam - 20225749

Assignment 1

Code:

```
.eqv SEVENSEG_LEFT 0xFFFF0010 # ??a ch? c?a ?n LED 7 ?o?n tr?i. # Bit 0 =
?o?n a; # Bit 1 = ?o?n b; ... # Bit 7 = d?u .
.eqv SEVENSEG_RIGHT 0xFFFF0011 # ??a ch? c?a ?n LED 7 ?o?n ph?i
.data
arr: .word 0x3F, 0x06, 0x5B, 0x4F, 0x66, 0x6D, 0x7D, 0x07, 0x7F, 0x6F, 0x77, 0x7C,
0x39, 0x5E, 0x79, 0x71
.text
main:
    # L?p t? 0 ?n 15
    la $t7, arr
    add $t8, $0, $t7
    add $t9, $0, $t7
    li $t1, 0 # Kh?i t?o bi?n ??m
loop:
    subi $t3, $t1, 10
    slt $t4, $t3, $0
    beq $t4, 0, exit
SO_TRUOC_9:
    lw $t2, 0($t8)
    addi $a0, $0, 0x3F
    jal SHOW_7SEG_RIGHT # Hi?n th?

    add $a0, $0, $t2
    jal SHOW_7SEG_LEFT
    addi $t8, $t8, 4
    addi $t1, $t1, 1
    li $v0, 32
    # ??t gi? tr? cho $a0 l? 1000 ?? ch? ??nh th?i gian ng? l? 1000 milliseconds (1
    gi?y)
    li $a0, 1000
    # G?i syscall ?? th?c hi?n ?? tr?
    syscall
```

j loop

Kĩ thuật tra n?u bi?n ??m v?n nh? h?n 16, ti?p t?c v?ng l?p
exit:

li \$v0, 10

syscall

endmain:

#-----

Function SHOW_7SEG_LEFT: B?t/t?t ?n LED 7 ?o?n tr?

param[in] \$a0 gi? tr? c?n hi?n th?

remark \$t0 thay ??i

#-----

SHOW_7SEG_LEFT:

li \$t0, SEVENSEG_LEFT # G?n ??a ch? c?ng

sb \$a0, 0(\$t0) # G?n gi? tr? m?

jr \$ra

#-----

Function SHOW_7SEG_RIGHT: B?t/t?t ?n LED 7 ?o?n ph?

param[in] \$a0 gi? tr? c?n hi?n th?

remark \$t0 thay ??i

#-----

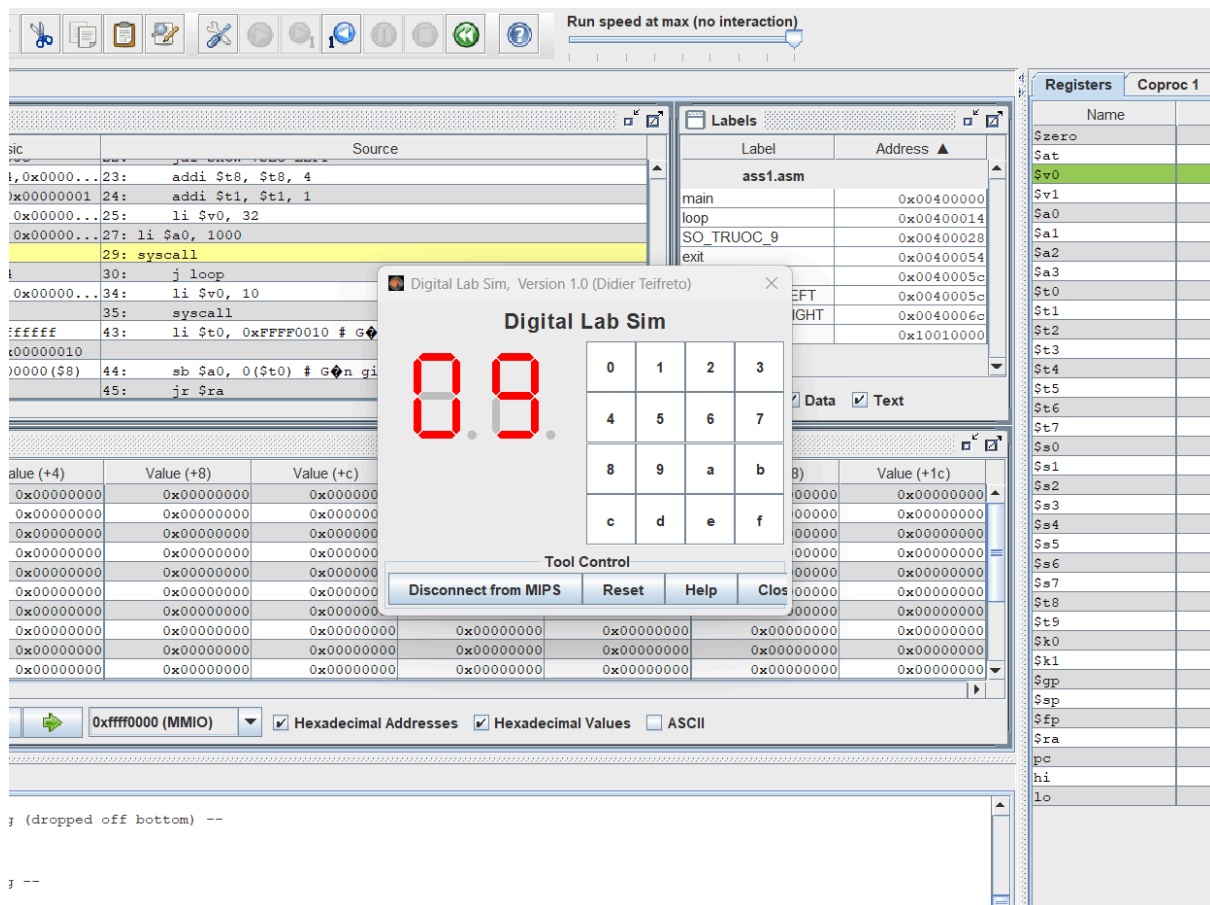
SHOW_7SEG_RIGHT:

li \$t0, SEVENSEG_RIGHT # G?n ??a ch? c?ng

sb \$a0, 0(\$t0) # G?n gi? tr? m?

jr \$ra

Result:



Assignment 2

Code:

```
.eqv MONITOR_SCREEN 0x10010000 #Dia chi bat dau cua bo nho man hinh
.eqv RED 0x00FF0000 #Cac gia tri mau thuong su dung
.eqv GREEN 0x0000FF00
.eqv BLUE 0x000000FF
.eqv WHITE 0x00FFFFFF
.eqv YELLOW 0x00FFFF00
.text
```

```
li $k0, MONITOR_SCREEN #Nap dia chi bat dau cua man hin
```

Loop1:

```
beq $t1, 256, end_loop1
nop
add $t2, $k0, $t1
li $t0, BLUE
sw $t0, 0($t2)
nop
```

```

        addi $t1, $t1, 32
        j Loop1
        nop
end_loop1:

        li $t1, 28

Loop2:
        beq $t1, 284, end_loop2
        nop
        add $t2, $k0, $t1

        li $t0, BLUE
        sw $t0, 0($t2)
        nop

        addi $t1, $t1, 32
        j Loop2
        nop
end_loop2:

        li $t1, 36

Loop3:
        beq $t1, 252, end_loop3
        nop
        add $t2, $k0, $t1

        li $t0, BLUE
        sw $t0, 0($t2)
        nop

        addi $t1, $t1, 36
        j Loop3
        nop

end_loop3:

```

Result:

recute

segment

Address	Code	Basic	Source
r00400000	0x3c011001	lui \$1,0x00001001	8: li \$k0, 0x10010000 #Nap dia chi bat dau cua man ..
r00400004	0x343a0000	ori \$26,\$1,0x00000000	
r00400008	0x20010100	addi \$1,\$0,0x000000	
r0040000c	0x10290008	beq \$1,\$9,0x000000	
r00400010	0x00000000	nop	
r00400014	0x03495020	add \$10,\$26,\$9	
r00400018	0x240800ff	addiu \$9,\$0,0x0000	
r0040001c	0xa4080000	sw \$20,0x00000000(\$	
r00400020	0x00000000	nop	
r00400024	0x21290020	addi \$9,\$9,0x0000	
r00400028	0x08100002	j 0x00400008	
r0040002c	0x00000000	nop	

segment

Address	Value (+0)	Value (+4)
t10010000	0x000000ff	0x0000
t10010020	0x000000ff	0x0000
t10010040	0x000000ff	0x0000
t10010060	0x000000ff	0x0000
t10010080	0x000000ff	0x0000
t100100a0	0x000000ff	0x0000
t100100c0	0x000000ff	0x0000
t100100e0	0x000000ff	0x0000
t10010100	0x00000000	0x0000
t10010120	0x00000000	0x0000

Labels

Label	Address
Assignment2.asm	

Registers

Coproc 1

Coproc 0

Name	Number
\$zero	0
\$at	1
\$v0	2
\$v1	3
	4
	5
	6
	7
	8
	9
	10
	11
	12
	13
	14
	15
	16
	17
	18
	19
	20
	21
	22
	23
	24
	25
	26
	27
\$w0	28
\$w1	29
\$w2	30
\$w3	31

Bitmap Display, Version 1.0

Bitmap Display

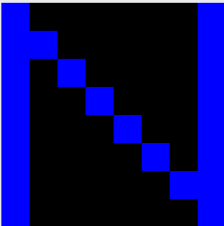
Unit Width in Pixels 32

Unit Height in Pixels 32

Display Width in Pixels 256

Display Height in Pixels 256

Base address for display 0x10010000 (static data)



Tool Control

Disconnect from MIPS

Reset

Help

Close

0x10010000 (.data)

☒ Hexadecimal Addresses

☒ Hexadecimal Values

☐ ASCII