**BÁO CÁO THỰC HÀNH KIẾN TRÚC MÁY TÍNH TUẦN 10 (P1)**

**Assignment 1**

* Code

.eqv SEVENSEG\_RIGHT 0xFFFF0010 # Dia chi cua den led 7 doan trai.

.eqv SEVENSEG\_LEFT 0xFFFF0011 # Dia chi cua den led 7 doan phai

#Nguyen Tuan Nam 20194629

.text

main:

li $a0, 91 #so 2 => 8 bit cuối thì phải là 01011011

jal SHOW\_7SEG\_LEFT # show

li $a0, 111 # so 9 => 8 bit cuối thì phải là 01101111

jal SHOW\_7SEG\_RIGHT # show

exit:

li $v0, 10

syscall

endmain:

SHOW\_7SEG\_LEFT:

li $t0, SEVENSEG\_LEFT # Gán địa chỉ

sb $a0, 0($t0) # Gán giá trị

jr $ra

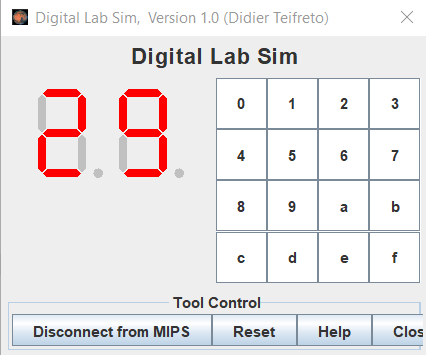
SHOW\_7SEG\_RIGHT:

li $t0, SEVENSEG\_RIGHT # Gán địa chỉ

sb $a0, 0($t0) # Gán giá trị

jr $ra

* Kết quả chạy chương trình:



**Assignment 2**

* **Code:**

.eqv SEVENSEG\_RIGHT 0xFFFF0010 # Dia chi cua den led 7 doan trai.

.eqv SEVENSEG\_LEFT 0xFFFF0011 # Dia chi cua den led 7 doan phai

.text

main:

input:

li $v0, 5 # Đọc số nguyên dương nhập vào

syscall

li $a0, 10

blt $v0, $a0, input # Nếu mà nó < 10 thì bắt nhập lại

end\_input:

div $v0, $a0 # lấy số vừa nhập chia cho 10 để lấy chữs ố cuối

mflo $v0

mfhi $s1

div $v0, $v0, $a0 # chia tiếp cho 10 để lấy chữ số gần cuối

mflo $v0

mfhi $s0

li $t0, 0 # Số để so sánh

beq $s0, $t0, set\_0l

addi $t0, $t0, 1

beq $s0, $t0, set\_1l

addi $t0, $t0, 1

beq $s0, $t0, set\_2l

addi $t0, $t0, 1

beq $s0, $t0, set\_3l

addi $t0, $t0, 1

beq $s0, $t0, set\_4l

addi $t0, $t0, 1

beq $s0, $t0, set\_5l

addi $t0, $t0, 1

beq $s0, $t0, set\_6l

addi $t0, $t0, 1

beq $s0, $t0, set\_7l

addi $t0, $t0, 1

beq $s0, $t0, set\_8l

addi $t0, $t0, 1

beq $s0, $t0, set\_9l

nextl:

jal SHOW\_7SEG\_LEFT # show

li $t0, 0

beq $s1, $t0, set\_0r

addi $t0, $t0, 1

beq $s1, $t0, set\_1r

addi $t0, $t0, 1

beq $s1, $t0, set\_2r

addi $t0, $t0, 1

beq $s1, $t0, set\_3r

addi $t0, $t0, 1

beq $s1, $t0, set\_4r

addi $t0, $t0, 1

beq $s1, $t0, set\_5r

addi $t0, $t0, 1

beq $s1, $t0, set\_6r

addi $t0, $t0, 1

beq $s1, $t0, set\_7r

addi $t0, $t0, 1

beq $s1, $t0, set\_8r

addi $t0, $t0, 1

beq $s1, $t0, set\_9r

nextr:

jal SHOW\_7SEG\_RIGHT # show

j exit

# Đặt chỉ số hiển thị cho led

set\_0l:

ori $a0, $0, 0x3f

j nextl

set\_1l:

ori $a0, $0, 0x06

j nextl

set\_2l:

ori $a0, $0, 0x5b

j nextl

set\_3l:

ori $a0, $0, 0x4f

j nextl

set\_4l:

ori $a0, $0, 0x66

j nextl

set\_5l:

ori $a0, $0, 0x6d

j nextl

set\_6l:

ori $a0, $0, 0x7d

j nextl

set\_7l:

ori $a0, $0, 0x07

j nextl

set\_8l:

ori $a0, $0, 0x7f

j nextl

set\_9l:

ori $a0, $0, 0x6f

j nextl

set\_0r:

ori $a0, $0, 0x3f

j nextr

set\_1r:

ori $a0, $0, 0x06

j nextr

set\_2r:

ori $a0, $0, 0x5b

j nextr

set\_3r:

ori $a0, $0, 0x4f

j nextr

set\_4r:

ori $a0, $0, 0x66

j nextr

set\_5r:

ori $a0, $0, 0x6d

j nextr

set\_6r:

ori $a0, $0, 0x7d

j nextr

set\_7r:

ori $a0, $0, 0x07

j nextr

set\_8r:

ori $a0, $0, 0x7f

j nextr

set\_9r:

ori $a0, $0, 0x6f

j nextr

exit:

li $v0, 10

syscall

endmain:

SHOW\_7SEG\_LEFT:

li $t1, SEVENSEG\_LEFT # Gán địa chỉ

sb $a0, 0($t1) # Gán giá trị

jr $ra

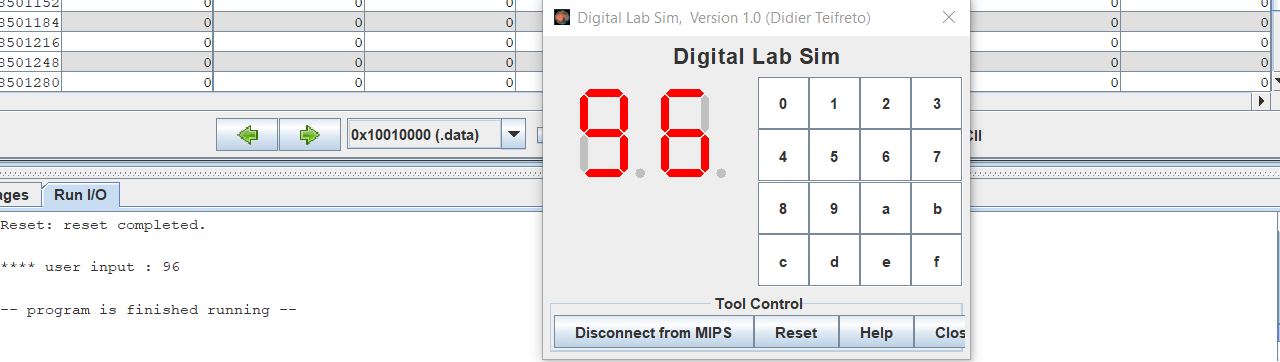
SHOW\_7SEG\_RIGHT:

li $t1, SEVENSEG\_RIGHT # Gán địa chỉ

sb $a0, 0($t1) # Gán giá trị

jr $ra

* **Kết quả:**

****

**Assignment 3**

* **Code:**

.eqv SEVENSEG\_RIGHT 0xFFFF0010 # Dia chi cua den led 7 doan trai.

.eqv SEVENSEG\_LEFT 0xFFFF0011 # Dia chi cua den led 7 doan phai

.text

main:

input:

li $v0, 12 # Đọc ký tự

syscall

li $a0, 10

end\_input:

div $v0, $a0 # lấy số vừa nhập chia cho 10 để lấy chữ số cuối

mflo $v0

mfhi $s1

div $v0, $v0, $a0 # chia tiếp cho 10 để lấy chữ số gần cuối

mflo $v0

mfhi $s0

li $t0, 0 # Số để so sánh

beq $s0, $t0, set\_0l

addi $t0, $t0, 1

beq $s0, $t0, set\_1l

addi $t0, $t0, 1

beq $s0, $t0, set\_2l

addi $t0, $t0, 1

beq $s0, $t0, set\_3l

addi $t0, $t0, 1

beq $s0, $t0, set\_4l

addi $t0, $t0, 1

beq $s0, $t0, set\_5l

addi $t0, $t0, 1

beq $s0, $t0, set\_6l

addi $t0, $t0, 1

beq $s0, $t0, set\_7l

addi $t0, $t0, 1

beq $s0, $t0, set\_8l

addi $t0, $t0, 1

beq $s0, $t0, set\_9l

nextl:

jal SHOW\_7SEG\_LEFT # show

li $t0, 0

beq $s1, $t0, set\_0r

addi $t0, $t0, 1

beq $s1, $t0, set\_1r

addi $t0, $t0, 1

beq $s1, $t0, set\_2r

addi $t0, $t0, 1

beq $s1, $t0, set\_3r

addi $t0, $t0, 1

beq $s1, $t0, set\_4r

addi $t0, $t0, 1

beq $s1, $t0, set\_5r

addi $t0, $t0, 1

beq $s1, $t0, set\_6r

addi $t0, $t0, 1

beq $s1, $t0, set\_7r

addi $t0, $t0, 1

beq $s1, $t0, set\_8r

addi $t0, $t0, 1

beq $s1, $t0, set\_9r

nextr:

jal SHOW\_7SEG\_RIGHT # show

j exit

# Đặt chỉ số hiển thị cho led

set\_0l:

ori $a0, $0, 0x3f

j nextl

set\_1l:

ori $a0, $0, 0x06

j nextl

set\_2l:

ori $a0, $0, 0x5b

j nextl

set\_3l:

ori $a0, $0, 0x4f

j nextl

set\_4l:

ori $a0, $0, 0x66

j nextl

set\_5l:

ori $a0, $0, 0x6d

j nextl

set\_6l:

ori $a0, $0, 0x7d

j nextl

set\_7l:

ori $a0, $0, 0x07

j nextl

set\_8l:

ori $a0, $0, 0x7f

j nextl

set\_9l:

ori $a0, $0, 0x6f

j nextl

set\_0r:

ori $a0, $0, 0x3f

j nextr

set\_1r:

ori $a0, $0, 0x06

j nextr

set\_2r:

ori $a0, $0, 0x5b

j nextr

set\_3r:

ori $a0, $0, 0x4f

j nextr

set\_4r:

ori $a0, $0, 0x66

j nextr

set\_5r:

ori $a0, $0, 0x6d

j nextr

set\_6r:

ori $a0, $0, 0x7d

j nextr

set\_7r:

ori $a0, $0, 0x07

j nextr

set\_8r:

ori $a0, $0, 0x7f

j nextr

set\_9r:

ori $a0, $0, 0x6f

j nextr

exit:

li $v0, 10

syscall

endmain:

SHOW\_7SEG\_LEFT:

li $t1, SEVENSEG\_LEFT # Gán địa chỉ

sb $a0, 0($t1) # Gán giá trị

jr $ra

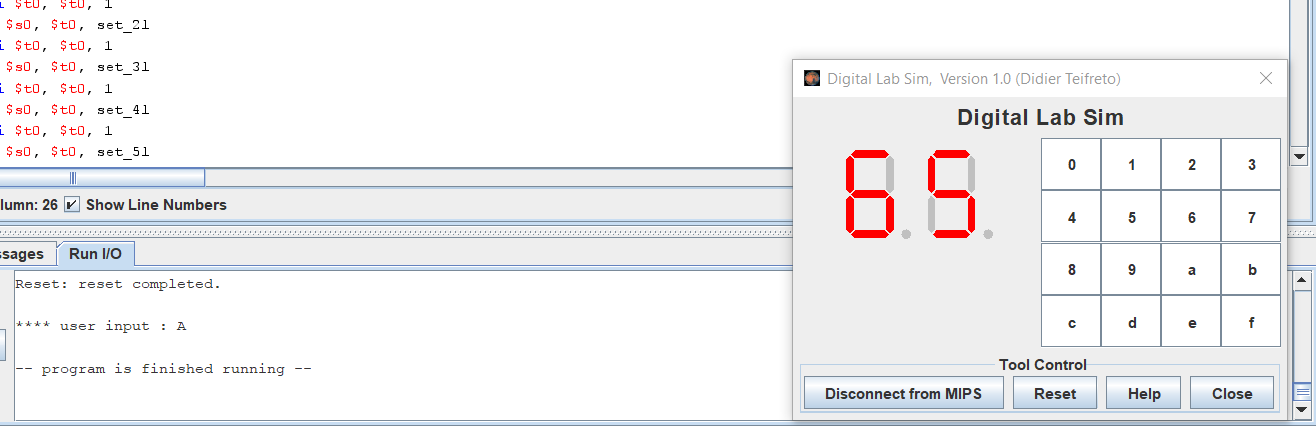
SHOW\_7SEG\_RIGHT:

li $t1, SEVENSEG\_RIGHT # Gán địa chỉ

sb $a0, 0($t1) # Gán giá trị

jr $ra

* **Kết quả**



**Assignment 4**

* **Code:**

.eqv MONITOR\_SCREEN 0x10010000

.eqv RED 0x00FF0000

.eqv GREEN 0x0000FF00

.eqv BLUE 0x000000FF

.eqv WHITE 0x00FFFFFF

.eqv YELLOW 0x00FFFF00

.text

li $k0, MONITOR\_SCREEN

li $s0, WHITE

li $t0, 0 # biến để đếm hàng

li $t2, 8 # số hàng và cột là 8

li $t3, 2

checkrow:

beq $t0, $t2, exit # hàng >= 8 thì dừng

li $t1, 0 # biến để đếm cột

div $t0, $t3 # kiểm tra xem hàng là chẵn hay lẻ

mfhi $t4

checkcol:

beq $t1, $t2, endcheckcol # cột >= 8 thì đến hàng tiếp theo

div $t1, $t3 # kiểm tra xem cột là chẵn hay lẻ

mfhi $t5

xor $t6, $t4, $t5 # nếu số hàng hoặc số cột là cùng chẵn hoặc cùng lẻ thì tô màu

beq $t6, $0, print

next:

addi $t1, $t1, 1

addi $k0, $k0, 4 # cộng thêm 4 đẻ vẽ ô tiếp theo

j checkcol

endcheckcol:

addi $t0, $t0, 1

j checkrow

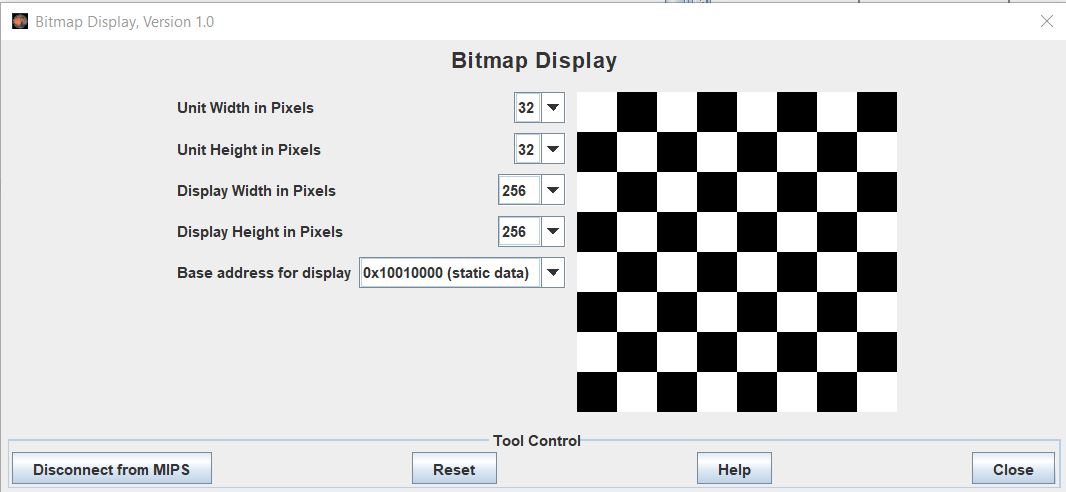
print:

sw $s0, 0($k0)

j next

exit:

* **Kết quả**

****

**Assignment 5**

.eqv MONITOR\_SCREEN 0x10010000

.eqv RED 0x00FF0000

.eqv GREEN 0x0000FF00

.text

# x1 < x2 y1 < y2

li $k0, MONITOR\_SCREEN

li $s0, RED

li $v0, 5

syscall

add $s1, $v0, $0 # x1

li $v0, 5

syscall

add $s2, $v0, $0 # y1

li $v0, 5

syscall

add $s3, $v0, $0 # x2

li $v0, 5

syscall

add $s4, $v0, $0 # y2

addi $t0, $s2, -1 # biến đếm hàng

loop11:

bgt $t0, $s4, next

addi $t1, $s1, -1 # biến đếm cột

loop21:

bgt $t1, $s3, endloop21

jal print

addi $t1, $t1, 1

j loop21

endloop21:

addi $t0, $t0, 1

j loop11

next:

li $s0, GREEN

add $t0, $0, $s2 # biến đếm hàng

loop1:

beq $t0, $s4, exit

add $t1, $0, $s1 # biến đếm cột

loop2:

beq $t1, $s3, endloop2

jal print

addi $t1, $t1, 1

j loop2

endloop2:

addi $t0, $t0, 1

j loop1

print:

addi $t2, $0, 64

mul $t2, $t2, $t0

add $t2, $t2, $t1

sll $t2, $t2, 2

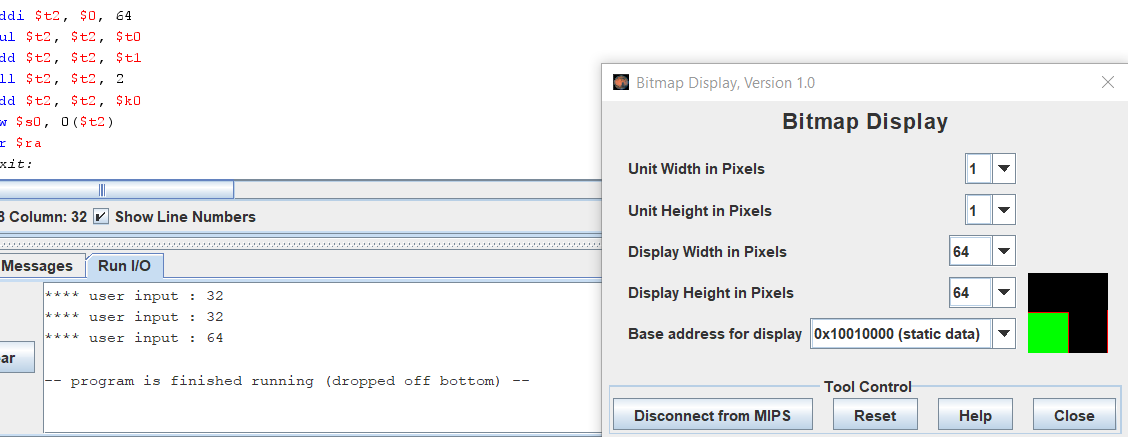
add $t2, $t2, $k0

sw $s0, 0($t2)

jr $ra

exit:

* **Kết quả**

****