**Thực hành kiến trúc máy tính tuần 36**

**Full name: Lê Quang Khải**

**Student ID: 20225638**

Assignment 1

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

                # Bit 0 = doan a;

                # Bit 1 = doan b; ...

                # Bit 7 = dau .

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

.text

main:

    li $a0, 0xCF # set value for segments

    jal SHOW\_7SEG\_LEFT # show

    nop

    li $a0, 0x7F # set value for segments

    jal SHOW\_7SEG\_RIGHT # show

    nop

exit:   li $v0, 10

    syscall

endmain:

#---------------------------------------------------------------

# Function SHOW\_7SEG\_LEFT : turn on/off the 7seg

# param[in] $a0 value to shown

# remark $t0 changed

#---------------------------------------------------------------

SHOW\_7SEG\_LEFT:

    li $t0, SEVENSEG\_LEFT # assign port's address

    sb $a0, 0($t0) # assign new value

    nop

    jr $ra

    nop

#---------------------------------------------------------------

# Function SHOW\_7SEG\_RIGHT : turn on/off the 7seg

# param[in] $a0 value to shown

# remark $t0 changed

#---------------------------------------------------------------

SHOW\_7SEG\_RIGHT:

    li $t0, SEVENSEG\_RIGHT # assign port's address

    sb $a0, 0($t0) # assign new value

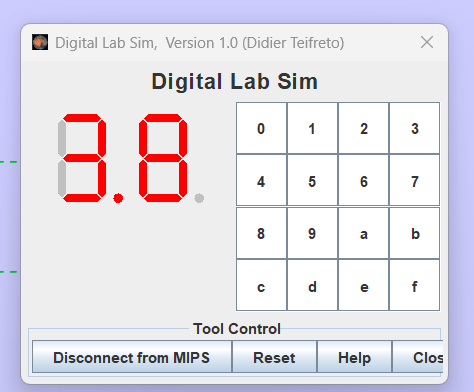
    nop

    jr $ra

    nop

**Hiển thị các số khác nhau:**

- Hiển thị 2 chữ số cuối của MSSV 20225638:



Assignment 2

.data

messenger: .asciiz "Nhap vao mot so nguyen: "

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

                # Bit 0 = doan a;

                # Bit 1 = doan b; ...

                # Bit 7 = dau .

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

.eqv SEVENSEG\_0 0x3F

.eqv SEVENSEG\_1 0x6

.eqv SEVENSEG\_2 0x5B

.eqv SEVENSEG\_3 0x4F

.eqv SEVENSEG\_4 0x66

.eqv SEVENSEG\_5 0x6D

.eqv SEVENSEG\_6 0x7D

.eqv SEVENSEG\_7 0x7

.eqv SEVENSEG\_8 0x7F

.eqv SEVENSEG\_9 0x6F

.text

input:

        li $v0, 4

        la $a0, messenger

        syscall

        li $v0, 5

        syscall

        move $t0, $v0

        li $s0, 10

main:

        div $t0, $s0

        mfhi $t1 # lấy ra chữ số cuối

        mflo $t0

        div $t0, $s0

        mfhi $t2 # lấy ra chữ số hàng chục

        move $a0, $t2  # set value for segments

        jal convert

    jal SHOW\_7SEG\_LEFT # show

    nop

        move $a0, $t1  # set value for segments

        jal convert

    jal SHOW\_7SEG\_RIGHT # show

    nop

exit:   li $v0, 10

    syscall

endmain:

SHOW\_7SEG\_LEFT:

    li $t0, SEVENSEG\_LEFT # assign port's address

    sb $a0, 0($t0) # assign new value

    nop

    jr $ra

    nop

SHOW\_7SEG\_RIGHT:

    li $t0, SEVENSEG\_RIGHT # assign port's address

    sb $a0, 0($t0) # assign new value

    nop

    jr $ra

    nop

convert:

    beq $a0, 0, zero

    beq $a0, 1, one

    beq $a0, 2, two

    beq $a0, 3, three

    beq $a0, 4, four

    beq $a0, 5, five

    beq $a0, 6, six

    beq $a0, 7, seven

    beq $a0, 8, eight

    beq $a0, 9, nine

zero:

    li $a0, SEVENSEG\_0

    jr $ra

one:

    li $a0, SEVENSEG\_1

    jr $ra

two:

    li $a0, SEVENSEG\_2

    jr $ra

three:

    li $a0, SEVENSEG\_3

    jr $ra

four:

    li $a0, SEVENSEG\_4

    jr $ra

five:

    li $a0, SEVENSEG\_5

    jr $ra

six:

    li $a0, SEVENSEG\_6

    jr $ra

seven:

    li $a0, SEVENSEG\_7

    jr $ra

eight:

    li $a0, SEVENSEG\_8

    jr $ra

nine:

    li $a0, SEVENSEG\_9

    jr $ra

**Kết quả:**

**A screenshot of a computer

Description automatically generated**

Assignment 3

.data

messenger: .asciiz "Nhap vao mot ki tu: "

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

                # Bit 0 = doan a;

                # Bit 1 = doan b; ...

                # Bit 7 = dau .

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

.eqv SEVENSEG\_0 0x3F

.eqv SEVENSEG\_1 0x6

.eqv SEVENSEG\_2 0x5B

.eqv SEVENSEG\_3 0x4F

.eqv SEVENSEG\_4 0x66

.eqv SEVENSEG\_5 0x6D

.eqv SEVENSEG\_6 0x7D

.eqv SEVENSEG\_7 0x7

.eqv SEVENSEG\_8 0x7F

.eqv SEVENSEG\_9 0x6F

.text

input:

        li $v0, 4

        la $a0, messenger

        syscall

        li $v0, 12

        syscall

        move $t0, $v0

        li $s0, 10

main:

        div $t0, $s0

        mfhi $t1 # lấy ra chữ số cuối

        mflo $t0

        div $t0, $s0

        mfhi $t2 # lấy ra chữ số hàng chục

        move $a0, $t2  # set value for segments

        jal convert

    jal SHOW\_7SEG\_LEFT # show

    nop

        move $a0, $t1  # set value for segments

        jal convert

    jal SHOW\_7SEG\_RIGHT # show

    nop

exit:   li $v0, 10

    syscall

endmain:

SHOW\_7SEG\_LEFT:

    li $t0, SEVENSEG\_LEFT # assign port's address

    sb $a0, 0($t0) # assign new value

    nop

    jr $ra

    nop

SHOW\_7SEG\_RIGHT:

    li $t0, SEVENSEG\_RIGHT # assign port's address

    sb $a0, 0($t0) # assign new value

    nop

    jr $ra

    nop

convert:

    beq $a0, 0, zero

    beq $a0, 1, one

    beq $a0, 2, two

    beq $a0, 3, three

    beq $a0, 4, four

    beq $a0, 5, five

    beq $a0, 6, six

    beq $a0, 7, seven

    beq $a0, 8, eight

    beq $a0, 9, nine

zero:

    li $a0, SEVENSEG\_0

    jr $ra

one:

    li $a0, SEVENSEG\_1

    jr $ra

two:

    li $a0, SEVENSEG\_2

    jr $ra

three:

    li $a0, SEVENSEG\_3

    jr $ra

four:

    li $a0, SEVENSEG\_4

    jr $ra

five:

    li $a0, SEVENSEG\_5

    jr $ra

six:

    li $a0, SEVENSEG\_6

    jr $ra

seven:

    li $a0, SEVENSEG\_7

    jr $ra

eight:

    li $a0, SEVENSEG\_8

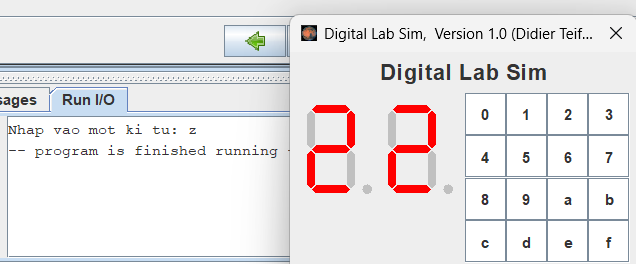
    jr $ra

nine:

    li $a0, SEVENSEG\_9

    jr $ra

**Kết quả:**

****

Assignment 4

.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

.eqv MAGENTA 0x00FF00FF

.eqv CYAN 0x0000FFFF

.eqv PURPLE 0x00800080

.text

    li $k0, MONITOR\_SCREEN #Nap dia chi bat dau cua man hinh

    li $t1, 0 # counter

    li $t2, 0

    li $t3, 2

    li $t7, 8

loop:

    beq $t1, $t7, next\_line

    li $t0, PURPLE

    sw $t0, 0($k0)

    addi $t1, $t1, 1

    addi $k0, $k0, 4

    li $t0, CYAN

    sw $t0, 0($k0)

    addi $t1, $t1, 1

    addi $k0, $k0, 4

    j loop

next\_line:

    beq $t2, $t7, endmain

    li $t1, 0

    addi $t2, $t2, 1

    div $t2, $t3

    mfhi $s0

    beq $s0, $zero, loop

    j loop\_2

loop\_2:

    beq $t1, $t7, next\_line

    li $t0, C

YAN

    sw $t0, 0($k0)

    addi $t1, $t1, 1

    addi $k0, $k0, 4

    li $t0, PURPLE

    sw $t0, 0($k0)

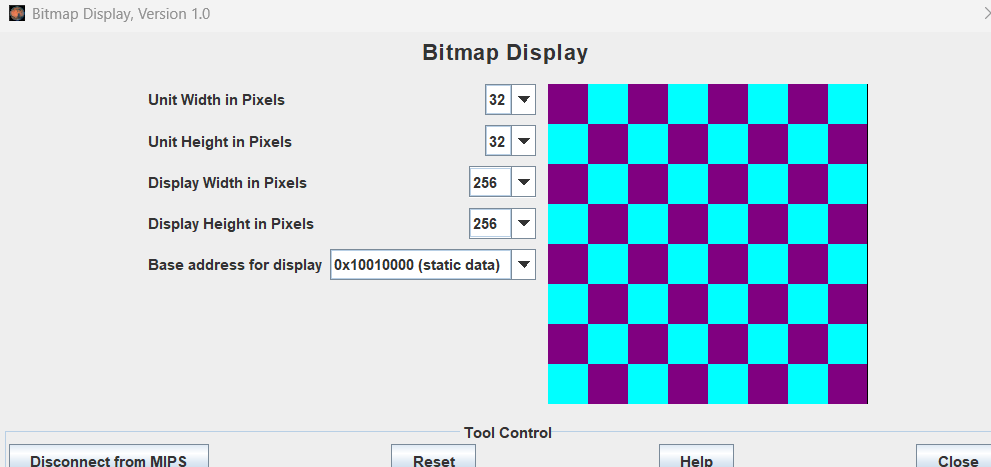
    addi $t1, $t1, 1

    addi $k0, $k0, 4

    j loop\_2

endmain:

**Kết quả:**

****

Assignment 5

.data

nhapX1: .asciiz "Nhap x1: "

nhapY1: .asciiz "Nhap y1: "

nhapX2: .asciiz "Nhap x2: "

nhapY2: .asciiz "Nhap y2: "

.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

.text

    li $k0, MONITOR\_SCREEN #Nap dia chi bat dau cua man hinh

    li $t1, 0

    li $t2, 0

input:

       li $v0, 4

       la $a0, nhapX1

       syscall

       li $v0, 5

       syscall

       move $s0, $v0    # x1 = s0

       li $v0, 4

       la $a0, nhapY1

       syscall

       li $v0, 5

       syscall

       move $s1, $v0    # y1 = s1

       li $v0, 4

       la $a0, nhapX2

       syscall

       li $v0, 5

       syscall

       move $s2, $v0    # x2 = s2

       li $v0, 4

       la $a0, nhapY2

       syscall

       li $v0, 5

       syscall

       move $s3, $v0    # y2 = s3

compare\_x:

    bgt $s0, $s2, set\_max\_x

compare\_y:

    bgt $s1, $s3, set\_max\_y

start:

    move $t0, $s0 # current x

    move $t1, $s1 # current y

draw\_border:

    bgt $t0, $s2, next\_line\_border

    bgt $t1, $s3, start\_fill

    mul $t2, $t1, 8

    add $t2, $t2, $t0

    mul $t2, $t2, 4

    add $k1, $k0, $t2

    li $t3, RED

    sw $t3, 0($k1)

    addi $t0, $t0, 1

    j draw\_border

next\_line\_border:

    move $t0, $s0

    addi $t1, $t1, 1

    j draw\_border

start\_fill:

    addi $s0, $s0, 1

    addi $s1, $s1, 1

    subi $s2, $s2, 1

    subi $s3, $s3, 1

    move $t0, $s0 # current x

    move $t1, $s1 # current y

fill:

    bgt $t0, $s2, next\_line\_fill

    bgt $t1, $s3, fill

    mul $t2, $t1, 8

    add $t2, $t2, $t0

    mul $t2, $t2, 4

    add $k1, $k0, $t2

    li $t3, GREEN

    sw $t3, 0($k1)

    addi $t0, $t0, 1

    j fill

next\_line\_fill:

    move $t0, $s0

    addi $t1, $t1, 1

    j fill

set\_max\_x:

    move $t0, $s0

    move $s0, $s2

    move $s2, $t0

    j compare\_y

set\_max\_y:

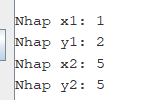
    move $t0, $s0

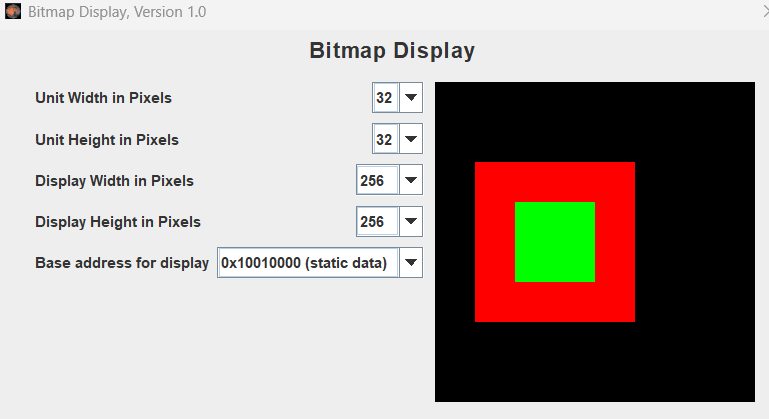
    move $s1, $s3

    move $s3, $t0

    j start

**Kết quả:**

****

****