**BÀI BÁO CÁO TUẦN 10**

Họ và tên: Nguyễn Trọng Khánh Duy

MSSV: 20210284

***Assignment 1:***

**CODE:**

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

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

.text

main:

# MSSV 20210284 => 2 so cuoi la 84

# so 8 khi chuyen sang den led 7 doan : 0111 1111 (bit) = 7F (hex)

# so 4 khi chuyen sang den led 7 doan : 0110 0110 (bit) = 66 (hex)

li $a0, 0x7F # set value for segments

jal SHOW\_7SEG\_LEFT # show

nop

li $a0, 0x66 # set value for segments

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

**Kết quả:**

A screenshot of a computer

Description automatically generated

***Assignment 2***

**CODE:**

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

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

.data

nhapso: .asciiz "Nhap so nguyen n = "

.text

main:

li $v0,4

la $a0,nhapso

syscall

li $v0, 5

syscall

add $s1, $v0, $zero # n = $s1

div $s2, $s1, 10

mfhi $t1 # so hang don vi

div $s3, $s2, 10

mfhi $t2 # so hang chuc

addi $s4, $zero, -1 # khoi tao check = -1

add $t3, $zero, $t2 # $t3 = so hang chuc

Check\_so:

So\_0: bne $t3, 0, So\_1

li $a0, 0x3F

j in\_so

So\_1: bne $t3, 1, So\_2

li $a0, 0x06

j in\_so

So\_2: bne $t3, 2, So\_3

li $a0, 0x5B

j in\_so

So\_3: bne $t3, 3, So\_4

li $a0, 0x4F

j in\_so

So\_4: bne $t3, 4, So\_5

li $a0, 0x66

j in\_so

So\_5: bne $t3, 5, So\_6

li $a0, 0x6D

j in\_so

So\_6: bne $t3, 6, So\_7

li $a0, 0x7D

j in\_so

So\_7: bne $t3, 7, So\_8

li $a0, 0x07

j in\_so

So\_8: bne $t3, 8, So\_9

li $a0, 0x7F

j in\_so

So\_9: bne $t3, 9, exit

li $a0, 0x6F

in\_so:

beq $s4, $zero, in\_so\_don\_vi# if check = -1 in so hang chuc

# if check = 0 in so hang don vi

in\_so\_chuc:

jal SHOW\_7SEG\_LEFT # show

nop

addi $s4, $s4, 1

add $t3, $zero, $t1 # $t3 = so hang don vi

j Check\_so

in\_so\_don\_vi:

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

A picture containing text, font, screenshot, line

Description automatically generated**Kết quả:**

TH1: Khi nhập số có 1 chữ số:

A screenshot of a computer

Description automatically generated

A picture containing text, font, white, typography

Description automatically generatedTH2: Khi nhập một số có 2 chữ số:

A screenshot of a computer

Description automatically generated

TH3: Khi nhập một số có 3 chữ số:

A picture containing text, font, white, typography

Description automatically generated

A screenshot of a computer

Description automatically generated

***Assignment 3***

**CODE:**

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

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

.data

nhapkytu: .asciiz "Nhap ky tu: "

.text

main:

li $v0,4

la $a0,nhapkytu

syscall

li $v0, 12

syscall

add $s1, $v0, $zero # n = $s1

div $s2, $s1, 10

mfhi $t1 # so hang don vi

div $s3, $s2, 10

mfhi $t2 # so hang chuc

addi $s4, $zero, -1 # khoi tao check = -1

add $t3, $zero, $t2 # $t3 = so hang chuc

Check\_so:

So\_0: bne $t3, 0, So\_1

li $a0, 0x3F

j in\_so

So\_1: bne $t3, 1, So\_2

li $a0, 0x06

j in\_so

So\_2: bne $t3, 2, So\_3

li $a0, 0x5B

j in\_so

So\_3: bne $t3, 3, So\_4

li $a0, 0x4F

j in\_so

So\_4: bne $t3, 4, So\_5

li $a0, 0x66

j in\_so

So\_5: bne $t3, 5, So\_6

li $a0, 0x6D

j in\_so

So\_6: bne $t3, 6, So\_7

li $a0, 0x7D

j in\_so

So\_7: bne $t3, 7, So\_8

li $a0, 0x07

j in\_so

So\_8: bne $t3, 8, So\_9

li $a0, 0x7F

j in\_so

So\_9: bne $t3, 9, exit

li $a0, 0x6F

in\_so:

beq $s4, $zero, in\_so\_don\_vi# if check = -1 in so hang chuc

# if check = 0 in so hang don vi

in\_so\_chuc:

jal SHOW\_7SEG\_LEFT # show

nop

addi $s4, $s4, 1

add $t3, $zero, $t1 # $t3 = so hang don vi

j Check\_so

in\_so\_don\_vi:

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

**Kết quả:**

Nhập ký tự thường:

A screenshot of a computer

Description automatically generated



Nhập ký tự in hoa:

A screenshot of a computer

Description automatically generated with medium confidence

***Assignment 4***

**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 hinh

li $t2, 2 # khoi tao 2

li $t4, 8 # khoi tao 8

li $t1, -1 # i = -1

li $t4, 0 # j = 0

li $s1, 4 # $s1 = 4

add $k0, $k0, -4 # $k0 = $k0 - 4

FOR: addi $t1, $t1, 1 # i = 0, i ++

addi $t4, $t4, 1 # j++

beq $t1, 72, EXIT # i = 72 stop

add $k0, $k0, $s1 # $k0 += 4

div $t1, $t2 # i / 2

mfhi $t3 # $t3 = i % 2

bne $t4, 8, continue # j = 8 => i++

li $t4, 0 # j = 0

addi $t1, $t1, 1 # i++

continue:

beq $t3, $zero, doi\_mau

li $t0, RED

sw $t0, 0($k0)

nop

j FOR

doi\_mau:

li $t0, BLUE

sw $t0, 0($k0)

nop

j FOR

EXIT: li $v0, 10

syscall

**Giải thích code:**

- Dùng một biến i chạy từ 0 – 71 thì dừng

- Dùng một biến j chạy từ 1 – 8

- Khi i /2 dư 1 thì sẽ in màu đỏ, i / 2 dư 0 thì in màu xanh

- Vì bảng là 8x8 nên mỗi lần j = 8 sẽ tăng i thêm 1 => i sẽ tăng 2 đơn vị khi xuống dòng => khi xuống dòng i sẽ giữ nguyên màu. Vì có 8 dòng nên sẽ tăng i lên 8 đơn vị nên phải đặt cho i dừng khi i = 72

**Kết quả:**

A screenshot of a computer

Description automatically generated with medium confidence

***Assignment 5***

**CODE:**

.eqv MONITOR\_SCREEN 0x10010000

.eqv RED 0x00FF0000

.eqv GREEN 0x0000FF00

.data

x1: .asciiz "Nhap x1: "

y1: .asciiz "Nhap y1: "

x2: .asciiz "Nhap x2: "

y2: .asciiz "Nhap y2: "

error1: .asciiz "Error: x2 phai khac x1. Moi nhap lai!\n"

error2: .asciiz "Error: y2 phai khac y1. Moi nhap lai!\n"

.text

li $k0, MONITOR\_SCREEN

li $v0, 4

la $a0, x1

syscall

li $v0, 5

syscall

move $s0, $v0

li $v0, 4

la $a0, y1

syscall

li $v0, 5

syscall

move $s1, $v0

NhapX2: li $v0, 4

la $a0, x2

syscall

li $v0, 5

syscall

move $s2, $v0

beq $s2, $s0, Error1

NhapY2: li $v0, 4

la $a0, y2

syscall

li $v0, 5

syscall

move $s3, $v0

beq $s3, $s1, Error2

j Tsugi

Error1: li $v0, 4

la $a0, error1

syscall

j NhapX2

Error2: li $v0, 4

la $a0, error2

syscall

j NhapY2

Tsugi:

slt $t0, $s0, $s2

slt $t1, $s1, $s3

beq $t0, 0, Case3

beq $t1, 0, Case2

Case1: add $v0, $s1, $zero

For1: bgt $v0, $s3, Exit

add $v1, $s0, $zero

For2: bgt $v1, $s2, EndFor2

beq $v0, $s1, InVien1

beq $v0, $s3, InVien1

beq $v1, $s0, InVien1

beq $v1, $s2, InVien1

sll $t8, $v0, 6

add $t8, $t8, $v1

sll $t8, $t8, 2

li $a1, GREEN

add $a2, $k0, $t8

sw $a1, 0($a2)

add $v1, $v1, 1

j For2

InVien1: sll $t8, $v0, 6

add $t8, $t8, $v1

sll $t8, $t8, 2

li $a1, RED

add $a2, $k0, $t8

sw $a1, 0($a2)

add $v1, $v1, 1

j For2

EndFor2:

add $v0, $v0, 1

j For1

Case2: add $v0, $s3, $zero

For3: bgt $v0, $s1, Exit

add $v1, $s0, $zero

For4: bgt $v1, $s2, EndFor4

beq $v0, $s1, InVien2

beq $v0, $s3, InVien2

beq $v1, $s0, InVien2

beq $v1, $s2, InVien2

sll $t8, $v0, 6

add $t8, $t8, $v1

sll $t8, $t8, 2

li $a1, GREEN

add $a2, $k0, $t8

sw $a1, 0($a2)

add $v1, $v1, 1

j For4

InVien2:sll $t8, $v0, 6

add $t8, $t8, $v1

sll $t8, $t8, 2

li $a1, RED

add $a2, $k0, $t8

sw $a1, 0($a2)

add $v1, $v1, 1

j For4

EndFor4:

add $v0, $v0, 1

j For3

Case3: beq $t1, 0, Case4

add $v0, $s1, $zero

For5: bgt $v0, $s3, Exit

add $v1, $s2, $zero

For6: bgt $v1, $s0, EndFor6

beq $v0, $s1, InVien3

beq $v0, $s3, InVien3

beq $v1, $s0, InVien3

beq $v1, $s2, InVien3

sll $t8, $v0, 6

add $t8, $t8, $v1

sll $t8, $t8, 2

li $a1, GREEN

add $a2, $k0, $t8

sw $a1, 0($a2)

add $v1, $v1, 1

j For6

InVien3:sll $t8, $v0, 6

add $t8, $t8, $v1

sll $t8, $t8, 2

li $a1, RED

add $a2, $k0, $t8

sw $a1, 0($a2)

add $v1, $v1, 1

j For6

EndFor6:

add $v0, $v0, 1

j For5

Case4: add $v0, $s3, $zero

For7: bgt $v0, $s1, Exit

add $v1, $s2, $zero

For8: bgt $v1, $s0, EndFor8

beq $v0, $s1, InVien4

beq $v0, $s3, InVien4

beq $v1, $s0, InVien4

beq $v1, $s2, InVien4

sll $t8, $v0, 6

add $t8, $t8, $v1

sll $t8, $t8, 2

li $a1, GREEN

add $a2, $k0, $t8

sw $a1, 0($a2)

add $v1, $v1, 1

j For8

InVien4:sll $t8, $v0, 6

add $t8, $t8, $v1

sll $t8, $t8, 2

li $a1, RED

add $a2, $k0, $t8

sw $a1, 0($a2)

add $v1, $v1, 1

j For8

EndFor8:

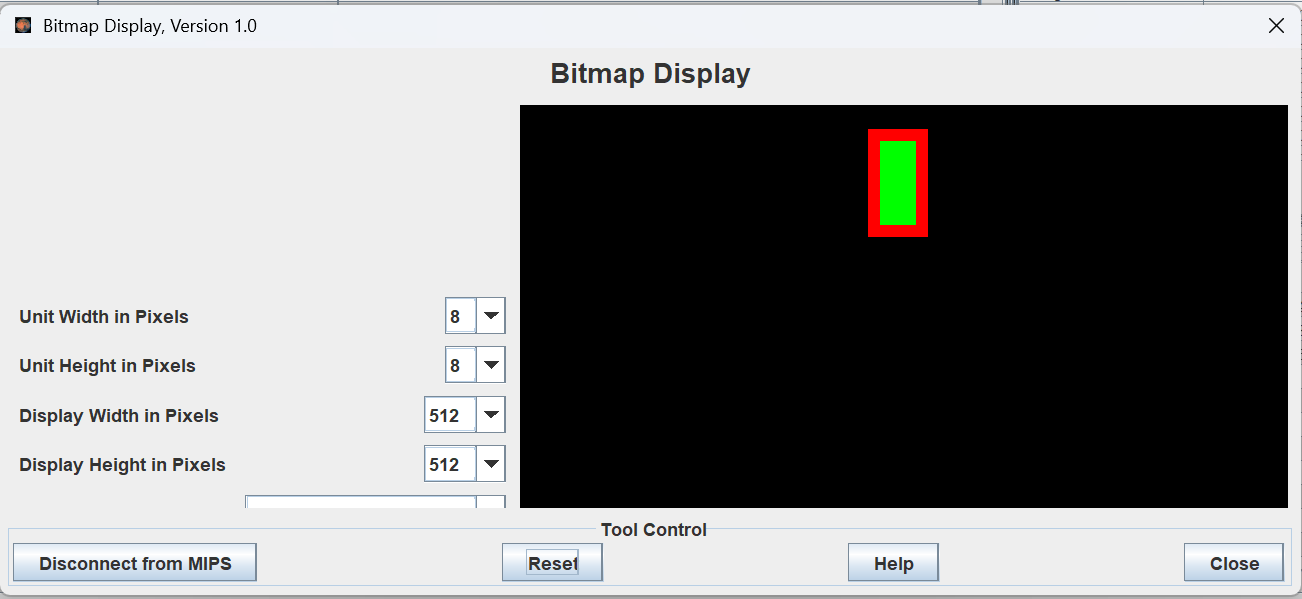
add $v0, $v0, 1

j For7

Exit: li $v0, 10

syscall

**Kết quả:**



A screenshot of a computer

Description automatically generated with medium confidence