**BÀI THỰC HÀNH TUẦN 10**

**KIẾN TRÚC MÁY TÍNH**

Họ và tên: Đinh Huy Dương

MSSV: 20215020

**Bài 1:**

.eqv SEVENSEG\_LEFT 0xFFFF0011

.eqv SEVENSEG\_RIGHT 0xFFFF0010 phai

.text

main:

li $a0, 0x5b # set value for segments

jal SHOW\_7SEG\_LEFT # show

nop

li $a0, 0x3F # 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

Kết quả:

A screenshot of a computer

Description automatically generated with medium confidence

**Bài 2:**

.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

.data

mess: .asciiz "Insert the integer: "

.text

main:

jal input

nop

add $t1,$0,$v0

li $s0,10

div $t1,$s0

mfhi $t3

mflo $t1

jal check

nop

jal SHOW\_7SEG\_RIGHT # show

nop

div $t1,$s0

mfhi $t3

jal check

nop

jal SHOW\_7SEG\_LEFT # 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

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

input:

li $v0,4

la $a0,mess

syscall

li $v0,5

syscall

jr $ra

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

check:

li $t2,0

beq $t3,$t2,case0

addi $t2,$t2,1

beq $t3,$t2,case1

addi $t2,$t2,1

beq $t3,$t2,case2

addi $t2,$t2,1

beq $t3,$t2,case3

addi $t2,$t2,1

beq $t3,$t2,case4

addi $t2,$t2,1

beq $t3,$t2,case5

addi $t2,$t2,1

beq $t3,$t2,case6

addi $t2,$t2,1

beq $t3,$t2,case7

addi $t2,$t2,1

beq $t3,$t2,case8

addi $t2,$t2,1

beq $t3,$t2,case9

case0:

li $a0,0x3f

j out

case1:

li $a0,0x6

j out

case2:

li $a0,0x5b

j out

case3:

li $a0,0x4f

j out

case4:

li $a0,0x66

j out

case5:

li $a0,0x6d

j out

case6:

li $a0,0x7d

j out

case7:

li $a0,0x7

j out

case8:

li $a0,0x7f

j out

case9:

li $a0,0x6f

j out

out:

jr $ra

Kết quả:

A screenshot of a computer

Description automatically generated with medium confidence

**Bài 3:**

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

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

.data

prompt: .asciiz "Enter a letter: "

array: .word 0x3f, 0x6, 0x5b, 0x4f, 0x66, 0x6d, 0x79, 0x7, 0x7f, 0x6f

.text

main:

# Display prompt to enter an integer

la $s0, array

li $v0, 4

la $a0, prompt

syscall

# Read integer from user

read\_letter:

li $v0, 12

syscall

move $s1, $v0

# Store the input value in $s1

decode: div $s1, $s1, 10

mfhi $t1 #Remainder stored in $t1

sll $t2, $t1, 2

add $t2, $s0, $t2

lw $a0, 0($t2)

add $t3, $t3, 1

beq $t3, 1, SHOW\_7SEG\_LEFT

nop

beq $t3, 2, SHOW\_7SEG\_RIGHT

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

j decode

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

# 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

j decode

# Exit the program

li $v0, 10

syscall

**A screenshot of a computer

Description automatically generated**

**Bài 4:**

**.**eqv MONITOR\_SCREEN 0x10010000

.eqv WHITE 0x00FFFFFF

.eqv GREEN 0x0000FF00

.text

li $k0, MONITOR\_SCREEN

li $t1, 0 # Counter for rows

draw\_chessboard:

li $t2, 0 # Counter for columns

inner\_loop:

mul $t3, $t1, 8

add $t3, $t3, $t2

sll $t3, $t3, 2

andi $t4, $t1, 1 # Check if row is even or odd

andi $t5, $t2, 1 # Check if column is even or odd

xor $t4, $t4, $t5

beqz $t4, blanc

nop

li $t0, GREEN

j set\_color

nop

blanc:

li $t0, WHITE

set\_color:

add $s0, $k0, $t3

sw $t0, 0($s0)

addi $t2, $t2, 1 # Increment column counter

slti $t4, $t2, 8 # Check if column < 8

bnez $t4, inner\_loop

nop

addi $t1, $t1, 1

slti $t4, $t1, 8

bnez $t4, draw\_chessboard

nop

end:

**A screenshot of a computer

Description automatically generated**