**THỰC HÀNH KIẾN TRÚC MÁY TÍNH**

**LÊ QUỐC ĐẢNG**

**20225801**

**Assignment 1**

Hiển thị số khác nhau

* Code

# Định nghĩa địa chỉ của các màn hình bảy đoạn

.eqv SEVENSEG\_LEFT 0xFFFF0011 # Địa chỉ của màn hình bảy đoạn bên trái

.eqv SEVENSEG\_RIGHT 0xFFFF0010 # Địa chỉ của màn hình bảy đoạn bên phải

.text

main:

show1: li $a0,0x06 # set value for segments

jal SHOW\_7SEG\_LEFT # show

nop

show5: li $a0,0x6D # set value for segments

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

sb $a0, 0($t0) # assign new value

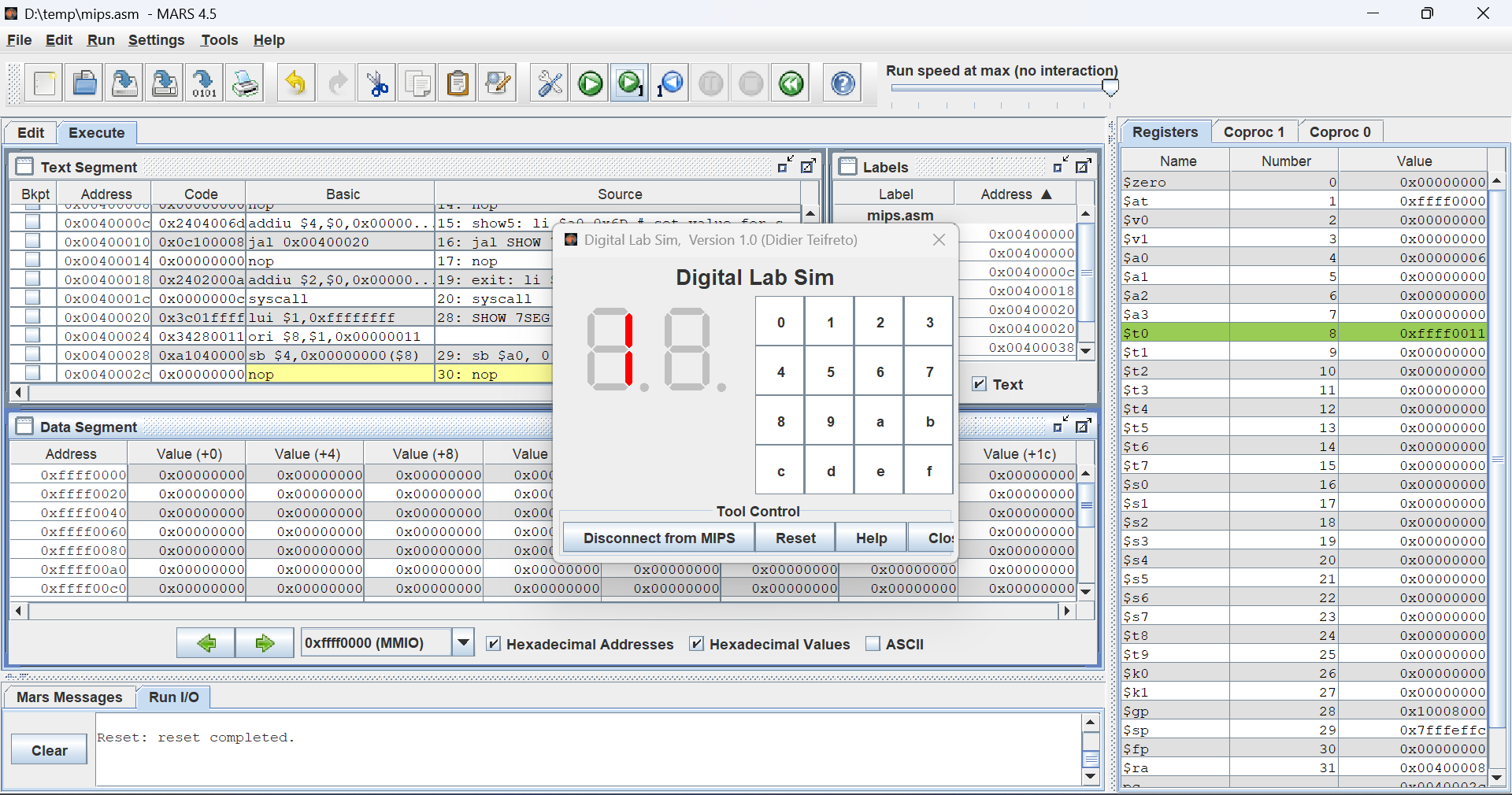
nop

jr $ra

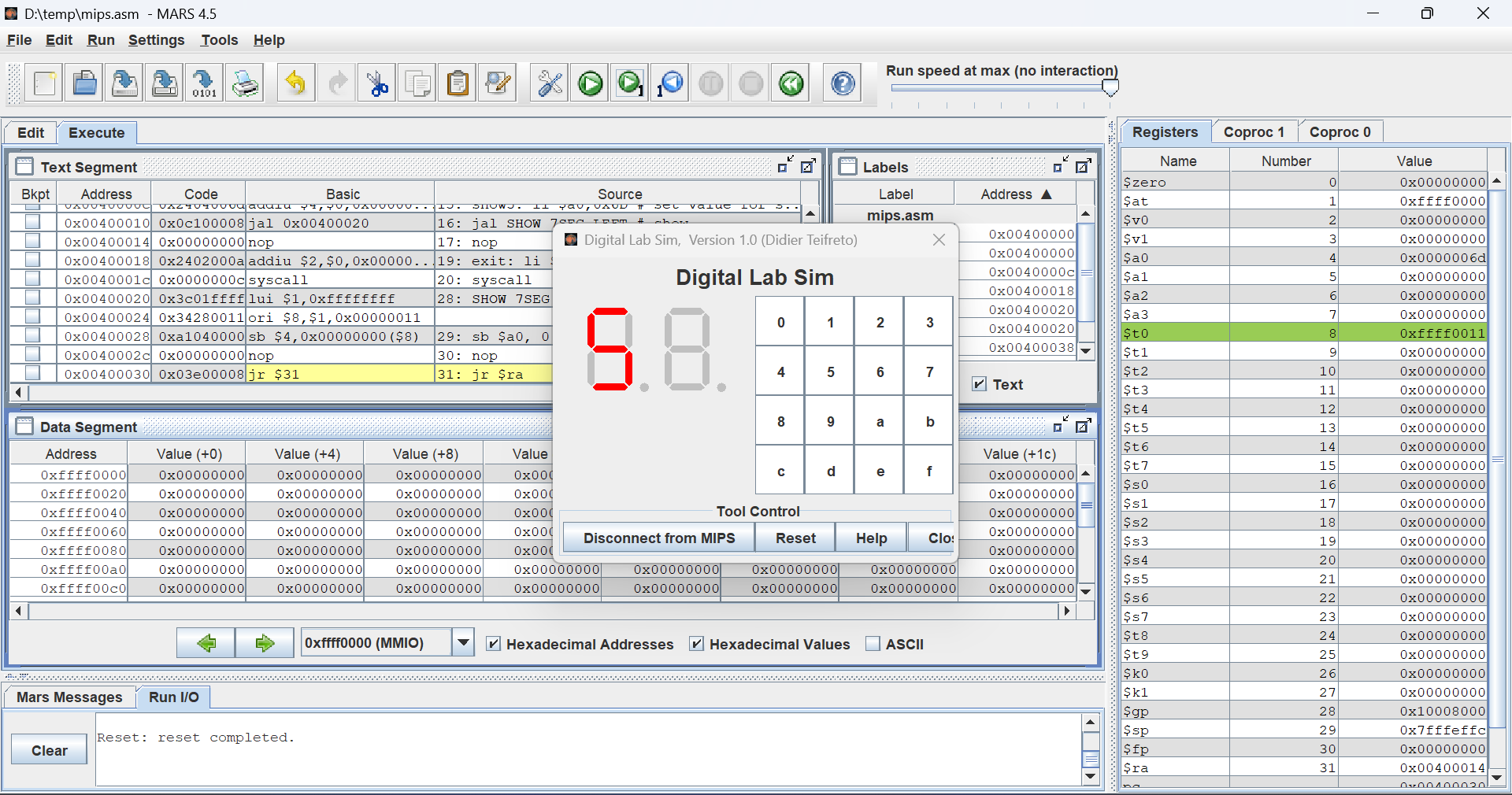
nop

* Kết quả

Hiển thị số 1



Hiển thị số 5



Chương trình đếm từ 0->9->0

* Code

# Định nghĩa địa chỉ của các màn hình bảy đoạn

.eqv SEVENSEG\_LEFT 0xFFFF0011 # Địa chỉ của màn hình bảy đoạn bên trái

.eqv SEVENSEG\_RIGHT 0xFFFF0010 # Địa chỉ của màn hình bảy đoạn bên phải

.data

# Định nghĩa các mẫu bit cho các số 0-9

SEVENSEG\_PATTERNS: .word 0x3F,0x06,0x5B,0x4F,0x66,0x6D,0x7D,0x07,0x7F,0x6F

Aend: .word

.text

main:

la $t1, SEVENSEG\_PATTERNS # Địa chỉ của mảng các mẫu bit

la $t2, Aend

reset: #Khởi tạo về giá trị 0

add $t3,$t1,$zero

loop:

beq $t3,$t2,reset #Nếu hết mảng thì reset

lw $a0,0($t3) # set value for segments

jal SHOW\_7SEG\_LEFT # show

nop

addi $t3,$t3,4 #Tăng lên số tiếp theo

j loop

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

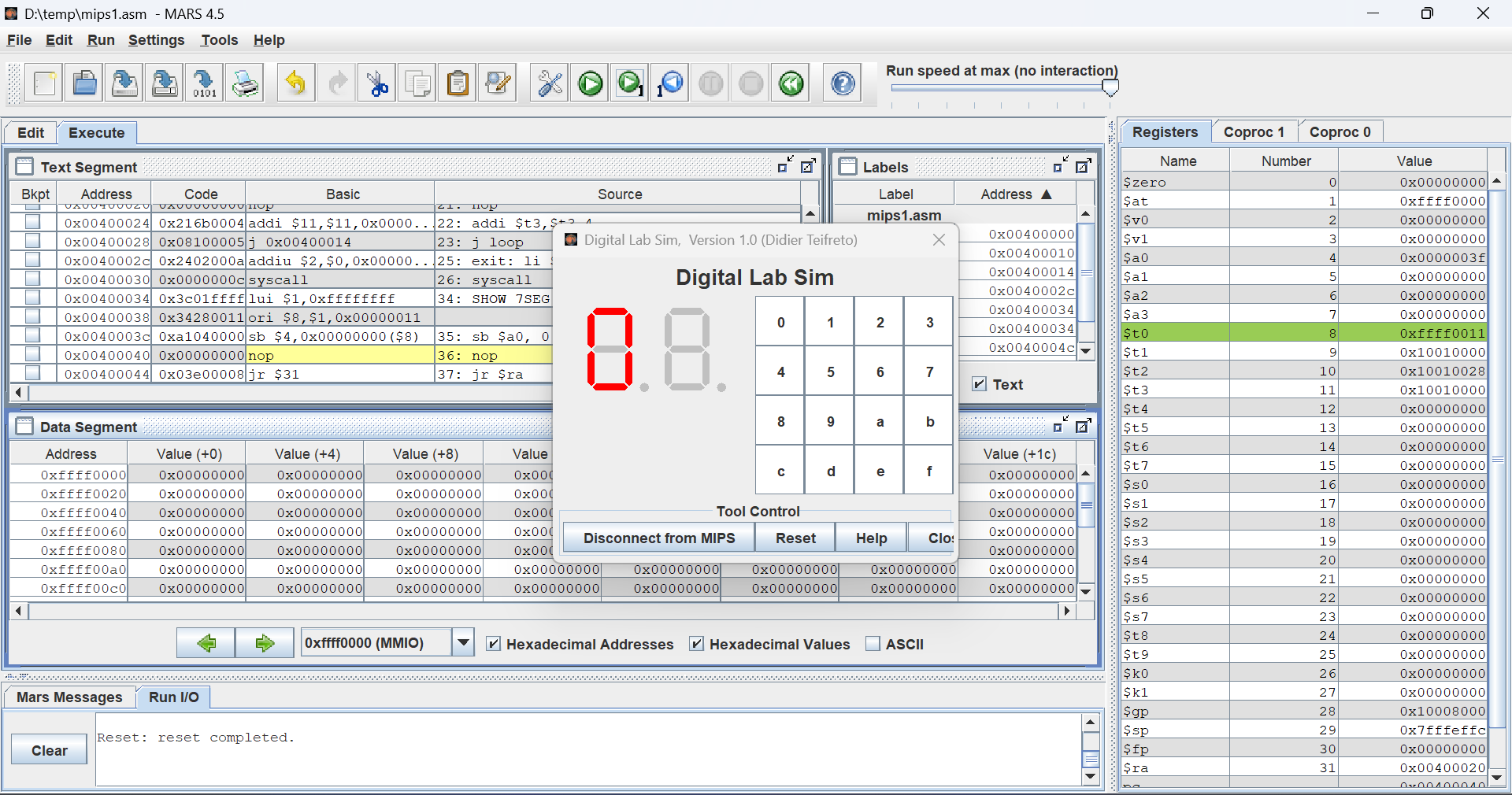
nop

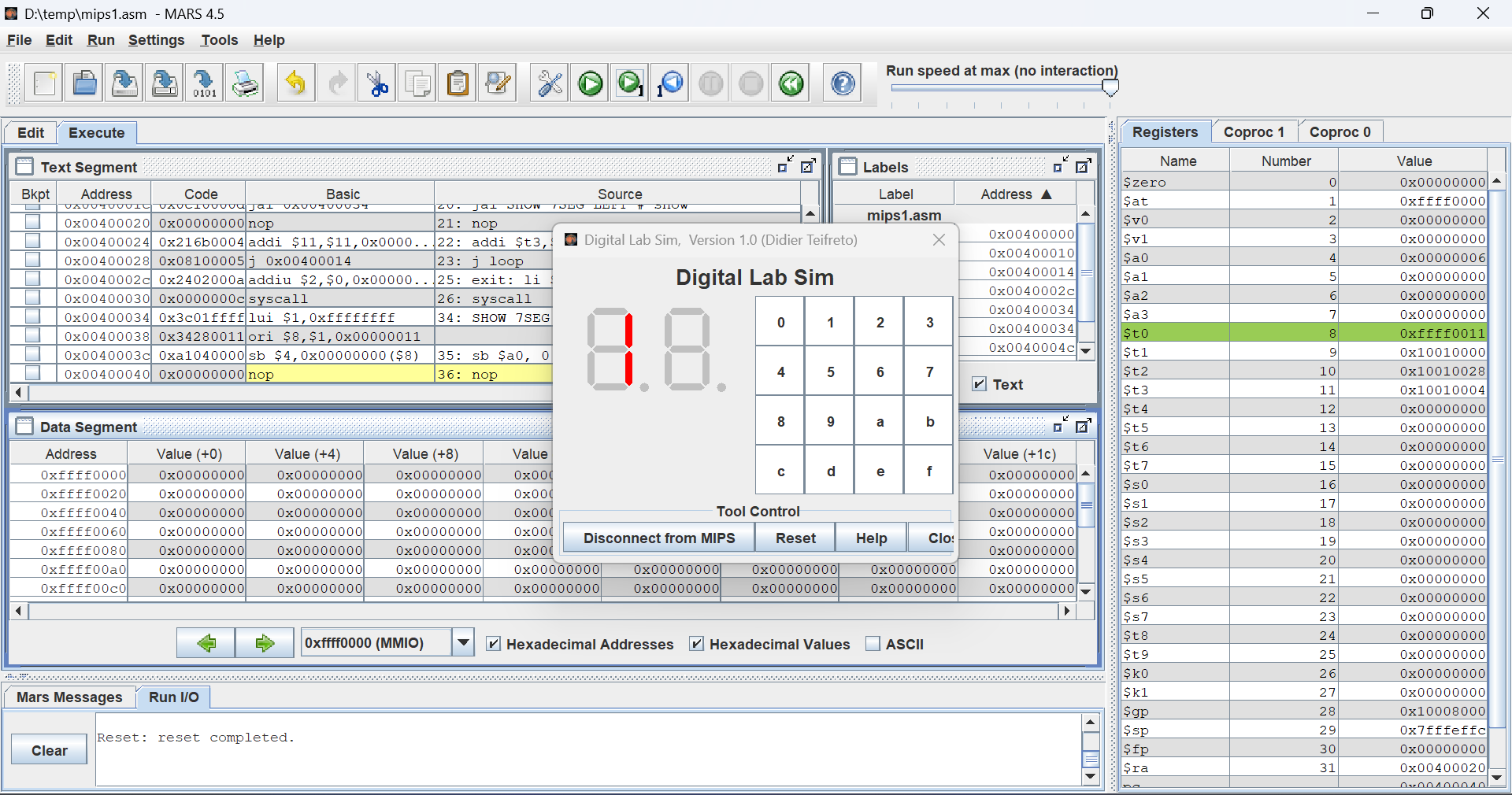
jr $ra

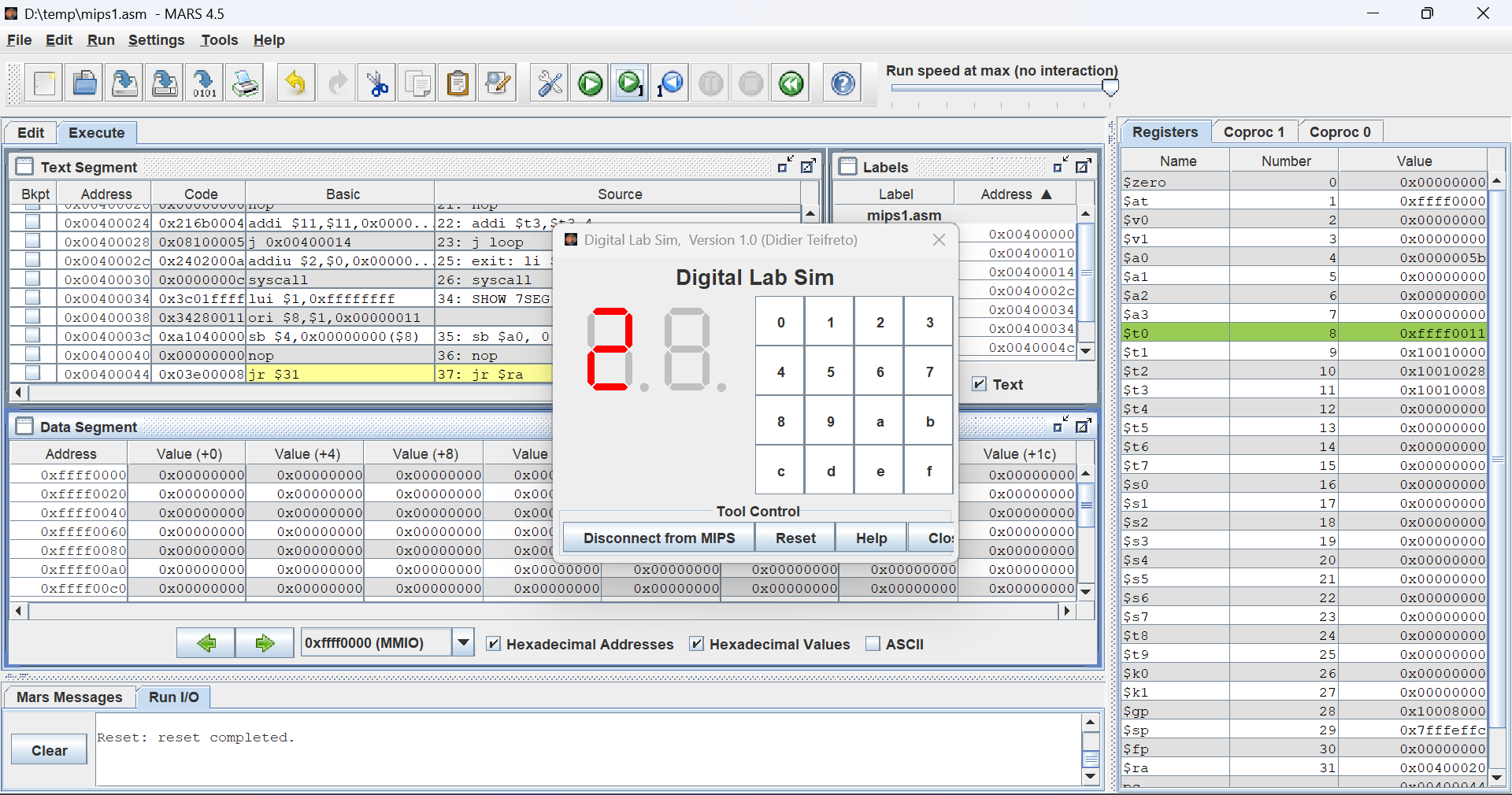
nop

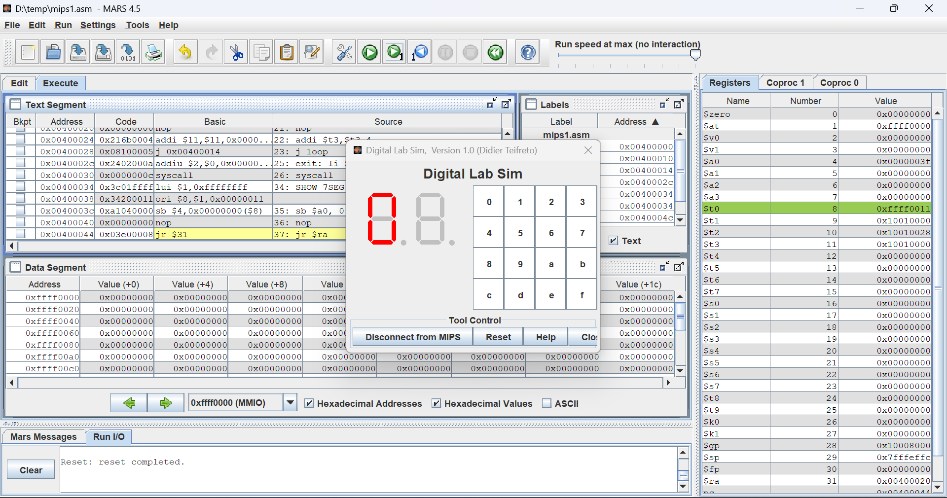
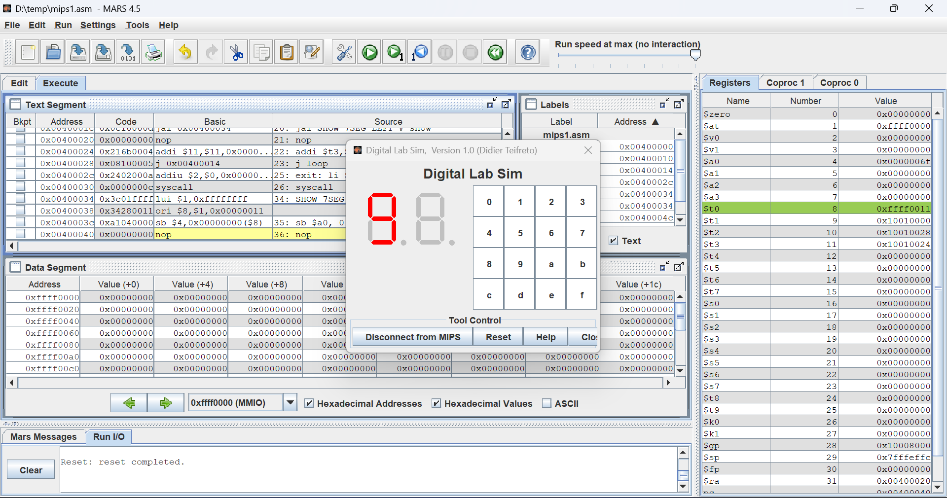
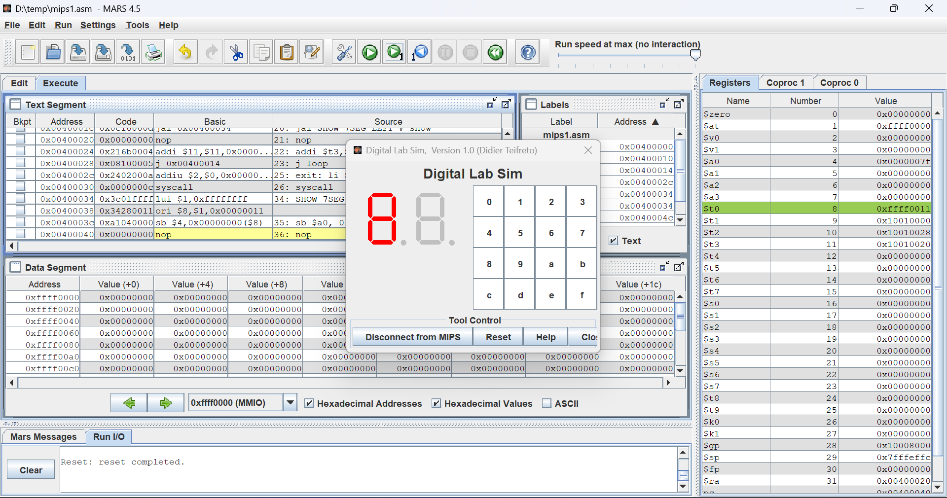
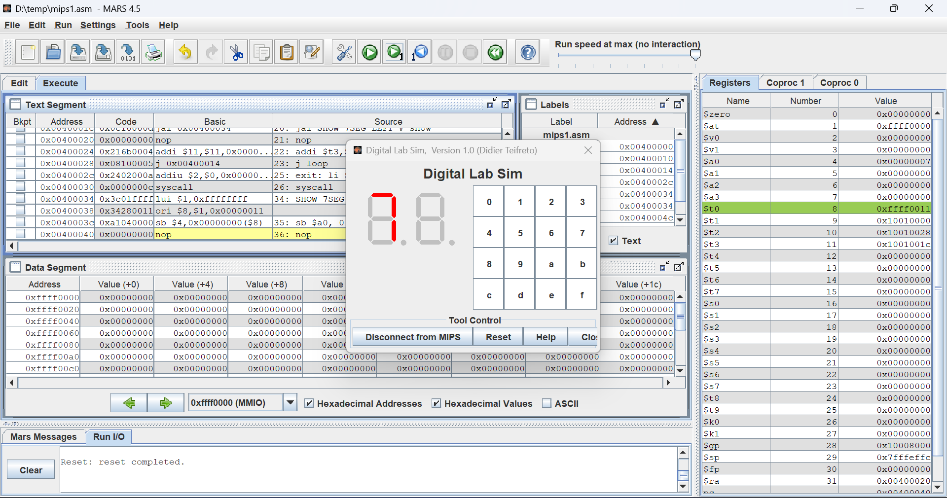
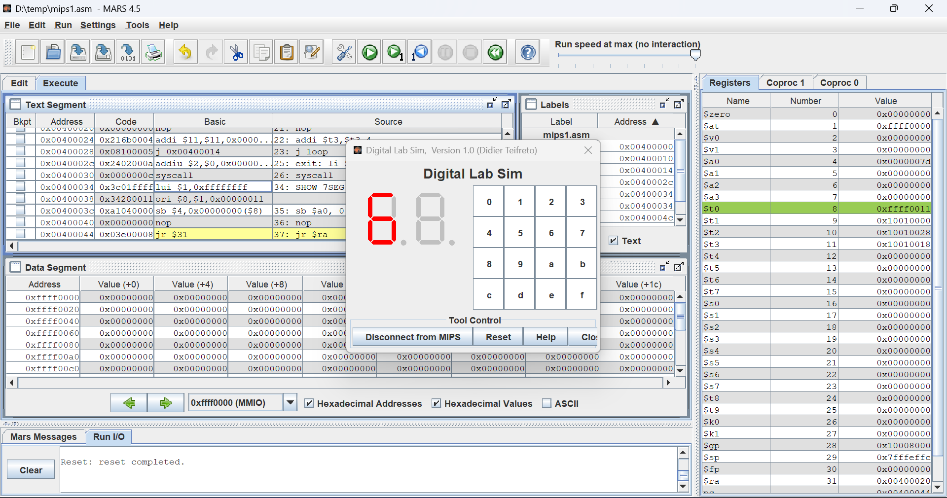
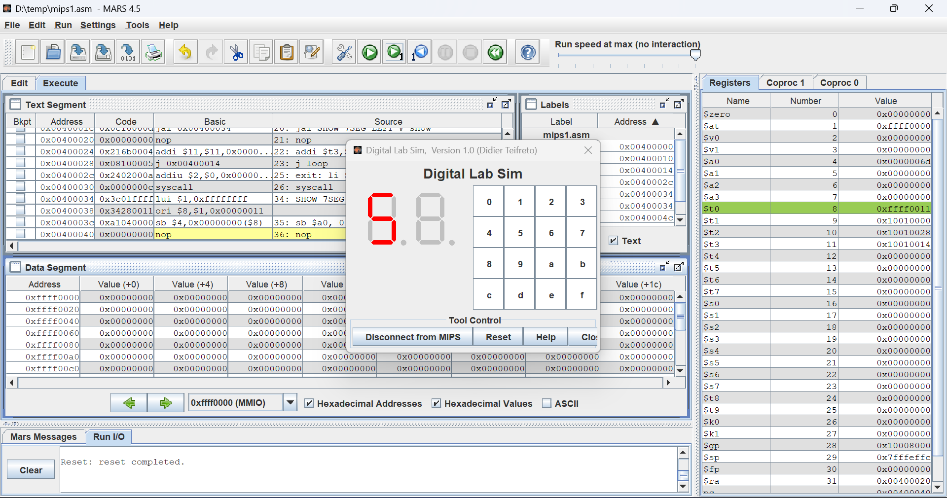
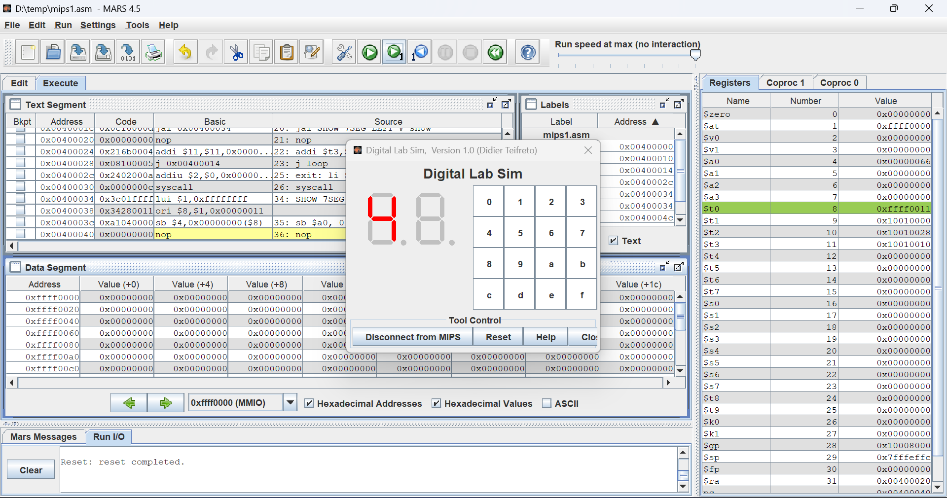
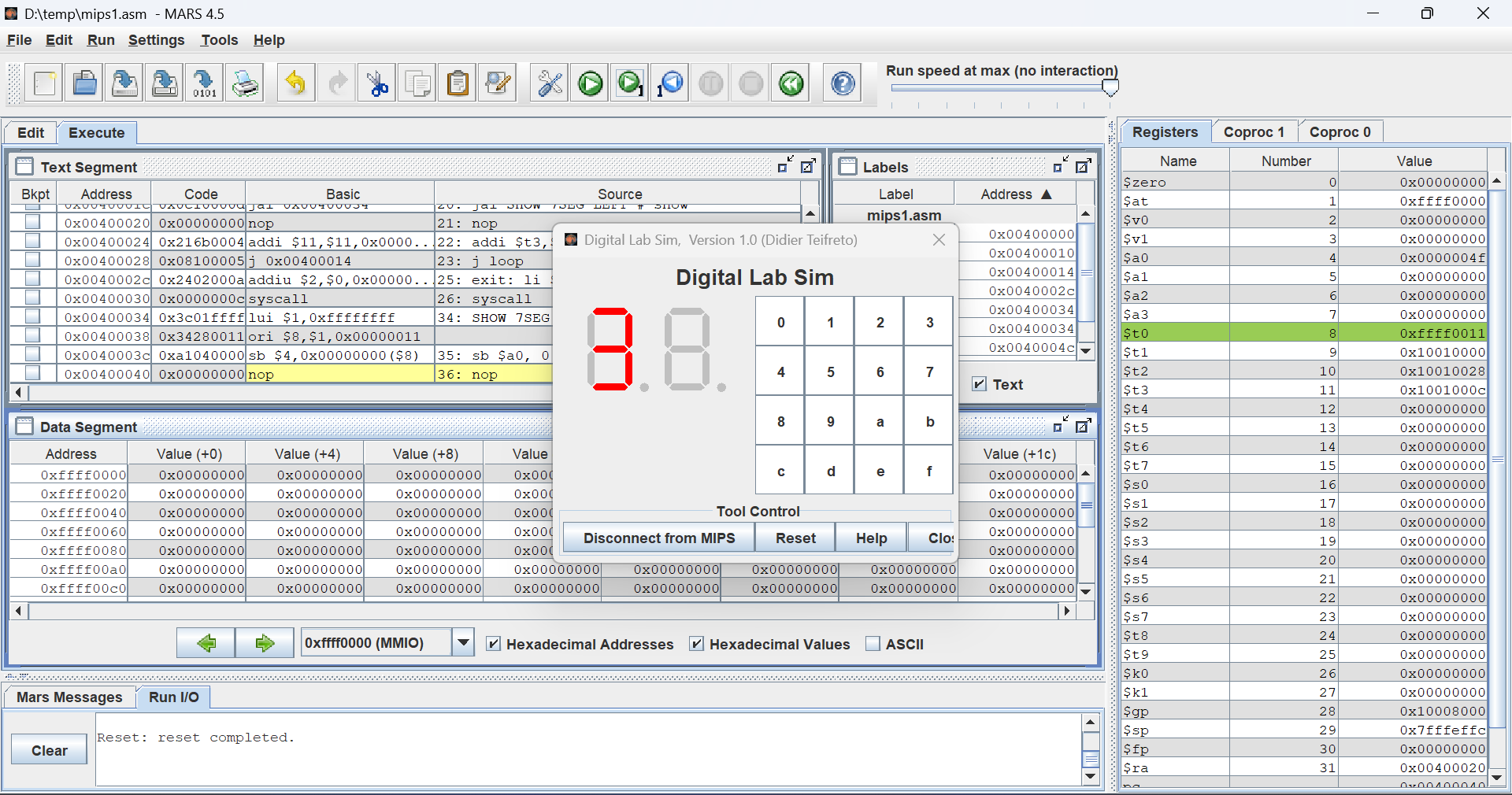
-Giải thích: Tạo 1 mảng để lưu các mẫu bit của số 0 – 9, tạo vòng lặp để hiển thị từ 0 – 9 và khi bằng 9 thì reset về 0.

* Kết quả









**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

.eqv BLACK 0x00000000

.text

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

addi $k0,$k0,60 #Dia chi cua mép bên phải màn hình

li $t0, RED #Mau can to

li $t1, BLACK #Mau cua nen

sw $t0, 0($k0)

nop

loop:

beq $k0, MONITOR\_SCREEN, end #Đến địa chỉ bắt đầu của màn hình thì kết thúc

addi $k0, $k0,-4 #Giảm về địa chỉ ô bên phải

sw $t0, 0($k0) #Tô màu đỏ ô hiện tại

sw $t1, 4($k0) #Tô màu nền của ô trước đó

j loop

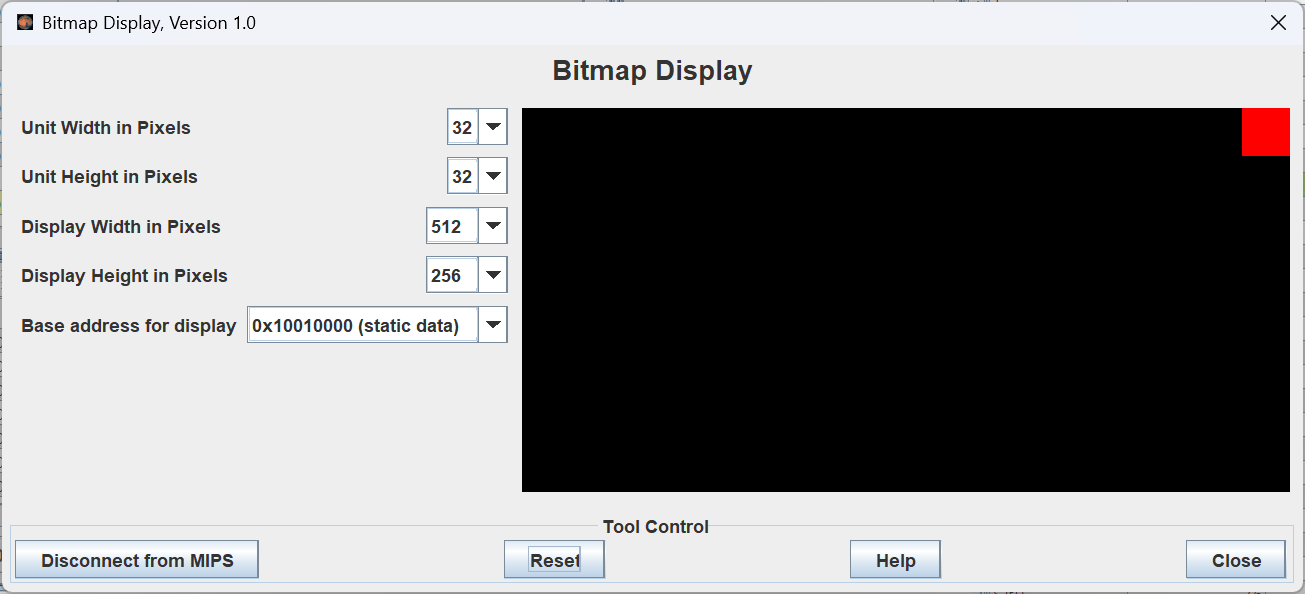
end:

li $v0,10

syscall

* Kết quả

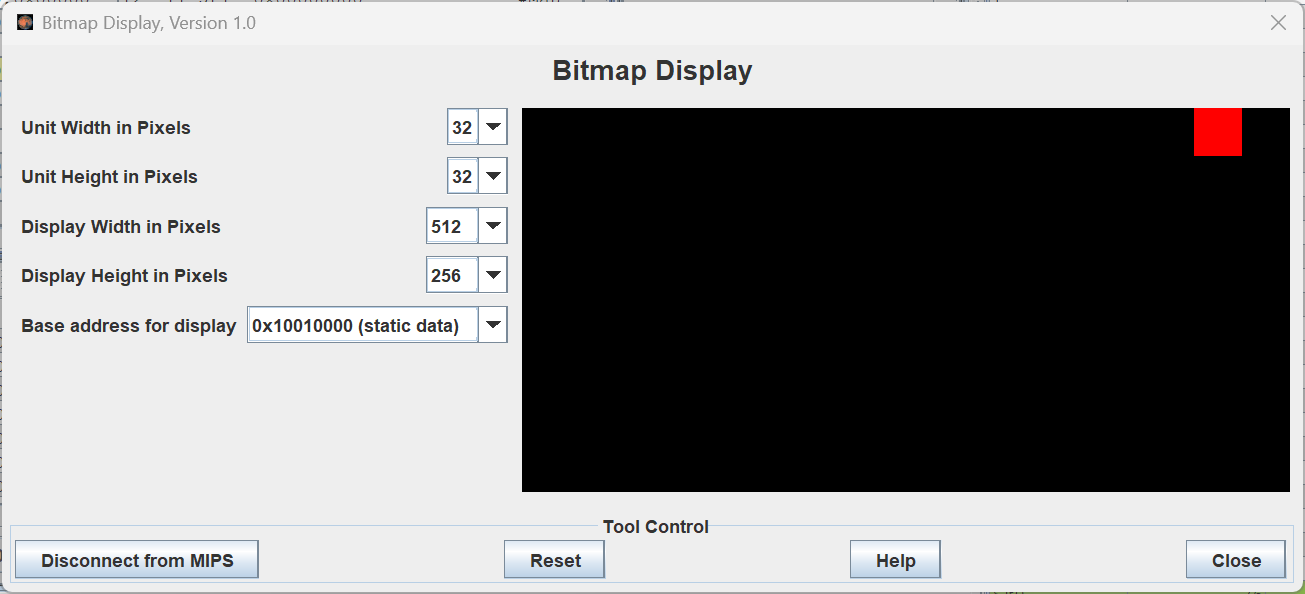
-Ban đầu ô đỏ ở bên trái màn hình



-Tô màu ô kế tiếp bên trái



-Xóa ô trước đó bằng cách tô màu nền



-Thực hiện tương tự đến khi kết thúc chương trình

-Kết thúc khi ô đỏ về địa chỉ bắt đầu của màn hình

