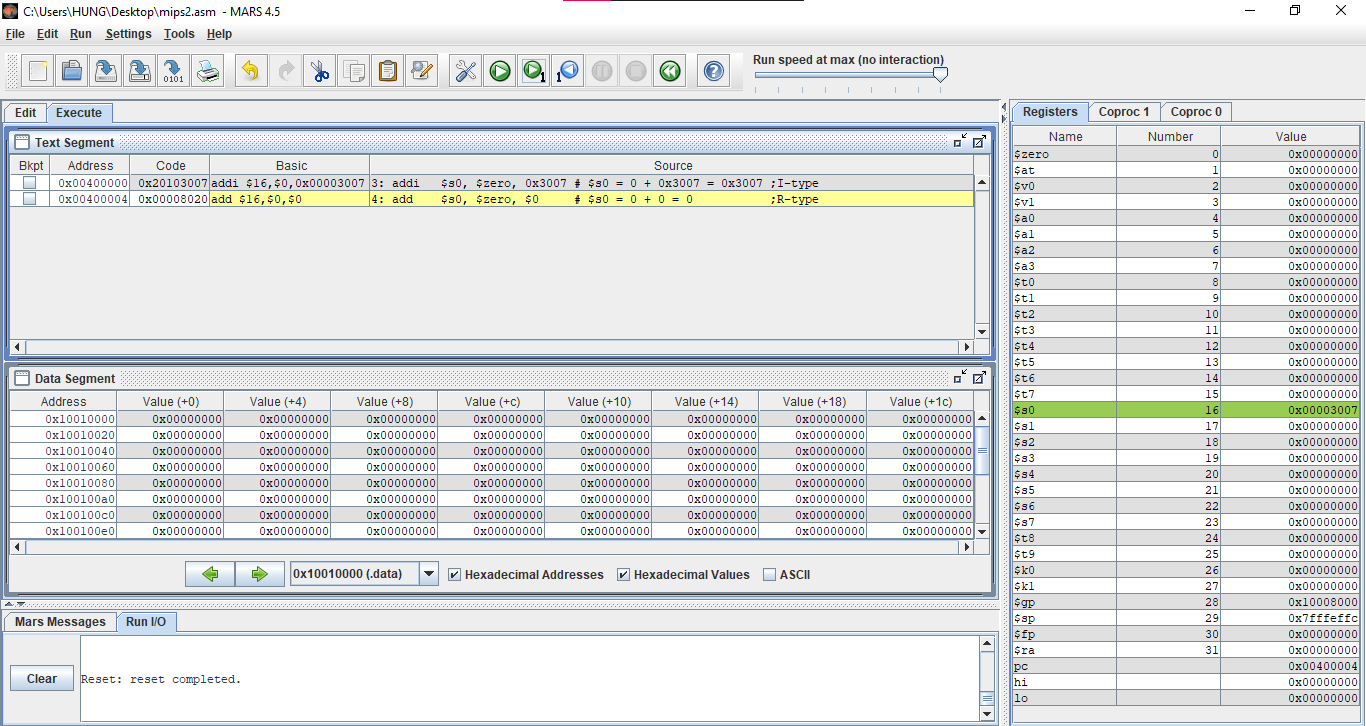
Assignment 1: lệnh gán số 16-bit

#Laboratory Exercise 2, Assignment 1

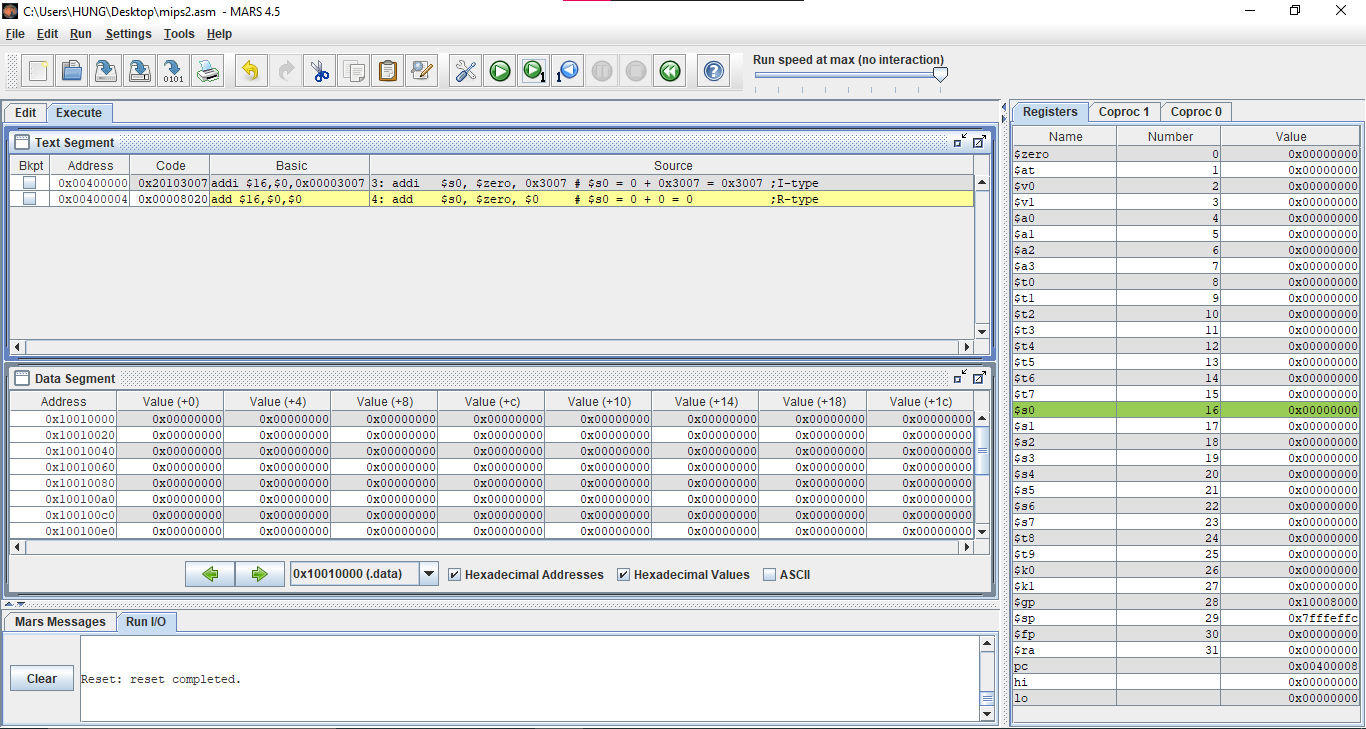
.text

addi $s0, $zero, 0x3007 # $s0 = 0 + 0x3007 = 0x3007 ;I-type

add $s0, $zero, $0 # $s0 = 0 + 0 = 0 ;R-type



* Sau lệnh: addi $s0, $zero, 0x3007 giá trị của thanh ghi $s0 trở thành:  
  0 + 0x3007 = 0x3007



* Sau lệnh: add $s0, $zero, $0 giá trị của thanh ghi $s0 trở thành:  
  0 + 0 = 0

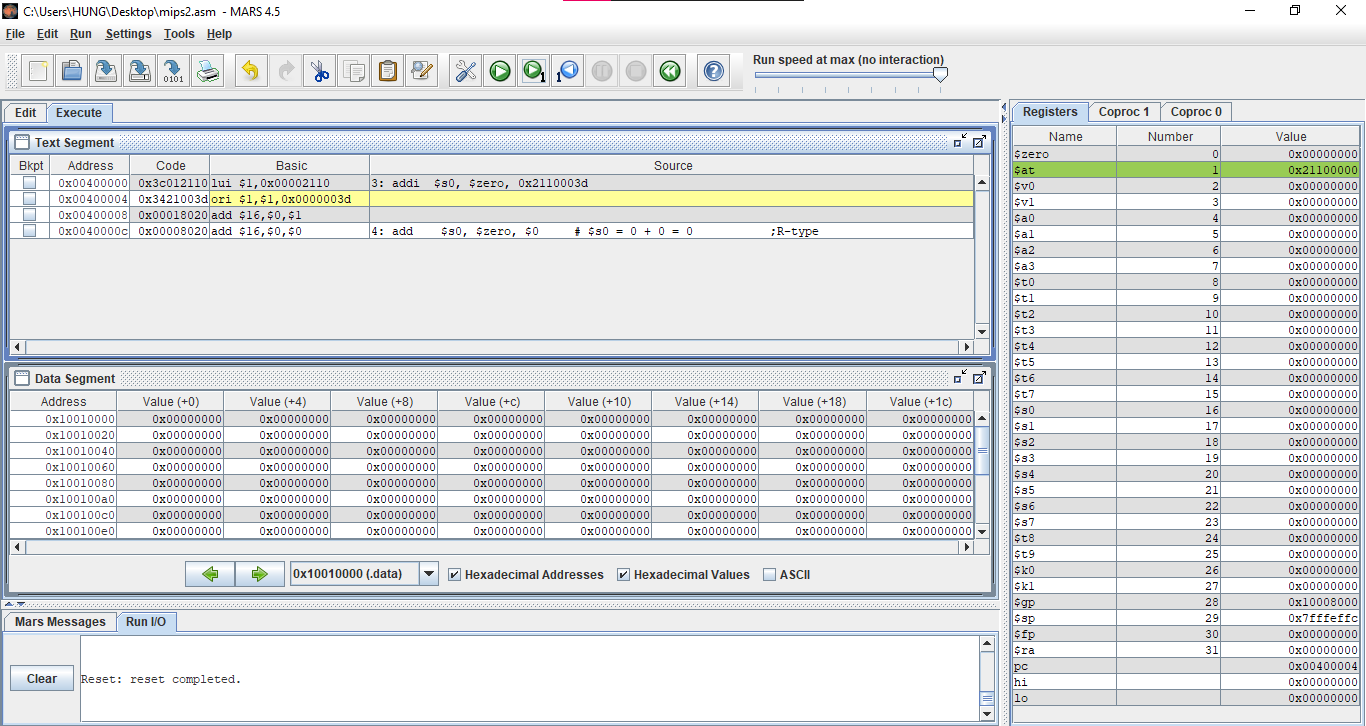
Sửa lại lệnh lui thành: addi $s0, $zero, 0x2110003d

#Laboratory Exercise 2, Assignment 1

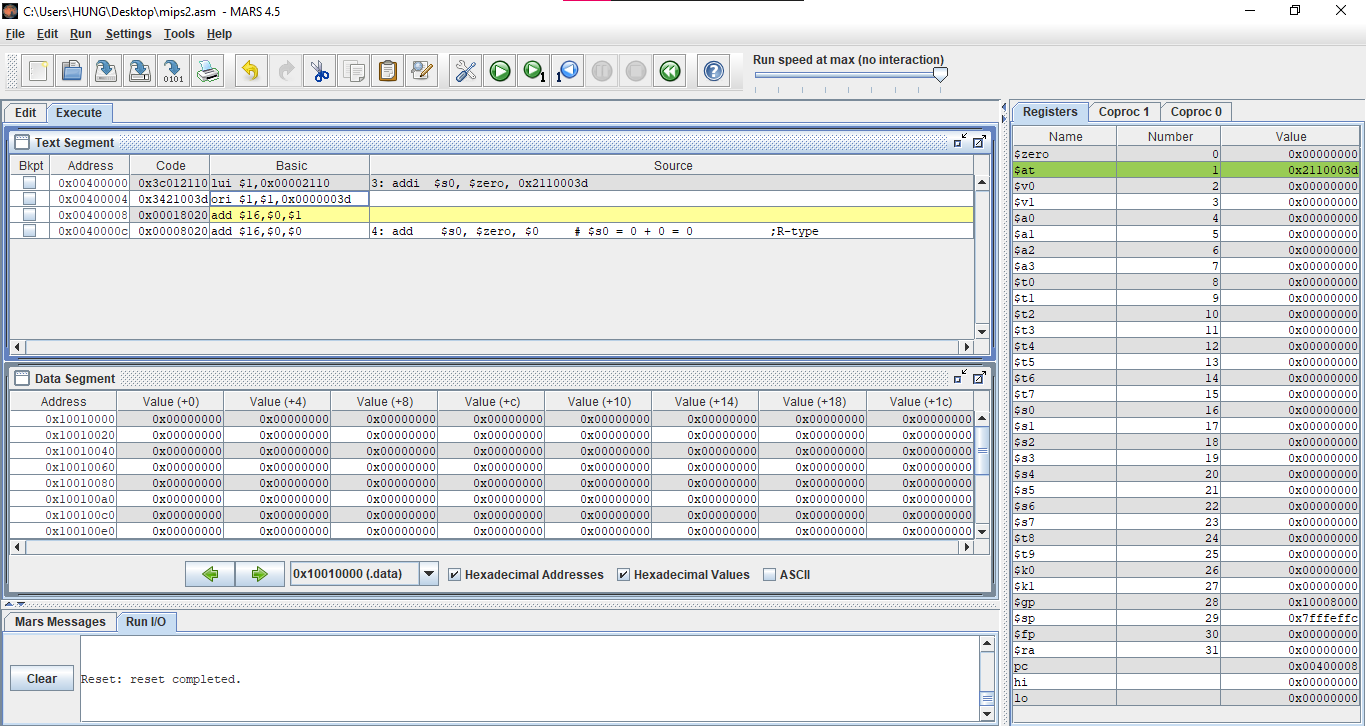
.text

addi $s0, $zero, 0x2110003d

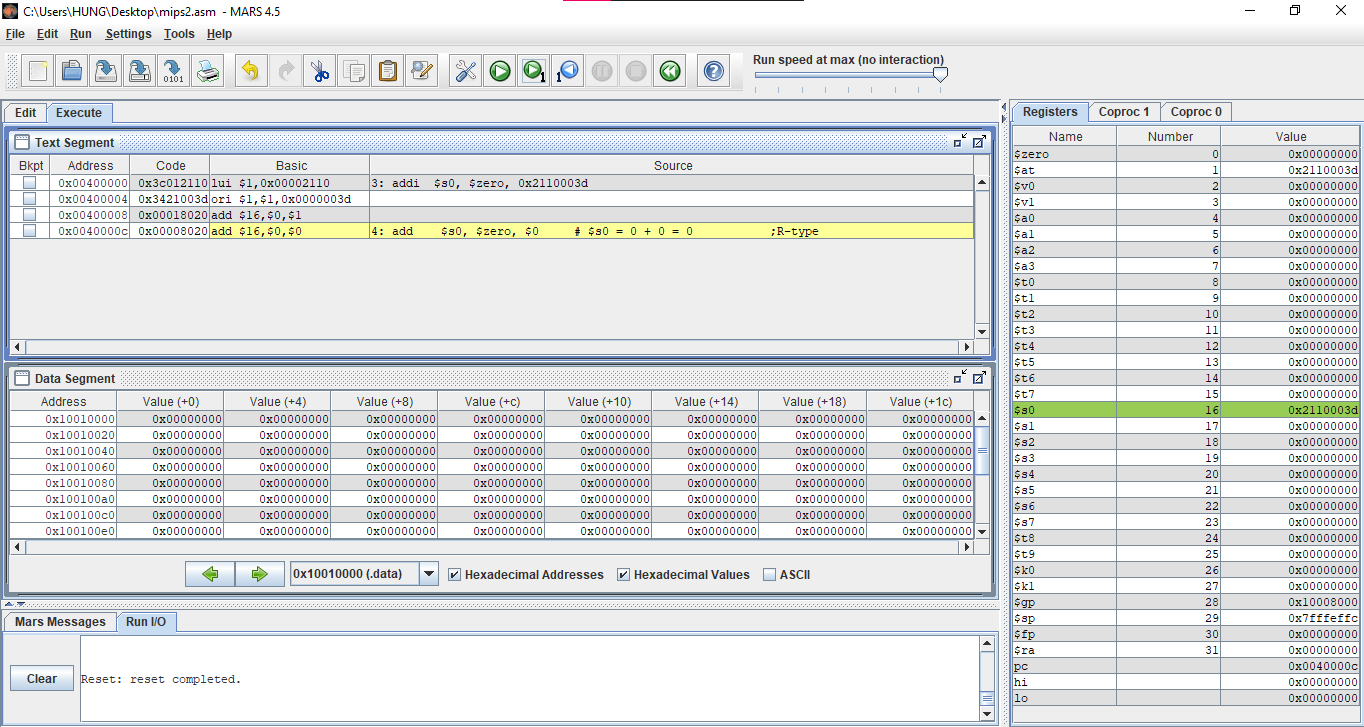
add $s0, $zero, $0 # $s0 = 0 + 0 = 0 ;R-type



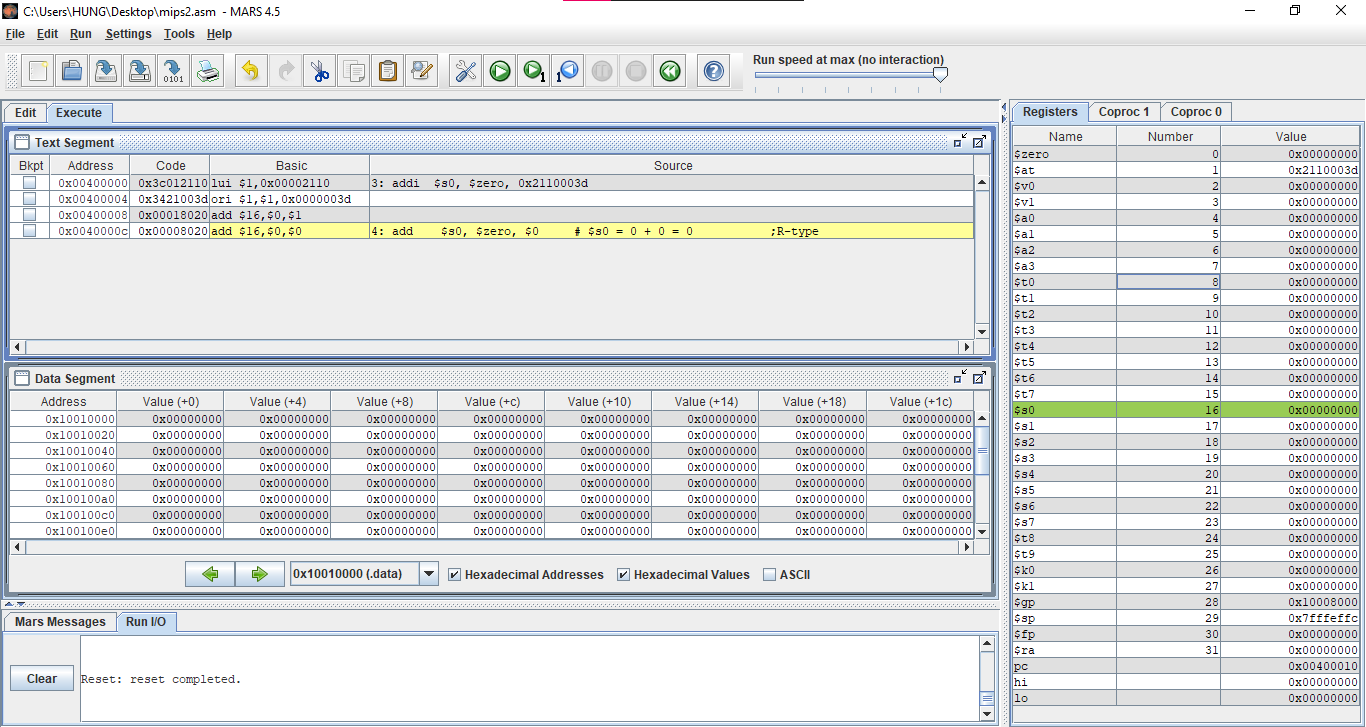
* Set 16 bits high của $1 thành 0x2110



* Lệnh ori biến 16 bits low của $1 thành 0x0000003d



* Add $16,$0,$1: biến giá trị của $16 thành $0+$1 = 0+0x2110003d=0x2110003d



* Add $16,$0,$0: biến giá trị của $16 thành $0+$0 = 0+0=0x00000000

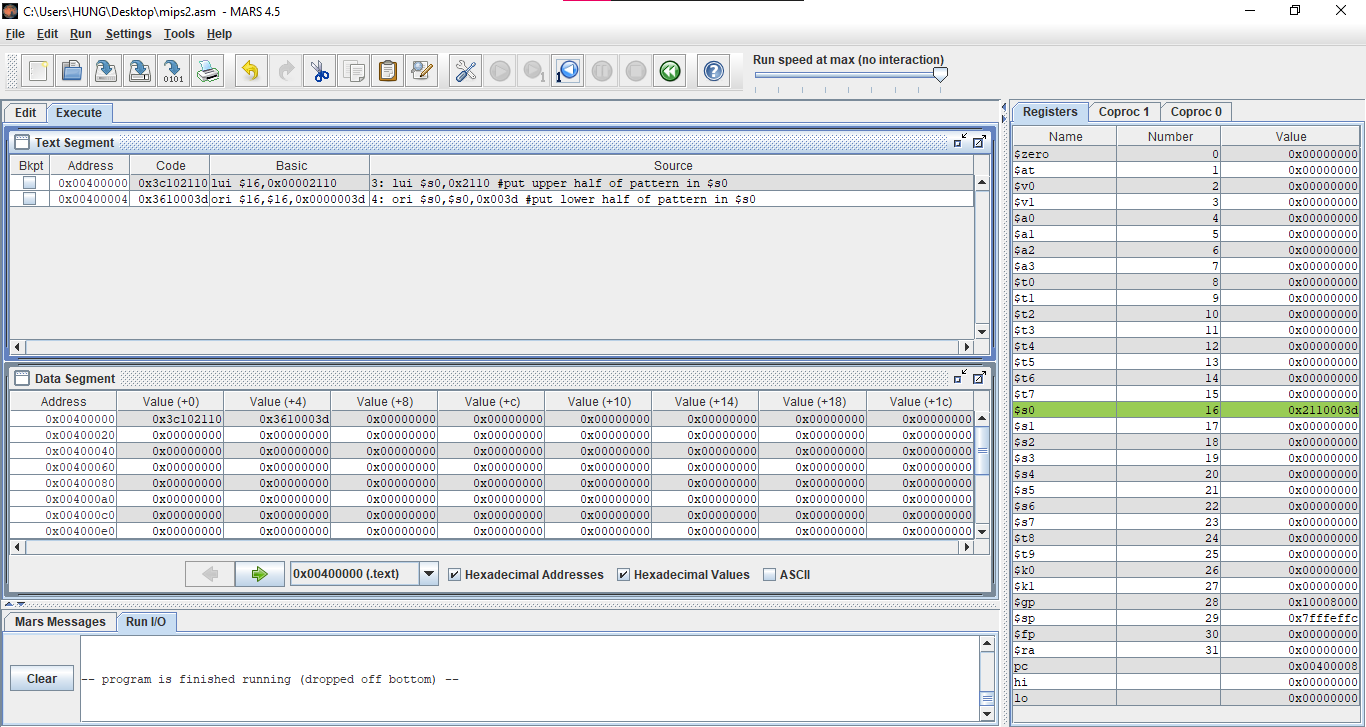
Assignment 2: lệnh gán số 32-bit

#Laboratory Exercise 2, Assignment 2

.text

lui $s0,0x2110 #put upper half of pattern in $s0

ori $s0,$s0,0x003d #put lower half of pattern in $s0



* lui $s0,0x2110: set 16 bits high của $s0 thành 0x2110
* ori $s0,$s0,0x003d: set 16 bits low của $s0 thành 0x003d
* Giá trị của $s0 trở thành 0x2110003d
* Giá trị của thanh ghi $pc thay đổi theo câu lệnh được thực hiện
* Các byte đầu tiên trong vùng lệnh Data segment trùng với các byte trên cột code của vùng lệnh Text Segment

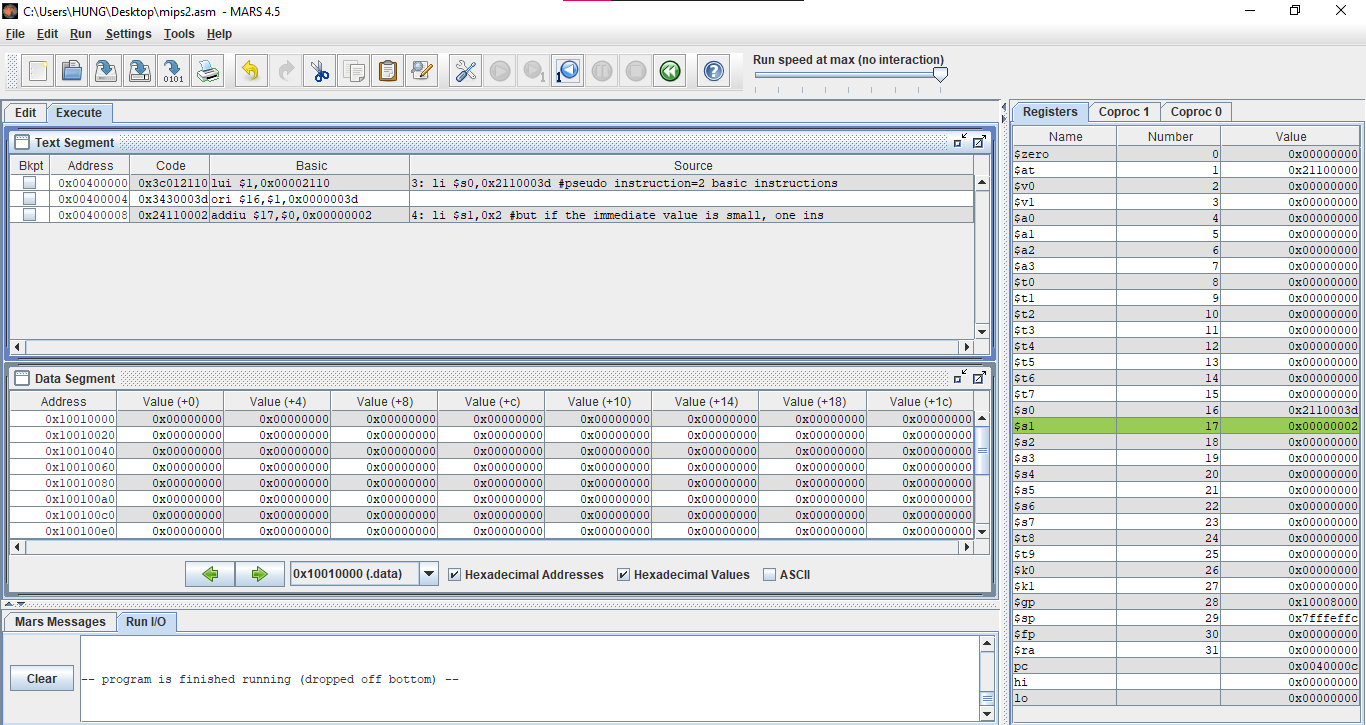
Assignment 3: lệnh gán (giả lệnh)

#Laboratory Exercise 2, Assignment 3

.text

li $s0,0x2110003d #pseudo instruction=2 basic instructions

li $s1,0x2 #but if the immediate value is small, one ins



* li $s0,0x2110003d: lệnh được chuyển thành 2 lệnh lui và ori
* li $s1,0x2: lệnh được chuyển thành addiu $17,$0, 0x00000002 do giá trị của 0x00000002 nhỏ

Assignment 4: tính biểu thức 2x + y = ?

#Laboratory Exercise 2, Assignment 4

.text

# Assign X, Y

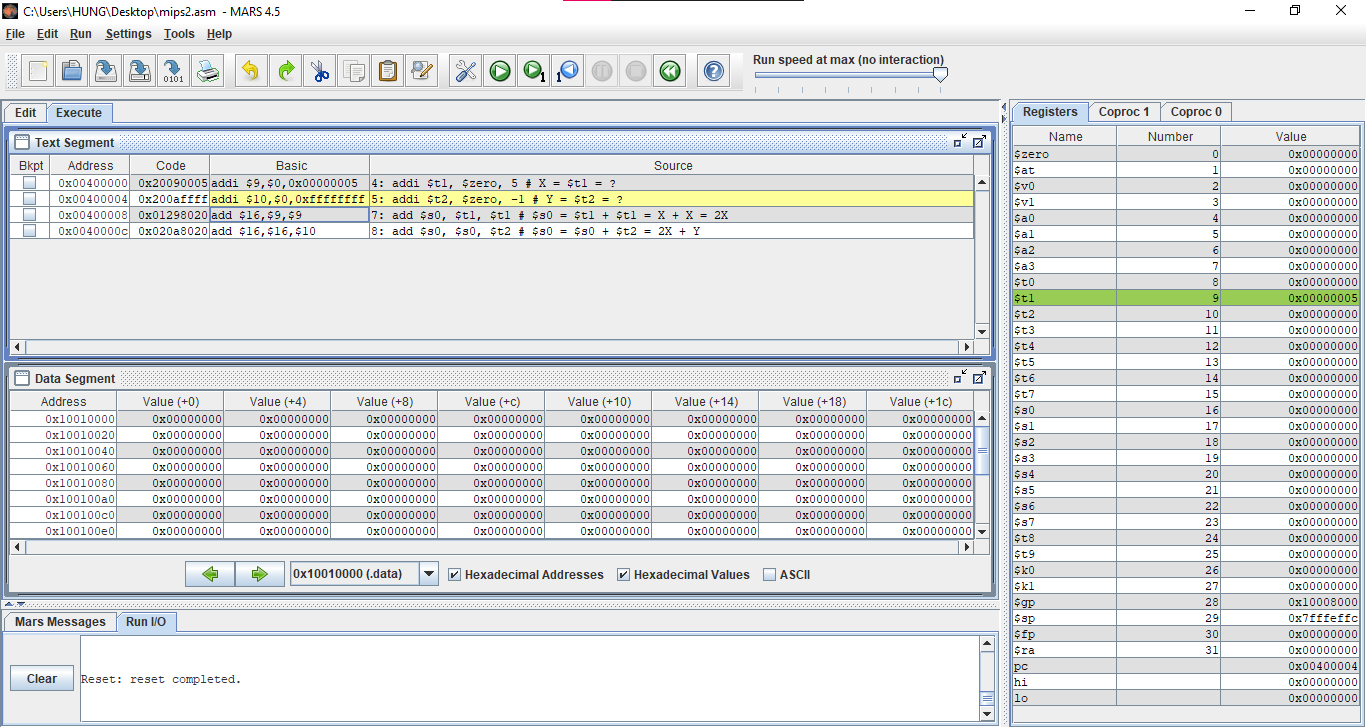
addi $t1, $zero, 5 # X = $t1 = ?

addi $t2, $zero, -1 # Y = $t2 = ?

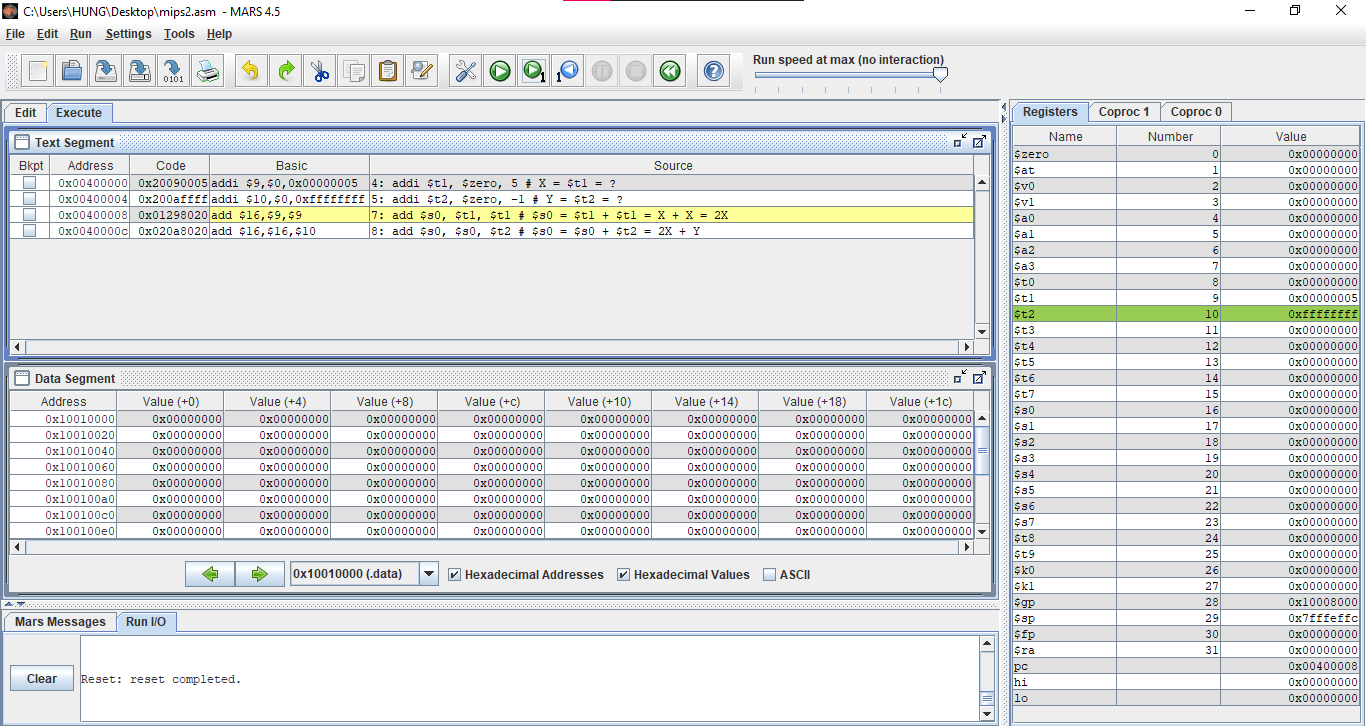
# Expression Z = 2X + Y

add $s0, $t1, $t1 # $s0 = $t1 + $t1 = X + X = 2X

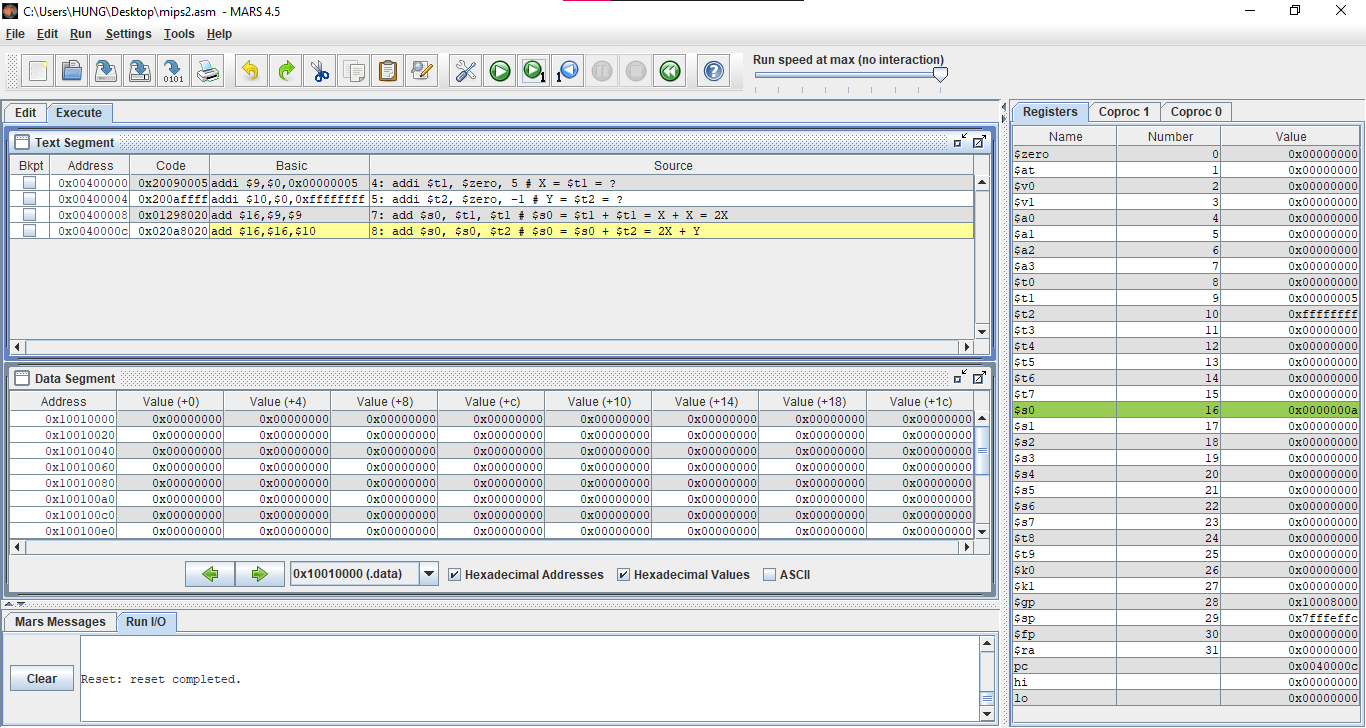
add $s0, $s0, $t2 # $s0 = $s0 + $t2 = 2X + Y



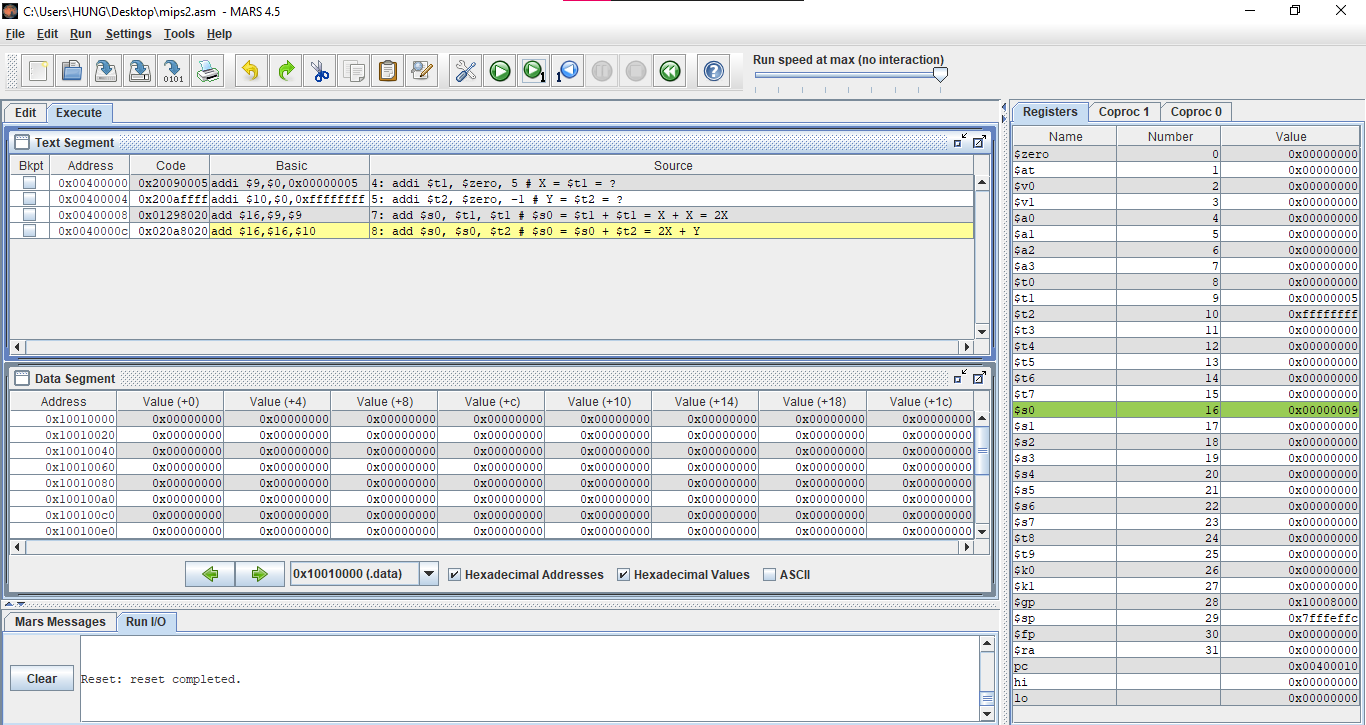
* addi $t1, $zero, 5: nạp giá trị X=5 vào $t1



* addi $t2, $zero, -1: nạp giá trị Y=-1 vào $t2



* add $s0, $t1, $t1: thực hiện phép tính 2X=X+X=$t1+$t1 rồi lưu kết quả vào $s0



* add $s0, $s0, $t2: thực hiện phép tính $s0+$t2=2X+Y=Z=9 rồi lưu giá trị Z vào $s0

Assignment 5: phép nhân

#Laboratory Exercise 2, Assignment 5

.text

# Assign X, Y

addi $t1, $zero, 4 # X = $t1 = ?

addi $t2, $zero, 5 # Y = $t2 = ?

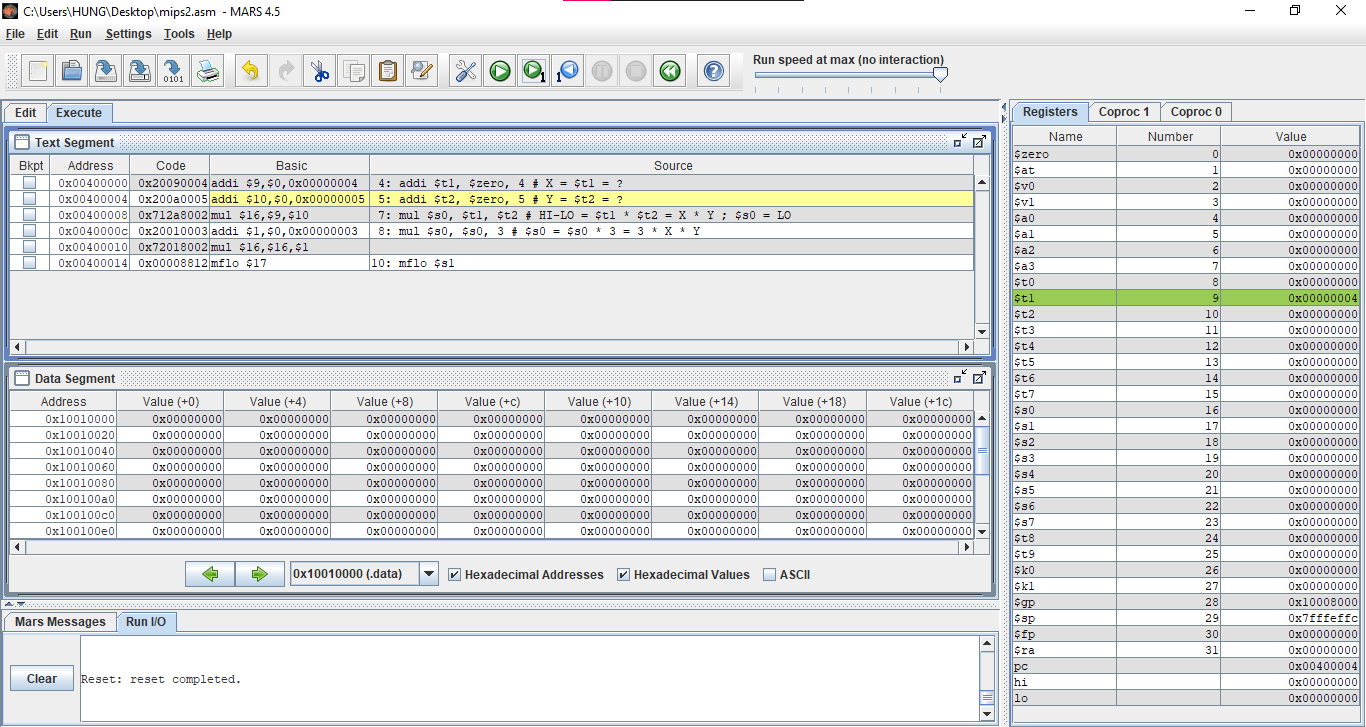
# Expression Z = 3\*XY

mul $s0, $t1, $t2 # HI-LO = $t1 \* $t2 = X \* Y ; $s0 = LO

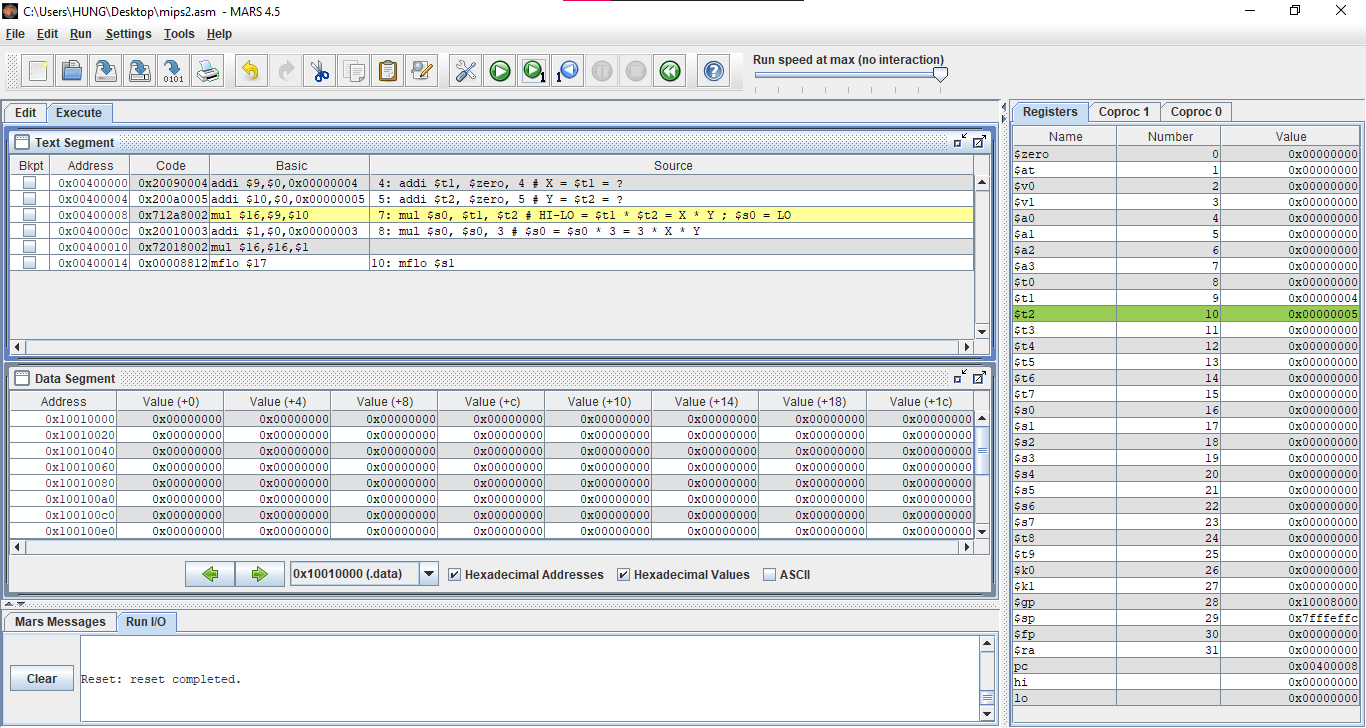
mul $s0, $s0, 3 # $s0 = $s0 \* 3 = 3 \* X \* Y

# Z' = Z

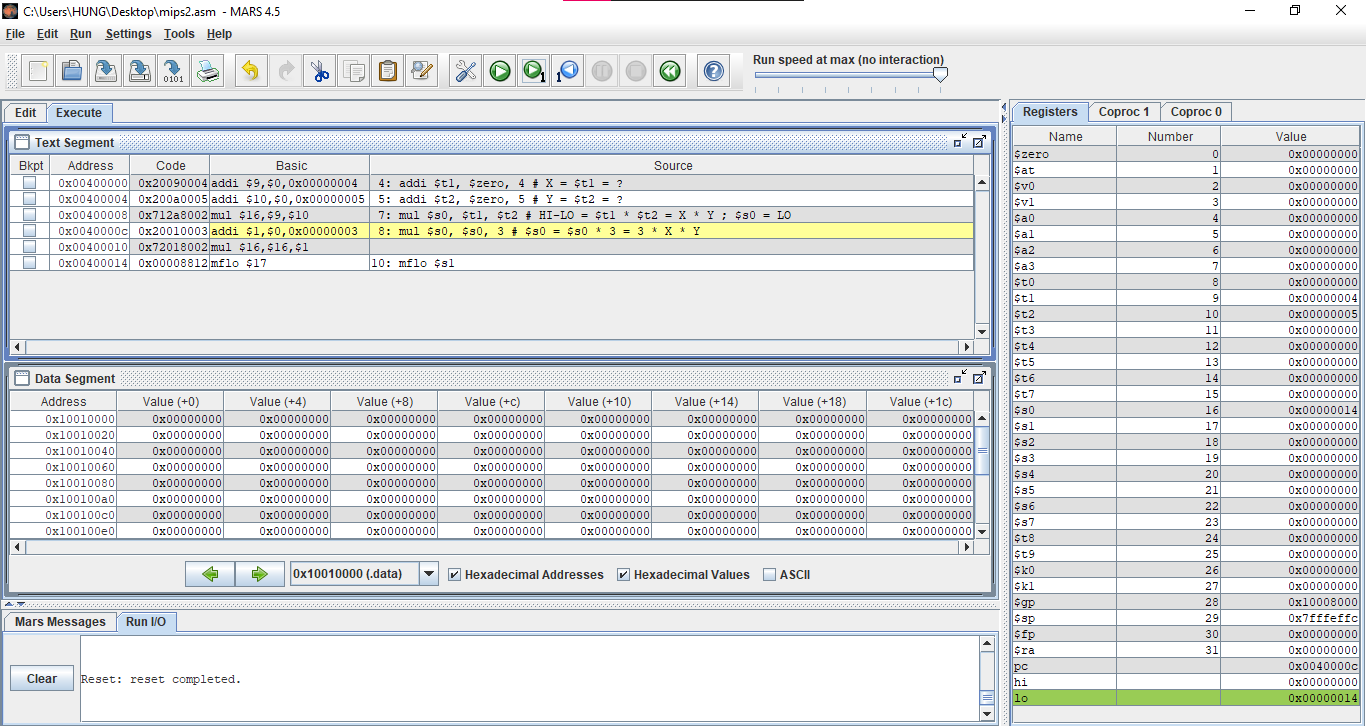
mflo $s1



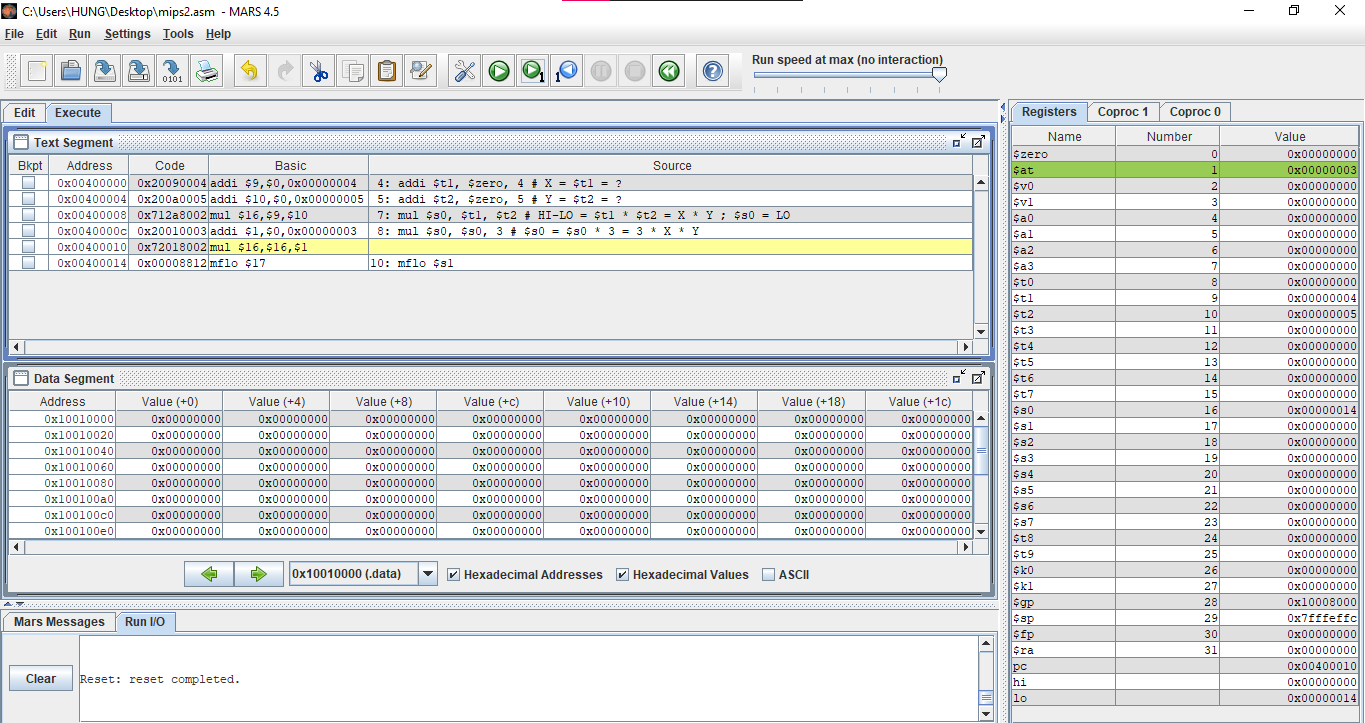
* addi $t1, $zero, 4: thực hiện cộng X+0=4+0 rồi lưu vào $t1



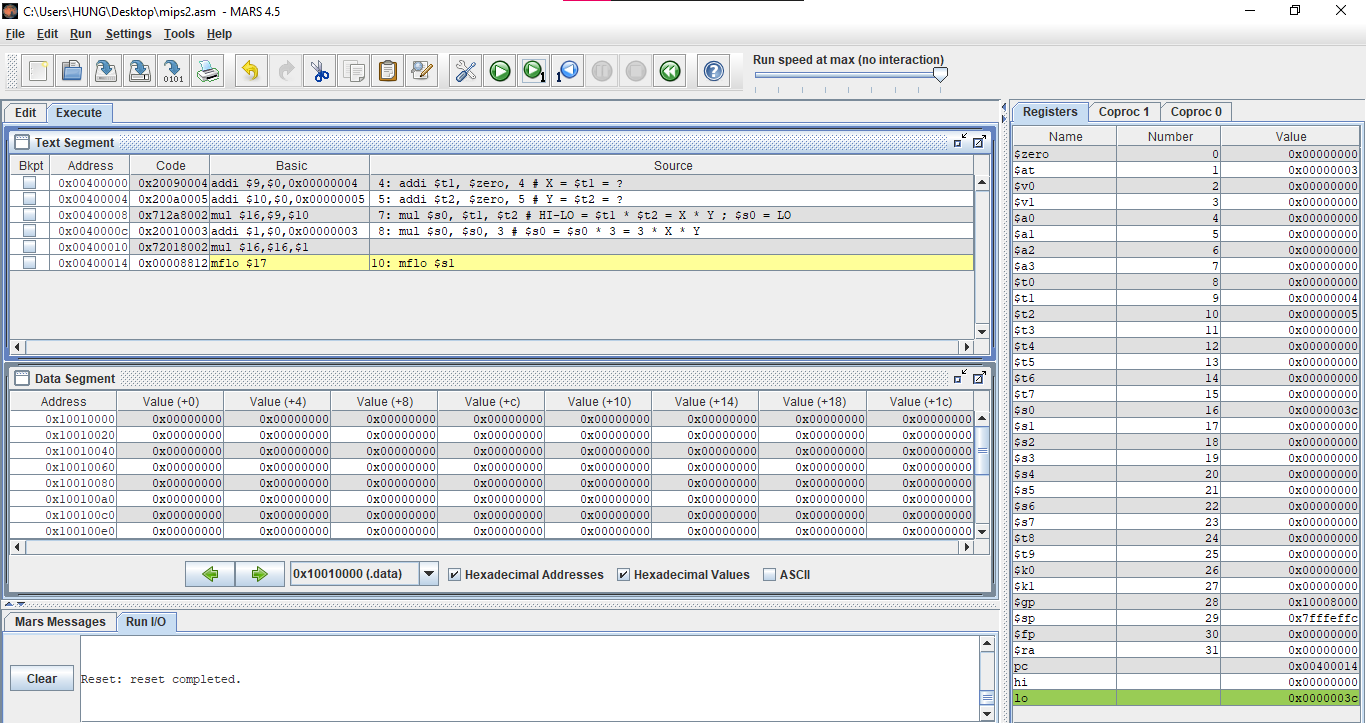
* addi $t2, $zero, 5: thực hiện cộng Y+0=5+0 rồi lưu vào $t2



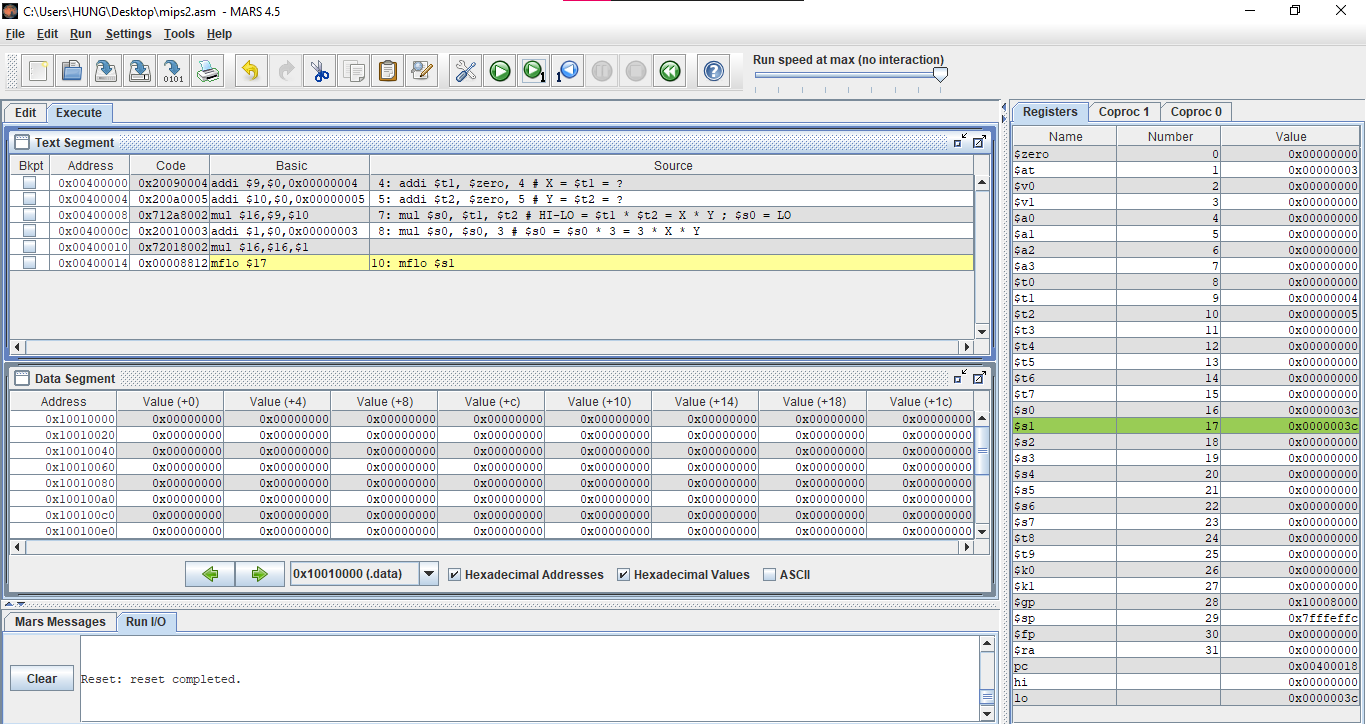
* mul $s0, $t1, $t2: thực hiện phép nhân X\*Y=$t1\*$t2 rồi lưu kết quả vào $s0 và LO



* mul $s0, $s0, 3:
  + addi $1,$0,0x00000003: lưu giá trị 3 vào $0



* + mul $16, $16, $1: thực hiện phép nhân 3\*X\*Y=$1\*$16 rồi lưu kết quả vào $16($0)



* mflo $s1: lưu giá trị của LO vào $s1

Sau khi kết thúc chương trình, xem kết quả đúng.

Assignment 6: tạo biến và truy cập biến

#Laboratory Exercise 2, Assignment 6

.data # DECLARE VARIABLES

X : .word 5 # Variable X, word type, init value =

Y : .word -1 # Variable Y, word type, init value =

Z : .word # Variable Z, word type, no init value

.text # DECLARE INSTRUCTIONS

# Load X, Y to registers

la $t8, X # Get the address of X in Data Segment

la $t9, Y # Get the address of Y in Data Segment

lw $t1, 0($t8) # $t1 = X

lw $t2, 0($t9) # $t2 = Y

# Calcuate the expression Z = 2X + Y with registers only

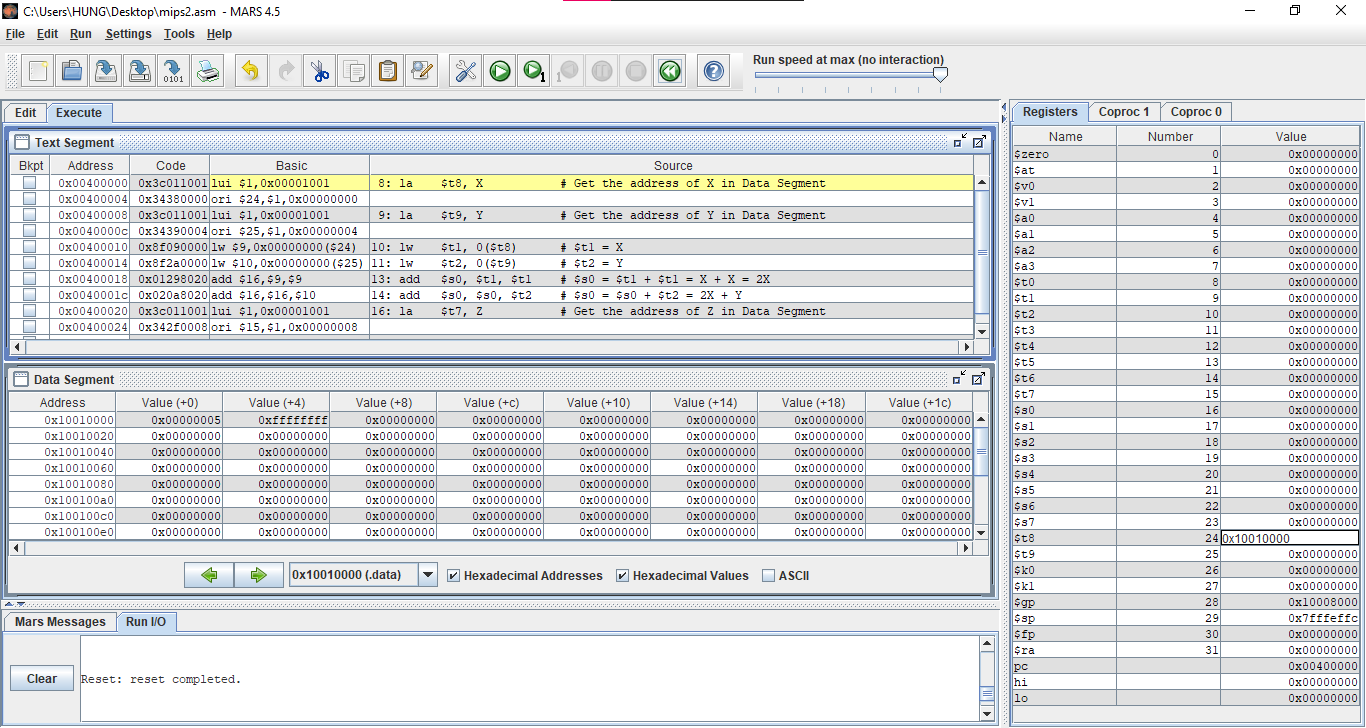
add $s0, $t1, $t1 # $s0 = $t1 + $t1 = X + X = 2X

add $s0, $s0, $t2 # $s0 = $s0 + $t2 = 2X + Y

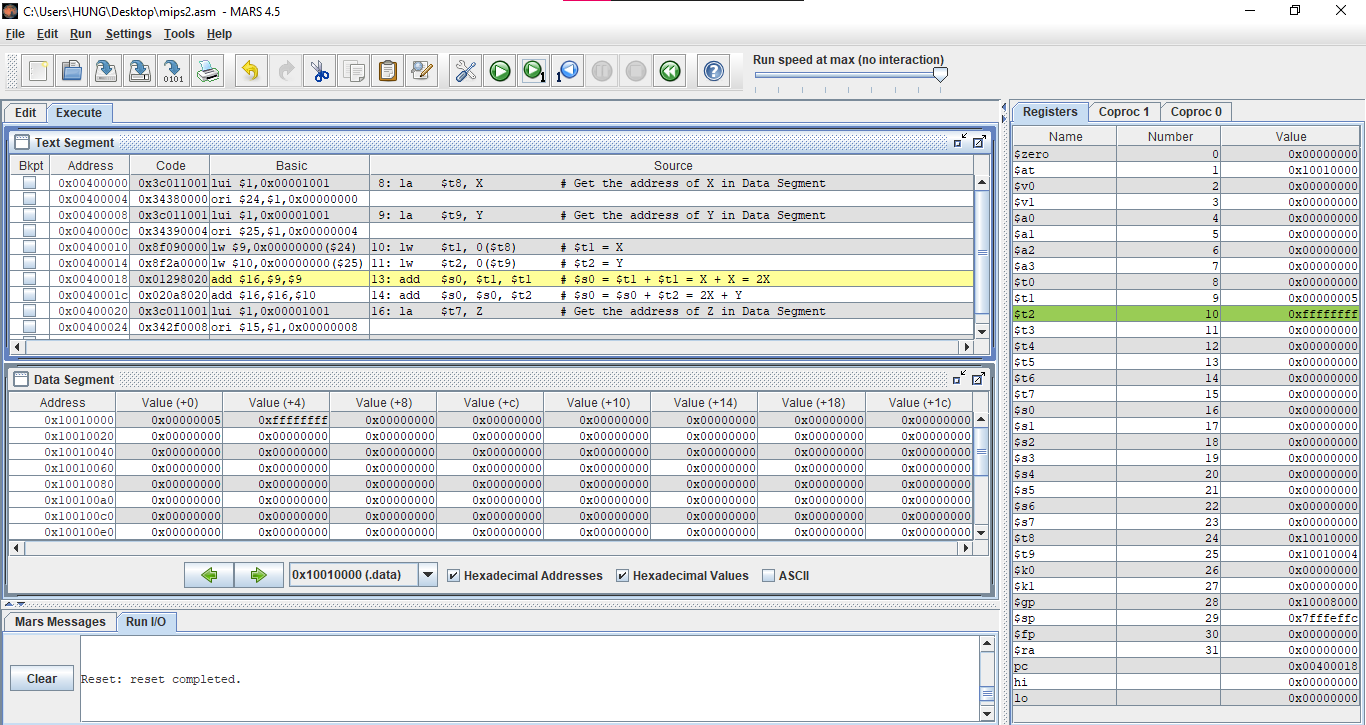
# Store result from register to variable Z

la $t7, Z # Get the address of Z in Data Segment

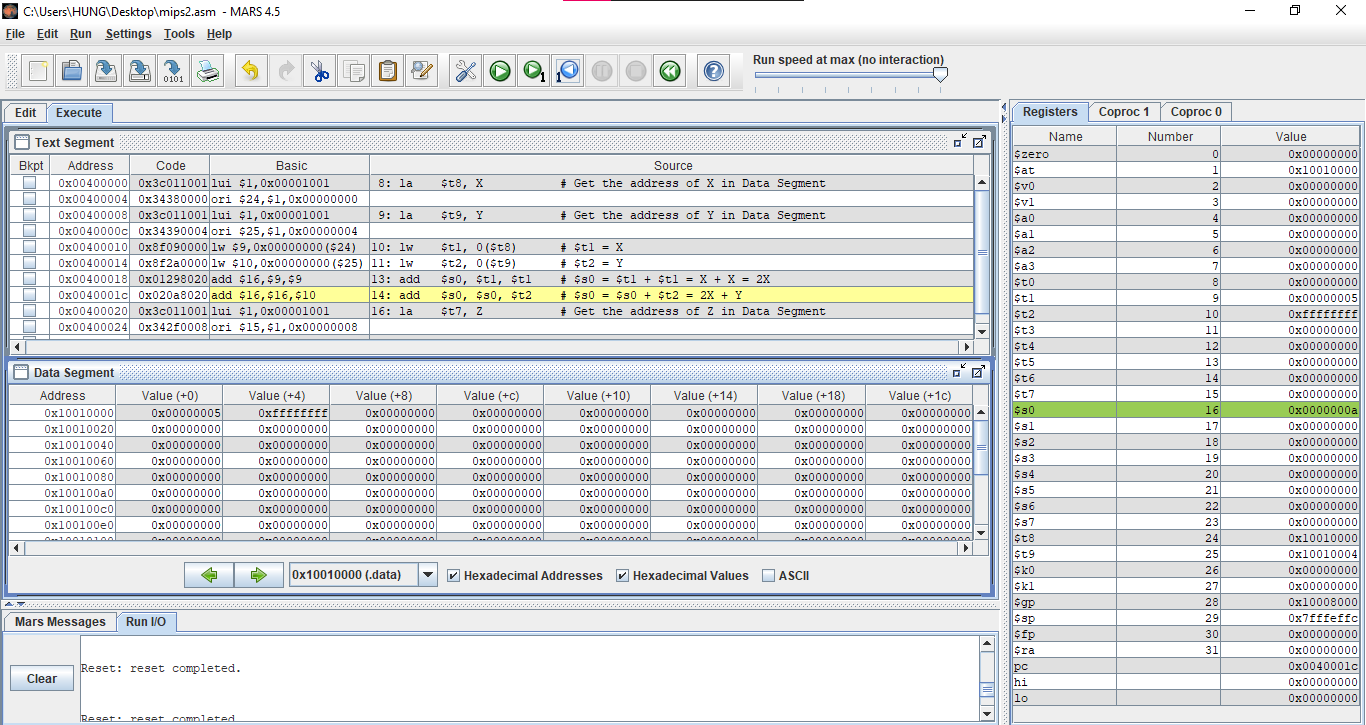
sw $s0, 0($t7) # Z = $s0 = 2X + Y



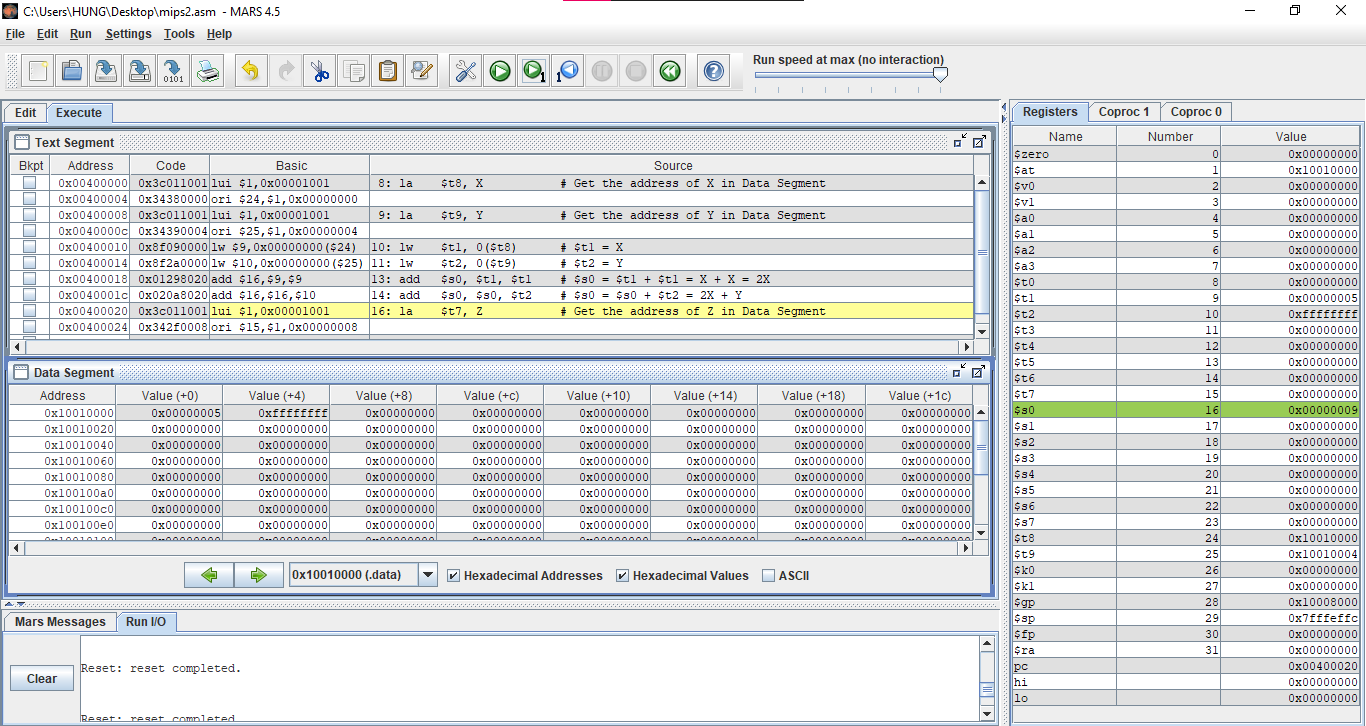
* Lệnh la được biên dịch thành lệnh lui và ori



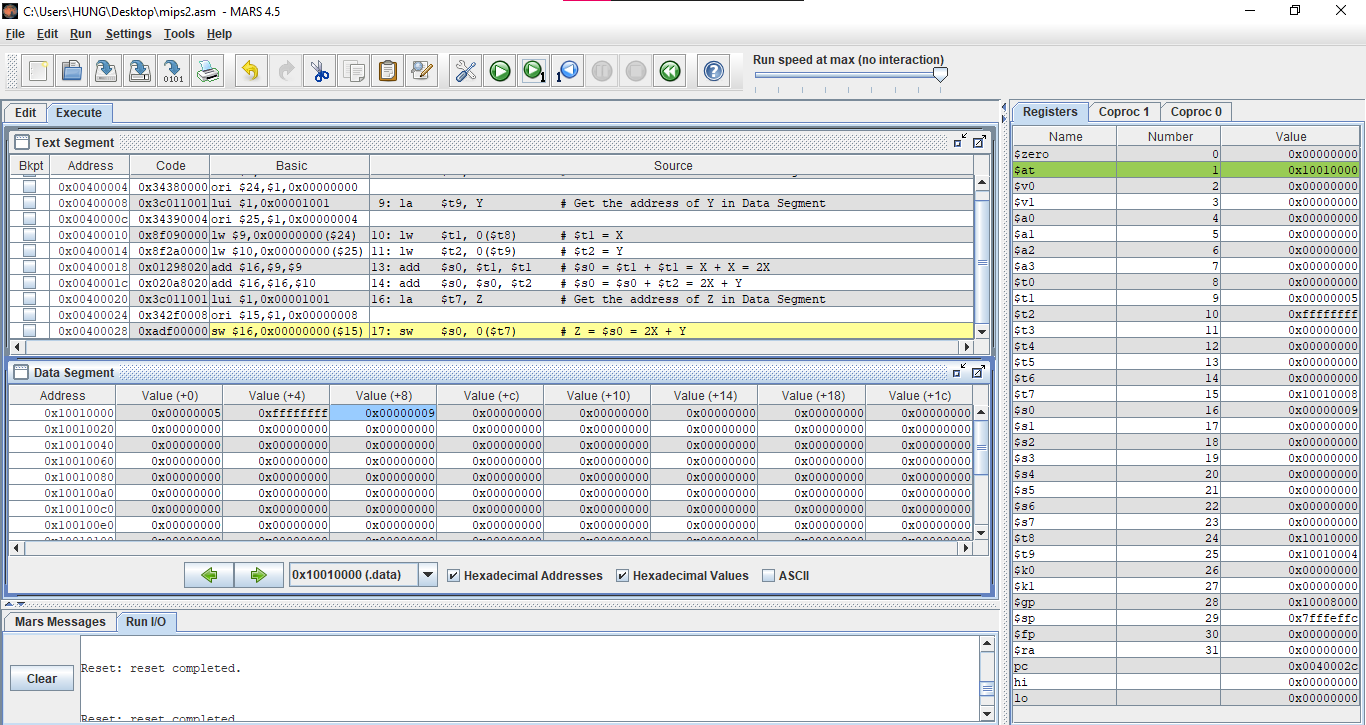
* lw $t1, 0($t8): nạp giá trị được lưu ở địa chỉ trong $t8 vào $t1
* lw $t2, 0($t9): nạp giá trị được lưu ở địa chỉ trong $t9 vào $t2



* add $s0, $t1, $t1: thực hiện phép cộng $s0=$t1+$t1=X+X



* add $s0, $s0, $t2: thực hiện phép cộng $s0=$s0+$t2=X+X+Y=2X+Y



* la $t7, Z: lưu địa chỉ của biến Z vào $t7
* sw $s0, 0($t7): lưu giá trị của $s0 vào địa chỉ của biến Z trong $t7