**Activity 1:**

;default bank 0

mov A, #0xAA ;set A = 0xAA

;move R0 - R7 = A = 0xAA in bank 0

mov R0, A

mov R1, A

mov R2, A

mov R3, A

mov R4, A

mov R5, A

mov R6, A

mov R7, A

setb RS0 ;set to bank 1

;move R0 - R7 = A = 0xAA in bank 1

mov R0, A

mov R1, A

mov R2, A

mov R3, A

mov R4, A

mov R5, A

mov R6, A

mov R7, A

setb RS1 ;set to bank 3

;move R0 - R7 = A = 0xAA in bank 3

mov R0, A

mov R1, A

mov R2, A

mov R3, A

mov R4, A

mov R5, A

mov R6, A

mov R7, A

clr RS0 ;set to bank 2

;move R0 - R7 = A = 0xAA in bank 2

mov R0, A

mov R1, A

mov R2, A

mov R3, A

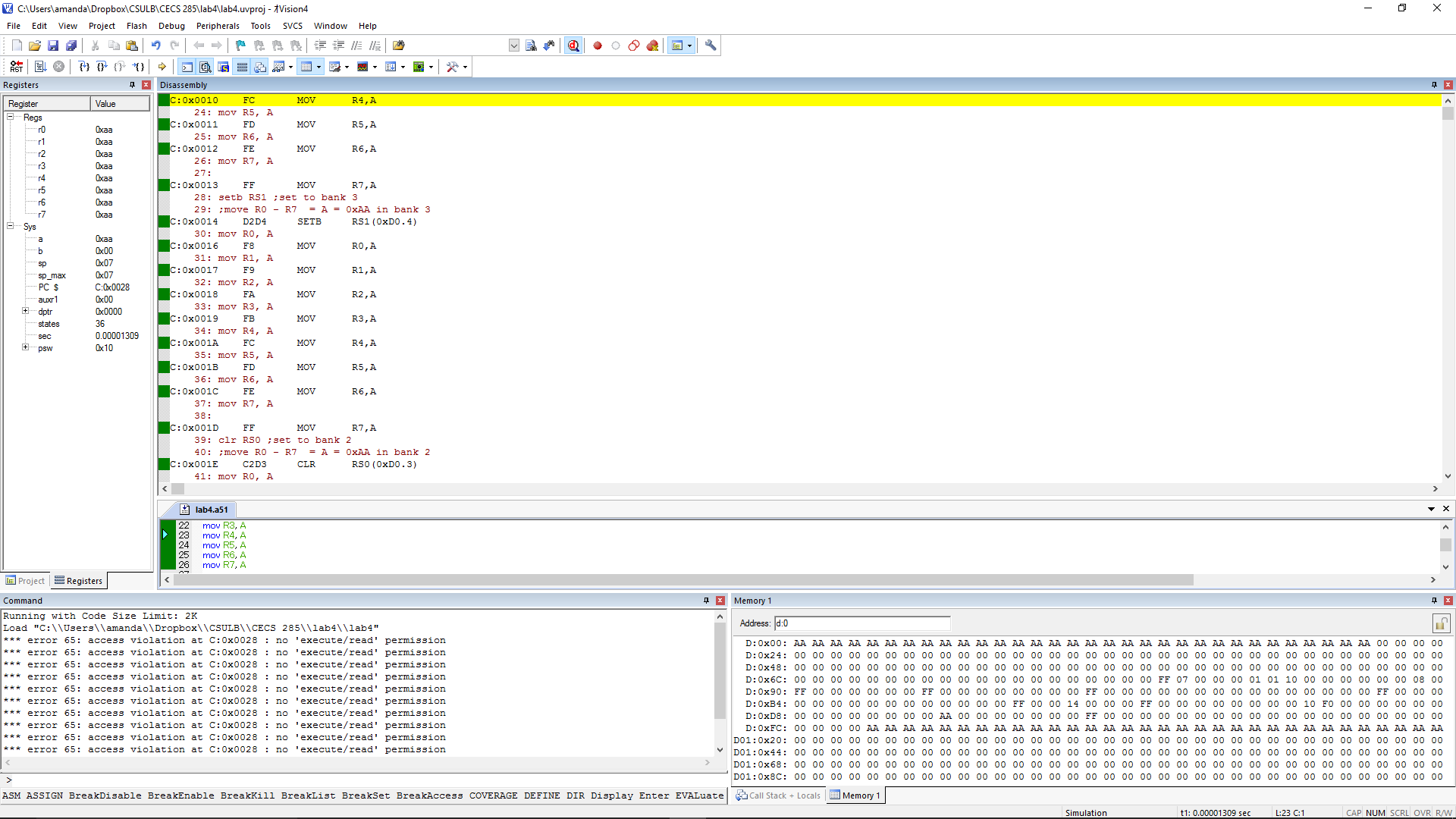
mov R4, A

mov R5, A

mov R6, A

mov R7, A

end



**Activity 2:**

num1 EQU 0x30 ;address of num1 = 0x30

num2 EQU 0x31 ;address of num2 = 0x31

num3 EQU 0x32 ;address of num3 = 0x32

mov num1, #21

mov num2, #21

;use A to compare num1 and num2

mov A, num1 ;A = num1

clr CY ;make sure A starts with a positive value

subb A, num2 ;A - num2

jz EQUAL ;if num1 == num2 jump tp EQUAL

jc LESS ;if A == negative number, jump to less

jnc MORE ;if A == positive number, jump to less

EQUAL:

mov num3, #0x22 ;num3 = 0x22

jmp CheckDone

LESS:

mov num3, #0x11 ;num3 = 0x11

jmp CheckDone

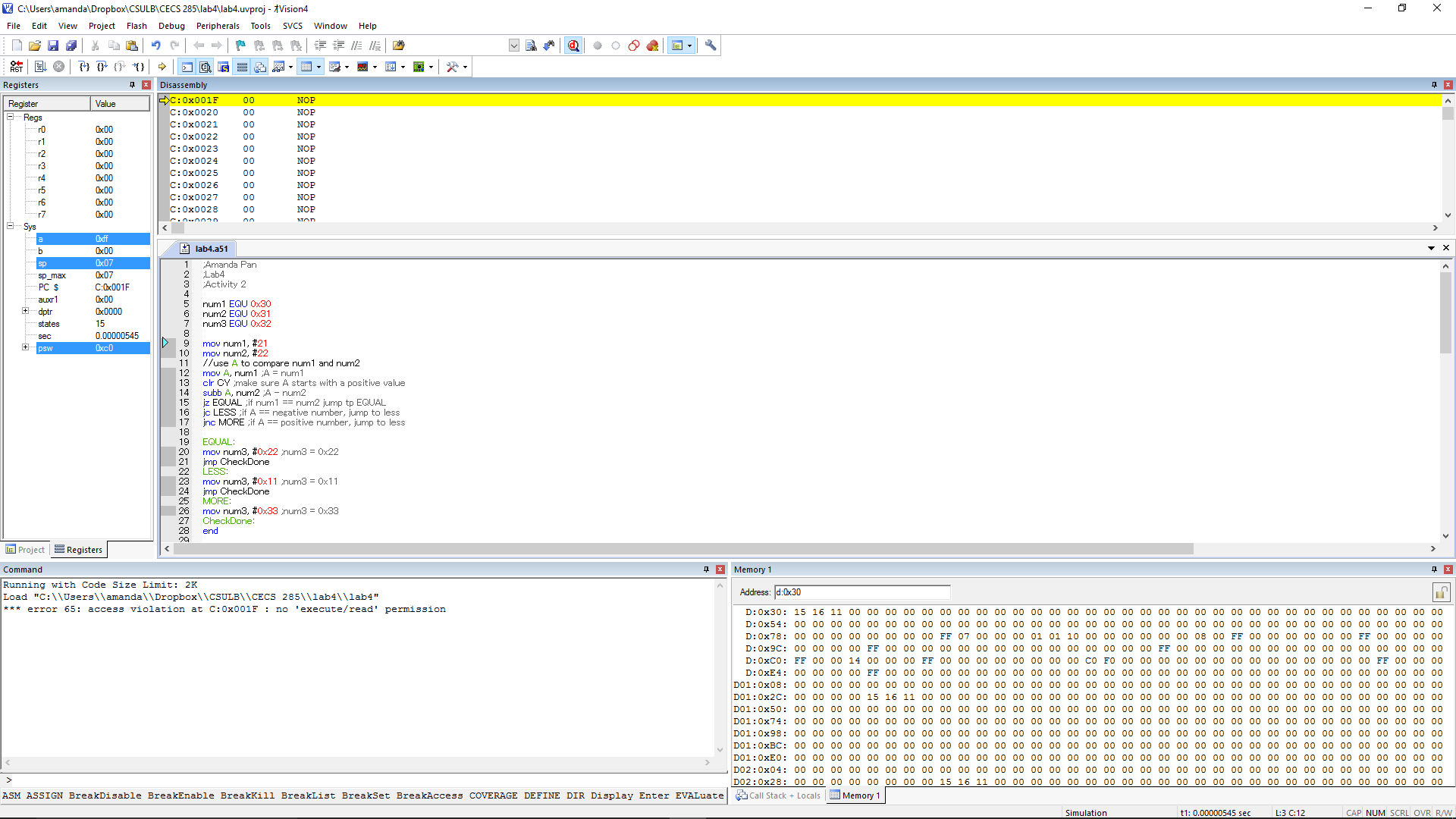
MORE:

mov num3, #0x33 ;num3 = 0x33

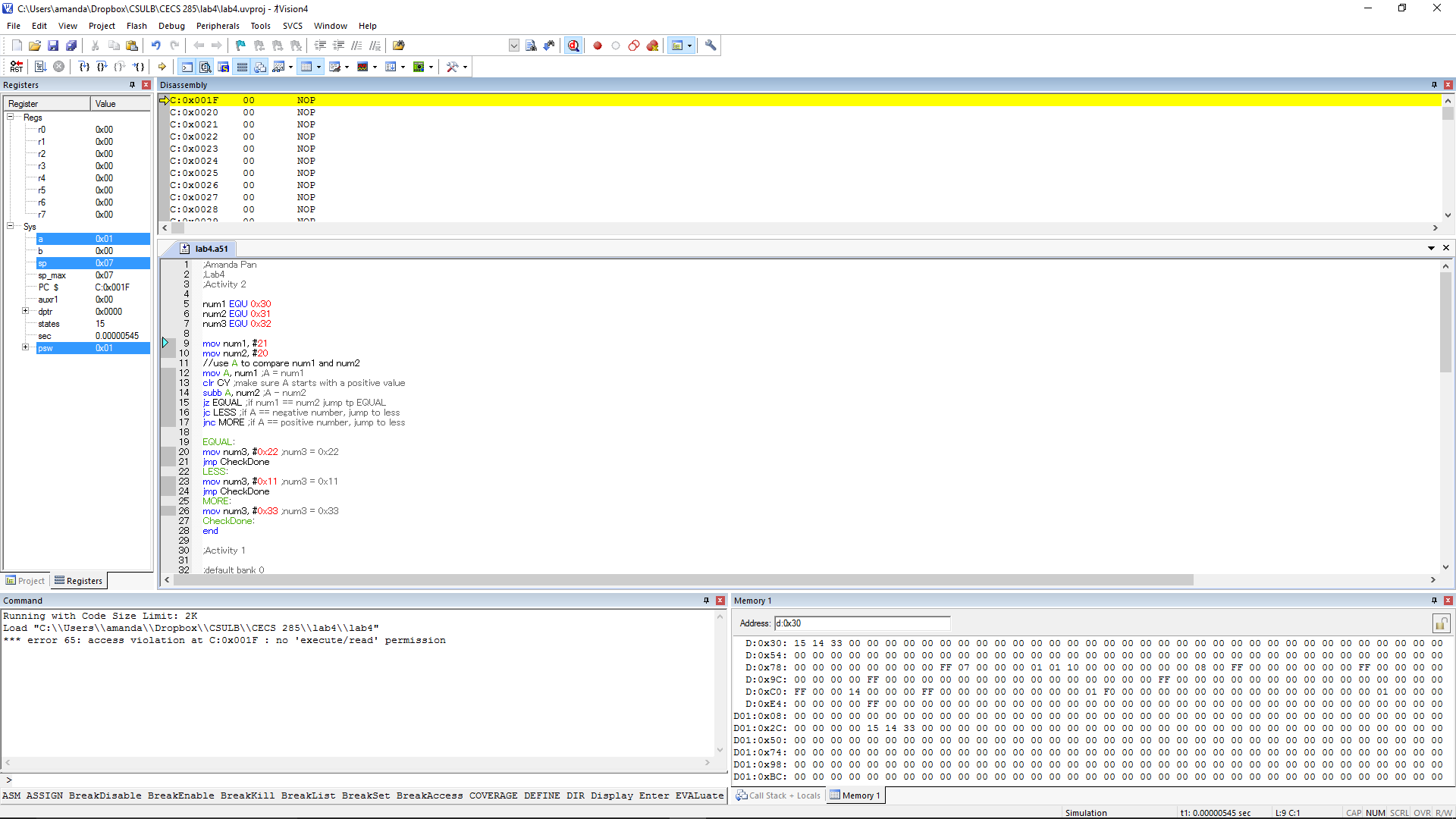
CheckDone:

end

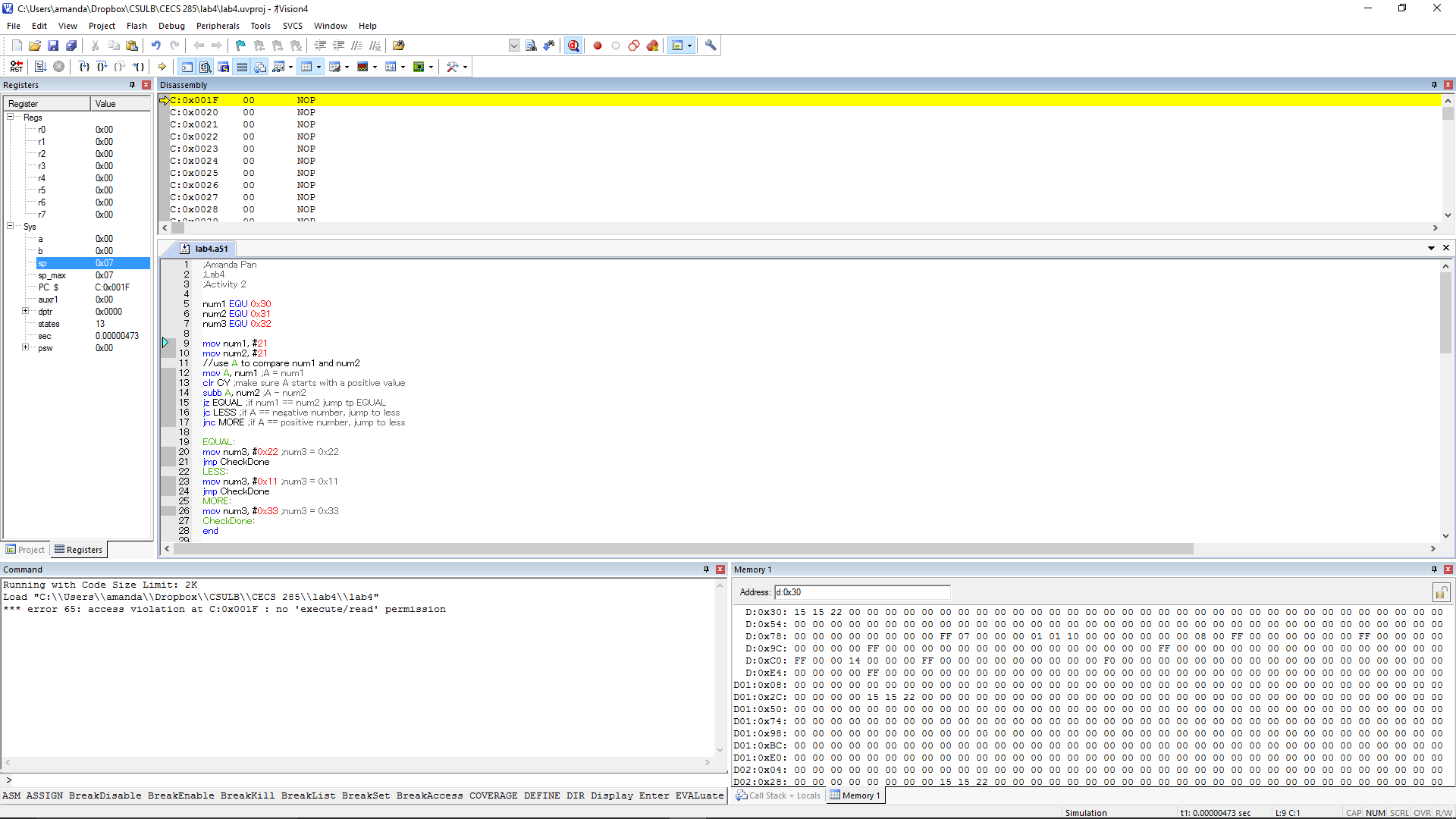
num1 = 21, num2 = 22



num1 = 21, num2 = 20



Num1=21, num2=21



**Activity 3:**

num1 EQU 0x30 ;address of num1 = 0x30

num2 EQU 0x31 ;address of num2 = 0x31

num3 EQU 0x32 ;address of num3 = 0x32

choice EQU 0x40 ; address of choice = 0x40

mov num1, #0x10 ;set num1 = 0x10

mov num2, #0x20 ;set num2 = 0x20

mov choice, #7 ;set choice = 1

mov R0, choice ; set R0 = choice = 1

cjne R0, #1, CHOICE2 ;jump to CHOICE2 if R0 is not 1

mov A, num1 ;set A = num1 = 0x010

add A, num2 ;A = num1 + num2 = 0x10 + 0x20

mov num3, A ;set num3 = A

jmp CheckDone ;jump to checkdone

CHOICE2:

cjne R0, #2, CHOICE3 ;jump to CHOICE3 if R0 is not 2

mov A, num1 ;set A = num1

subb A, num2 ;A = num1 - num2

mov num3, A ;set num3 = A

jmp CheckDone ;jump to checkdone

CHOICE3:

cjne R0, #3, OTHERS ;jump to OTHERS if R0 is not 3

mov A, num2 ;set A = num2

subb A, num1 ;A = num2 - num1 = 0x10 - 0x20

mov num3, A ;set num3 = A

jmp CheckDone ;jump to checkdone

OTHERS:

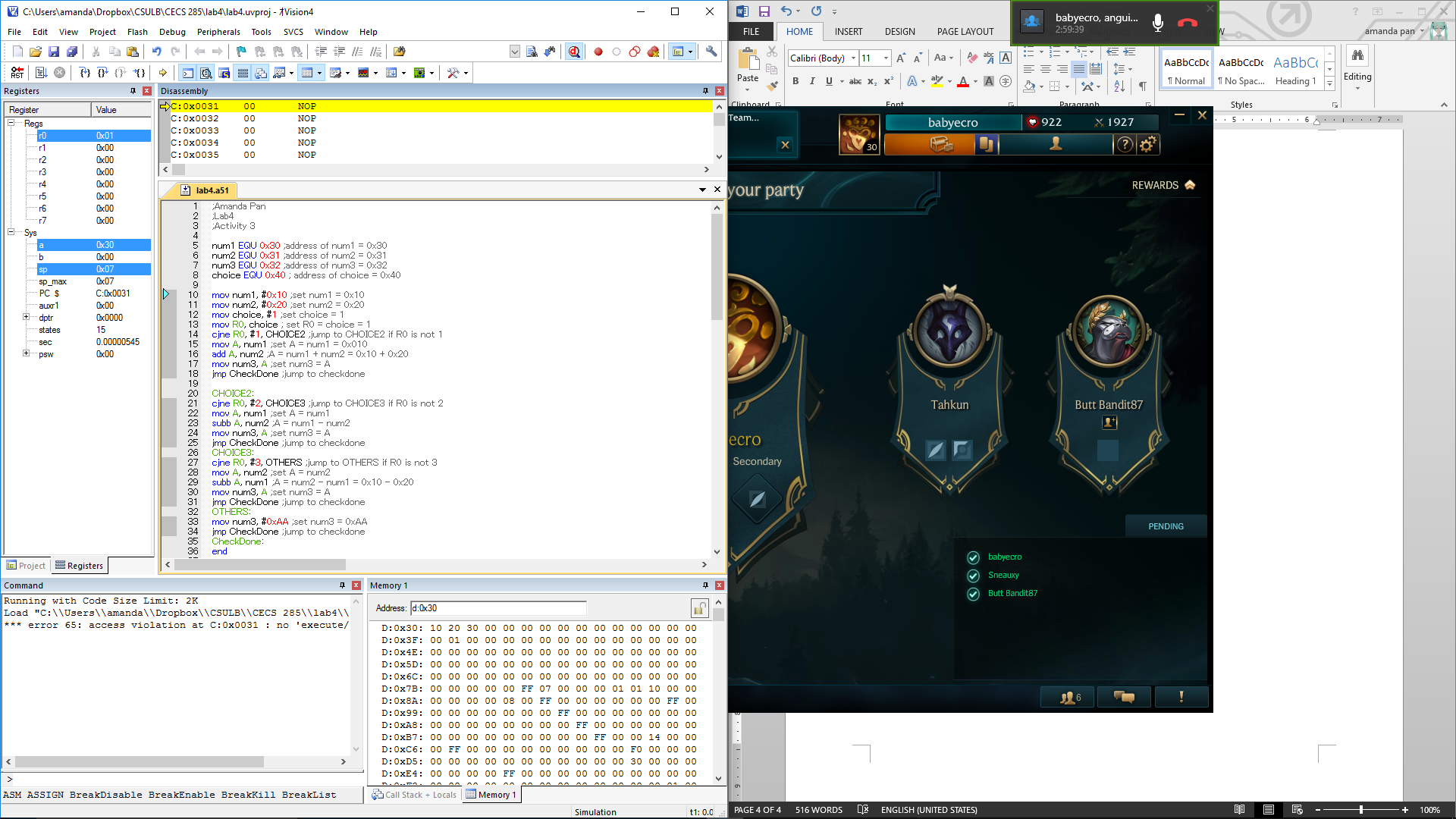
mov num3, #0xAA ;set num3 = 0xAA

jmp CheckDone ;jump to checkdone

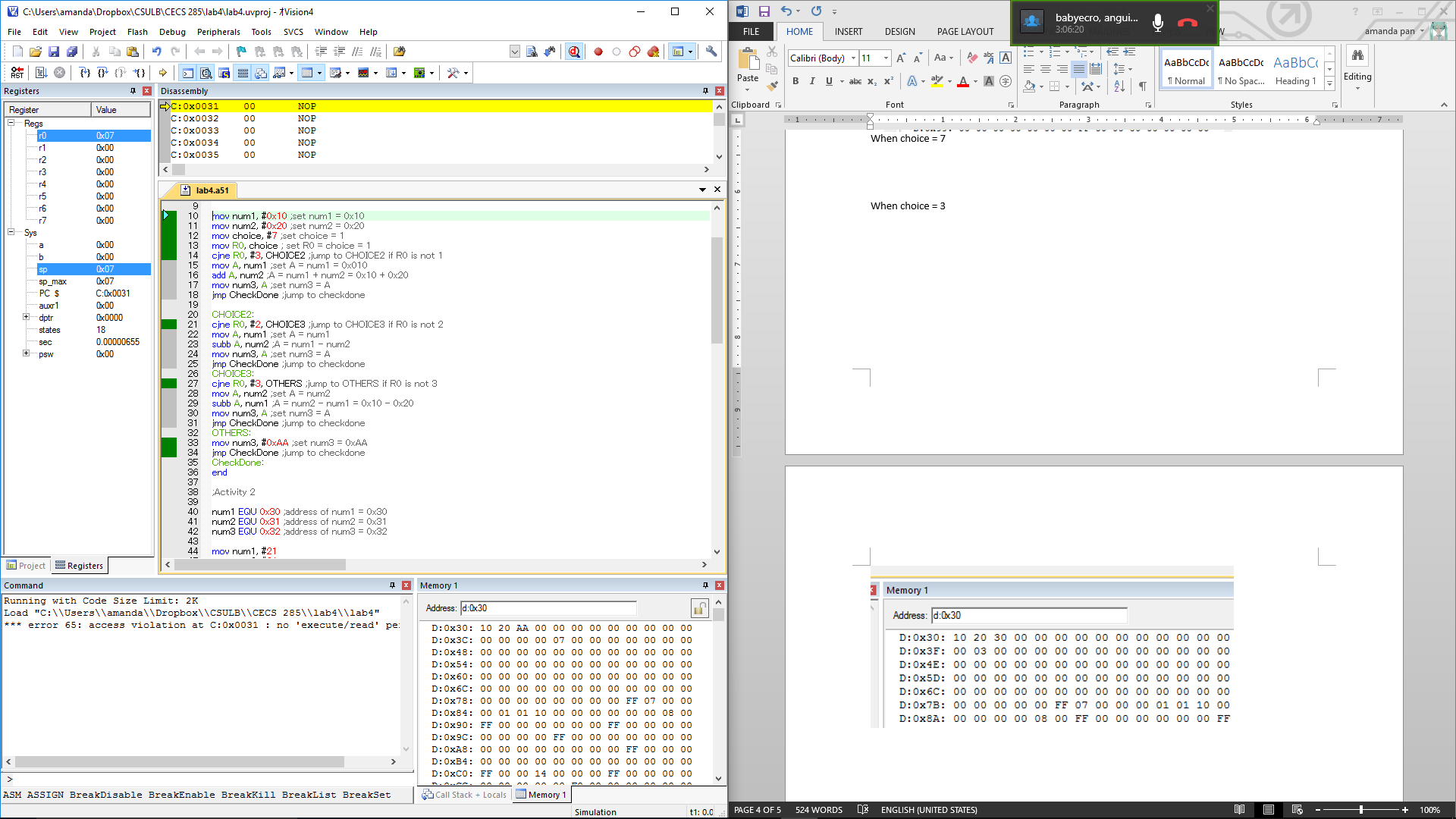
CheckDone:

end

when choice = 1



When choice = 7



When choice = 3

