**Bài thực hành week10+11**

**Asignment 4(week10):**

*Mã nguồn :*

.eqv KEY\_CODE 0xFFFF0004 # ASCII code from keyboard, 1 byte

.eqv KEY\_READY 0xFFFF0000 # =1 if has a new keycode ?

# Auto clear after lw

.eqv DISPLAY\_CODE 0xFFFF000C # ASCII code to show, 1 byte

.eqv DISPLAY\_READY 0xFFFF0008 # =1 if the display has already to do

# Auto clear after sw

.text

li $k0, KEY\_CODE

li $k1, KEY\_READY

li $s0, DISPLAY\_CODE

li $s1, DISPLAY\_READY

loop: nop

WaitForKey: lw $t1, 0($k1) # $t1 = [$k1] = KEY\_READY

nop

beq $t1, $zero, WaitForKey # if $t1 == 0 then Polling

nop

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

ReadKey: lw $t0, 0($k0) # $t0 = [$k0] = KEY\_CODE

nop

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

WaitForDis: lw $t2, 0($s1) # $t2 = [$s1] = DISPLAY\_READY

nop

beq $t2, $zero, WaitForDis # if $t2 == 0 then Polling

nop

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

Encrypt: addi $t0, $t0, 1 # change input key

beq $t0, 69, end

nop

beq $t0, 101, end

nop

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

ShowKey: sw $t0, 0($s0) # show key

nop

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

j loop

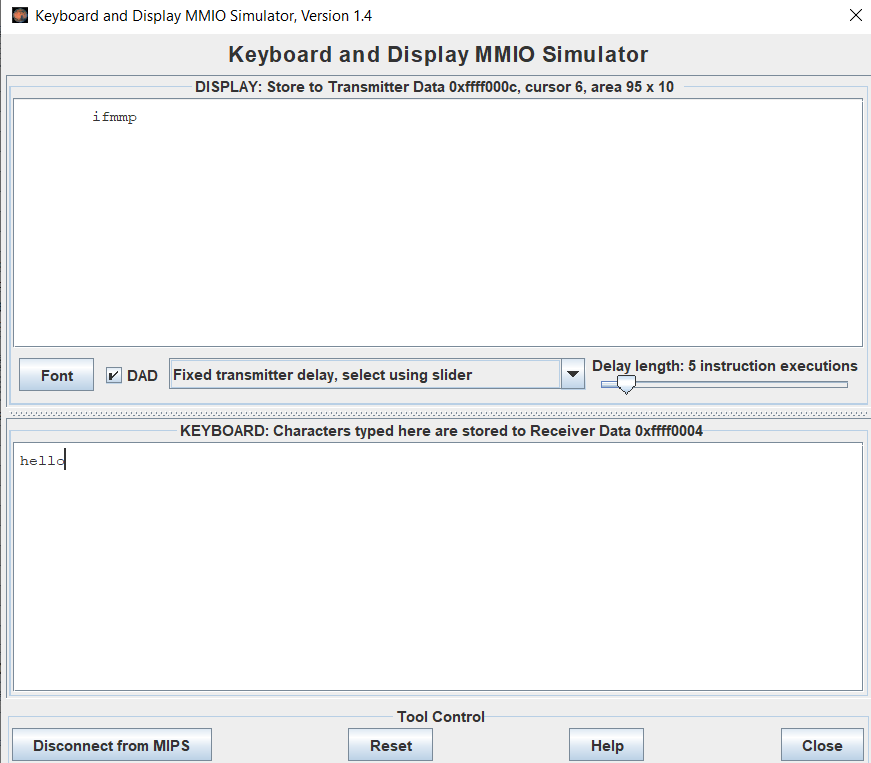
nop

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

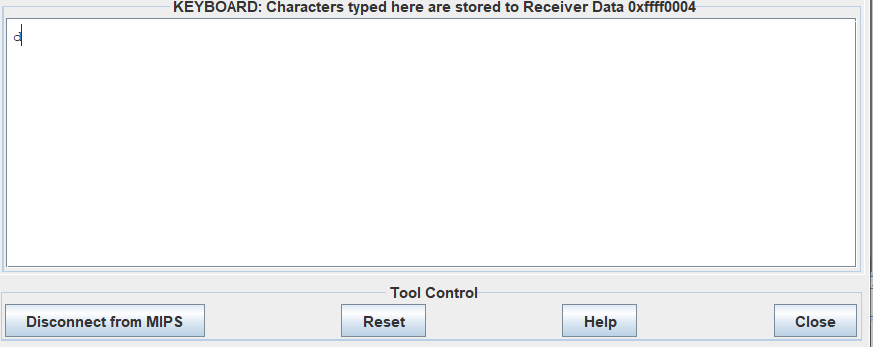
end: li $v0, 10

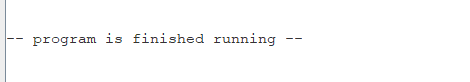
syscall

*Kết quả :*



Khi chạy kí tự ‘d’ thì chương trình kết thúc





**Assignment 1(week11):**

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

# col 0x1 col 0x2 col 0x4 col 0x8

#

# row 0x1 0 1 2 3

# 0x11 0x21 0x41 0x81

#

# row 0x2 4 5 6 7

# 0x12 0x22 0x42 0x82

#

# row 0x4 8 9 a b

# 0x14 0x24 0x44 0x84

#

# row 0x8 c d e f

# 0x18 0x28 0x48 0x88

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

# command row number of hexadecimal keyboard (bit 0 to 3)

# Eg. assign 0x1, to get key button 0,1,2,3

# assign 0x2, to get key button 4,5,6,7

# NOTE must reassign value for this address before reading,

# eventhough you only want to scan 1 row

.eqv IN\_ADRESS\_HEXA\_KEYBOARD 0xFFFF0012

# receive row and column of the key pressed, 0 if not key pressed

# Eg. equal 0x11, means that key button 0 pressed.

# Eg. equal 0x28, means that key button D pressed.

.eqv OUT\_ADRESS\_HEXA\_KEYBOARD 0xFFFF0014

.text

main: li $t1, IN\_ADRESS\_HEXA\_KEYBOARD

li $t2, OUT\_ADRESS\_HEXA\_KEYBOARD

return:

li $t3, 0x08 # check row 4 with key C, D,E, F

sb $t3, 0($t1 ) # must reassign expected row

lb $a0, 0($t2) # read scan code of key button

li $v0, 34 # print integer (hexa)

syscall

li $a0, 100 # sleep 100ms

li $v0, 32

syscall

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

li $t4, 0x04 # check row 3 with key 8,9,a,b

sb $t4, 0($t1 ) # must reassign expected row

lb $a0, 0($t2) # read scan code of key button

li $v0, 34 # print integer (hexa)

syscall

li $a0, 100 # sleep 100ms

li $v0, 32

syscall

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

li $t5, 0x02 # check row 2 with key 4,5,6,7

sb $t5, 0($t1 ) # must reassign expected row

lb $a0, 0($t2) # read scan code of key button

li $v0, 34 # print integer (hexa)

syscall

li $a0, 100 # sleep 100ms

li $v0, 32

syscall

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

li $t6, 0x01 # check row 1 with key 0,1,2,3

sb $t6, 0($t1 ) # must reassign expected row

lb $a0, 0($t2) # read scan code of key button

li $v0, 34 # print integer (hexa)

syscall

li $a0, 100 # sleep 100ms

li $v0, 32

syscall

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

j return