Laboratory Exercise 11 – Report:

Nguyễn Hải Dương- 20194530

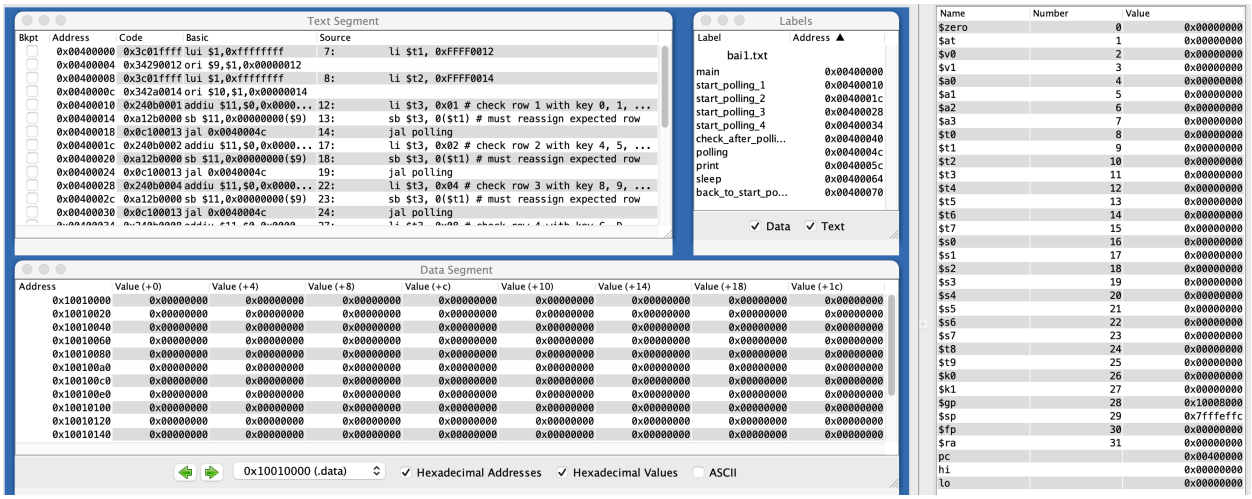
1. Assignment 1

- Mã nguồn:

```
.eqv IN_ADRESS_HEXA_KEYBOARD 0xFFFF0012
.eqv OUT_ADRESS_HEXA_KEYBOARD 0xFFFF0014


.text
main:
        li $t1, IN_ADRESS_HEXA_KEYBOARD
        li $t2, OUT_ADRESS_HEXA_KEYBOARD


start_polling_1:
        li $t3, 0x01 # check row 1 with key 0, 1, 2, 4
        sb $t3, 0($t1) # must reassign expected row
        jal polling

start_polling_2:
        li $t3, 0x02 # check row 2 with key 4, 5, 6, 7
        sb $t3, 0($t1) # must reassign expected row
        jal polling

start_polling_3:
        li $t3, 0x04 # check row 3 with key 8, 9, A, B
        sb $t3, 0($t1) # must reassign expected row
        jal polling

start_polling_4:
        li $t3, 0x08 # check row 4 with key C, D, E, F
        sb $t3, 0($t1) # must reassign expected row
        jal polling

check_after_polling_4:
        beq $a0, 0x0, print
        j start_polling_1
```

```
35  polling:
36          lb $a0, 0($t2) # read scan code of key button
37          bne $a0, 0x0, print
38          jr $ra
39
40  print:
41          li $v0, 34 # print integer (hexa)
42          syscall
43
44  sleep:
45          li $a0, 3000 # sleep 100ms
46          li $v0, 32
47          syscall
48
49  back_to_start_polling:
50          j start_polling_1       # back to check row 1
51
```

- Kết quả chạy:



MSSV: 20194530

- Giải thích:

Yêu cầu: Check toàn bộ các ký tự từ 0 -> F

In ra kết quả khi nhập mã số sinh viên từ bàn phím

0x41 là số 2;

0x11 là số 0;

0x21 là số 1;

0x24 là số 9;

0x12 là số 4;

0x22 là số 5;

0x41 là số 3;

0x11 là số 0;

MSSV: 20194530 → Kết quả như ảnh trên

2. Assignment 2

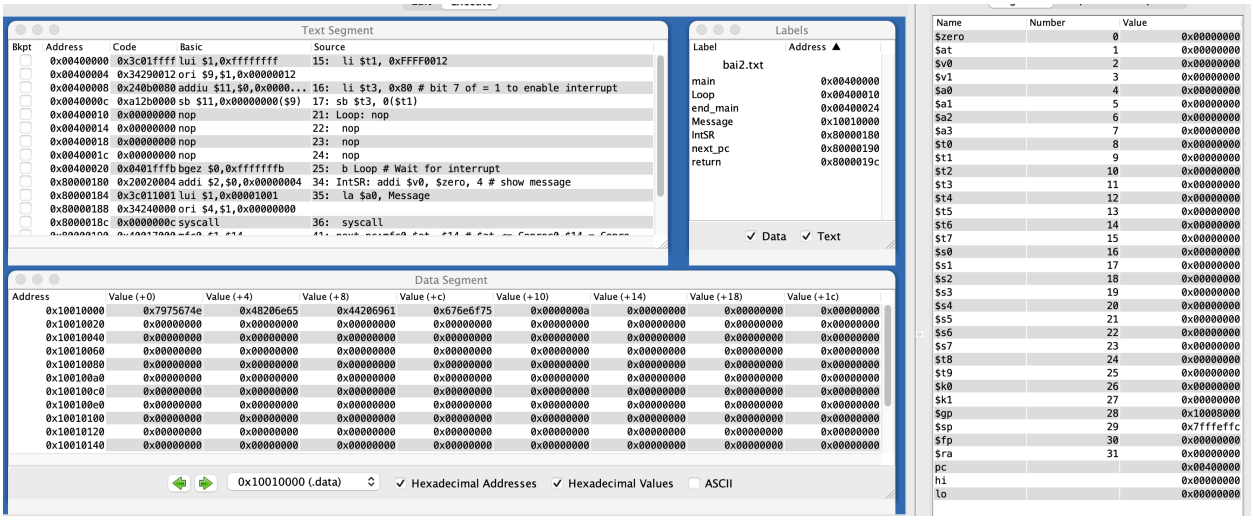
- Mã nguồn:

```
#home assignment 2

.eqv IN_ADRESS_HEXA_KEYBOARD 0xFFFF0012
.data
Message: .asciiz "Nguyen Hai Duong\n"
#~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
# MAIN Procedure
#~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
.text
main:
 #-------------------------------------------------------------
 # Enable interrupts you expect
 #-------------------------------------------------------------
 # Enable the interrupt of Keyboard matrix 4x4 of Digital Lab Sim
 li $t1, IN_ADRESS_HEXA_KEYBOARD
 li $t3, 0x80 # bit 7 of = 1 to enable interrupt
sb $t3, 0($t1)
 #-------------------------------------------------------------
 # No-end loop, main program, to demo the effective of interrupt
 #-------------------------------------------------------------
Loop: nop
 nop
 nop
 nop
 b Loop # Wait for interrupt
end_main:
#~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~ ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
# GENERAL INTERRUPT SERVED ROUTINE for all interrupts
#~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
.ktext 0x80000180
 #-------------------------------------------------------------
 # Processing
 #-------------------------------------------------------------
IntSR: addi $v0, $zero, 4 # show message
```

```
34  IntSR: addi $v0, $zero, 4 # show message
35   la $a0, Message
36   syscall
37   #---------------------------------------------------------------
38   # Evaluate the return address of main routine
39   # epc <= epc + 4
40   #---------------------------------------------------------------
41  next_pc:mfc0 $at, $14 # $at <= Coproc0.$14 = Coproc0.epc
42   addi $at, $at, 4 # $at = $at + 4 (next instruction)
43   mtc0 $at, $14 # Coproc0.$14 = Coproc0.epc <= $at
44  return: eret # Return from exception
45
```

- Kết quả chạy mô phỏng:

```
Clear

Reset: reset completed.

Reset: reset completed.

Nguyen Hai Duong
```



- Giải thích:

Khi nhấn phím bất kì từ 0 -> F thì sẽ hiện ra tên và mssv

3. Assignment 3

- Mã nguồn:

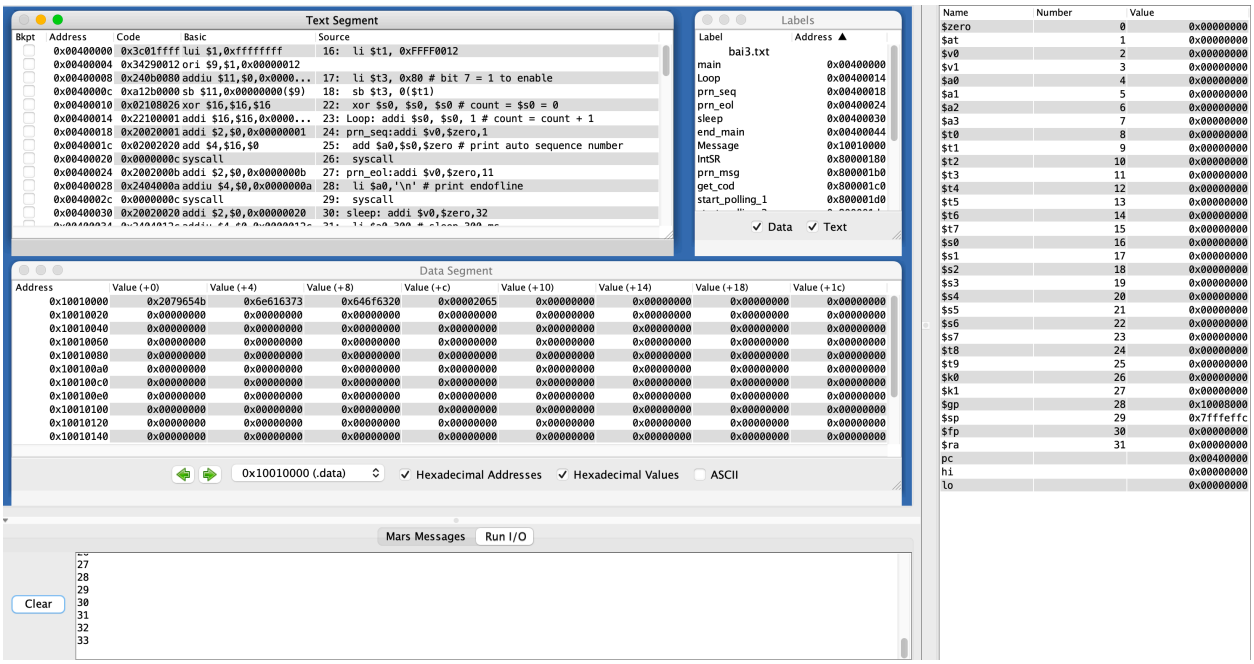
```
# home assignment 3

.eqv IN_ADRESS_HEXA_KEYBOARD 0xFFFF0012
.eqv OUT_ADRESS_HEXA_KEYBOARD 0xFFFF0014
.data
Message: .asciiz "Key scan code "
#~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
# MAIN Procedure
#~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
.text
main:
 #-------------------------------------------------------------
 # Enable interrupts you expect
 #-------------------------------------------------------------
 # Enable the interrupt of Keyboard matrix 4x4 of Digital Lab Sim
 li $t1, IN_ADRESS_HEXA_KEYBOARD
 li $t3, 0x80 # bit 7 = 1 to enable
 sb $t3, 0($t1)
#-------------------------------------------------------------
 # Loop an print sequence numbers
 #-------------------------------------------------------------
 xor $s0, $s0, $s0 # count = $s0 = 0
Loop: addi $s0, $s0, 1 # count = count + 1
prn_seq:addi $v0,$zero,1
 add $a0,$s0,$zero # print auto sequence number
 syscall
prn_eol:addi $v0,$zero,11
 li $a0,'\n' # print endofline
 syscall
sleep: addi $v0,$zero,32
 li $a0,300 # sleep 300 ms
 syscall
 nop # WARNING: nop is mandatory here.
 b Loop # Loop
```

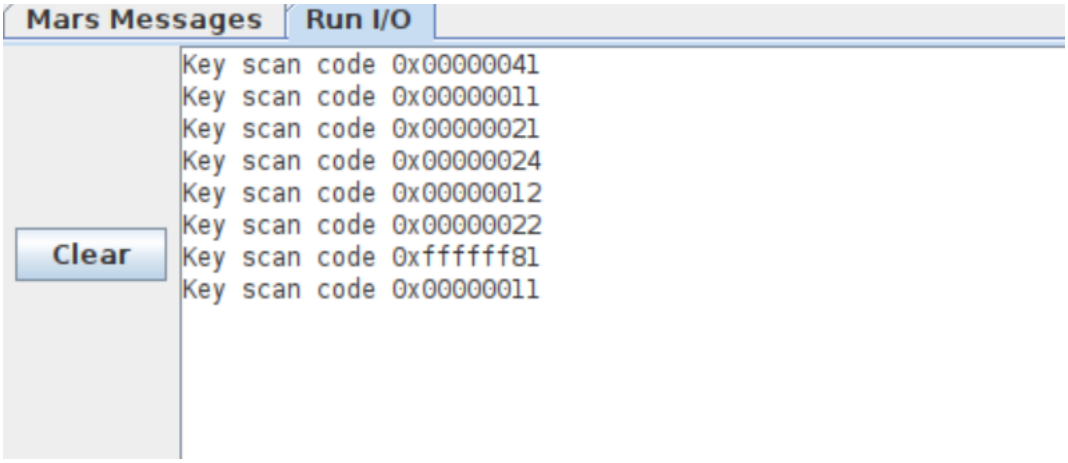
```
34   b Loop # Loop
35  end_main:
36  #~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
37  # GENERAL INTERRUPT SERVED ROUTINE for all interrupts
38  #~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
39  .ktext 0x80000180
40   #---------------------------------------------------------
41   # SAVE the current REG FILE to stack
42   #---------------------------------------------------------
43  IntSR: addi $sp,$sp,4 # Save $ra because we may change it later
44   sw $ra,0($sp)
45   addi $sp,$sp,4 # Save $ra because we may change it later
46   sw $at,0($sp)
47   addi $sp,$sp,4 # Save $ra because we may change it later
48   sw $v0,0($sp)
49   addi $sp,$sp,4 # Save $a0, because we may change it later
50   sw $a0,0($sp)
51   addi $sp,$sp,4 # Save $t1, because we may change it later
52   sw $t1,0($sp)
53   addi $sp,$sp,4 # Save $t3, because we may change it later
54   sw $t3,0($sp)
55   #---------------------------------------------------------
56   # Processing
57   #---------------------------------------------------------
58  prn_msg:addi $v0, $zero, 4
59   la $a0, Message
60   syscall
61  get_cod:
```

```
get_cod:
 li $t1, IN_ADRESS_HEXA_KEYBOARD
 li $t2, OUT_ADRESS_HEXA_KEYBOARD
start_interrupt_1:
        li $t3, 0x81 # check row 1 with key 0, 1, 2, 4
        sb $t3, 0($t1) # must reassign expected row
        jal interrupt

start_interrupt_2:
        li $t3, 0x82 # check row 2 with key 4, 5, 6, 7
        sb $t3, 0($t1) # must reassign expected row
        jal interrupt

start_interrupt_3:
        li $t3, 0x84 # check row 3 with key 8, 9, A, B
        sb $t3, 0($t1) # must reassign expected row
        jal interrupt

start_interrupt_4:
        li $t3, 0x88 # check row 4 with key C, D, E, F
        sb $t3, 0($t1) # must reassign expected row
        jal interrupt

check_after_interrupt_4:
        beq $a0, 0x0, prn_cod
        j next_pc

interrupt:
        lb $a0, 0($t2) # read scan code of key button
        bne $a0, 0x0, prn_cod
        jr $ra
prn_cod:li $v0,34
```

```
 92  prn_cod:li $v0,34
 93   syscall
 94   li $v0,11
 95   li $a0,'\n' # print endofline
 96   syscall
 97   #-----------------------------------------------------------
 98   # Evaluate the return address of main routine
 99   # epc <= epc + 4
100   #-----------------------------------------------------------
101  next_pc:mfc0 $at, $14 # $at <= Coproc0.$14 = Coproc0.epc
102   addi $at, $at, 4 # $at = $at + 4 (next instruction)
103   mtc0 $at, $14 # Coproc0.$14 = Coproc0.epc <= $at
104   #-----------------------------------------------------------
105   # RESTORE the REG FILE from STACK
106   #-----------------------------------------------------------
107  restore:lw $t3, 0($sp) # Restore the registers from stack
108   addi $sp,$sp,-4
109   lw $t1, 0($sp) # Restore the registers from stack
110   addi $sp,$sp,-4
111   lw $a0, 0($sp) # Restore the registers from stack
112   addi $sp,$sp,-4
113   lw $v0, 0($sp) # Restore the registers from stack
114   addi $sp,$sp,-4
115   lw $ra, 0($sp) # Restore the registers from stack
116   addi $sp,$sp,-4
117  return: eret # Return from exception
118
```

- Kết quả chạy:





- Giải thích:

In ra kết quả khi nhập MSSV vào Lab Sim

Kết quả ra như sau:

0x41 là số 2;

0x11 là số 0;

0x21 là số 1;

0x24 là số 9;

0x12 là số 4;

0x22 là số 5;

0x41 là số 3;

0x11 là số 0;

MSSV: 20194530 → Kết quả như ảnh trên