Sinh viên : Vũ Văn Trọng

Mã Số Sinh viên : 20184206

ASSIGNMENT 1

|  |
| --- |
| #------------------------------------------------------ |
|  | # 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 |
|  | polling1: |
|  | li $t3, 0x1 # 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 |
|  | bne $a0, $zero, print |
|  | polling2: li $t3, 0x2 # 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 |
|  | bne $a0, $zero, print |
|  | polling3: li $t3, 0x4 # 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 |
|  | bne $a0, $zero, print |
|  | polling4: li $t3, 0x8 # 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 |
|  | bne $a0, $zero, print |
|  | print: |
|  | li $v0, 34 # print integer (hexa) |
|  | syscall |
|  | sleep: |
|  | li $a0, 100 # sleep 100ms |
|  | li $v0, 32 |
|  | syscall |
|  | back\_to\_polling1: |
|  | j polling1 # continue polling |

* **Kết quả**

