**BÁO CÁO THỰC HÀNH**

**KIẾN TRÚC MÁY TÍNH LAB**

**Assignment 1:**

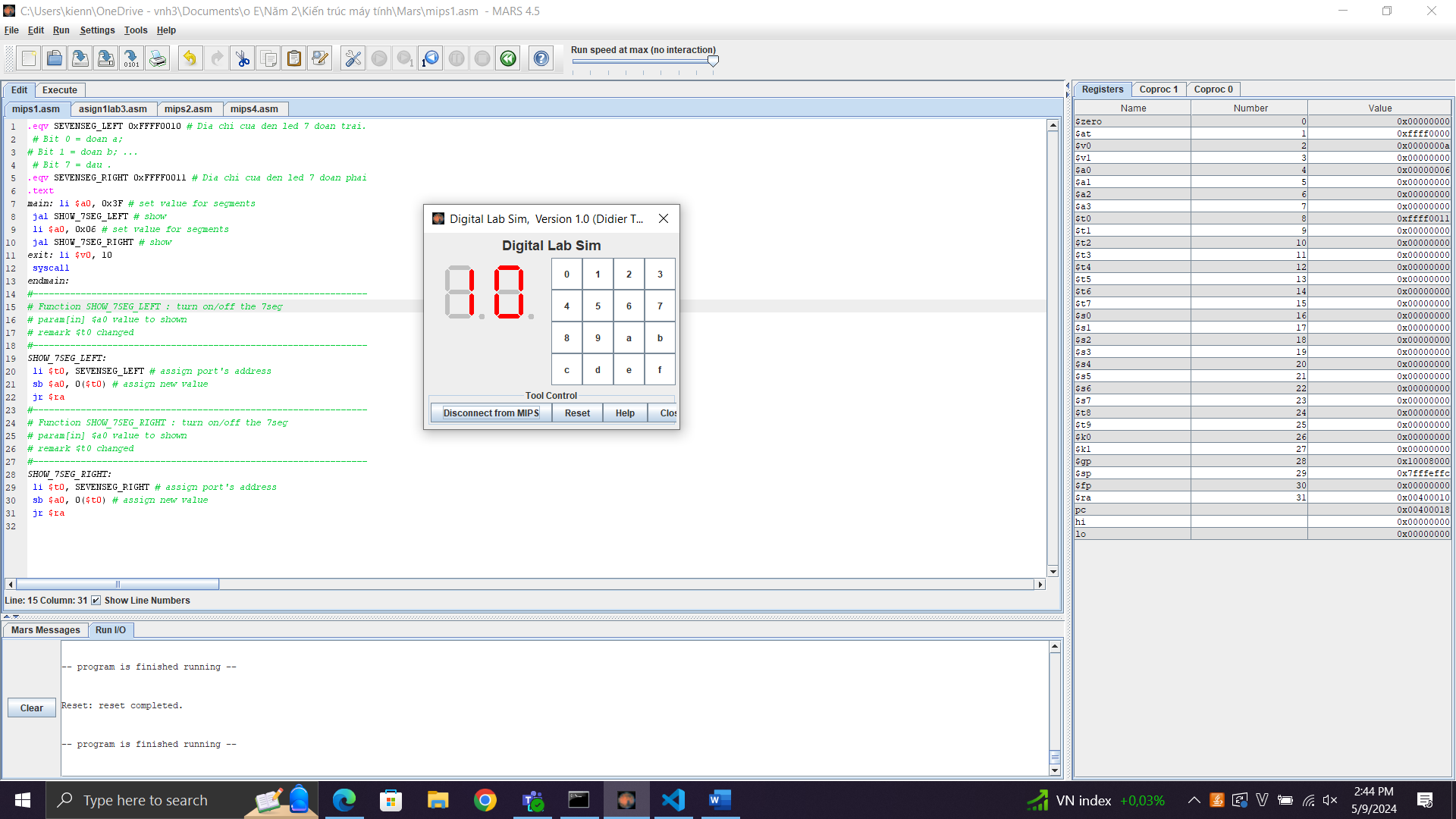
**Code:**

|  |
| --- |
| . eqv SEVENSEG\_LEFT 0xFFFF0010 # Dia chi cua den led 7 doan trai.  # Bit 0 = doan a;  # Bit 1 = doan b; ...  # Bit 7 = dau .  .eqv SEVENSEG\_RIGHT 0xFFFF0011 # Dia chi cua den led 7 doan phai  .text  main: li $a0, 0x3F # set value for segments  jal SHOW\_7SEG\_LEFT # show  li $a0, 0x06 # set value for segments  jal SHOW\_7SEG\_RIGHT # show  exit: li $v0, 10  syscall  endmain:  #---------------------------------------------------------------  # Function SHOW\_7SEG\_LEFT : turn on/off the 7seg  # param[in] $a0 value to shown  # remark $t0 changed  #---------------------------------------------------------------  SHOW\_7SEG\_LEFT:  li $t0, SEVENSEG\_LEFT # assign port's address  sb $a0, 0($t0) # assign new value  jr $ra  #---------------------------------------------------------------  # Function SHOW\_7SEG\_RIGHT : turn on/off the 7seg  # param[in] $a0 value to shown  # remark $t0 changed  #---------------------------------------------------------------  SHOW\_7SEG\_RIGHT:  li $t0, SEVENSEG\_RIGHT # assign port's address  sb $a0, 0($t0) # assign new value  jr $ra |

**Nhận xét:**

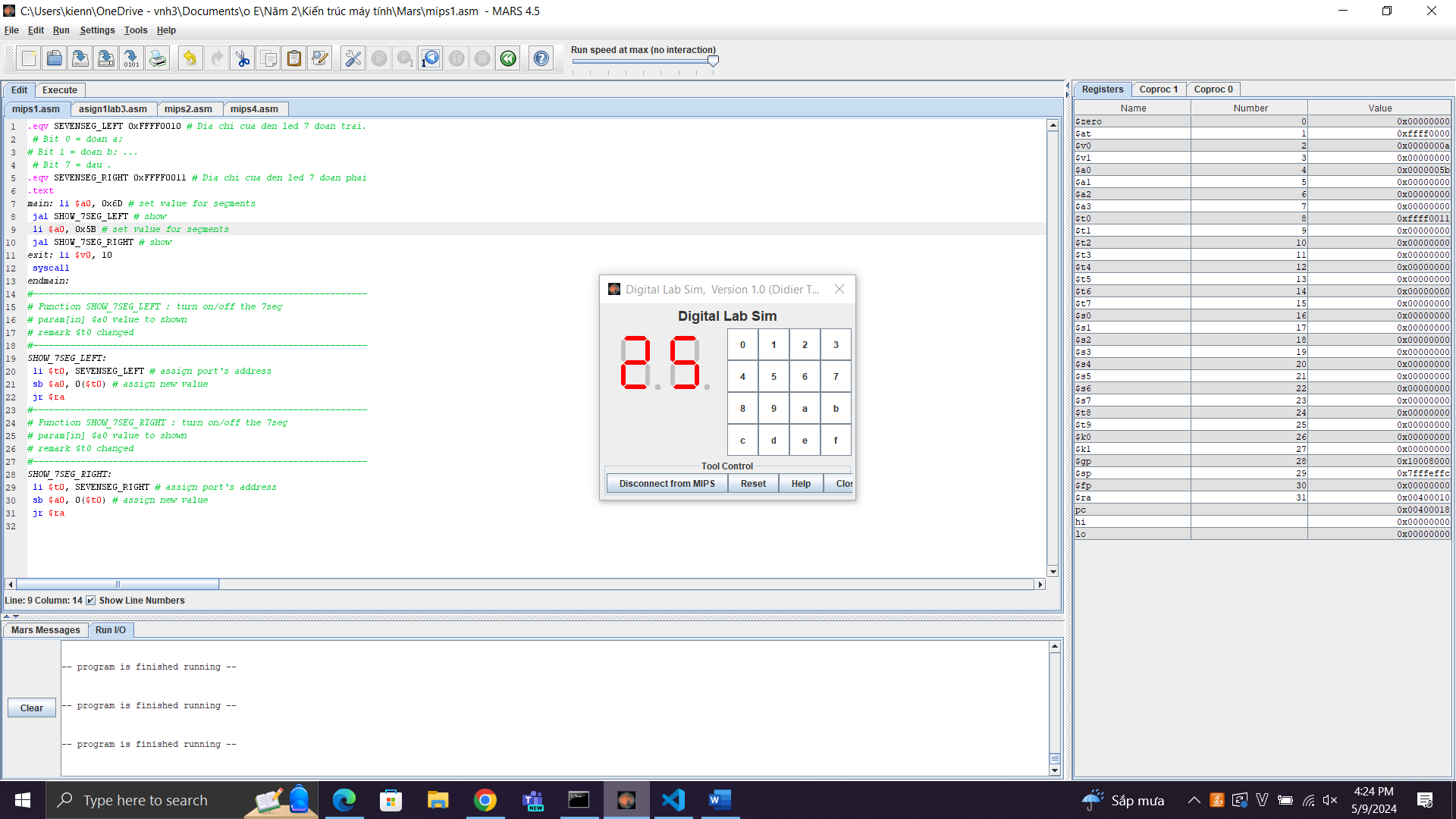
* Đoạn code trên dùng để hiển thị 2 số cuối MSSV trên LED 7 thanh
* MSSV = 20226110 => Hiển thị số 10 trên LED 7 thanh
  + Hiển thị số 1 => thanh b,c sáng => $a0 = 0x06
  + Hiển thị số 0 => thanh a,b,c,d,e,f => $a0 = 0x3F

**Kết quả:**



Ta chạy tương tự với 1 số hiển thị mong muốn: số 25 là ngày sinh của Nguyễn Trung Kiên

* + Hiển thị số 2 => thanh a,b,d,e,g sáng => $a0 = 0x5B
  + Hiển thị số 5 => thanh a,b,c,d,e,f => $a0 = 0x6D



**Assignment 2:**

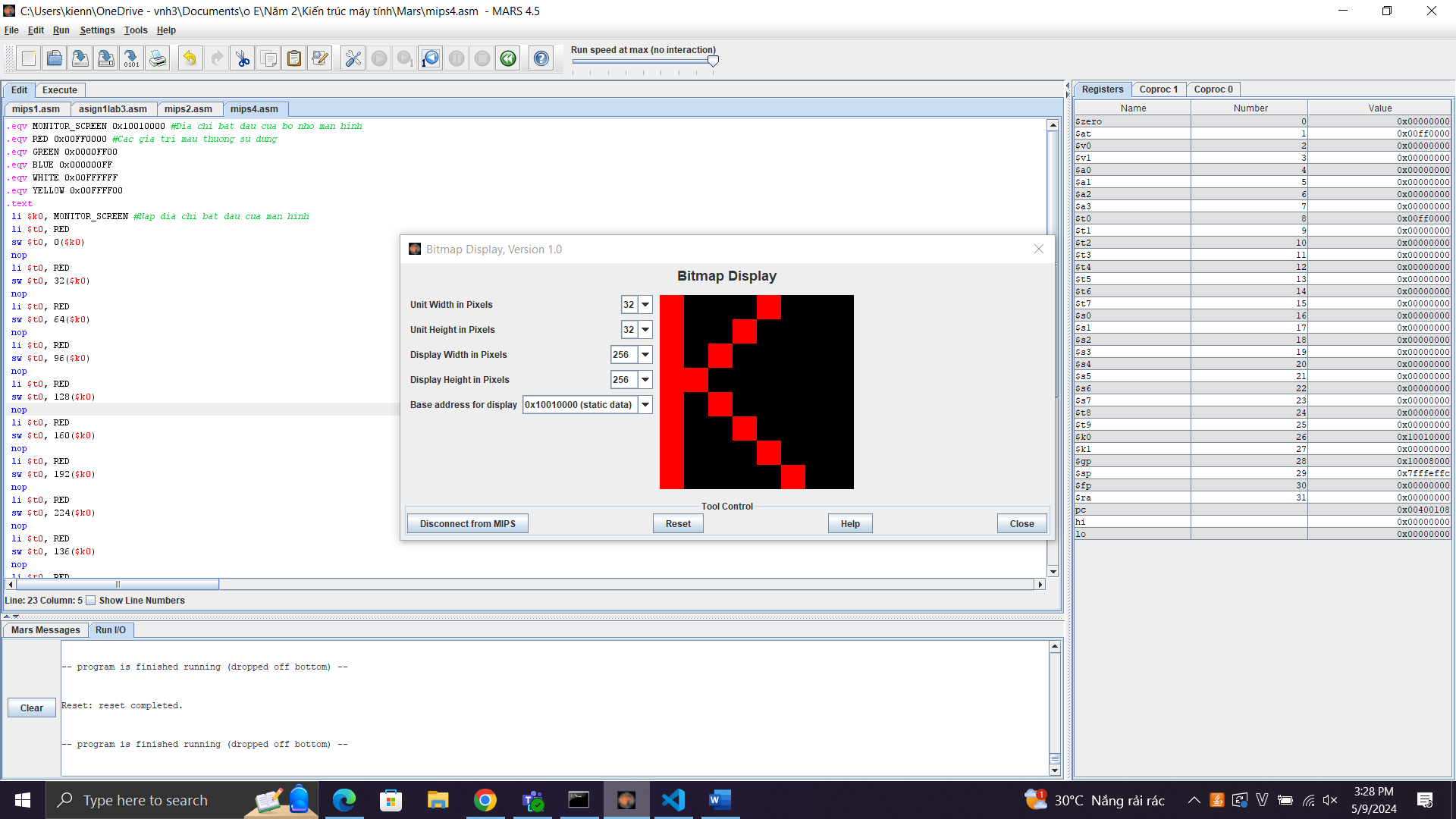
**Code:**

|  |
| --- |
| .eqv MONITOR\_SCREEN 0x10010000 #Dia chi bat dau cua bo nho man hinh  .eqv RED 0x00FF0000 #Cac gia tri mau thuong su dung  .eqv GREEN 0x0000FF00  .eqv BLUE 0x000000FF  .eqv WHITE 0x00FFFFFF  .eqv YELLOW 0x00FFFF00  .text  li $k0, MONITOR\_SCREEN #Nap dia chi bat dau cua man hinh  li $t0, RED  sw $t0, 0($k0)  nop  li $t0, RED  sw $t0, 32($k0)  nop  li $t0, RED  sw $t0, 64($k0)  nop  li $t0, RED  sw $t0, 96($k0)  nop  li $t0, RED  sw $t0, 128($k0)  nop  li $t0, RED  sw $t0, 160($k0)  nop  li $t0, RED  sw $t0, 192($k0)  nop  li $t0, RED  sw $t0, 224($k0)  nop  li $t0, RED  sw $t0, 136($k0)  nop  li $t0, RED  sw $t0, 100($k0)  nop  li $t0, RED  sw $t0, 172($k0)  nop  li $t0, RED  sw $t0, 208($k0)  nop  li $t0, RED  sw $t0, 244($k0)  nop  li $t0, RED  sw $t0, 72($k0)  nop  li $t0, RED  sw $t0, 44($k0)  nop  li $t0, RED  sw $t0, 16($k0)  nop |

**Nhận xét:**

* Đoạn code trên dùng để hiển thị chữ K, chữ cái đầu tiên của Nguyễn Trung Kiên
* Ta hiển thị chữ K bằng cách kẻ 1 đường dọc và 2 đường chéo. Đường dọc là các vị trí có offset chia hết cho 32 trong mảng và đường chéo ta tính toán vị trí hơn kém nhau 28 (đối với đường chéo lên) và 36 (đối với đường chéo dưới).

**Kết quả:**



**Assignment 3:**

**Code:**

|  |
| --- |
| .eqv HEADING 0xffff8010 # Integer: An angle between 0 and 359  # 0 : North (up)  # 90: East (right)  # 180: South (down)  # 270: West (left)  .eqv MOVING 0xffff8050 # Boolean: whether or not to move  .eqv LEAVETRACK 0xffff8020 # Boolean (0 or non-0):  # whether or not to leave a track  .eqv WHEREX 0xffff8030 # Integer: Current x-location of MarsBot  .eqv WHEREY 0xffff8040 # Integer: Current y-location of MarsBot  .text  main: jal TRACK # draw track line  nop  addi $a0, $zero, 90 # Marsbot rotates 90\* and start running  jal ROTATE  nop  jal GO  nop  sleep1: addi $v0,$zero,32 # Keep running by sleeping in 1000 ms  li $a0,1000  syscall    jal UNTRACK # keep old track  nop  jal TRACK # and draw new track line  nop  goDOWN: addi $a0, $zero, 180 # Marsbot rotates 180\*  jal ROTATE  nop    sleep2: addi $v0,$zero,32 # Keep running by sleeping in 5000 ms  li $a0,5000  syscall  jal UNTRACK # keep old track  nop  jal TRACK # and draw new track line  nop  goUP: addi $a0, $zero, 0 # Marsbot rotates 0\*  jal ROTATE  nop    sleep3: addi $v0,$zero,32 # Keep running by sleeping in 2500 ms  li $a0,2500  syscall  jal UNTRACK # keep old track  nop  jal TRACK # and draw new track line  nop    goASKEW:addi $a0, $zero, 120 # Marsbot rotates 120\*  jal ROTATE  nop    sleep4: addi $v0,$zero,32 # Keep running by sleeping in 4500 ms  li $a0, 4500  syscall    jal UNTRACK # keep old track  nop  jal TRACK # and draw new track line  nop    goASKEW1:addi $a0, $zero, 300 # Marsbot rotates 300\*  jal ROTATE  nop    sleep5: addi $v0,$zero,32 # Keep running by sleeping in 4500 ms  li $a0,4500  syscall    jal UNTRACK # keep old track  nop  jal TRACK # and draw new track line  nop    goASKEW3:addi $a0, $zero, 30 # Marsbot rotates 30\*  jal ROTATE  nop    sleep6: addi $v0,$zero,32 # Keep running by sleeping in 4500 ms  li $a0,4500  syscall    jal UNTRACK # keep old track  nop  jal TRACK # and draw new track line  nop    j endgame  end\_main:    #-----------------------------------------------------------  # GO procedure, to start running  # param[in] none  #-----------------------------------------------------------  GO: li $at, MOVING # change MOVING port  addi $k0, $zero,1 # to logic 1,  sb $k0, 0($at) # to start running  nop  jr $ra  nop  #-----------------------------------------------------------  # STOP procedure, to stop running  # param[in] none  #-----------------------------------------------------------  STOP: li $at, MOVING # change MOVING port to 0  sb $zero, 0($at) # to stop  nop  jr $ra  nop  #-----------------------------------------------------------  # TRACK procedure, to start drawing line  # param[in] none  #-----------------------------------------------------------  TRACK: li $at, LEAVETRACK # change LEAVETRACK port  addi $k0, $zero,1 # to logic 1,  sb $k0, 0($at) # to start tracking  nop  jr $ra  nop  #-----------------------------------------------------------  # UNTRACK procedure, to stop drawing line  # param[in] none  #-----------------------------------------------------------  UNTRACK:li $at, LEAVETRACK # change LEAVETRACK port to 0  sb $zero, 0($at) # to stop drawing tail  nop  jr $ra  nop  #-----------------------------------------------------------  # ROTATE procedure, to rotate the robot  # param[in] $a0, An angle between 0 and 359  # 0 : North (up)  # 90: East (right)  # 180: South (down)  # 270: West (left)  #-----------------------------------------------------------  ROTATE: li $at, HEADING # change HEADING port  sw $a0, 0($at) # to rotate robot  nop  jr $ra  nop    endgame: |

**Nhận xét:**

* Đoạn code trên dùng MarsBot để vẽ chữ K, chữ cái đầu tiên của tên Nguyễn Trung Kiên
* Ta hiển thị chữ K bằng cách kẻ 1 đường dọc và 2 đường chéo.
* Đầu tiên ta thiết lập Marsbot 90 độ (hướng phải) và di chuyển ra phải 1000ms.
* Ta thiết lập Marsbot 180 độ (hướng xuống dưới) và di chuyển xuống dưới 5000ms.
* Ta thiết lập Marsbot 0 độ (hướng lên trên) và di chuyển xuống dưới 2500ms. Mục đích là vào giữa đoạn thẳng rồi kẻ đường chéo.
* Ta thiết lập Marsbot 120 độ (hướng chéo xuống) và di chuyển 4500ms.
* Ta thiết lập Marsbot 300 độ (hướng ngược lại) và di chuyển 4500ms để Bot quay lại vị trí ban đầu.
* Ta thiết lập Marsbot 30 độ (hướng chéo lên) và di chuyển 4500ms.

**Kết quả:**

