

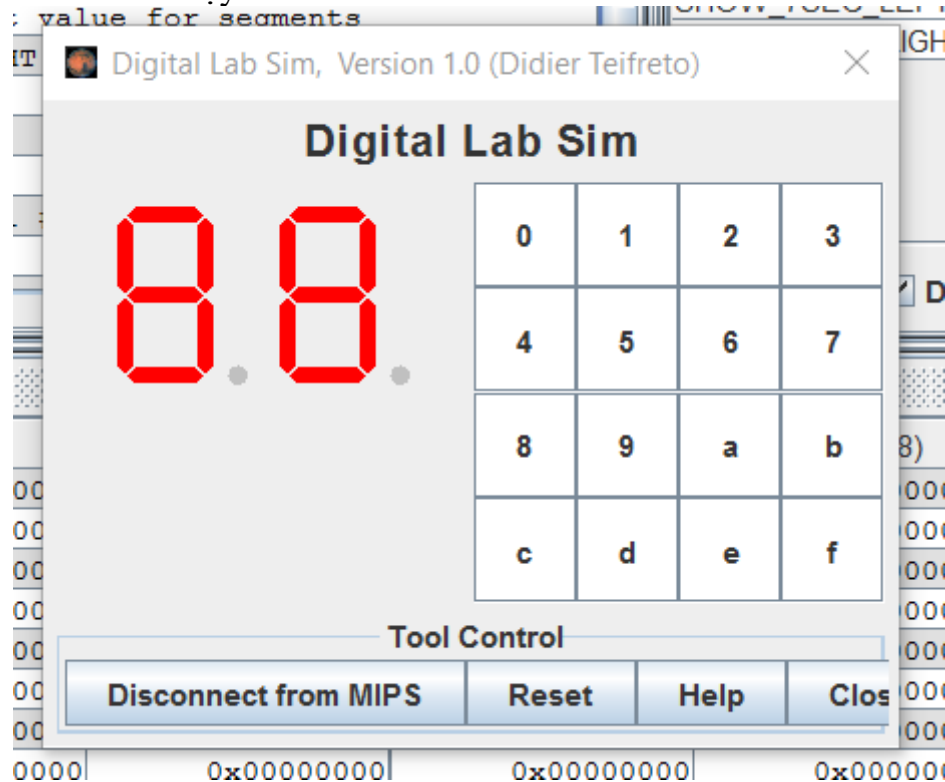
# Báo cáo tuần 10

## Bài 1

- Mã nguồn:

```
1  .eqv SEVENSEG_LEFT 0xFFFF0011      # Địa chỉ của đèn led 7 đoạn trái.
2                                     # Bit 0 = đoạn a;
3                                     # Bit 1 = đoạn b; ...
4                                     # Bit 7 = dấu .
5  .eqv SEVENSEG_RIGHT 0xFFFF0010     # Địa chỉ của đèn led 7 đoạn phải
6  .text
7  main:
8      li $a0, 0x7F # set value for segments
9      jal SHOW_7SEG_LEFT # show
10     nop
11     li $a0, 0x7F # set value for segments
12     jal SHOW_7SEG_RIGHT # show
13     nop
14 exit:  li $v0, 10
15         syscall
16 endmain:
17 #-----
18 # Function SHOW_7SEG_LEFT : turn on/off the 7seg
19 # param[in] $a0 value to shown
20 # remark $t0 changed
21 #-----
22 SHOW_7SEG_LEFT:
23     li $t0, SEVENSEG_LEFT # assign port's address
24     sb $a0, 0($t0) # assign new value
25     nop
26     jr $ra
27     nop
28 #-----
29 # Function SHOW_7SEG_RIGHT : turn on/off the 7seg
30 # param[in] $a0 value to shown
31 # remark $t0 changed
32 #-----
33 SHOW_7SEG_RIGHT:
34     li $t0, SEVENSEG_RIGHT # assign port's address
35     sb $a0, 0($t0) # assign new value
36     nop
37     jr $ra
38     nop
```

- Màn hình chạy:



- Giải thích:
  - Hàm main (Dòng 7-13):
    - li, a0, 0x7F: Do số 8 phải hiện tất cả các thanh -> 0x7F
    - jal đến hàm Show\_7Seg\_Left hoặc Show\_7Leg\_Right
    - Theo sau jal là nop
  - Hàm exit: li v0 10 => 10 là lệnh exit
  - 2 hàm Show\_7Seg\_Left, Show\_7Leg\_Right: Cố định như nhau
    - Dòng 23: Gán địa chỉ port cho t0
    - Dòng 24: Gán giá trị tại địa chỉ thứ 0 trong t0 cho a0
    - Dòng 25-26: nop và quay trở lại địa chỉ cũ (\$ra)

## Bài 2

- Mã nguồn

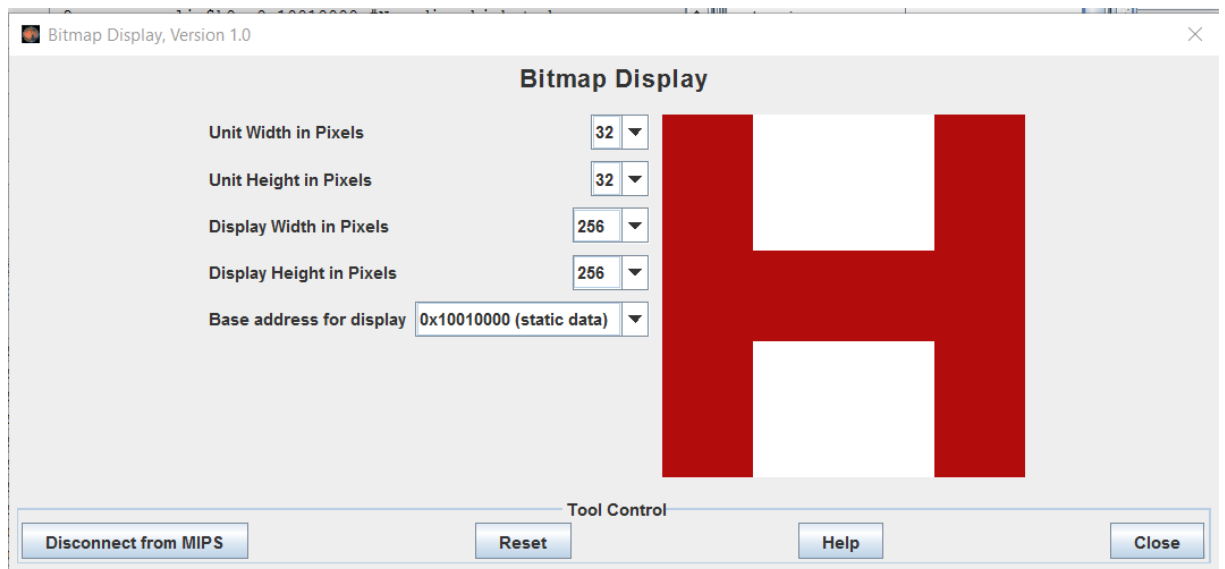
```
1 .eqv MONITOR_SCREEN 0x10010000 #Dia chi bat dau cua bo nho man hinh
2 .eqv RED 0x00FF0000 #Cac gia tri mau thuong su dung
3 .eqv GREEN 0x0000FF00
4 .eqv BLUE 0x000000FF
5 .eqv WHITE 0x00FFFFFF
6 .eqv YELLOW 0x00FFFF00
7 .eqv WRED 0x00B20C0C
8 .text
9     li $k0, MONITOR_SCREEN #Nap dia chi bat dau cua man hinh
10 main:    addi $s1, $0, 8      #Size
11         jal display_col
12         nop
13         jal display_row
14         nop
15 end_main:
16         li $v0, 10
17         syscall
18 display_col:
19         addi $t0, $0, 0      #i
20 col_i:    beq $t0, 0, blue_col
21         beq $t0, 1, blue_col
22         beq $t0, 6, blue_col
23         beq $t0, 7, blue_col
24         j white_col
25         nop
26 blue_col:
27         mul $t2, $t0, 4
28         add $t2, $t2, $k0
29         li $t1, 0            #j
30 col_display_blue:
31         mul $t3, $t1, 32
32         add $t3, $t3, $t2
33         li $a0, WRED
34         sw $a0, 0($t3)
35         addi $t1, $t1, 1
36         slt $t4, $t1, $s1
37         bne $t4, 0, col_display_blue
38         j check_col_i
39         nop
```

```

40 white_col:
41     mul $t2, $t0, 4
42     add $t2, $t2, $k0
43     li $t1, 0                #j
44 col_display_white:
45     mul $t3, $t1, 32
46     add $t3, $t3, $t2
47     li $a0, WHITE
48     sw $a0, 0($t3)
49     addi $t1, $t1, 1
50     slt $t4, $t1, $s1
51     bne $t4, 0, col_display_white
52     j check_col_i
53     nop
54 check_col_i: addi $t0, $t0, 1
55     slt $t4, $t0, $s1
56     bne $t4, 0, col_i
57     jr $ra
58 display_row:
59     addi $t0, $0, 0          #i
60 row_i:  beq $t0, 3, blue_row
61     beq $t0, 4, blue_row
62     j check_row_i
63     nop
64 blue_row:
65     mul $t2, $t0, 32
66     add $t2, $t2, $k0
67     li $t1, 0                #j
68 row_display_blue:
69     mul $t3, $t1, 4
70     add $t3, $t3, $t2
71     li $a0, WRED
72     sw $a0, 0($t3)
73     addi $t1, $t1, 1
74     slt $t4, $t1, $s1
75     bne $t4, 0, row_display_blue
76     j check_row_i
77     nop
78 check_row_i:
79     addi $t0, $t0, 1
80     slt $t4, $t0, $s1
81     bne $t4, 0, row_i
82     jr $ra
83

```

- Màn hình chạy



- Giải thích
  - Thực hiện in cột:
    - biến chạy  $i$  biểu diễn cột, nếu  $i=0,1,6,7$ 
      - ⇒ thực hiện display WRED (nếu không, thực hiện display white) với biến chạy  $j$  ( $0 \rightarrow 7$ ) biểu diễn hàng
  - Thực hiện in hàng:
    - biến chạy  $i$  biểu diễn hàng, nếu  $i=3,4$ 
      - ⇒ thực hiện display WRED (nếu không, thực hiện display white) với biến chạy  $j$  ( $0 \rightarrow 7$ ) biểu diễn cột
  - Mỗi hàng trên bitmap display có 32 byte
    - ⇒ cộng 32 byte vào địa chỉ hiện tại để xuống dòng