



Ε.Μ.Π. - ΣΧΟΛΗ ΗΛΕΚΤΡΟΛΟΓΩΝ ΜΗΧ. ΚΑΙ ΜΗΧΑΝΙΚΩΝ ΥΠΟΛΟΓΙΣΤΩΝ
ΤΟΜΕΑΣ ΤΕΧΝΟΛΟΓΙΑΣ ΠΛΗΡΟΦΟΡΙΚΗΣ ΚΑΙ ΥΠΟΛΟΓΙΣΤΩΝ
ΕΡΓΑΣΤΗΡΙΟ ΜΙΚΡΟΥΠΟΛΟΓΙΣΤΩΝ ΚΑΙ ΨΗΦΙΑΚΩΝ ΣΥΣΤΗΜΑΤΩΝ
ΑΚΑΔ. ΕΤΟΣ 2019-2020

Συστήματα Μικροϋπολογιστών, Ροή Υ, εξάμηνο 6^ο

3^η Σειρά Αναλυτικών Ασκήσεων

Στοιχεία φοιτητών:

Όνομα: Μαντζόυτας Ανδρέας, ΑΜ: 03117108, Εξάμηνο: 6^ο

Όνομα: Τσιτσής Αντώνιος, ΑΜ: 03117045, Εξάμηνο: 6^ο

Ημερομηνία Παράδοσης: 01/06/2020

1^η Άσκηση

Παραθέτουμε τον κώδικα της πρώτης άσκησης, η οποία και επισυνάπτεται με όνομα **Ask1Seira3.8085**.

```
IN 10H
LXI H,0B02H
MVI M,10H
LXI H,0B03H
MVI M,10H
LXI H,0B04H
MVI M,10H
LXI H,0B05H
MVI M,10H

START:
    MVI A,0DH
    SIM
    EI           ;RTS 6.5 routine

WAIT:
    JMP WAIT

INTR_ROUTINE:
    MVI C,61H

ON_LEDS:
    MVI B,00H      ;1/2sec delay
    MVI A,00H
    STA 3000H      ;turn off for 1/2 sec
    EI             ;enable interrupts

DECIMAL_TIMER:
    DCR C
    MOV A,C
    CPI 00H
    JZ WAIT
    MOV A,C      ;if sub leads to F last digit decrease to have
    ANI 0FH      ;9 as last to make it behave like a decimal
    CPI 0FH
    JZ DECREASE_DECIMAL
    JMP DISPLAY

DECREASE_DECIMAL:
    MOV A,C
    SUI 06H
    MOV C,A

DISPLAY:
    INR B
    MOV A,B
    CPI 10H
    JZ OFF_LEDS
    CPI 20H
    JZ ON_LEDS
    MVI B,02H
    CALL DELB
    MOV B,A
    LXI H,0B00H
    MOV A,C
    ANI 0FH      ;isolate4 LSB'S
    MOV M,A
    INX H        ;point to 0A01H
    MOV A,C
    ANI F0H      ;isolate 4 MSB'S
    RRC          ;shift LSB'S
```

```

    RRC
    RRC
    RRC
    MOV M,A
    LXI D,0B00H      ;display them
    CALL STDM
    CALL DCD
    JMP DISPLAY

OFF_LEDs:
    MVI B,11H
    MVI A,FFH
    STA 3000H      ;turn off for 1/2 sec
    JMP DISPLAY

END

```

2^η Άσκηση

```

    IN 10H
    MVI A,0DH
    MVI B,00H
    SIM
    EI
    LXI H,0A00H
    MVI M,10H
    LXI H,0A01H
    MVI M,10H
    LXI H,0A02H
    MVI M,10H
    LXI H,0A03H
    MVI M,10H

START:
    JMP START

INTR_ROUTINE:
    EI
    JMP READING

READING:
    CALL KIND      ;first digit
    RAL
    RAL
    RAL
    RAL
    MOV B,A
    CALL KIND      ;second digit
    ADD B
    MOV B,A

DECISION:
    MVI C,32H      ;50
    CMP C
    JC LED1
    JZ LED1      ;If input belongs [0,50]
    MVI D,64H      ;100
    CMP D
    JC LED2
    JZ LED2      ;If input belongs (50,100]
    MVI E,96H      ;150
    CMP E
    JC LED3
    JZ LED3      ;If input belongs (100,150]
    JMP LED4      ;If input belongs (150,255]

```

```

LED1: MVI A,FEH      ;1st LED to the right
      STA 3000H
      JMP DISPLAY

LED2: MVI A,FDH      ;2st LED to the right
      STA 3000H
      JMP DISPLAY

LED3: MVI A,FBH      ;3st LED to the right
      STA 3000H
      JMP DISPLAY

LED4: MVI A,F7H      ;4st LED to the right
      STA 3000H
      JMP DISPLAY

DISPLAY:
      LXI H,0A04H
      MOV A,B
      ANI 0FH          ;isolate 4 LSB'S
      MOV M,A
      INX H            ;point to 0A05H
      MOV A,B
      ;
      ANI F0H          ;isolate 4 MSB'S
      RRC
      RRC
      RRC
      RRC
      MOV M,A

      LXI D,0A00H      ;display them
      CALL STDM

HHH:  CALL DCD
      JMP HHH

END

```

Η άσκηση επισυνάπτεται με όνομα Ask2Seira3.8085.

3^η Άσκηση

α)

INR16 MACRO ADDR:

```
PUSH H  
PUSH B  
  
LXI H,ADDR      ; (H)(L) <- ADDR  
MOV C,M        ; C <- Xlow  
INX H  
MOV B,M        ; B <- Xhigh  
INX B          ; BC <- BC + 1  
MOV M,B        ; (ADDR+1) <- B  
DCX H          ;  
MOV M,C        ; (ADDR) <- C  
  
POP B  
POP H  
ENDM
```

β)

FILL MACRO ADDR,K:

```
PUSH H  
PUSH PSW  
  
MOV A,K  
LXI H,ADDR  
  
CPI 00H  
JNZ K_NOT_ZERO  
  
MVI M,00H ;IF WE ARE HERE, K IS 0  
INX H  
MVI A,FFH  
  
K_NOT_ZERO:  
    MOV M,A  
    DCR A  
    INX H  
    MOV M,A  
    CPI 00H  
    JNZ K_NOT_ZERO  
  
    POP PSW  
    POP H  
ENDM
```

γ)

```
RHLR MACRO Q,R  
    MOV A,R  
    RAL ;CY= 7o bit tou A  
    MOV R,A  
    MOV A,Q  
    RAL  
    MOV Q,A ;LSB(Q)=MSB(R)  
ENDM
```

Σημειώνεται ότι δεν μπορούμε να κάνουμε PUSH και POP τον καταχωρητή A και τα flags, γιατί όταν κάνουμε pop θα χαθεί το κρατούμενο CY, το οποίο χρειαζόμαστε.

4^η Άσκηση

Εφόσον η εντολή JMP 0900H έχει ξεκινήσει να εκτελείται ήδη πριν τη διακοπή, η διακοπή θα εκτελεστεί, αφού αναγνωριστεί και ο τελευταίος κύκλος της JMP. Αρχικά, ο PC θα πάρει την τιμή 0900H, που του ορίζει η εντολή JMP 0900H, ενώ αμέσως μετά θα αναγνωριστεί η διακοπή, θα απενεργοποιηθούν οι διακοπές, και θα αποθηκευτεί η τιμή του PC στην στοίβα SP. Αφού η τιμή του PC αποτελείται από 16 bits, οι αλλαγές που θα συμβούν στη στοίβα είναι οι εξής:

Στη θέση SP_1 θα αποθηκευτούν τα 8 MSB, δηλαδή η τιμή 09H, ενώ στην θέση SP_2 τα 8 LSB, δηλαδή η τιμή 00H. Επομένως, η τιμή του δείκτη στοίβας θα μειωθεί κατά 2, ώστε να δείχνει στην αμέσως επόμενη κενή θέση. Άρα η νέα τιμή SP θα είναι:

- **SP = 1FEEH**

Τέλος, ο PC θα πάρει την διεύθυνση της ρουτίνας RST6.5, και επομένως:

- **PC = 0034H**

5^η Άσκηση

Κύριο πρόγραμμα

```
MVI A,0EH
SIM
LXI H,0000H
MVI C,20H
EI
ADDR:
MOV A,C
CPI 00H
JNZ ADDR ;WAIT TO FINISH ADDING
DI
MOV A,H ;TO MULTIPLY WITH 16 WE HAVE TO SHIFT 4 TIMES
ANI C0H ;SO WE MOVE THE FIRST 2 BITS OF H TO THE LAST POSITIONS
RAR ;END THE NEXT 6 TO THE FIRST POSITIONS OF L
RAR
RAR
RAR
RAR
RAR
MOV H,A
MOV A,L
ANI 3FH
RAL
RAL
MOV L,A
HLT
```

Ποτίνα εξυπηρέτησης

```
002C:
JMP RST5.5
RST5.5:
PUSH PSW
IN 20H
MOV B,A
MOV A,C
ANI 01H
CPI 00H ;ON EVEN NUMBERS IGNORE
JC GO
MOV A,B
MVI D,00H
MOV E,A
DAD D ;ADD TO TOTAL
GO: DCR C ;DECREASE COUNTER
POP PSW
EI
RET
END
```