

Όνοματεπώνυμο: Κωνσταντίνος Σιδέρης  
ΑΜ: 03118134



**Εργαστήριο Μικροϋπολογιστών**  
**1η Εργασία**

## Άσκηση 1

Ακολουθεί ο κώδικας της άσκησης 1 με σχόλια:

```
LXI B,03E8H ;Αρχικοποίηση χρονοκαθυστέρησης σε 1000ms=1sec

START:  IN 20H      ;Διάβασμα εισόδου
        RAL        ;Ολίσθηση και έλεγχος του MSB
        JNC START  ;Επανάληψη ελέγχου όσο το MSB είναι ίσο με 0
        RAR
        ANI 0FH    ;Επαναφορά της εισόδου και απομόνωση των τεσσάρων LSB
        CPI 00H    ;Αν η είσοδος είναι 0 λαμβάνουμε νέα είσοδο μέχρι να μην είναι 0
        JZ START
        MOV D,A    ;Αποθήκευση του αριθμού στον D και αύξηση κατά 1
        INR D
        MVI A,00H  ;Μηδενισμός του A

CNTUP:  CMA
        OUT 30H    ;Συμπλήρωμα ως προς το 1 και απεικόνιση χρόνου σε δυαδική
μορφή

        CALL DELB
        CMA
        INR A      ;Αύξηση του A κατά 1
        CALL STATIC;Κλήση ρουτίνας ελέγχου λειτουργίας
        CMP D      ;Έλεγχος εάν ο A είναι ίσος με την είσοδο
        JC CNTUP   ;Αν όχι συνεχίζεται η μέτρηση προς τα πάνω αλλιώς ξεκινά η
αντίστροφη μέτρηση

        DCR A      ;Μείωση του A κατά δύο ώστε να ισούται με την είσοδο-1
        DCR A
CNTDN:  CMA
        OUT 30H    ;Συμπλήρωμα ως προς το 1 και απεικόνιση χρόνου σε δυαδική
μορφή

        CALL DELB
        CMA
        DCR A      ;Μείωση του A κατά 1
        CALL STATIC;Κλήση ρουτίνας ελέγχου λειτουργίας
        CPI 00H    ;Έλεγχος εάν ο A είναι ίσος με 0
        JNZ CNTDN  ;Αν όχι συνεχίζεται η μέτρηση προς τα κάτω αλλιώς επανεκκίνηση
του προγράμματος

        JMP START

STATIC: PUSH PSW    ;Ρουτίνα ελέγχου λειτουργίας προγράμματος (έλεγχος MSB)
STOP:   IN 20H      ;Διάβασμα εισόδου
        RAL        ;Ολίσθηση και έλεγχος του MSB
        JNC STOP   ;Επανάληψη ελέγχου όσο το MSB είναι ίσο με 0
        POP PSW
        RET

END
```

## Άσκηση 2

Ακολουθεί ο κώδικας της άσκησης 2 με σχόλια:

	IN 10H	;Απενεργοποίηση προστασίας μνήμης
	MVI A,10H	
	STA 0903H	;Βάζουμε κενά στα ψηφία του 7-segment display
	STA 0904H	;που δεν χρησιμοποιούμε
	STA 0905H	
INPUT1:	CALL KIND	;Λαμβάνουμε είσοδο x από το πληκτρολόγιο
	CPI 00H	;Ελέγχουμε ότι η είσοδος είναι δεκαεξαδικό ψηφίο
	JC INPUT1	
	CPI 10H	
	JNC INPUT1	
	RLC	;Κάνουμε 4 περιστροφές ώστε η είσοδος να πολλαπλασιαστεί με
16		
	RLC	
	RLC	
	RLC	
	MOV B,A	;Αποθηκεύουμε το 16x στον B
INPUT2:	CALL KIND	;Λαμβάνουμε είσοδο y από το πληκτρολόγιο
	CPI 00H	;Ελέγχουμε ότι η είσοδος είναι δεκαεξαδικό ψηφίο
	JC INPUT2	
	CPI 10H	
	JNC INPUT2	
	ADD B	;Προσθέτουμε το 16x στο y
	MOV B,A	;Αποθηκεύουμε τον τελικό αριθμό στον A
	MVI D,10H	
	CPI 64H	;Ελέγχουμε αν ο αριθμός είναι μεγαλύτερος του 99D
	JC TENS	
	MVI D,00H	
HUNDREDS:	SUI 64H	;Αν είναι αφαιρούμε 100 από αυτόν μέχρι να μην είναι
	INR D	;Μετράμε πόσες εκατοντάδες περιέχει ο αριθμός
	CPI 64H	
	JC TENS	;Αν είναι μικρότερος του 100 ελέγχουμε τις δεκάδες
	JMP HUNDREDS	
TENS:	MOV B,A	
	MOV A,D	
	STA 0902H	;Αποθηκεύουμε τις εκατοντάδες στην κατάλληλη θέση μνήμης
	MOV A,B	
	MVI D,00H	
	CPI 0AH	;Ελέγχουμε αν ο αριθμός είναι μεγαλύτερος του 9D
	JC ONES	
CONT:	SUI 0AH	;Αν είναι αφαιρούμε 10 από αυτόν μέχρι να μην είναι
	INR D	;Μετράμε πόσες δεκάδες περιέχει ο αριθμός
	CPI 0AH	
	JC ONES	;Αν είναι μικρότερος του 100 ελέγχουμε τις μονάδες
	JMP CONT	

ONES:	MOV B,A	
	MOV A,D	
	STA 0901H	;Αποθηκεύουμε τις δεκάδες στην κατάλληλη θέση μνήμης
	MOV A,B	;Ο αριθμός που παραμένει στον A είναι οι μονάδες
	STA 0900H	;Αποθηκεύουμε τις μονάδες στην κατάλληλη θέση μνήμης
	LDA 0902H	
	CPI 10H	
	JNZ PRINT	
	LDA 0901H	;Ελέγχουμε τις εκατοντάδες και τις δεκάδες και βάζουμε κενά όπου
χρειάζεται	CPI 00H	
	JNZ PRINT	
	MVI A,10H	
	STA 0901H	
PRINT:	CALL SCREEN	;Καλούμε την ρουτίνα απεικόνισης στο 7-segment display
	JMP INPUT1	;Επανακινούμε το πρόγραμμα
SCREEN:	PUSH D	;Ρουτίνα απεικόνισης στο 7-segment display
	LXI D,0900H	;Αρχή διευθύνσεων αποθήκευσης για το 7-segment display
	CALL STDM	
	CALL DCD	
	POP D	
	RET	
	END	

### Άσκηση 3

Ακολουθεί ο κώδικας της άσκησης 3 με σχόλια:

```
LXI B,01F4H ;Αρχικοποίηση χρονοκαθυστέρησης σε 500ms=0,5sec
IN 20H
ANI 01H
MOV E,A ;Αρχικοποίηση E στον οποίο αποθηκεύουμε την τιμή του LSB
MVI A,FEH ;Αρχικοποίηση της θέσης του βαγονέτου
OUT 30H
CALL DELB

LEFT: RLC ;Αριστερή περιστροφή
OUT 30H
CALL DELB
CALL CHECK ;Κλήση ρουτίνας ελέγχου αλλαγής κατεύθυνσης
MOV D,A
MOV A,H
CPI 00H ;Έλεγχος σημαίας αλλαγής κατεύθυνσης η οποία ανανεώνεται
από την ρουτίνα ελέγχου
MOV A,D
JZ LEFT ;Αν η σημαία είναι 0 δεν αλλάζει η κατεύθυνση
JMP RIGHT ;Αν είναι 1 αλλάζουμε σε δεξιά περιστροφή

RIGHT: RRC ;Δεξιά περιστροφή
OUT 30H
CALL DELB
CALL CHECK ;Κλήση ρουτίνας ελέγχου αλλαγής κατεύθυνσης
MOV D,A
MOV A,H
CPI 00H ;Έλεγχος σημαίας αλλαγής κατεύθυνσης η οποία ανανεώνεται
από την ρουτίνα ελέγχου
MOV A,D
JZ RIGHT ;Αν η σημαία είναι 0 δεν αλλάζει η κατεύθυνση
JMP LEFT ;Αν είναι 1 αλλάζουμε σε αριστερή περιστροφή

CHECK: PUSH PSW ;Ρουτίνα ελέγχου αλλαγής κατεύθυνσης
MOV L,A
STATIC: IN 20H ;Έλεγχος λειτουργίας προγράμματος (έλεγχος MSB)
MOV D,A
RAL
JNC STATIC
MOV A,L
MVI H,00H
CPI 7FH ;Έλεγχος αν βρισκόμαστε στο αριστερό άκρο
JNZ C2
MVI H,01H ;Αν βρισκόμαστε τότε το βαγονέτο πρέπει να αλλάξει κατεύθυνση
ανεξάρτητα του LSB οπότε η σημαία αλλαγής γίνεται 1
CALL DELB ;Έξτρα αναμονή 0,5sec για συνολική αναμονή 1sec στα άκρα
JMP EXIT ;Έξοδος από την ρουτίνα ελέγχου

C2: CPI FEH ;Έλεγχος αν βρισκόμαστε στο δεξί άκρο
JNZ CONT
```

	MVI H,01H	;Αν βρισκόμαστε τότε το βαγονέτο πρέπει να αλλάξει κατεύθυνση ανεξάρτητα του LSB οπότε η σημαία αλλαγής γίνεται 1
	CALL DELB	;Έξτρα αναμονή 0,5sec για συνολική αναμονή 1sec στα άκρα
	JMP EXIT	;Έξοδος από την ρουτίνα ελέγχου
CONT:	MOV A,D	;Αν δεν βρισκόμαστε στα άκρα ελέγχουμε το LSB
	ANI 01H	
	MOV L,A	;Κάνουμε XOR του LSB με την τελευταία τιμή του
	XRA E	;Αν είναι διαφορετικό δηλαδή ζητήθηκε αλλαγή κατεύθυνσης η σημαία γίνεται 1( $1 \text{ XOR } 0 = 1$ )
	MOV E,L	;Αν δεν είναι διαφορετικό δηλαδή δεν ζητήθηκε αλλαγή κατεύθυνσης η σημαία γίνεται 0( $0 \text{ XOR } 0 = 1 \text{ XOR } 1 = 0$ )
EXIT:	MOV H,A	
	POP PSW	
	RET	
	END	