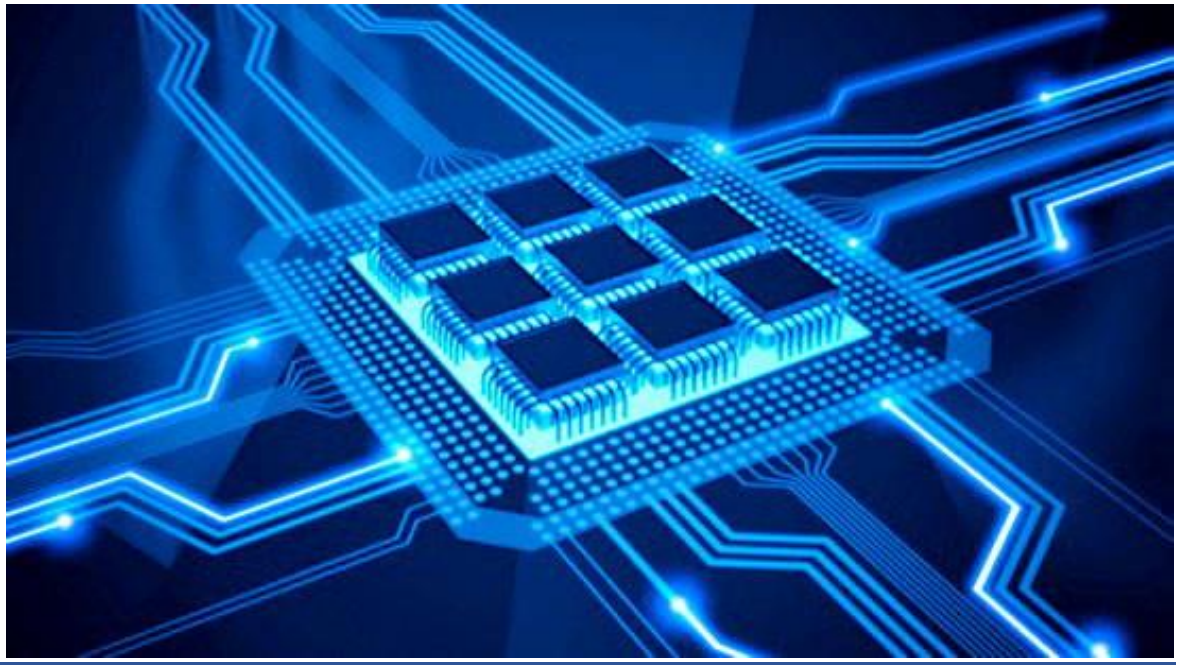
2η ΕΡΓΑΣΤΗΡΙΑΚΗ ΆΣΚΗΣΗ
ΓΙΑ ΤΟ ΜΑΘΗΜΑ "Εργαστήριο Μικροϋπολογιστών"



Ομάδα: 16
Ημερομηνία επίδειξης: 25/10/2017
Μέλη:
Κερασιώτης Ιωάννης, Α.Μ.:03114951
Πευκιανάκης Κωνσταντίνος, Α.Μ.: 03114897
Ραφτόπουλος Ευάγγελος, Α.Μ.:03114743

ΖΗΤΗΜΑ 2:

Η άσκηση αυτή ζητάει υλοποίηση σε assembly 8085 μια αριθμομηχανής αριθμών για πρόσθεση και αφαίρεση. Δέχεται δύο διψήφιους δεκαδικούς αριθμούς και το είδος της πράξης (πρόσθεση ή αφαίρεση). Πατώντας το FETCH PC πραγματοποιείται η πράξη της πρόσθεσης ενώ πατώντας το πλήκτρο DECR πραγματοποιείται αφαίρεση. Θα πρέπει να εμφανίσει το αποτέλεσμα σε δεκαδική μορφή.

Αρχικά αρχικοποιούμε τις θέσεις που χρησιμοποιεί η DCD με 00H γιατί θα εμφανίζει η 7-segment display όλες τις ενδείξεις. Έπειτα γίνεται τοι διάβασμα του πρώτου διψήφιου αριθμού. Διαβάζει δύο ψηφία ελέγχοντας αν είναι δεκαδικοί αριθμοί (0-9) και τα αποθηκεύει στις κατάλληλες θέσεις για να τα εμφανίσει η ρουτίνα DCD. Αυτές οι θέσεις είναι οι 0BF1H για τις δεκάδες και η 0BF0H για τις μονάδες. Έχουμε κάνει την παραδοχή ότι αν δοθεί μη αποδεκτός χαρακτήρας να ξαναδιαβάζει και τα δύο ψηφία του αριθμού. Έπειτα σχηματίζουμε τον 1ο διψήφιο αριθμό με την πράξη 10*δεκάδες+μονάδες. Ακριβώς την ίδια διαδικασία κάνουμε και για τον δεύτερο διψήφιο αριθμό.

Στην συνέχεια ανάλογα με το πλήκτρο που πατήθηκε μετά την συμπλήρωση του 1ου διψήφιου αριθμού, γίνεται η κατάλληλη πράξη. Αν έχει πατηθεί το πλήκτρο για πρόσθεσης θα γίνει η πρόσθεση και θα γίνει έλεγχος για το αν έχουμε overfloat. Αν έχουμε τότε εμφανίζει --, αλλιώς εμφανίζει το αποτέλεσμα της πρόσθεσης το οποίο έχει αποθηκευτεί στις θέσεις οι οποίες εμφανίζουν στις μεσαίες θέσεις της 7-segment display.

Αν πατηθεί το πλήκτρο για αφαίρεση τότε θα βρούμε τον μεγαλύτερο από τους δυο αριθμούς ώστε να βρεθεί η απόλυτη τιμή και να εμφανίσει το αποτέλεσμα. Η εμφάνιση του αποτελέσματος θα γίνει μετά την εύρεση των μονάδων και των δεκάδων.

Ο κώδικας της άσκησης παρατίθεται παρακάτω :

```
NOP     ;αρχικοποίηση των θέσεων 0800H και 0800H για την
NOP     ;αποθήκευση των δύο διψήφιων αριθμών
IN 10H

MVI A,00H
LXI H,0BF0H     ;Αρχικοποίηση του τμήματος μνήμης που
MVI M,00H       ;χρησιμοποιείται από τη ρουτίνα DCD
INX H
MVI M,00H       ;θέλουμε να εμφανίζονται όλες οι ενδείξεις της
INX H           ;7 segment-dislpay άρα βάζουμε την τιμή 00H
MVI M,00H
INX H
MVI M,00H
INX H
MVI M,00H
INX H
MVI M,00H

START:

                                ;ΓΙΑ ΤΟ ΔΙΑΒΑΣΜΑ ΤΟΥ 1ΟΥ ΔΙΨΗΦΙΟΥ ΑΡΙΘΜΟΥ
READ_FIRST_NUMBER:      ;διάβασμα 1ου ψηφίου (δεκάδες)
CALL KIND               ;Διαβάζουμε από το πληκτρολόγιο
CPI 0AH                 ;Κοιτάμε αν είναι δεκαδικό το ψηφίο (0-9)
JNC READ_FIRST_NUMBER   ;Περιμένουμε μέχρι να έρθει ένα έγκυρο ψηφίο

STA 0BF1H               ;αποθηκεύουμε τις μονάδες του αριθμού στην θέση OBF1H ώστε να
CALL DCD                ;την εμφανίσει η DCD στο δεξί μέρος της οθόνης (προτελευταία ένδειξη)

READ_SECOND_NUMBER:     ;διάβασμα του 2ου ψηφίου (μονάδες)
CALL KIND               ;Διαβάζουμε από το πληκτρολόγιο
CPI 0AH                 ;Κοιτάμε αν είναι δεκαδικό το ψηφίο (0-9)
JNC READ_FIRST_NUMBER   ;Περιμένουμε μέχρι να έρθει ένα έγκυρο ψηφίο

STA 0BF0H               ;αποθηκεύουμε τις μονάδες του αριθμού στην θέση OBF0H ώστε να
CALL DCD                ;την εμφανίσει η DCD στο δεξί μέρος της οθόνης (τελευταία ένδειξη)

LDA 0BF1H               ;σχηματισμός του 1ου διψήφιου αριθμού.
ADD A                   ;A+A=2A
MOV B,A                 ;B<-2A
ADD A                   ;2A+2A=4A
ADD A                   ;4A+4A=8A
ADD B                   ;8A+2A=10A
MOV B,A

LDA 0BF0H
ADD B                   ;Γίνεται η πράξη 10*δεκάδες + μονάδες
STA 0800H               ;στην θέση 0800H αποθηκεύεται 1ος 2-ψηφιος αριθμός !!

READ_PRAKSH:            ;ΔΙΑΒΑΣΜΑ ΤΗΣ ΠΡΑΞΗΣ
CALL KIND               ;Διαβάζουμε από το πληκτρολόγιο
CPI 81H                 ;ελέγχουμε αν πατήθηκε το πλήκτρο FETCH PC ή το DECR
JZ OK                   ;αν δεν πατήθηκε κανένα από τα προηγούμενα τότε ξαναδιάβασε
CPI 85H
JNZ READ_PRAKSH
```

```
OK:
MOV E,A                 ;στον Ε έχουμε την τιμή του κουμπιού που πατήθηκε που καθορίζει την πράξη
                        ;που θα εκτελεστεί
                        ;ΓΙΑ ΤΟ ΔΙΑΒΑΣΜΑ ΤΟΥ 2ΟΥ ΔΙΨΗΦΙΟΥ ΑΡΙΘΜΟΥ
READ_FIRST_NUMBER1:     ;διάβασμα 1ου ψηφίου (δεκάδες)
CALL KIND               ;Διαβάζουμε από το πληκτρολόγιο
CPI 0AH                 ;Κοιτάμε αν είναι δεκαδικό το ψηφίο (0-9)
JNC READ_FIRST_NUMBER1  ;Περιμένουμε μέχρι να έρθει ένα έγκυρο ψηφίο


STA 0BF5H               ;αποθηκεύουμε τις μονάδες του αριθμού στην θέση OBF5H ώστε να
CALL DCD                ;την εμφανίσει η DCD στο αριστερό μέρος της οθόνης (πρώτη από την αρχή
ένδειξη)

READ_SECOND_NUMBER1:    ;διάβασμα 2ου ψηφίου (μονάδες)
CALL KIND               ;Διαβάζουμε από το πληκτρολόγιο
CPI 0AH                 ;Κοιτάμε αν είναι δεκαδικό το ψηφίο (0-9)
JNC READ_FIRST_NUMBER1  ;Περιμένουμε μέχρι να έρθει ένα έγκυρο ψηφίο

STA 0BF4H               ;αποθηκεύουμε τις μονάδες του αριθμού στην θέση OBF5H ώστε να
CALL DCD                ;την εμφανίσει η DCD στο αριστερό μέρος της οθόνης (δεύτερη από την αρχή
                        ; ένδειξη)

LDA 0BF5H               ;σχηματισμός του 1ου διψήφιου αριθμού. όμοια με πριν
ADD A
MOV B,A
ADD A
ADD A
ADD B
MOV B,A

LDA 0BF4H
ADD B                   ;Γίνεται η πράξη 10*δεκάδες + μονάδες
STA 0801H               ;στην θέση μνήμης 0801H Ο 2ος 2-ψηφιος αριθμός !!

MOV A,E                 ;στον Ε έχουμε την πράξη που θα κάνει
CPI 85H                 ;αν έχει την τιμή 85H θα κάνει αφαίρεση αλλιώς πρόσθεση
JNZ NEXT

LDA 0800H       ;εκτελείται η πράξη: 1ος αριθμός + 2ος αριθμός
MOV B,A
LDA 0801H
ADD B
CPI 64H         ;αν το αποτέλεσμα είναι μεγαλύτερο από 99 τότε έχουμε overfloat
JNC OVERFLOAT
CALL DCD
JMP EYRESH_MONADON_DEKADON

NEXT:
LDA 0800H
MOV B,A
LDA 0801H
CMP B           ;αν ο αφαιρετέος είναι μικρότερος από τον αφαιρέτη βρες την απόλυτη τιμή
JNC AFAIR       ;της διαφοράς
STA 0800H       ;κώδικας που αντιμεταθέτει τους αριθμούς ώστε η διαφορά να είναι
MOV A,B         ;μεγαλύτερη του 0
```

```
STA 0801H

AFAIR:          ;εκτελείται η πράξη της αφαίρεσης
LDA 0800H
MOV B,A
LDA 0801H
SUB B

EYRESH_MONADON_DEKADON:    ;ανάλυση του αριθμού σε μονάδες και δεκάδες
MVI B,FFH
DECA:       ;για δεκάδες
INR B       ;αφαιρούμε από το B το 0AH=10D μέχρι να γίνει αρνητικός και μετράμε το πλήθος των
            ;αφαιρέσεων
SUI 0AH
JNC DECA   ;αν εξακολουθεί να είναι θετικός πήγαινε στην DECA
ADI 0AH     ;προσθέτουμε το 0AH για να διορθώσουμε το αρνητικό υπόλοιπο
STA 0BF2H  ;αποθήκευσε τις μονάδες στην κατάλληλη θέση για την DCD
MOV A,B
STA 0BF3H  ;αποθήκευσε τις δεκάδες στην κατάλληλη θέση για την DCD
CALL DCD
JMP START

OVERFLOAT:
MVI A,1CH   ;η περίπτωση της υπερχείλισης. σε αυτή την περίπτωση η άσκηση ζητάει να
STA 0BF3H   ;εμφανίσει την τιμή -- . ο χαρακτήρας - αναπαρίσταται από την τιμή 1CH
STA 0BF2H
CALL DCD

JMP START  ;το πρόγραμμα είναι συνεχής λειτουργίας
END
```