

Εργαστήριο Μικροϋπολογιστών

3η εργαστηριακή άσκηση

Τμήμα: Β Ομάδα: 15

Συνεργάτες: Μαρουφίδης Ιωάννης (03113506),

Περράκης Γεώργιος (03113511) ,

Σοφινίδης Γεώργιος (03113179)

1^η Άσκηση

START:

```
MVI A,10H           ;code 10H is blank
STA 0B00H           ;and is stored to the 7
STA 0B01H           ;segment displays
STA 0B02H           ;as a starting condition
STA 0B03H
STA 0B04H
STA 0B05H
LXI D,0B00H         ;register D has the starting
CALL STDM           ;address for routine STDM
CALL DCD
CALL KIND           ;while routine KIND is waiting
                   ;for a value, routine DCD
                   ;is executed by default.
                   ;Code of the button to
                   ;register A.

CALL FIND_CODE      ;routine that returns code of
MOV A,L             ;pressed button to registers HL
STA 0B00H           ;first number to the rightest
LXI D,0B00H         ;seven segment digit
CALL STDM
CALL DCD
CALL KIND
CALL FIND_CODE
MOV A,L
STA 0B01H           ;and second number next to the
LXI D,0B00H         ;first
CALL STDM
CALL DCD
LDA 0B00H           ;first number is stored
MOV B,A             ;to register B
LDA 0B01H           ;and second to register C
```

```

        MOV C,A
LOOP1:
        CALL KIND
        CALL FIND_CODE
        MOV A,L                ;if F button is pressed we
        CPI 0FH                ;start all again
        JZ START
        MOV A,H
        CPI 08H                ;if left digit of the code is
        JNZ LOOP1              ;not 8 then a number is pressed
        MOV A,L                ;so we ignore it
        CPI 03H                ;code digit = 83
        JZ ADDITION
        CPI 01H                ;code digit = 81
        JZ SUBTRACTION
        CPI 05H                ;code digit = 85
        JZ MULTIPLICATION
        JMP LOOP1              ;if any of the upper buttons
                                ;is not pressed we continue
                                ;till it is

FIND_CODE:                      ;routine that returns the code
        MOV H,A                ;of pressed button
        ANI 0FH
        MOV L,A                ;right digit of the code
        MOV A,H                ;to register L
        ANI F0H                ;and left to register H
        RAR
        RAR
        RAR
        RAR
        MOV H,A
        RET

ADDITION:                      ;addition routine
        LDA 0B00H
        MOV B,A
        LDA 0B01H
        MOV C,A
        MOV A,C
        ADD B
        MOV C,A
        ANI 0FH                ;we separate the two digits
        MOV L,A                ;of the result
        MOV A,C
        ANI F0H
        RAR
        RAR
        RAR
        RAR
        MOV H,A                ;and we store them to the

```

```

        STA 0B03H           ;2nd and 3rd 7 segment digit
        MOV A,L
        STA 0B02H
        LXI D,0B00H
        CALL STDM
        CALL DCD
        JMP LOOP1

SUBTRACTION:                ;subtraction routine
        LDA 0B00H
        MOV B,A
        LDA 0B01H
        MOV C,A
        MOV A,B
        CMP C
        JC NEGATIVE        ;we check if the result is
                             ;negative or positive
                             ;(or zero)
        MOV A,B             ;if its positive we just
                             ;execute 1st minus 2nd number
        SUB C
        STA 0B02H
        MVI A,00H
        STA 0B03H
        LXI D,0B00H
        CALL STDM
        CALL DCD
        JMP LOOP1

NEGATIVE:                   ;if the result is negative
                             ;we execute 2nd minus 1st
                             ;number and we put a minus(-)
                             ;just before the result
        MOV A,C
        SUB B
        STA 0B02H
        MVI A,1CH
        STA 0B03H
        LXI D,0B00H
        CALL STDM
        CALL DCD
        JMP LOOP1

MULTIPLICATION:            ;multiplication routine
        LDA 0B00H
        MOV B,A
        LDA 0B01H
        MOV C,A
        MVI A,00H
        MVI D,00H

LOOP2:                      ;we add the 2nd number A times
                             ;where A is the 1st number
        ADD B
        INR D
        MOV H,A
        MOV A,D
        CMP C
        MOV A,H
        JNZ LOOP2

```

```

MOV C,A
ANI 0FH
MOV L,A
MOV A,C
ANI F0H
RAR
RAR
RAR
RAR
MOV H,A
STA 0B03H
MOV A,L
STA 0B02H
LXI D,0B00H
CALL STDM
CALL DCD
JMP LOOP1

```

END

Στην παραπάνω άσκηση, υλοποιούμε μία αριθμομηχανή τριών πράξεων: πρόσθεσης, αφαίρεσης και πολλαπλασιασμού μονοψήφιων δεκαεξαδικών αριθμών (0 – A). Δίνονται δύο αριθμοί που αναπαρίστανται στα 2 δεξιότερα 7 segment displays και το αποτέλεσμα αναπαρίσταται στα επόμενα 2 displays αντίστοιχα. Το πρόγραμμα μας αγνοεί τυχόν πλήκτρα που πατήσει ο χρήστης μετά την εισαγωγή όπως επιπλέον αριθμούς ή πλήκτρα λειτουργίας που δεν συμβαδίζουν με την εκφώνηση. Αν το αποτέλεσμα προκύψει αρνητικό προσθέτουμε ένα (-) αριστερά από αυτό. Γενικότερα τα αποτελέσματα των πράξεων μας ‘χωράνε’ σε δύο 7 segment displays χωρίς τον κίνδυνο υπερχείλισης.