

3^η Ομάδα Ασκήσεων στα Συστήματα Μικροϋπολογιστών

6^ο Εξάμηνο, Ακαδημαϊκή Περίοδος 2020 – 2021

Ονοματεπώνυμο	Αριθμός Μητρώου
Μπούφαλης Οδυσσεύς – Δημήτριος	el18118
Στεφανάκης Γεώργιος	el18436

Άσκηση 1^η, 2^η

Μαζί με την παρούσα αναφορά επισυνάπτονται και τα αρχεία πηγαίου κώδικα που περιέχουν τις απαντήσεις στις ασκήσεις υπ' αριθμόν 1 και 2. Τα αρχεία αυτά μπορούν να εκτελεστούν και να επαληθευτεί η ορθότητά τους στον προσομοιωτή TSIK MicroLab 8085 Simulator.

- ask1.8085
- ask2.8085

Άσκηση 3^η

α)

```
SWAP Nibble MACRO Q
    PUSH PSW
    MOV A, Q
    RLC
    RLC
    RLC
    RLC
    MOV Q, A
    MOV A, M
    RRC
    RRC
    RRC
    RRC
    MOV M, A
    POP PSW
ENDM
```

β)

```
FILL MACRO RP, X, K
    PUSH PSW
    PUSH H
    MOV A, X
    LXI H, RP
START:
    MVI M, K
    INX H
    DCR A
    JNZ START
    POP H
    POP PSW
ENDM
```

γ)

```

RHLL MACRO n
    PUSH PSW
    PUSH B

    MVI A,n
    CPI 00H
    JZ FINISH
    MVI B,n
START:
    MOV A,L
    RAL
    MOV L,A
    MOV A,H
    RAL
    MOV H,A
    DCR B
    JNZ START
FINISH:
    POP B
    POP PSW
ENDM

```

Άσκηση 4^η

Η διακοπή συμβαίνει στο μέσο της εντολής CALL 0880H, άρα θα ολοκληρωθεί η εκτέλεση της τρέχουσας εντολής: η τρέχουσα τιμή του μετρητή προγράμματος (0800H) αποθηκεύεται στην στοίβα, ο δείκτης στοίβας ανεβαίνει 2 θέσεις πάνω και στον μετρητή προγράμματος καταχωρείται η διεύθυνση 0880H. Έπειτα σώζεται η τιμή του μετρητή προγράμματος και η κατάσταση του επεξεργαστή και εκτελείται η ρουτίνα εξυπηρέτησης της διακοπής RST 7.5. Αυτό σημαίνει ότι η τιμή του μετρητή προγράμματος αποθηκεύεται ξανά στην στοίβα, ο δείκτης στοίβας ανεβαίνει άλλες 2 θέσεις πάνω και στον μετρητή προγράμματος καταχωρείται η διεύθυνση της διακοπής για να εκτελεστεί η σχετική ρουτίνα. Όταν ολοκληρωθεί η εκτέλεση της ρουτίνας εξυπηρέτησης της διακοπής, η διεύθυνση που βρίσκεται στην κορυφή της στοίβας (0880H) επανέρχεται στον μετρητή προγράμματος, ο δείκτης στοίβας κατεβαίνει 2 θέσεις κάτω και εκτελείται η ρουτίνα που αρχίζει από τη διεύθυνση 0880H, σύμφωνα με την εντολή CALL 0880H. Όταν ολοκληρωθεί η εκτέλεση και της τελευταίας ρουτίνας, η διεύθυνση στην κορυφή της στοίβας (0800H) επαναφέρεται στον μετρητή προγράμματος, ο δείκτης στοίβας κατεβαίνει άλλες 2 θέσεις κάτω και συνεχίζεται η εκτέλεση του προγράμματος από τη διεύθυνση 0801H.

Η όλη διαδικασία φαίνεται σχηματικά στον παρακάτω πίνακα, όπου δίνονται τα περιεχόμενα του μετρητή προγράμματος και της στοίβας αρχικά (1), μετά την εκτέλεση της εντολής CALL 0880H (2), μετά την πραγματοποίηση της διακοπής RST 7.5 (3), μετά την εκτέλεση της ρουτίνας εξυπηρέτησης της διακοπής (4) και μετά την εκτέλεση της ρουτίνας που καλεί η εντολή CALL 0880H (5).

1		2		3		4		5	
PC	0800H	PC	0880H	PC	(RST 7.5)	PC	0880H	PC	0800H
SP	00H	SP	00H	SP	80H	SP	00H	SP	00H
SP+1	30H	SP+1	08H	SP+1	08H	SP+1	08H	SP+1	30H
		SP+2	00H	SP+2	00H	SP+2	00H		
		SP+3	30H	SP+3	08H	SP+3	30H		
				SP+4	00H				
				SP+5	30H				

Άσκηση 5^η

EXERCISE_5I:

```
IN 10H
MVI A,0DH           ; init interrupt mask
SIM
MVI B,00H
LXI H,0000H         ; init counter for mean value calculation
MVI C,01H           ; init data counter
EI

LOOP1:
MOV A,C             ; check if all data has been fetched
CPI 41H
JNZ LOOP1
DI
DAD H               ; calculate mean value
DAD H
DAD H
MOV A,L
ANI 80H
CPI 80H
JNZ FIN
INR H

FIN:  HLT
```

002C: JMP RST5.5

RST5.5:

```
PUSH PSW           ; store A and flags
MOV A,B
CPI 00H            ; check if LSB or MSB
JZ  LSB
IN 20H
RLC
RLC
RLC
RLC                ; place MSBs in correct positions
ADD E              ; A now contains BCD number
MOV E,A
MVI D,00H
DAD D              ; sum of data on HL
MVI B,00H          ; store 0 because after comes LSB
INR C              ; increase data counter
POP PSW
EI
RET

LSB:
IN 20H
MOV E,A            ; E <- LSB
INR B              ; increase B because after comes MSB
INR C              ; increase data counter
POP PSW
EI
RET

END
```

EXERCISE_5II:

```

MAIN:      LXI H,0000H      ; init buffer at zero, stores mean value
           MVI C,40H        ; C <- data counter
           MVI B,00H        ; store on B previous value of x7 from PORT IN
                           ; wste na boresoume na edopisoume to thetiko metopo

CHECK:     MOV A,C          ; loop until all data read
           CPI 00H
           JZ  CALC         ; calculate mean
           IN  20H          ; read from input
           ANI 80H          ; isolate MSB = x7
           CMP B
           MOV B,A          ; update B
           JZ  CHECK        ; An einai idia den exoume kapoio metwpo
           JNC INPUT        ; An A>B exoume thetiko metwpo
           JMP CHECK        ; Periptosi A<B.

CALC:      DAD H
           DAD H            ; to calculate mean divide by 32 (right shift 5 times)
           DAD H            ; instead we can left shift 3 times
           MOV A,L          ; then result will be available on reg H instead of L
           ANI 80H          ; L will hold decimal part
           CPI 80H          ; so in order to round number we have to check L's MSB
           JNZ FIN          ; if it is equal to 1 increase H else leave it as it is
           INR H

FIN:       HLT

INPUT:     IN  20H          ; read input
           ANI 0FH          ; isolate 4 lsb's
           MVI D,00H
           MOV E,A

           MOV A,C
           RRC              ; right shift once to divide A by 2
           JNC CONT        ; if divisible then LSB was read else MSB was read
           MOV A,E          ; if it's the MSB shift left 4 times
           RLC
           RLC
           RLC
           RLC
           MOV E,A

CONT:      DAD D            ; add D to buffer
           DCR C            ; decrease counter
           JMP CHECK

           END

```