

Ε23 - Θέμα 3

Συστήματα Μικροπολογιστών - Ελένη Νασοπούλου

(β) Στο μΕ 8085 όταν εκτελείται η εντολή **JMP 0820H**, ο μετρητής προγράμματος είναι (PC)=0780H και ο δείκτης στοίβας (SP)=900H. Στο μέσον της εκτέλεσης της εντολής συμβαίνει διακοπή RST 6.5. Δώστε τις νέες τιμές των PC, SP, το περιεχόμενο της στοίβας καθώς και τις λειτουργίες που συμβαίνουν στην αρχή και στην επιστροφή από την ρουτίνα εξυπηρέτησης. (0.5 ΜΟΝΑΔΕΣ)

RESET	0000
...	...
RST1	0008
...	...
RST2	0010
...	...
RST3	0018
RST4	0020
...	...
TRAP	0024
...	...
RST5	0028
...	...
RST5.5	002C
...	...
RST6	0030
...	...
RST6.5	0034
...	...
RST7	0038
...	...
RST7.5	003C
...	...

Οι λειτουργίες που επικελούνται ως ανάρτηση στη software διακοπή RST 6.5 είναι οι εξής:

1. Ολοκλήρωση τρέχουσας εντολής:

πριν: $\left\{ \begin{array}{l} (PC) = 0780H \\ (SP) = 0900H \end{array} \right\}$ μετά: $\left\{ \begin{array}{l} (PC) = 0820H \\ (SP) = 0900H \end{array} \right\}$

2. Σωζεται ο PC και η κατάσταση του μΕ και το PC μεταβαίνει στη ρουτίνα εξυπηρέτησης

$\left\{ \begin{array}{l} (PC) = 0034H \\ (SP) = 08FEH \end{array} \right\}$ $(08FEH) = 20H$
 $(08FFH) = 08H$

3. Μετά την ολοκλήρωση της ρουτίνας διακοπής ανακατάει η κατάσταση του μΕ και εκτελεί την επόμενη εντολή

$\left\{ \begin{array}{l} (PC) = 0820H \\ (SP) = 0900H \end{array} \right\}$

Ε23 - Θέμα 4

ΘΕΜΑ 4* : (1.5 ΜΟΝΑΔΕΣ): Απαντήστε στα παρακάτω ερωτήματα (σύντομα και αιτιολογημένα):

(α) Ποιος είναι ο ρόλος της στοίβας σε έναν μικροεπεξεργαστή; Αναφέρετε δυο παραδείγματα χρήσης της. (0.3 ΜΟΝΑΔΕΣ)

(β) Έστω ο παρακάτω κώδικας προγράμματος σε assembly μΥ-Σ 8085. Να εξηγήσετε και να υπολογίσετε το άνω όριο του αριθμού ενεργοποιήσεων της διακοπής RST 5.5, ώστε να έχουμε ασφαλή λειτουργία του προγράμματος. Να προτείνετε βελτιώσεις για την ασφαλή λειτουργία του κώδικα assembly μΥ-Σ 8085, ανεξαρτήτως του αριθμού ενεργοποιήσεων των διακοπών RST 5.5. (0.6 ΜΟΝΑΔΕΣ)

→ Interrupt
slide 18

```
RST5.5: JMP LOOP1
...
START: LXI SP, LOOP2+9
MVI B, FFH
LOOP1: INR B
MVI A, 0EH
SIM
EI
LOOP2: JMP LOOP2
```

```
RST5.5: JMP LOOP1
...
START: MVI B, 0H
MVI A, 0EH
SIM
EI
LOOP2: JMP LOOP2
...
LOOP1: INR B
EI
RET
```

	JMP	LOOP2	0EH → 0000 1110 → b+3=1
εναν 4 {	(LO BYTE)	LOOP2 + 1	αφαιρέση διακοπής RST 5.5
	(HI BYTE)	LOOP2 + 2	
εναν 3 {	(LO BYTE)	LOOP2 + 3	στη 5 ^η επανάληψη ο σωρός
	(HI BYTE)	LOOP2 + 4	θα επικαλύψει τον κώδικα του
εναν 2 {	(LO BYTE)	LOOP2 + 5	προγράμματος
	(HI BYTE)	LOOP2 + 6	
εναν 1 {	PCL	LOOP2 + 7	
	PCH	LOOP2 + 8	
SP →		LOOP2 + 9	

Ορθή μαρρη:

```
RST5.5: JMP LOOP1
...
START: LXI SP, LOOP2+9
MVI B, FFH
LOOP1: INR B
MVI A, 0EH
SIM
EI
LOOP2: JMP LOOP2
```

Ε23 - Θέμα 4

(γ) Έστω ένα $\mu\text{Υ-}\Sigma$ με τα εξής χαρακτηριστικά: 1) Μονάδα μνήμης ROM 2K (Διευθύνσεις 0000H – 07FFH), 2) Μονάδα μνήμης RAM 4K (Διευθύνσεις 0800H – 17FFH) και 3) Μονάδα μνήμης RAM 2K (Διευθύνσεις 1800H – 1FFFH). Με βάση τον χάρτη διευθύνσεων, να σχεδιάσετε το λογικό κύκλωμα που επιλέγει την κάθε μονάδα μνήμης (0.6 ΜΟΝΑΔΕΣ). Η υλοποίηση να γίνει αποκλειστικά με τη χρήση λογικών πυλών.

A_{15-12}	A_{11-8}	A_{7-4}	A_{3-0}	addr	mem
0000	0000	0000	0000	0000H	ROM
0000	0111	1111	1111	07FFH	
0000	1000	0000	0000	0800H	RAM1
0000	1111	1111	1111	0FFFH	
0001	0000	0000	0000	1000H	RAM2
0001	0111	1111	1111	17FFH	
0001	1000	0000	0000	1800H	
0001	1111	1111	1111	1FFFH	

A_{15}, A_{14}, A_{13} : επιρροών

A_{11}, A_{10} : bit επιλογής

ROM: (A_{12}, A_{11}) = (0,0) και addr: A_{10-0}

RAM1: (A_{12}, A_{11}) = (0,1) η (A_{12}, A_{11}) = (1,0) και addr: A_{10-0}

RAM2: (A_{12}, A_{11}) = (1,1) και addr: A_{10-0}

