

K23 - Θέμα 3

Συστήματα Μικροπολογιστών - Ελένη Νασοπούλου

(β) Στο μΕ 8085 όταν εκτελείται η εντολή CALL 0530H, ο μετρητής προγράμματος είναι (PC)=0630H και ο δείκτης σωρού (SP)=0740H. Στο μέσον της εκτέλεσης της εντολής συμβαίνει διακοπή RST 5.5. Δώστε τις νέες τιμές των PC, SP, το περιεχόμενο του σωρού καθώς και τις λειτουργίες που συμβαίνουν στην αρχή και στην επιστροφή από την ρουτίνα εξυπηρέτησης. (0.5 ΜΟΝΑΔΕΣ)

Οι λειτουργίες που επιτελούνται ως ανάρτηση στη software διακοπή RST 5.5 είναι οι εξής:

1. Αποκλήρωση τρέχουσας εντολής:

πριν: $\left\{ \begin{array}{l} (PC) = 0630H \\ (SP) = 0740H \end{array} \right\}$ μετά: $\left\{ \begin{array}{l} (PC) = 0530H \\ (SP) = 073EH \end{array} \right\}$ $(073EH) = 30H$
 $(073FH) = 06H$

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
...	...

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

$\left\{ \begin{array}{l} (PC) = 002CH \\ (SP) = 073CH \end{array} \right\}$ $(073CH) = 30H$
 $(073DH) = 05H$
 $(073EH) = 30H$
 $(073FH) = 06H$

3. μετά την αποκλήρωση της ρουτίνας διακοπής ανακαίεται η κατάσταση του μΕ και εκτελεί την επόμενη εντολή

$\left\{ \begin{array}{l} (PC) = 0530H \\ (SP) = 073EH \end{array} \right\}$ $(073EH) = 30H$
 $(073FH) = 06H$

K23 - Θέμα 4

(α) Ποιος είναι ο ρόλος της στοιβας σε έναν μικροελεξεργαστή? Αναφέρετε δυο παραδείγματα χρήσης της.

Η στοιβας (stack) σε έναν μικροεπεξεργαστή, όπως ο 8085, χρησιμοποιείται για την προσωρινή αποθήκευση δεδομένων και διευθύνσεων.

Ο ρόλος της περιλαμβάνει διαχείριση υπορουτινών, αποθήκευση δεδομένων και διαχείριση διακοπών.

Η στοιβας λειτουργεί ως μια περιοχή μνήμης LIFO (Last In, First Out), όπου τα δεδομένα προστίθενται και ανακτώνται με τη χρήση των εντολών PUSH και POP.

(β) Ποια η διαφορά ρουτίνας και μακροεντολής; Σε ποιες περιπτώσεις θα προτιμούσατε την χρήση της μιας μεθόδου και σε ποιες της άλλης; (0.3 ΜΟΝΑΔΕΣ)

- Μακροεντολή: Κατά τη δημιουργία του κώδικα, τοποθετείται στη θέση της όλο το σώμα της μακροεντολής. Προτιμάται για εξοικονόμηση χρόνου εκτέλεσης.
- Ρουτίνα: Κατά τη δημιουργία του κώδικα, μεταβαίνουμε σε ένα τμήμα έξω από το κυρίως πρόγραμμα, διακόπτεται η εκτέλεση του και ο μετρητής προγράμματος μεταβαίνει στην πρώτη εντολή της ρουτίνας. Προτιμάται για εξοικονόμηση μνήμης.

(γ) Για ποιο λόγο η μνήμη των ενός μικροελεγκτή είναι χωρισμένη σε τμήματα με διαφορετική διευθυνσιοδότηση;

Η μνήμη RAM ενός μικροελεγκτή είναι συνήθως οργανωμένη σε διαφορετικά τμήματα με διαφορετική διευθυνσιοδότηση για να εξυπηρετεί διάφορους σκοπούς. Κάθε τμήμα μπορεί να χρησιμοποιείται για τη διαχείριση μεταβλητών, στοιβών και κώδικα προγράμματος, επιτρέποντας αποδοτική πρόσβαση από τον επεξεργαστή και προγραμματιζόμενη προστασία για ανάγνωση ή εγγραφή δεδομένων. Επιπλέον, η διαίρεση σε τμήματα επιτρέπει την απενεργοποίηση μέρους της μνήμης για εξοικονόμηση ενέργειας σε χαμηλή κατανάλωση εφαρμογών.

K23 - Θέμα 4

(δ) Έστω ένα $\mu\text{Υ-}\Sigma$ με τα παρακάτω χαρακτηριστικά: 1) Μονάδα μνήμης ROM 8K (Διευθύνσεις 0000H – 1FFFH); 2) Μονάδα μνήμης RAM 4K (Διευθύνσεις 2000H – 2FFFH); 3) Μονάδα μνήμης RAM 2K (Διευθύνσεις 3000H – 37FFH);
Να σχεδιάσετε το λογικό κύκλωμα που επιλέγει την κάθε μονάδα μνήμης ανάλογα με την εκάστοτε ζητούμενη διεύθυνση. Η ολοκλήρωση να γίνει με τη χρήση λογικών πυλών.

$A_{15}-A_8$	A_7-A_4	A_3-A_0	addr	mem
0000	0000	0000	0000H	ROM
0001	1111	1111	1FFFH	
0010	0000	0000	2000H	RAM1
0010	1111	1111	2FFFH	
0011	0000	0000	3000H	RAM2
0011	0111	1111	37FFH	

A_{15}, A_{14} : επιρροών

A_{15}, A_{12}, A_{11} : bit διαδρομής

ROM: $A_{15}=0$ και addr: $A_{12}=0$

RAM1: $(A_{15}, A_{12})=(1,0)$ και addr: $A_{11}=0$

RAM2: $(A_{15}, A_{12}, A_{11})=(1,1,0)$ και addr: $A_{10}=0$

