



ΕΘΝΙΚΟ ΜΕΤΣΟΒΙΟ ΠΟΛΥΤΕΧΝΕΙΟ

ΣΧΟΛΗ ΗΜ&ΜΥ
Εργαστήριο Μικροϋπολογιστών

2^η Εργαστηριακή Άσκηση
Ακ. έτος 2011-2012

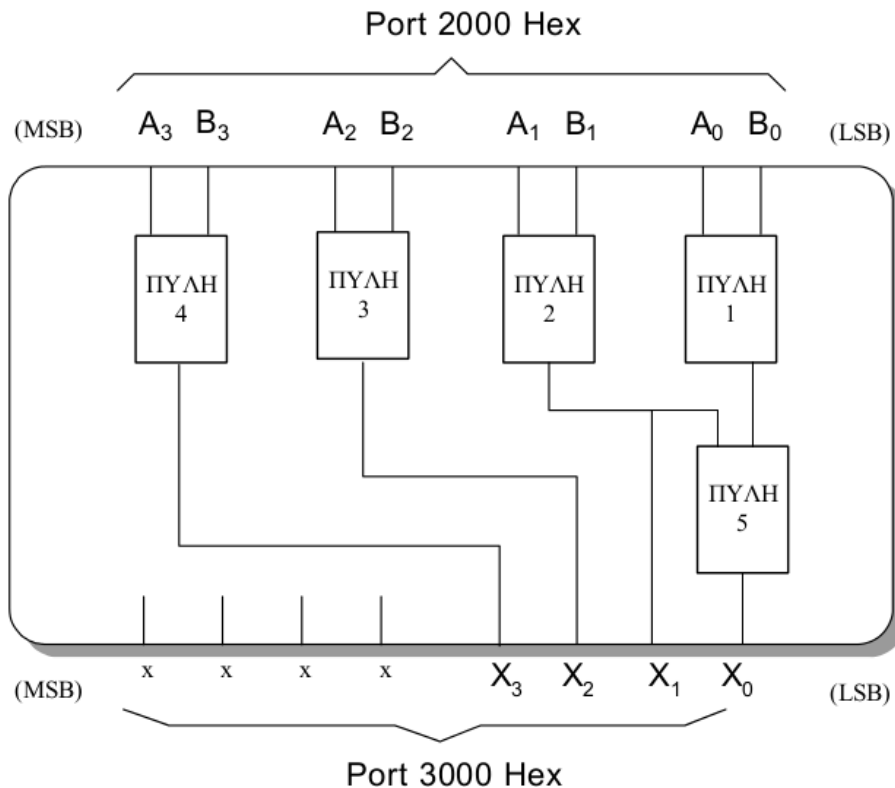
Ομάδα C07:

Ελένη Ευαγγελάτου	A.M.: 03108050
Γρηγόρης Λύρας	A.M.: 03109687
Βασιλεία Φραγκιαδάκη	A.M.: 03108026

11 Δεκεμβρίου 2011

Άσκηση 3(i)

Σε αυτήν την άσκηση υλοποιούμε την εξομοίωση του εξής ολοκληρωμένου κυκλώματος: Για την διαδικασία



Σχήμα 1: Το IC του θέματος

που ακολουθήσαμε αποθηκεύσαμε το αποτέλεσμα μας στον C, έχοντας ως βοηθητικό buffer τον B και κάνοντας πράξεις στον A. Αρχικά παίρνουμε ένα ένα ψηφίο από την είσοδο μας απομονώνοντάς το με RAL και βλέπουμε το κρατούμενο. Υλοποιούμε σε πρώτο βήμα τις 4 πύλες XOR. Για να υλοποιήσουμε την κάθε πύλη ουσιαστικά κοιτάμε αν έχουμε 0 με 1 (ή 1 με 0) στα συγκρινόμενα ψηφία, οπότε και η XOR δίνει αποτέλεσμα TRUE (δηλαδή 1). Αυτήν την πράξη επαναλαμβάνουμε και για τα 8 bits της εισόδου (φυσικά σε ζεύγη, όπως δίνονται στο σχήμα). Το αποτέλεσμα αποθηκεύεται στον C. Σε δεύτερο στάδιο αποθηκεύουμε τα δύο τελευταία least significant bits και τα βάζουμε ως τελεστέους σε μία OR. Αφού προηγουμένως "αδειάζουμε" το lsb του C, αποθηκεύουμε εκεί το αποτέλεσμα της OR. Το τελικό αποτέλεσμα περνιέται εκ νέου στον C και τα περιεχόμενα του οποίου εμφανίζουμε στα LEDS.

Ο κώδικάς μας είναι ο εξής:

```

1  START:
2  LDA 2000H
3  CALL XOR
4  CALL XOR
5  CALL XOR
6  CALL XOR
7  MOV D,C
8  MOV A,C
9  RAR
10 MOV C,A
11 MOV A,D
12 RAL
13 RAL
14 RAL
15 RAL
16 RAL
17 RAL
18 CALL OR
19 MOV A,C
20 ANI 0FH
21 CMA

```

```

22     STA 3000H
23     JMP START
24
25 XOR:
26     RAL          ;CALL XOR
27     JC ONE       ;XY
28     RAL          ;OY
29     JC TRUE      ;OY
30     JMP FALSE    ;00
31 ONE:
32     RAL          ;1Y
33     JC FALSE     ;11
34     JMP TRUE     ;10
35
36 OR:
37     RAL          ;CALL OR;XY
38     JC TRUE      ;1Y
39     RAL          ;OY
40     JC TRUE      ;11
41     JMP FALSE    ;00
42
43 TRUE:
44     MOV B,A      ;USE B AS BUFFER
45     MOV A,C      ;LOAD PREVIOUS OUTCOME FROM C
46     RAL          ;SHIFT A
47     ORI 01H      ;ADD 1 IN THE END
48     JMP RESTORE
49
50 FALSE:
51     MOV B,A      ;USE B AS BUFFER
52     MOV A,C      ;LOAD PREV FROM C
53     RAL          ;SHIFT A
54     ANI FEH      ;ADD 0 IN THE END
55     JMP RESTORE
56
57 RESTORE:
58     MOV C,A      ;RESTORE BUFFERS AS THEY WERE
59     MOV A,B
60     RET
61 END

```

Άσκηση 4(ii)

Σε αυτή την άσκηση ζητείται να απεικονίσουμε στα δύο αριστερότερα displays την τιμή του κωδικού του πλήκτρου που πατήθηκε σύμφωνα με τον πίνακα 1 (στη σελ. 82 του βιβλίου). Γι' αυτό το σκοπό πρώτα σβήνουμε τελείως όλα τα displays, φορτώνοντας στις θέσεις μνήμης 0BB0H-0BB5H τη δεκαεξαδική τιμή 10H (αφού έχουμε άρει την προστασία μνήμης) και καλώντας τη ρουτίνα STDM που μεταφέρει τα δεδομένα από εκεί που τα έχουμε αποθηκεύσει, στις θέσεις μνήμης που θα τα βρει η DCD. Έπειτα η KIND φορτώνει στον A τον κωδικό του πλήκτρου που πατήθηκε. Για να τον απεικονίσουμε, απομονώνουμε τα τέσσερα lsb και τα φορτώνουμε στη θέση μνήμης που αντιστοιχεί στο 5ο ψηφίο και τα msb που φορτώνουμε στη θέση μνήμης που αντιστοιχεί στο 6ο ψηφίο. Έπειτα, με την κλήση διαδοχικά των STDM και DCD απεικονίζεται στα displays ο ζητούμενος κωδικός. Να σημειώσουμε ότι η KIND στο σώμα της, όπου υπάρχει αναμονή, καλεί συνεχώς τη DCD και έτσι "φρεσκάρεται" η οθόνη.

Ο κώδικάς μας φαίνεται παρακάτω:

```

1     IN 10H
2     MVI A,10H    ;fortunoume to 10H
3     STA 0BB0H    ;wste na ka8arisei
4     STA 0BB1H    ;to display
5     STA 0BB2H
6     STA 0BB3H
7     STA 0BB4H
8     STA 0BB5H
9     LXI D,0BB0H
10    CALL STDM
11 LP:
12    CALL KIND     ;h kind 8a ferei ton
13    MOV B,A       ;kwdiko pshfiou ston A
14    ANI 0FH       ;xrhsimopoioume ton B ws buffer
15    STA 0BB4H     ;grafoume ta 4LSB sthn 8esh tou pemptou pshfiou
16    MOV A,B
17    RLC           ;me kuklikh olis8ish 4fores
18    RLC           ;fernoume ta 4MSB sta 4LSB
19    RLC           ;gia na ta grapsoume sth dieu8unsh

```

```

20 RLC ;του εκτου pshfiou
21 ANI OFH
22 STA OBB5H
23 LXI D,OBB0H ;dinoume ston DE th 8esh pou 8a
24 CALL STDM ;xrhsimopoihsei h STDM gia na kanei to refresh
25 JMP LP ;tou display
26 END

```

Άσκηση 4(iv)

Η λογική μας σε αυτήν την άσκηση είναι ότι από τον δοθέντα αριθμό μετρήσαμε τις εκατοντάδες (που μπορεί να είναι μία ή μηδέν, διότι το πολύ μέχρι 127 θα μετρήσουμε κατά την εκφώνηση της άσκησης), τις δεκάδες και τις μονάδες και το πρόσημο του αριθμού. Οι εκατοντάδες, δεκάδες και μονάδες του αριθμού αποθηκεύονται στους καταχωρητές D, B, C αντίστοιχα, ενώ το πρόσημο στον E. Οι αριθμοί που δίνονται είναι από -128 έως 127.

MSB								
0	1	1	1	1	1	1	1	= 127
0	1	1	1	1	1	1	0	= 126
0	0	0	0	0	0	1	0	= 2
0	0	0	0	0	0	0	1	= 1
0	0	0	0	0	0	0	0	= 0
1	1	1	1	1	1	1	1	= -1
1	1	1	1	1	1	1	0	= -2
1	0	0	0	0	0	0	1	= -127
1	0	0	0	0	0	0	0	= -128

Σχήμα 2: Πίνακας τιμών

Για να γίνει αυτή η διαδικασία, βλέπουμε αν ο αριθμός μας είναι θετικός ή αρνητικός από το msb του. Εφόσον είναι θετικός αποθηκεύουμε στον E τον κωδικό της DCD ,10, ο οποίος αντιστοιχεί σε (κενό). Αν είναι αρνητικός τότε αποθηκεύουμε τον κωδικό 1C στον E, γιατί αυτός αντιστοιχεί σε μείον (-) στην DCD . Επίσης για να χειριστούμε τον υπόλοιπο αρνητικό αριθμό (ο οποίος είναι σε συμπλήρωμα ως προς 2), παίρνουμε το συμπλήρωμα ως προς 1 και προσθέτουμε 1. Αυτό το κάνουμε διότι το συμπλήρωμα ως προς 2 του συμπληρώματος ως προς 2 μας δίνει τον αρχικό (θετικό) αριθμό. Στην συνέχεια για να μετρήσουμε τις εκατοντάδες, δεκάδες και μονάδες, κοιτάμε αν ο αριθμός μας είναι μεγαλύτερος του 100. Αν ναι μετράμε μία εκατοντάδα και αφαιρούμε από τον αριθμό 100. Έπειτα (ή και αν ο αριθμός είναι <100) μετράμε πόσες δεκάδες έχουμε. Αφαιρούμε 10 έως ότου να έχουμε αρνητικό αριθμό και μετράμε πόσες φορές αφαιρέσαμε 10. μετά προσθέτουμε 10 για την διόρθωση του υπολοίπου. Το διορθωμένο υπόλοιπο είναι οι μονάδες. Αποθηκεύουμε τους καταχωρητές σε 6 συνεχόμενες θέσεις μνήμης (4 οι καταχωρητές + 2 με κωδικούς 10 για κενό, ώστε να μην εμφανίζουν τίποτε). Πλέον καλούμε την STDM , αφού της “γνωστοποιούμε” που είναι αυτές οι 6 θέσεις μνήμης και καλούμε και την DCD για την εμφάνιση στα 7segments και έχουμε τον αριθμό σε δεκαδική μορφή μαζί με το πρόσημο στα 7 segments.

```

1 START:
2 IN 10H
3
4 READ_BINARY:
5 LDA 2000H
6 RAL
7 JC NEGATIVE
8 JNC POSITIVE
9
10 ;ARNHTIKOS EINAI MONO OTAN MSB==1 , OPOTE KAI TON "METATREPW"
11
12 NEGATIVE:
13 RAR ;GIA UNDO THN RAL
14 CMA ;TO SUMPLHRWMA WS PROS 2 TOY SUMPLHRWMATOS WS PROS 2 MAS DINEI TON ARX ARI8MO
15 INR A
16 MVI E,1CH ;O E EXEI TO PROSHMO. 17 GIATI AYTO ANTISTOIXEI SE - STHN DCD
17 ;CMA ;TO CMA XREIAZETAI GIATI PLEON OLOI ARITHMOI MOU EINAI 8ETIKOI
18 JMP DEC_CONVERSION
19
20 POSITIVE:
21 RAR ;GIA UNDO RAL
22 MVI E,10H ;O E EXEI TO PROSHMO , TO 10 GIATI AUTO ANTISTOIXEI SE KENO STHN DCD
23
24 DEC_CONVERSION:

```

```

25
26     MVI C,00H ;STON C OI MONADES
27     MVI B,FFH ;STON B OI DECADES , EBALA FF GIATI META KSEKINAW KANONTAS INR
28     MVI D,00H ;STON D OI EKATONTADES
29
30 BCD_FINDER:
31     CPI 64H ;SUGKRINW ME TO 100
32     JC ELEGX_DEC ;AN EINAI DHL. NUMBER<100 H' ALLIWS (A)-100<0 TOTE ELEGXEI DECADES
33     MVI D,01H ;ALLIWS O ARITHMOS MAS EXEI MIA EKATONTADA
34     SUI 64H ;AFAIRW TO 100 KAI SUNEXIZW GIA DECADES
35
36 ELEGX_DEC:
37     INR B
38     SUI 0AH ;AFAIRW 10 KAI METRAW POSES DECADES AFAIRESA
39     JNC ELEGX_DEC ;MEXRI NA MEINOUNE MONADES
40     ADI 0AH ;DIORTHWSH YPOLOIPOU, PROSTHETW 10
41     MOV C,A ;STON C OI MONADES
42
43 EMFANISH_STA_3DEKSIA_7SEGM:
44
45     ;===1o LSB PSHFIO===
46     LXI H,0990H ;BAZW STON H::=C =LEAST SIGN BIT, TIS MONADES @ THESH MNMHMS 0990
47     MOV M,C
48
49     ;===2o LSB PSHFIO===
50     LXI H,0991H ;BAZW STON H::=B =2o LSB TIS DEKADES @ THESH MNMHMS 0991
51     MOV M,B
52
53     ;===3o LSB PSHFIO===
54     LXI H,0992H ;BAZW STON H::=L THN EKATONTADA (AN YPARXEI) @ THESH MNMHMS 0991
55     MOV M,D
56
57     ;===4o LSB PSHFIO===
58     LXI H,0993H ;BAZW STON H::=E TO PROSHMO
59     MOV M,E
60
61     ;===5o LSB PSHFIO===
62     LXI H,0994H ;EMFANIZEI KENA TA YPOLOIPA BITS
63     MVI M,10H
64
65     ;===6o (MSB)PSHFIO===
66     LXI H,0995H
67     MVI M,10H
68
69
70     LXI D,0990H ; BAZW STON D THN THESH APO OPOU ARXIZEI TO LSB
71     CALL STDM ; TA METAFEREI TA BITS NA TA PAREI H DCD KAI NA TA PAEI STA 7SEG
72     CALL DCD
73     JMP START ; H DCD PAIRNEI TA PSHFIA KAI TA BAZEI STOUS 7SEGMENTS
74 END

```