



ΠΑΝΕΠΙΣΤΗΜΙΟ
ΠΑΤΡΩΝ
UNIVERSITY OF PATRAS

ΤΜΗΜΑ ΜΗΧΑΝΙΚΩΝ Η/Υ ΚΑΙ ΠΛΗΡΟΦΟΡΙΚΗΣ
ΠΟΛΥΤΕΧΝΙΚΗ ΣΧΟΛΗ
ΜΑΘΗΜΑ: ΣΥΓΧΡΟΝΑ ΘΕΜΑΤΑ ΑΡΧΙΤΕΚΤΟΝΙΚΗΣ ΥΠΟΛΟΓΙΣΤΩΝ – ΕΡΓΑΣΤΗΡΙΟ

10

Ονοματεπώνυμο: ΜΗΛΤΙΑΔΗΣ ΜΑΝΤΕΣ

A.M.: 1084661

E-mail: up1084661@upnet.gr

Ονοματεπώνυμο: ΦΙΛΙΠΠΟΣ-ΠΑΡΑΣΚΕΥΑΣ ΖΥΓΟΥΡΗΣ

A.M.: 1084660

E-mail: up1084660@upnet.gr

ΑΝΑΦΟΡΑ ΑΣΚΗΣΗΣ 2

Accumulator: 0011

Program Counter: 1100

1 Βοηθητικός Καταχωρητής X: 0110

Ζητούμενο

Να γραφεί πρόγραμμα στο οποίο θα καλούνται όλες οι παρακάτω εντολές επιλέγοντας διάφορες τιμές για τα έντελα K. Επίσης μετά από κάθε εντολή LDA και LDX να τοποθετούνται οι κατάλληλες εντολές και μια εντολή STA, ώστε να μπορεί εύκολα να επιβεβαιώνεται η σωστή λειτουργία της αντίστοιχης LDA και LDX εντολής. Οι τιμές των καταχωρητών ACC, PC και X (συσσωρευτή, Μετρητή Προγράμματος και βοηθητικού καταχωρητή) θα σας δοθούν κατά την διάρκεια του εργαστηρίου.

LDA #K	Φόρτωσε στο συσσωρευτή τον διψήφιο δεκαεξαδικό αριθμό K.
LDX #K	Φόρτωσε στον βοηθητικό καταχωρητή X τον διψήφιο δεκαεξαδικό αριθμό K.
LDA (\$K)	Φόρτωσε στο συσσωρευτή το περιεχόμενο της θέσης μνήμης με διεύθυνση το περιεχόμενο της θέσης μνήμης με διεύθυνση K.
LDAX	Φόρτωσε στο συσσωρευτή το περιεχόμενο της θέσης μνήμης με διεύθυνση το περιεχόμενο του καταχωρητή X.
LDA K(X)	Φόρτωσε στο συσσωρευτή το περιεχόμενο της θέσης μνήμης με διεύθυνση ίση με άθροισμα του διψήφιου δεκαεξαδικού αριθμού K και του περιεχομένου του βοηθητικού καταχωρητή X.
MOVEAX	Μετέφερε το περιεχόμενο του καταχωρητή X στον συσσωρευτή.
STA \$K	Αποθήκευσε το περιεχόμενο του συσσωρευτή στη θέση μνήμης με διεύθυνση K.

Πρόγραμμα σε συμβολική γλώσσα

LDA #K :

PC + 1 → PC , MAR //έντελο εντολής
MDR + 0 → ACC //μεταφορά εντέλου (K) σε acc
PC + 1 → PC, MAR //επόμενη εντολή
NEXT(PC)

LDX #K :

PC + 1 → PC , MAR //έντελο εντολής
MDR + 0 → X //μεταφορά εντέλου (K) σε X
PC + 1 → PC, MAR //επόμενη εντολή
NEXT(PC)

LDA (\$K) :

PC + 1 → PC , MAR //έντελο εντολής
MDR + 0 → MAR //μεταφορά εντέλου (K) σε MAR
MDR + 0 → X //μεταφορά δεδομένου της K σε X
X + 0 → MAR //αποθήκευση διεύθυνσης που είναι
το περιεχόμενο του X σε MAR
MDR + 0 → ACC //μεταφορά δεδομένου της
διεύθυνσης που είναι περιεχόμενο του X σε acc
PC + 1 → PC, MAR //επόμενη εντολή
NEXT(PC)

2

LDAX :

X + 0 → MAR //μεταφορά
δεδομένου του X σε MAR
MDR + 0 → ACC //μεταφορά
δεδομένου της K σε X
PC + 1 → PC, MAR //επόμενη
εντολή
NEXT(PC)

LDA K(X) :

PC + 1 → PC , MAR //έντελο
εντολής
MDR + X → MAR //μεταφορά
εντέλου (K + X) σε MAR
MDR + 0 → ACC //μεταφορά
δεδομένου της διεύθυνσης K + X σε
acc
PC+1 → PC,MAR //επόμενη εντολή
NEXT(PC)

MOVEAX :

X + 0 → ACC //μεταφορά
δεδομένου του X σε acc
PC + 1 → PC, MAR //επόμενη
εντολή
NEXT(PC)

STA \$K :

PC + 1 → PC , MAR //έντελο
εντολής
MDR + 0 → NOP,MAR
ACC + 0 → NOP, MWE~
PC + 1 → PC, MAR //επόμενη
εντολή
NEXT(PC)

Πρόγραμμα σε γλώσσα μηχανής

BOOTSTRAP	BRA	BIN	CON	I	I	I	APOINT	BPORT	DDATA	SH~	SELB	MWE~	MARCLK	MSTATUS	LDS~	PCE~	CARRYE~	MDE~	DDATAE~	ADDRESS
	(4:0)	(2:0)	(2:0)	(2:0)	(5:3)	(8:6)	(3:0)	(3:0)	(1:0)											
SW+0->PC,MAR	XXXXX	000	XXX	111	000	011	XXXX	1100	XX	X	1	1	1	0	1	0	1	1	1	m00
NEXT(PC)	XXXXX	000	XXX	XXX	XXX	001	XXXX	XXXX	XX	X	1	1	0	0	0	1	1	1	1	m01

LDA #K	BRA	BIN	CON	I	I	I	APOINT	BPORT	DDATA	SH~	SELB	MWE~	MARCLK	MSTATUS	LDS~	PCE~	CARRYE~	MDE~	DDATAE~	ADDRESS
	(4:0)	(2:0)	(2:0)	(2:0)	(5:3)	(8:6)	(3:0)	(3:0)	(1:0)											
PC+1->PC,MAR	XXXXX	000	XXX	101	000	011	1100	1100	01	1	1	1	1	0	1	1	1	1	0	m02
MDR+0->ACC	XXXXX	000	XXX	111	000	011	XXXX	0011	XX	1	1	1	0	0	1	1	1	0	1	m03
PC+1->PC,MAR	XXXXX	000	XXX	101	000	011	1100	1100	01	1	1	1	1	0	1	1	1	1	0	m04
NEXT(PC)	XXXXX	000	XXX	XXX	XXX	001	XXXX	XXXX	XX	X	1	1	0	0	0	1	1	1	1	m05

3

STA \$K	BRA	BIN	CON	I	I	I	APOINT	BPORT	DDATA	SH~	SELB	MWE~	MARCLK	MSTATUS	LDS~	PCE~	CARRYE~	MDE~	DDATAE~	ADDRESS
	(4:0)	(2:0)	(2:0)	(2:0)	(5:3)	(8:6)	(3:0)	(3:0)	(1:0)											
PC+1->PC,MAR	XXXXX	000	XXX	101	000	011	1100	1100	01	1	1	1	1	0	1	1	1	1	0	m06
MDR+0->X	XXXXX	000	XXX	111	000	011	XXXX	0110	XX	1	1	1	0	0	1	1	1	0	1	m07
X+0->NOP,MAR	XXXXX	000	XXX	100	000	001	0110	XXXX	XX	1	1	1	1	0	1	1	1	1	1	m08
ACC+0->NOP,MWE	XXXXX	000	XXX	100	000	001	0011	XXXX	XX	1	1	1	0	0	1	1	1	1	1	m09
PC+1->PC,MAR	XXXXX	000	XXX	101	000	011	1100	1100	01	1	1	1	1	0	1	1	1	1	0	m0a
NEXT(PC)	XXXXX	000	XXX	XXX	XXX	001	XXXX	XXXX	XX	X	1	1	0	0	0	1	1	1	1	m0b

LDX #K	BRA	BIN	CON	I	I	I	APOINT	BPORT	DDATA	SH~	SELB	MWE~	MARCLK	MSTATUS	LDS~	PCE~	CARRYE~	MDE~	DDATAE~	ADDRESS
	(4:0)	(2:0)	(2:0)	(2:0)	(5:3)	(8:6)	(3:0)	(3:0)	(1:0)											
PC+1->PC,MAR	XXXXX	000	XXX	101	000	011	1100	1100	01	1	1	1	1	0	1	1	1	1	0	m0c
MDR+0->X	XXXXX	000	XXX	111	000	011	XXXX	0110	XX	1	1	1	0	0	1	1	1	0	1	m0d
PC+1->PC,MAR	XXXXX	000	XXX	101	000	011	1100	1100	01	1	1	1	1	0	1	1	1	1	0	m0e
NEXT(PC)	XXXXX	000	XXX	XXX	XXX	001	XXXX	XXXX	XX	X	1	1	0	0	0	1	1	1	1	m0f

STA \$K	BRA	BIN	CON	I	I	I	APORT	BPORT	DDATA	SH~	SELB	MWE~	MARCLK	MSTATUS	LDS~	PCE~	CARRYE~	MDE~	DDATAE~	ADDRESS
	(4:0)	(2:0)	(2:0)	(2:0)	(5:3)	(8:6)	(3:0)	(3:0)	(1:0)											
PC+1->PC,MAR	XXXXX	000	XXX	101	000	011	1100	1100	01	1	1	1	1	0	1	1	1	1	0	m10
MDR+0->X	XXXXX	000	XXX	111	000	011	XXXX	0110	XX	1	1	1	0	0	1	1	1	0	1	m11
X+0->NOP,MAR	XXXXX	000	XXX	100	000	001	0110	XXXX	XX	1	1	1	1	0	1	1	1	1	1	m12
ACC+0->NOP,MWE	XXXXX	000	XXX	100	000	001	0011	XXXX	XX	1	1	1	0	0	1	1	1	1	1	m13
PC+1->PC,MAR	XXXXX	000	XXX	101	000	011	1100	1100	01	1	1	1	1	0	1	1	1	1	0	m14
NEXT(PC)	XXXXX	000	XXX	XXX	XXX	001	XXXX	XXXX	XX	X	1	1	0	0	0	1	1	1	1	m15

LDA (\$K)	BRA	BIN	CON	I	I	I	APORT	BPORT	DDATA	SH~	SELB	MWE~	MARCLK	MSTATUS	LDS~	PCE~	CARRYE~	MDE~	DDATAE~	ADDRESS
	(4:0)	(2:0)	(2:0)	(2:0)	(5:3)	(8:6)	(3:0)	(3:0)	(1:0)											
PC+1->PC,MAR	XXXXX	000	XXX	101	000	011	1100	1100	01	1	1	1	1	0	1	1	1	1	0	m16
MDR+0->ACC	XXXXX	000	XXX	111	000	011	XXXX	0011	XX	1	1	1	0	0	1	1	1	0	1	m17
ACC+0->NOP,MAR	XXXXX	000	XXX	100	000	001	0011	XXXX	XX	1	1	1	1	0	1	1	1	1	1	m18
MDR+0->X	XXXXX	000	XXX	111	000	011	XXXX	0110	XX	1	1	1	0	0	1	1	1	0	1	m19
X+0->MAR	XXXXX	000	XXX	100	000	001	0110	XXXX	XX	1	1	1	1	0	1	1	1	1	1	m1a
MDR+0->ACC	XXXXX	000	XXX	111	000	011	XXXX	0011	XX	1	1	1	0	0	1	1	1	0	1	m1b
PC+1->PC,MAR	XXXXX	000	XXX	101	000	011	1100	1100	01	1	1	1	1	0	1	1	1	1	0	m1c
NEXT(PC)	XXXXX	000	XXX	XXX	XXX	001	XXXX	XXXX	XX	X	1	1	0	0	0	1	1	1	1	m1d

STA \$K	BRA	BIN	CON	I	I	I	APORT	BPORT	DDATA	SH~	SELB	MWE~	MARCLK	MSTATUS	LDS~	PCE~	CARRYE~	MDE~	DDATAE~	ADDRESS
	(4:0)	(2:0)	(2:0)	(2:0)	(5:3)	(8:6)	(3:0)	(3:0)	(1:0)											
PC+1->PC,MAR	XXXXX	000	XXX	101	000	011	1001	1001	01	1	1	1	1	0	1	1	1	1	0	m1e
MDR+0->X	XXXXX	000	XXX	111	000	011	XXXX	1010	XX	1	1	1	0	0	1	1	1	0	1	m1f
X+0->NOP,MAR	XXXXX	000	XXX	100	000	001	1010	XXXX	XX	1	1	1	1	0	1	1	1	1	1	m20
ACC+0->NOP,MWE	XXXXX	000	XXX	100	000	001	1000	XXXX	XX	1	1	1	0	0	1	1	1	1	1	m21
PC+1->PC,MAR	XXXXX	000	XXX	101	000	011	1001	1001	01	1	1	1	1	0	1	1	1	1	0	m22
NEXT(PC)	XXXXX	000	XXX	XXX	XXX	001	XXXX	XXXX	XX	X	1	1	0	0	0	1	1	1	1	m23

LDAX	BRA	BIN	CON	I	I	I	APOINT	BPORT	DDATA	SH~	SELB	MWE~	MARCLK	MSTATUS	LDS~	PCE~	CARRYE~	MDE~	DDATAE~	ADDRESS
	(4:0)	(2:0)	(2:0)	(2:0)	(5:3)	(8:6)	(3:0)	(3:0)	(1:0)											
X+0->MAR	XXXXX	000	XXX	100	000	001	1010	XXXX	XX	1	1	1	1	0	1	1	1	1	1	m24
MDR+0->ACC	XXXXX	000	XXX	111	000	011	XXXX	0011	XX	1	1	1	0	0	1	1	1	0	1	m25
PC+1->PC,MAR	XXXXX	000	XXX	101	000	011	1100	1100	01	1	1	1	1	0	1	1	1	1	0	m26
NEXT(PC)	XXXXX	000	XXX	XXX	XXX	001	XXXX	XXXX	XX	X	1	1	0	0	0	1	1	1	1	m27

STA \$K	BRA	BIN	CON	I	I	I	APOINT	BPORT	DDATA	SH~	SELB	MWE~	MARCLK	MSTATUS	LDS~	PCE~	CARRYE~	MDE~	DDATAE~	ADDRESS
	(4:0)	(2:0)	(2:0)	(2:0)	(5:3)	(8:6)	(3:0)	(3:0)	(1:0)											
PC+1->PC,MAR	XXXXX	000	XXX	101	000	011	1100	1100	01	1	1	1	1	0	1	1	1	1	0	m28
MDR+0->X	XXXXX	000	XXX	111	000	011	XXXX	0110	XX	1	1	1	0	0	1	1	1	0	1	m29
X+0->NOP,MAR	XXXXX	000	XXX	100	000	001	0110	XXXX	XX	1	1	1	1	0	1	1	1	1	1	m2a
ACC+0->NOP,MWE	XXXXX	000	XXX	100	000	001	0011	XXXX	XX	1	1	1	0	0	1	1	1	1	1	m2b
PC+1->PC,MAR	XXXXX	000	XXX	101	000	011	1100	1100	01	1	1	1	1	0	1	1	1	1	0	m2c
NEXT(PC)	XXXXX	000	XXX	XXX	XXX	001	XXXX	XXXX	XX	X	1	1	0	0	0	1	1	1	1	m2d

5

LDA K(X)	BRA	BIN	CON	I	I	I	APOINT	BPORT	DDATA	SH~	SELB	MWE~	MARCLK	MSTATUS	LDS~	PCE~	CARRYE~	MDE~	DDATAE~	ADDRESS
	(4:0)	(2:0)	(2:0)	(2:0)	(5:3)	(8:6)	(3:0)	(3:0)	(1:0)											
X+0->MAR	XXXXX	000	XXX	100	000	001	1010	XXXX	XX	1	1	1	1	0	1	1	1	1	1	m2e
MDR+0->ACC	XXXXX	000	XXX	111	000	011	XXXX	0011	XX	1	1	1	0	0	1	1	1	0	1	m2f
PC+1->PC,MAR	XXXXX	000	XXX	101	000	011	1100	1100	01	1	1	1	1	0	1	1	1	1	0	m30
NEXT(PC)	XXXXX	000	XXX	XXX	XXX	001	XXXX	XXXX	XX	X	1	1	0	0	0	1	1	1	1	m31

STA \$K	BRA	BIN	CON	I	I	I	APOINT	BPORT	DDATA	SH~	SELB	MWE~	MARCLK	MSTATUS	LDS~	PCE~	CARRYE~	MDE~	DDATAE~	ADDRESS
	(4:0)	(2:0)	(2:0)	(2:0)	(5:3)	(8:6)	(3:0)	(3:0)	(1:0)											
PC+1->PC,MAR	XXXXX	000	XXX	101	000	011	1100	1100	01	1	1	1	1	0	1	1	1	1	0	m32
MDR+0->X	XXXXX	000	XXX	111	000	011	XXXX	0110	XX	1	1	1	0	0	1	1	1	0	1	m33
X+0->NOP,MAR	XXXXX	000	XXX	100	000	001	0110	XXXX	XX	1	1	1	1	0	1	1	1	1	1	m34
ACC+0->NOP,MWE	XXXXX	000	XXX	100	000	001	0011	XXXX	XX	1	1	1	0	0	1	1	1	1	1	m35
PC+1->PC,MAR	XXXXX	000	XXX	101	000	011	1100	1100	01	1	1	1	1	0	1	1	1	1	0	m36
NEXT(PC)	XXXXX	000	XXX	XXX	XXX	001	XXXX	XXXX	XX	X	1	1	0	0	0	1	1	1	1	m37

MOVEAX	BRA	BIN	CON	I	I	I	APOINT	BPORT	DDATA	SH~	SELB	MWE~	MARCLK	MSTATUS	LDS~	PCE~	CARRYE~	MDE~	DDATAE~	ADDRESS
	(4:0)	(2:0)	(2:0)	(2:0)	(5:3)	(8:6)	(3:0)	(3:0)	(1:0)											
X+0->MAR	XXXXX	000	XXX	100	000	001	1010	XXXX	XX	1	1	1	1	0	1	1	1	1	1	m38
MDR+0->ACC	XXXXX	000	XXX	111	000	011	XXXX	0011	XX	1	1	1	0	0	1	1	1	0	1	m39
PC+1->PC,MAR	XXXXX	000	XXX	101	000	011	1100	1100	01	1	1	1	1	0	1	1	1	1	0	m3a
NEXT(PC)	XXXXX	000	XXX	XXX	XXX	001	XXXX	XXXX	XX	X	1	1	0	0	0	1	1	1	1	m3b

STA \$K	BRA	BIN	CON	I	I	I	APOINT	BPORT	DDATA	SH~	SELB	MWE~	MARCLK	MSTATUS	LDS~	PCE~	CARRYE~	MDE~	DDATAE~	ADDRESS
	(4:0)	(2:0)	(2:0)	(2:0)	(5:3)	(8:6)	(3:0)	(3:0)	(1:0)											
PC+1->PC,MAR	XXXXX	000	XXX	101	000	011	1100	1100	01	1	1	1	1	0	1	1	1	1	0	m3c
MDR+0->X	XXXXX	000	XXX	111	000	011	XXXX	0110	XX	1	1	1	0	0	1	1	1	0	1	m3d
X+0->NOP,MAR	XXXXX	000	XXX	100	000	001	0110	XXXX	XX	1	1	1	1	0	1	1	1	1	1	m3e
ACC+0->NOP,MWE	XXXXX	000	XXX	100	000	001	0011	XXXX	XX	1	1	1	0	0	1	1	1	1	1	m3f
PC+1->PC,MAR	XXXXX	000	XXX	101	000	011	1100	1100	01	1	1	1	1	0	1	1	1	1	0	m40
NEXT(PC)	XXXXX	000	XXX	XXX	XXX	001	XXXX	XXXX	XX	X	1	1	0	0	0	1	1	1	1	m41

6

Χρήση Don't care bits:

1. Μειώνουν την αλλαγή των καταστάσεων κάτι το οποίο συνεπάγεται με την μείωση του χώρου μνήμης που απαιτείται για την αναπαράσταση ενός δεδομένου ψηφιακού κυκλώματος που με τη σειρά του έχει ως αποτέλεσμα λιγότερη κατανάλωση ενέργειας.
2. Απλοποίηση εξόδου κυκλώματος.
3. Αποτρέπουν κινδύνους.
4. Διευκολύνουν σε μετατροπές κώδικα.

Σύντομη επεξήγηση Bootstrap:

SW+0->PC,MAR

Η μικροεντολή αυτή τελεί αρχικοποίηση του PC μέσω DIP SWITCHES της πλακέτας. Ξεκινώντας εκχωρούμε στις εισόδους *DDATAE~* και *MDE~* τιμή ίση με 1 (τα θέλουμε κλειστά), ενώ στην *PCE~* την τιμή 0. Η είσοδος *DDATA* θα λάβει αδιάφορους όρους (XX) διότι δεν μας είναι χρήσιμη. Στον Selector με είσοδο *I(2:0)* χρειαζόμαστε τον συνδυασμό DZ (Switches + 0), επομένως *I(2:0)* = 111. Ο Selector με εισόδους *SELB* & *BPORT* θα λάβει στις εισόδους του *SELB* = 1 για να περνάει το *BPORT*, το οποίο είναι ίσο με τον PC (1100). Για να πραγματοποιηθεί πρέπει η είσοδος *I(8:6)* του Register File να είναι ίση με 011. Στην αριθμητική μονάδα (ALU) ενεργοποιούμε την πρόσθεση εισάγοντας στην είσοδο *I(5:3)* την τιμή 000. Τέλος, το *MARCLK* λαμβάνει την τιμή 1 αφού επιθυμούμε να κλειδωθεί η διεύθυνση του καταχωρητή, ενώ καθώς δεν απαιτούμε εγγραφή στη μνήμη πρέπει *MWE~* = 1.

NEXT(PC)

Ενεργοποιείται η εντολή LDS~ ώστε α αρχίσει να τρέχει ο καινούργιος opcode τον οποίο διαβάσαμε (μεταφερόμαστε στο μικροπρόγραμμα της επόμενης μακροεντολής του μακροπρογράμματος).

- Εντελώς αυθαίρετα, ορίζουμε ένα μοναδικό **opcode**, διεύθυνση μικρομνήμης και **έντελο** για κάθε μία από τις μακροεντολές:

Mapper		
Κώδικας εντολής	Opcode/Θέση	Περιεχόμενα
LDA #01	00000000	00000010
STA \$0a	00000001	00000110
LDX #02	00000010	00001100
STA \$0b	00000011	00010000
LDA (\$03)	00000100	00010110
STA \$0c	00000101	00011110
LDAX	00000110	00100100
STA \$0d	00000111	00101000
LDA 04(X)	00001000	00101110
STA \$0e	00001001	00110010
MOVEAX	00001010	00111000
STA \$0f	00001011	00111100

↑ ↑
Opcodes Micromemory addresses

Main Memory		
Κώδικας εντολής	Θέση	Περιεχόμενο
LDA #01	00000000	00000000
	00000001	00000001
STA \$0a	00000010	00000001
	00000011	00001010
LDX #02	00000100	00000010
	00000101	00000010
STA \$0b	00000110	00000011
	00000111	00001011
LDA (\$03)	00001000	00000100
	00001001	00000011
STA \$0c	00001010	00000101
	00001011	00001100
LDAX	00001100	00000110
	00001101	
STA \$0d	00001110	00000111
	00001111	00001101
LDA 04(X)	00010000	00001000
	00010001	00000100
STA \$0e	00010010	00001001
	00010011	00001110
MOVEAX	00010100	00001010
	00010101	
STA \$0f	00010110	00001011
	00010111	00001111

← Opcode LDA #K
← Έντελο 1
← Opcode STA \$K
← Έντελο 10
← Opcode LDX #K
← Έντελο 2
← Opcode STA \$K
← Έντελο 11
← Opcode LDA (\$K)
← Έντελο 3
← Opcode STA \$K
← Έντελο 12
← Opcode LDAX
← Opcode STA \$K
← Έντελο 13
← Opcode LDA K(X)
← Έντελο 4
← Opcode STA \$K
← Έντελο 14
← Opcode MOVEAX
← Opcode STA \$K
← Έντελο 15