



ΤΕΙ ΣΕΡΡΩΝ

Τμήμα Πληροφορικής & Επικοινωνιών

Τομέας Αρχιτεκτονικής Υπολογιστών & Βιομηχανικών Εφαρμογών

ΕΡΓΑΣΤΗΡΙΟ ΑΡΧΙΤΕΚΤΟΝΙΚΗΣ Η/Υ

4^ο Εξάμηνο

Μαδεμλής Ιωάννης



ΤΕΙ ΣΕΡΡΩΝ

Τμήμα Πληροφορικής & Επικοινωνιών

Τομέας Αρχιτεκτονικής Υπολογιστών & Βιομηχανικών Εφαρμογών

ΕΡΓΑΣΤΗΡΙΟ 9

ΛΟΓΙΚΕΣ ΠΡΑΞΕΙΣ

Οι λογικές πράξεις που υποστηρίζει η Assembly του 8088 είναι :

- Πράξη AND
- Πράξη OR
- Πράξη NOT
- Πράξη XOR

Με τις λογικές πράξεις μπορούμε να:

- Να υπολογίσουμε τιμές λογικών συναρτήσεων (συναρτήσεων Boole)
- Να αλλάξουμε την τιμή ενός μόνο bit σε ένα καταχωρητή ή σε μια θέση μνήμης, χωρίς να επηρεαστούν τα υπόλοιπα bits
- Να ελέγξουμε την τιμή ενός bit ενός καταχωρητή ή μιας θέσης μνήμης



ΤΕΙ ΣΕΡΡΩΝ

Τμήμα Πληροφορικής & Επικοινωνιών

Τομέας Αρχιτεκτονικής Υπολογιστών & Βιομηχανικών Εφαρμογών

ΕΡΓΑΣΤΗΡΙΟ 9

ΛΟΓΙΚΕΣ ΠΡΑΞΕΙΣ

- Μηδενισμός bit. Για να μηδενίσουμε κάποια bit μέσα σε καταχωρητή ή θέση μνήμης, χωρίς να επηρεαστούν τα υπόλοιπα, κάνουμε **AND** τον καταχωρητή ή μνήμη με μια **μάσκα** που περιέχει «0» στις θέσεις των bit που θέλουμε να μηδενίσουμε και «1» στις υπόλοιπες.

Π.χ. θέλουμε να μηδενίσουμε τα LSB και MSB του AL

AL		X	X	X	X	X	X	X
Μάσκα	AND	0	1	1	1	1	1	0
AL		0	X	X	X	X	X	0



ΤΕΙ ΣΕΡΡΩΝ

Τμήμα Πληροφορικής & Επικοινωνιών

Τομέας Αρχιτεκτονικής Υπολογιστών & Βιομηχανικών Εφαρμογών

ΕΡΓΑΣΤΗΡΙΟ 9

ΛΟΓΙΚΕΣ ΠΡΑΞΕΙΣ

- Τοποθέτηση μονάδας σε **bit**. Για να κάνουμε 1 κάποια **bit** μέσα σε καταχωρητή ή θέση μνήμης, χωρίς να επηρεαστούν τα υπόλοιπα, κάνουμε **OR** τον καταχωρητή ή μνήμη με μια **μάσκα** που περιέχει «1» στις θέσεις των **bit** που θέλουμε να κάνουμε 1 και «0» στις υπόλοιπες.

Π.χ. θέλουμε να κάνουμε 1 τα **LSB** και **MSB** του **AL**

AL		X	X	X	X	X	X	X
Μάσκα	OR	1	0	0	0	0	0	1
AL		1	X	X	X	X	X	1



ΤΕΙ ΣΕΡΡΩΝ

Τμήμα Πληροφορικής & Επικοινωνιών

Τομέας Αρχιτεκτονικής Υπολογιστών & Βιομηχανικών Εφαρμογών

ΕΡΓΑΣΤΗΡΙΟ 9

ΛΟΓΙΚΕΣ ΠΡΑΞΕΙΣ

- Λήψη συμπληρώματος bit. Για να πάρουμε το συμπλήρωμα κάποιων bit μέσα σε καταχωρητή ή θέση μνήμης, χωρίς να επηρεαστούν τα υπόλοιπα, κάνουμε **XOR** τον καταχωρητή ή μνήμη με μια **μάσκα** που περιέχει «1» στις θέσεις των bit που θέλουμε να πάρουμε το συμπλήρωμα και «0» στις υπόλοιπες.

Π.χ. θέλουμε να πάρουμε το συμπλήρωμα των 4 λιγότερο σημαντικών bit του AL

AL		X	X	X	X	X	X	X	X
Μάσκα	XOR	0	0	0	0	1	1	1	1
AL		X	X	X	X	X'	X'	X'	X'



ΤΕΙ ΣΕΡΡΩΝ

Τμήμα Πληροφορικής & Επικοινωνιών

Τομέας Αρχιτεκτονικής Υπολογιστών & Βιομηχανικών Εφαρμογών

ΕΡΓΑΣΤΗΡΙΟ 9

ΛΟΓΙΚΕΣ ΠΡΑΞΕΙΣ

- Έλεγχος τιμής ενός bit. Για να ελέγξουμε αν κάποιο bit μέσα μέσα σε καταχωρητή ή θέση μνήμης είναι 0 ή 1, χρησιμοποιούμε την εντολή **TEST** μεταξύ του καταχωρητή ή μνήμης και μιας **μάσκας** που περιέχει «1» στη θέση του bit που θέλουμε να ελέγξουμε και «0» στις υπόλοιπες. Αν το αποτέλεσμα είναι 0 το bit που εξετάσαμε είναι 0, αλλιώς είναι 1.

Π.χ. θέλουμε να ελέγξουμε το MSB του AL

AL		X	X	X	X	X	X	X
Μάσκα	TEST	1	0	0	0	0	0	0
Αποτέλεσμα		?	0	0	0	0	0	0

Αν αποτέλεσμα = 0 (JZ = True) τότε **MSB=0**

Αν αποτέλεσμα \neq 0 (JNZ = True) τότε **MSB=1**



ΤΕΙ ΣΕΡΡΩΝ

Τμήμα Πληροφορικής & Επικοινωνιών

Τομέας Αρχιτεκτονικής Υπολογιστών & Βιομηχανικών Εφαρμογών

ΕΡΓΑΣΤΗΡΙΟ 9

ΕΝΤΟΛΕΣ ΛΟΓΙΚΩΝ ΠΡΑΞΕΩΝ

■ Εντολή **AND**

■ Λογικό ΚΑΙ, $AND\ a, b \rightarrow a = a\ AND\ b$

■ Εντολή **OR**

■ Λογικό Ή, $OR\ a, b \rightarrow a = a\ OR\ b$

■ Εντολή **XOR**

■ Αποκλειστικό Ή, $XOR\ a, b \rightarrow a = a\ XOR\ b$

■ Εντολή **TEST**

■ Λογικό ΚΑΙ χωρίς καταχώρηση αποτελέσματος, $TEST\ a, b \rightarrow a\ AND\ b$

■ Εντολή **NOT**

■ Λογικό ΌΧΙ, $NOT\ \text{καταχωρητής} \rightarrow \text{καταχωρητής} =$

Οι παράμετροι των εντολών, εκτός της **NOT**, είναι όπως της **MOV**



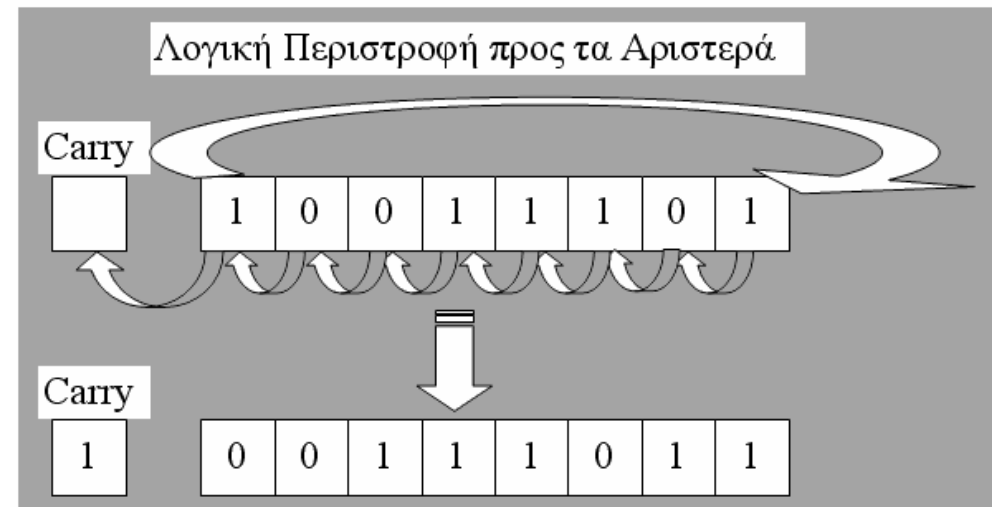
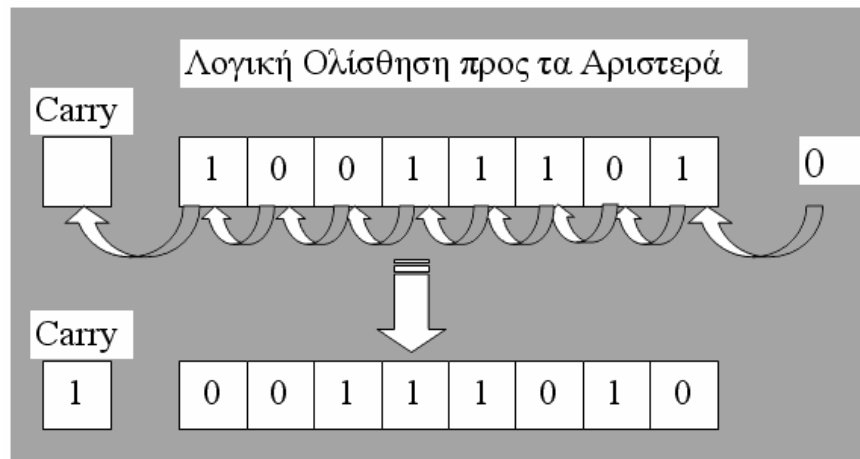
ΤΕΙ ΣΕΡΡΩΝ

Τμήμα Πληροφορικής & Επικοινωνιών

Τομέας Αρχιτεκτονικής Υπολογιστών & Βιομηχανικών Εφαρμογών

ΕΡΓΑΣΤΗΡΙΟ 9

ΟΛΙΣΘΗΣΗ & ΠΕΡΙΣΤΡΟΦΗ





ΤΕΙ ΣΕΡΡΩΝ

Τμήμα Πληροφορικής & Επικοινωνιών

Τομέας Αρχιτεκτονικής Υπολογιστών & Βιομηχανικών Εφαρμογών

ΕΡΓΑΣΤΗΡΙΟ 9

ΕΝΤΟΛΕΣ ΟΛΙΣΘΗΣΗΣ

■ Εντολή **SHL/SAL**, αριστερή ολίσθηση

- **SHL/SAL** καταχωρητής , 1

ολισθαίνει τα **bit** του καταχωρητή 1 θέση αριστερά, το υπερχειλίζον **bit** πάει στο Carry

- **SHL/SAL** καταχωρητής , CL

ολισθαίνει τα **bit** του καταχωρητή αριστερά όσο η τιμή του CL, το υπερχειλίζον **bit** πάει στο Carry

■ Εντολή **SHR**, δεξιά ολίσθηση

- **SHR** καταχωρητής , 1

ολισθαίνει τα **bit** του καταχωρητή 1 θέση δεξιά, το υπερχειλίζον **bit** πάει στο Carry

- **SHR** καταχωρητής , CL

ολισθαίνει τα **bit** του καταχωρητή δεξιά όσο η τιμή του CL, το υπερχειλίζον **bit** πάει στο Carry



ΤΕΙ ΣΕΡΡΩΝ

Τμήμα Πληροφορικής & Επικοινωνιών

Τομέας Αρχιτεκτονικής Υπολογιστών & Βιομηχανικών Εφαρμογών

ΕΡΓΑΣΤΗΡΙΟ 9

ΕΝΤΟΛΕΣ ΟΛΙΣΘΗΣΗΣ

■ Εντολή **SAR**, αριθμητική δεξιά ολίσθηση

■ SAR καταχωρητής, 1

ολισθαίνει τα **bit** του καταχωρητή 1 θέση δεξιά, το **MSB** παραμένει, το υπερχειλίζον **bit** πάει στο **Carry**

■ SAR καταχωρητής, CL

ολισθαίνει τα **bit** του καταχωρητή αριστερά όσο η τιμή του **CL**, το **MSB** παραμένει, το υπερχειλίζον **bit** πάει στο **Carry**



ΤΕΙ ΣΕΡΡΩΝ

Τμήμα Πληροφορικής & Επικοινωνιών

Τομέας Αρχιτεκτονικής Υπολογιστών & Βιομηχανικών Εφαρμογών

ΕΡΓΑΣΤΗΡΙΟ 9

ΕΝΤΟΛΕΣ ΠΕΡΙΣΤΡΟΦΗΣ

■ Εντολή **ROR/ROL**, δεξιά/αριστερή περιστροφή

■ ROR/ROL καταχωρητής , 1

περιστρέφει τα **bit** του καταχωρητή 1 θέση δεξιά/αριστερά, το υπερχειλίζον **bit** πάει στο Carry και ταυτόχρονα εισάγεται από αριστερά/δεξιά

■ ROR/ROL καταχωρητής , CL

περιστρέφει τα **bit** του καταχωρητή δεξιά/αριστερά όσο η τιμή του CL, το υπερχειλίζον **bit** πάει στο Carry και ταυτόχρονα εισάγεται από αριστερά/δεξιά

■ Εντολή **RCR/RCL**, δεξιά/αριστερή περιστροφή μέσω Carry

■ RCR/RCL καταχωρητής , 1

περιστρέφει τα **bit** του καταχωρητή 1 θέση δεξιά/αριστερά, το υπερχειλίζον **bit** πάει στο Carry και το Carry εισάγεται από αριστερά/δεξιά

■ RCR/RCL καταχωρητής , CL

περιστρέφει τα **bit** του καταχωρητή δεξιά/αριστερά όσο η τιμή του CL, το υπερχειλίζον **bit** πάει στο Carry και το Carry εισάγεται από αριστερά/δεξιά



ΤΕΙ ΣΕΡΡΩΝ

Τμήμα Πληροφορικής & Επικοινωνιών

Τομέας Αρχιτεκτονικής Υπολογιστών & Βιομηχανικών Εφαρμογών

ΕΡΓΑΣΤΗΡΙΟ 9

ΑΣΚΗΣΗ 9.1

Να γραφεί πρόγραμμα το οποίο θα μετατρέπει έναν αριθμό που βρίσκεται στη διεύθυνση μνήμης **0200** σε δυαδικό αριθμό βάζοντας κάθε ψηφίο του δυαδικού αριθμού σε διαδοχικά bytes με αρχή την διεύθυνση μνήμης **0208 (0208..020F)**, ώστε ο δυαδικός αριθμός να είναι αναγνώσιμος από τον χρήστη με την εντολή **D200**.

(Υπόδειξη : αρχικά φτιάξτε μία μάσκα με τιμή **10000000** η οποία σε κάθε επανάληψη θα ολισθαίνει δεξιά. Κάθε φορά συγκρίνετε τη μάσκα με τον αριθμό και αναλόγως τοποθετείτε **0** ή **1** στην κατάλληλη θέση μνήμης)

0200

FA

Αριθμός $(FA)_{16} = (11111010)_2$

Αποτέλεσμα :

0208 0209 020A 020B 020C 020D 020E 020F

01

01

01

01

01

00

01

00



ΤΕΙ ΣΕΡΡΩΝ

Τμήμα Πληροφορικής & Επικοινωνιών

Τομέας Αρχιτεκτονικής Υπολογιστών & Βιομηχανικών Εφαρμογών

ΕΡΓΑΣΤΗΡΙΟ 9

ΑΣΚΗΣΗ 9.2

Να γραφεί πρόγραμμα το οποίο θα μετατρέπει έναν αριθμό που βρίσκεται αναπτυγμένος ως δυαδικός αριθμός στις διευθύνσεις μνήμης **0208..020F** (ένα ψηφίο σε κάθε byte) σε κανονικό ακεραίο αριθμό του ενός byte που θα τοποθετείται στη διεύθυνση μνήμης **0210**.

(Υπόδειξη : αρχικά φτιάξτε μία μάσκα με τιμή 10000000 η οποία σε κάθε επανάληψη θα ολισθαίνει δεξιά και μηδενίστε έναν καταχωρητή ή απευθείας τη θέση μνήμης. Αν η κατάλληλη θέση μνήμης έχει «1» τότε προστίθεται λογικά το bit της μάσκας στον καταχωρητή/μνήμη)

0210

AB

Αποτέλεσμα : Αριθμός $(AB)_{16} = (10101011)_2$

0208	0209	020A	020B	020C	020D	020E	020F
01	00	01	00	01	00	01	01



ΤΕΙ ΣΕΡΡΩΝ

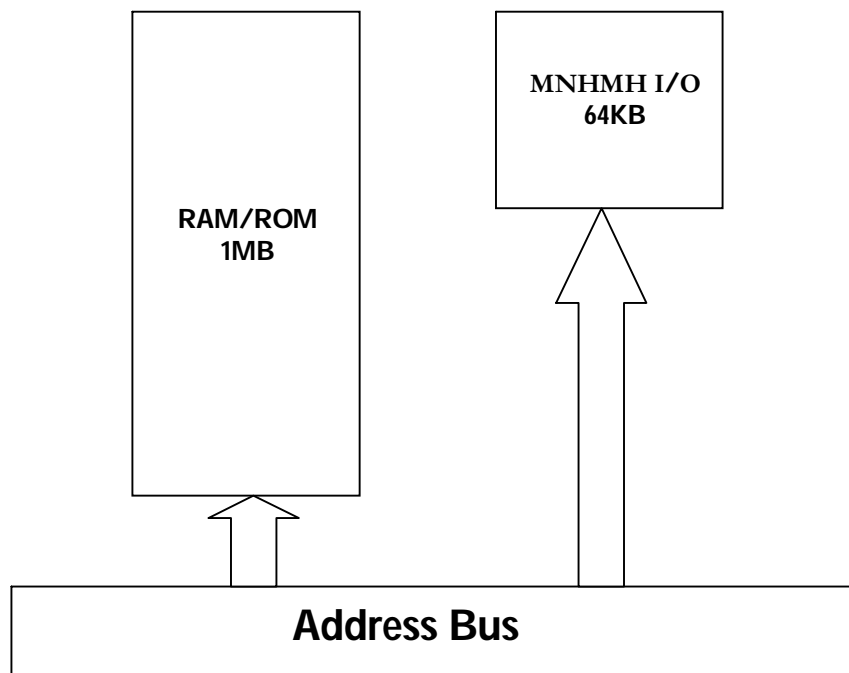
Τμήμα Πληροφορικής & Επικοινωνιών

Τομέας Αρχιτεκτονικής Υπολογιστών & Βιομηχανικών Εφαρμογών

ΕΡΓΑΣΤΗΡΙΟ 10

ΘΥΡΕΣ I/O

- Ο 8088 «βλέπει» τις περιφερειακές συσκευές του σαν διευθύνσεις μνήμης



Πως ξεχωρίζει ο 8088 για ποια από τις 2 μνήμες προορίζεται μια προσπέλαση μνήμης;

- Χρησιμοποιεί ειδικές εντολές για προσπέλαση μνήμης I/O
- Μέσω 3 σημάτων, S0, S1, S2



ΤΕΙ ΣΕΡΡΩΝ

Τμήμα Πληροφορικής & Επικοινωνιών

Τομέας Αρχιτεκτονικής Υπολογιστών & Βιομηχανικών Εφαρμογών

ΕΡΓΑΣΤΗΡΙΟ 10

ΘΥΡΕΣ I/O

S2	S2	S0	Σήμα Ελέγχου
0	0	0	Interrupt Acknowledge (Αποδοχή Διακοπής)
0	0	1	Read I/O Port (Ανάγνωση Θύρας I/O)
0	1	0	Write I/O Port (Εγγραφή Θύρας I/O)
0	1	1	Halt (Διακοπή λειτουργίας)
1	0	0	Code Access (Προσκόμιση opcode)
1	0	1	Read Memory (Ανάγνωση Μνήμης)
1	1	0	Write Memory (Εγγραφή Μνήμης)
1	1	1	Passive (Καμία Λειτουργία)

Ο 8088 μπορεί να επικοινωνήσει με 65.536 (2^{16}) διαφορετικές θύρες I/O (καταχωρητές συσκευών εισόδου/εξόδου). Για την επιλογή θύρας I/O χρησιμοποιούνται τα 16 λιγότερο σημαντικά ψηφία της διεύθυνσης (A0..A15) ενώ τα 4 περισσότερα σημαντικά (A16..A19) παραμένουν ίσα με 0.

Στον BGC-8088 υπάρχουν 8 συσκευές εισόδου/εξόδου στις οποίες έχουν παραχωρηθεί 16 διευθύνσεις για κάθε συσκευή, και άρα υπάρχουν συνολικά 128 διευθύνσεις καταχωρητών I/O. Οι διευθύνσεις αυτές είναι στην περιοχή 0FF00 .. 0FF7F. Η αντιστοιχία διευθύνσεων και συσκευών είναι η εξής :



ΤΕΙ ΣΕΡΡΩΝ

Τμήμα Πληροφορικής & Επικοινωνιών

Τομέας Αρχιτεκτονικής Υπολογιστών & Βιομηχανικών Εφαρμογών

ΕΡΓΑΣΤΗΡΙΟ 10

ΘΥΡΕΣ I/O ΣΤΟΝ BGC-8088

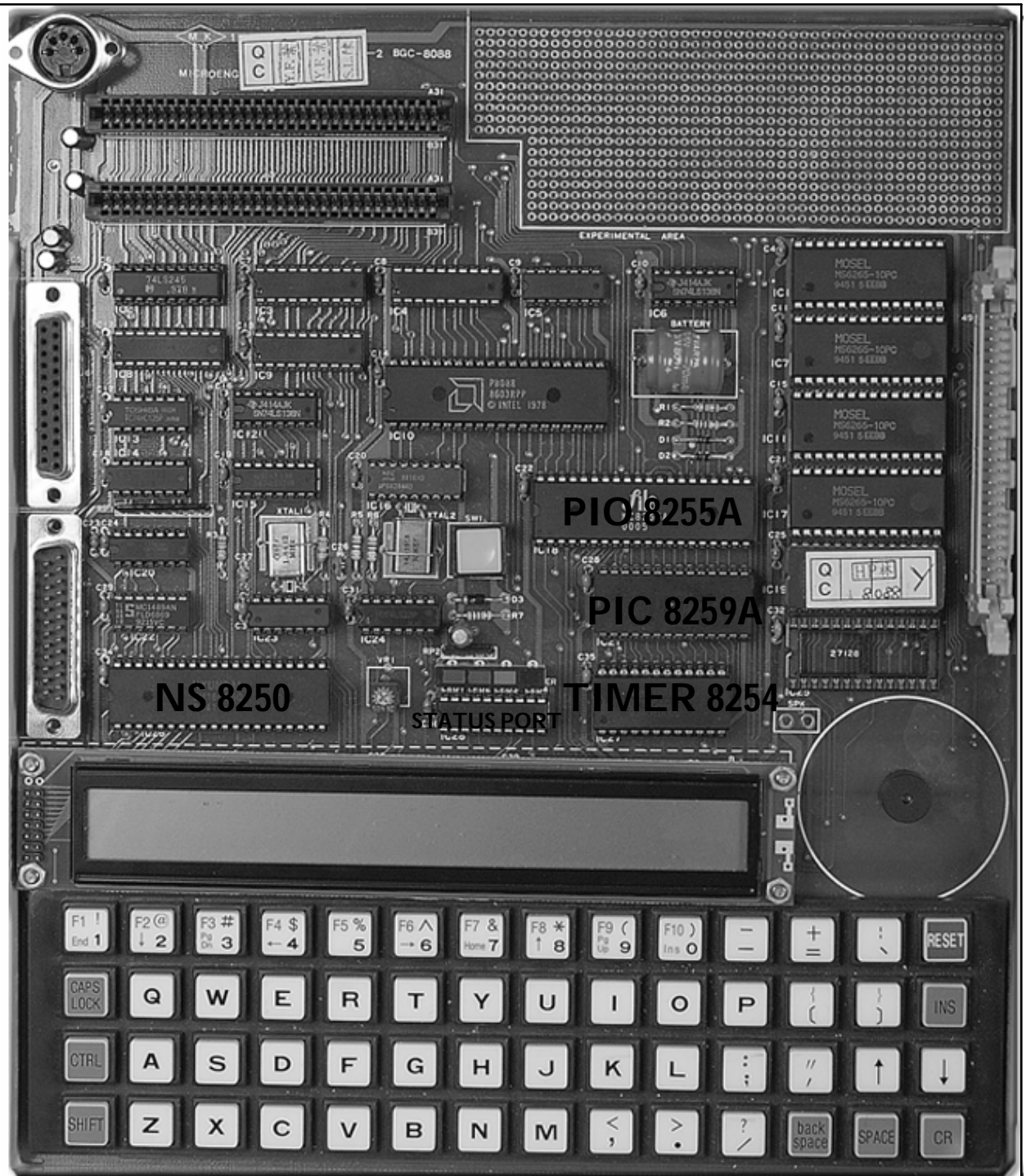
α/α	Περιοχή Δ/νσεων	Συσκευή	
1	FF00 – FF0F	8254	Programmable Timer (PT).
2	FF10 – FF1F	8255A	Programmable Peripheral Interface (PPI).
3	FF20 – FF2F	8259A	Programmable Interrupt Controller (PIC).
4	FF30 – FF3F	PRINTER	Παράλληλη θύρα
5	FF40 – FF4F	LCD	LCD Screen
6	FF50 – FF5F	8279	Keyboard controller
7	FF60 – FF6F	NS 8250	RS-232C Interface
8	FF70 – FF7F	STATUS PORT	8bit Data Latch, INS, CAPS, PRINT leds, speaker



ΤΕΙ ΣΕΡΡΩΝ

Τμήμα Πληροφορικής & Επικοινωνιών

Τομέας Αρχιτεκτονικής Υπολογιστών & Βιομηχανικών Εφαρμογών





ΤΕΙ ΣΕΡΡΩΝ

Τμήμα Πληροφορικής & Επικοινωνιών

Τομέας Αρχιτεκτονικής Υπολογιστών & Βιομηχανικών Εφαρμογών

ΕΡΓΑΣΤΗΡΙΟ 10

ΕΝΤΟΛΕΣ I/O

■ Η Εντολή IN

Είσοδος δεδομένων από θύρα I/O

■ IN AL, <αριθμός θύρας> π.χ. IN AL, 04

Διαβάζει το **byte** που βρίσκεται στην θύρα εισόδου που προσδιορίζεται από τον <αριθμό θύρας>, και το τοποθετεί στον **AL**. Η εντολή μπορεί να εφαρμοστεί μόνο για τις θύρες 0000..00FF

■ IN AL, DX π.χ. IN AL, DX

Διαβάζει το **byte** που βρίσκεται στην θύρα εισόδου που δίνουμε ως τιμή του καταχωρητή **DX**, και το τοποθετεί στον **AL**. Χρησιμοποιείται όταν έχουμε μέχρι 65.536 θύρες I/O στο σύστημα (όπως στα PC και τον BGC-8088)



ΤΕΙ ΣΕΡΡΩΝ

Τμήμα Πληροφορικής & Επικοινωνιών

Τομέας Αρχιτεκτονικής Υπολογιστών & Βιομηχανικών Εφαρμογών

ΕΡΓΑΣΤΗΡΙΟ 10

ΕΝΤΟΛΕΣ I/O

■ Η Εντολή **OUT**

Έξοδος δεδομένων προς θύρα I/O

■ **OUT** <αριθμός θύρας>, **AL** π.χ. **OUT FA,DX**

Εξάγει το **byte** που περιέχεται στον **AL**, στην θύρα που προσδιορίζεται από τον <αριθμό θύρας>. Η εντολή μπορεί να εφαρμοστεί μόνο για τις θύρες **0000..00FF**.

■ **OUT DX,AL** π.χ. **OUT DX,AL**

Εξάγει το **byte** που βρίσκεται στον **AL** στη θύρα εισόδου που δίνουμε ως τιμή του καταχωρητή **DX**. Χρησιμοποιείται όταν έχουμε μέχρι **65.536** θύρες I/O στο σύστημα (όπως στα **PC** και τον **BGC-8088**)



ΤΕΙ ΣΕΡΡΩΝ

Τμήμα Πληροφορικής & Επικοινωνιών

Τομέας Αρχιτεκτονικής Υπολογιστών & Βιομηχανικών Εφαρμογών

ΕΡΓΑΣΤΗΡΙΟ 10

ΤΟ STATUS PORT

Το Status Port αποτελείται από ένα 8-bit data latch (buffer) 74LS374 που είναι χαρτογραφημένο στην διεύθυνση FF70. Από τα 8 bits του latch χρησιμοποιούνται μόνο τα 4. Η σημασία των bits είναι η ακόλουθη :

Bit :	7	6	5	4	3	2	1	0
Σημασία:	X	X	X	X	Speaker	Print led	Ins Led	Caps lock

- Τα led ανάβουν με '0' και σβήνουν με '1' στα αντίστοιχα bit
- Το speaker παράγει ήχους όταν εναλλάσσεται '0' και '1' στο bit αυτό. Η συχνότητα εναλλαγής καθορίζει τη συχνότητα του ήχου. Η ένταση δεν ρυθμίζεται



ΤΕΙ ΣΕΡΡΩΝ

Τμήμα Πληροφορικής & Επικοινωνιών

Τομέας Αρχιτεκτονικής Υπολογιστών & Βιομηχανικών Εφαρμογών

ΕΡΓΑΣΤΗΡΙΟ 10

ΑΣΚΗΣΗ 10.1

Να αναπτυχθεί πρόγραμμα που θα παράγει έναν ήχο από το μεγαφωνάκι του BGC-8088. Ο ήχος να ξεκινάει από χαμηλή συχνότητα (μπάσος) και σταδιακά να αυξάνει συχνότητα (πρίμα). Στη συνέχεια και αφού φτάσει σε μία μέγιστη συχνότητα, να ξαναελαττώνει την συχνότητα του προς χαμηλότερες συχνότητες (ανέβασμα και κατέβασμα κλίμακας). Η διαδικασία αυτή να επαναλαμβάνεται 10 φορές.

ΑΣΚΗΣΗ 10.2

Να αναπτυχθεί πρόγραμμα που θα προκαλεί το σειριακό άναμμα και σβήσιμο των LEDES του πληκτρολογίου (CAPS, INS, PRINT) με μεταβλητό ρυθμό ο οποίος θα ξεκινάει αργά και θα επιταχύνει σταδιακά. Όταν φτάσει την μέγιστη συχνότητα θα επιβραδύνει και πάλι μέχρι το κάτω όριο (αργός ρυθμός). Η διαδικασία αυτή να επαναληφθεί 3 φορές.