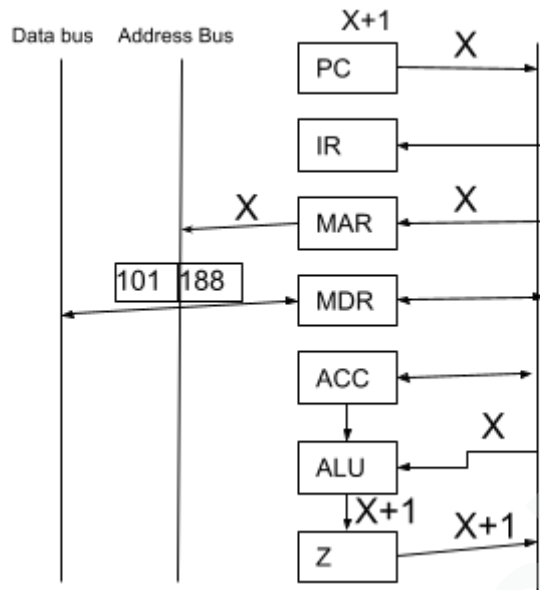


Αρχιτεκτονική Διάλεξη 6

Address :Διεύθυνση μνήμης όπου βρίσκονται τα δεδομένα της εντολής

OPCODE	Address
--------	---------



- Address→ Παράγοντας Εντολή ενός παράγοντα

- OPCODE: Κωδικός εντολής. Δείχνει την εντολή προς εκτέλεση

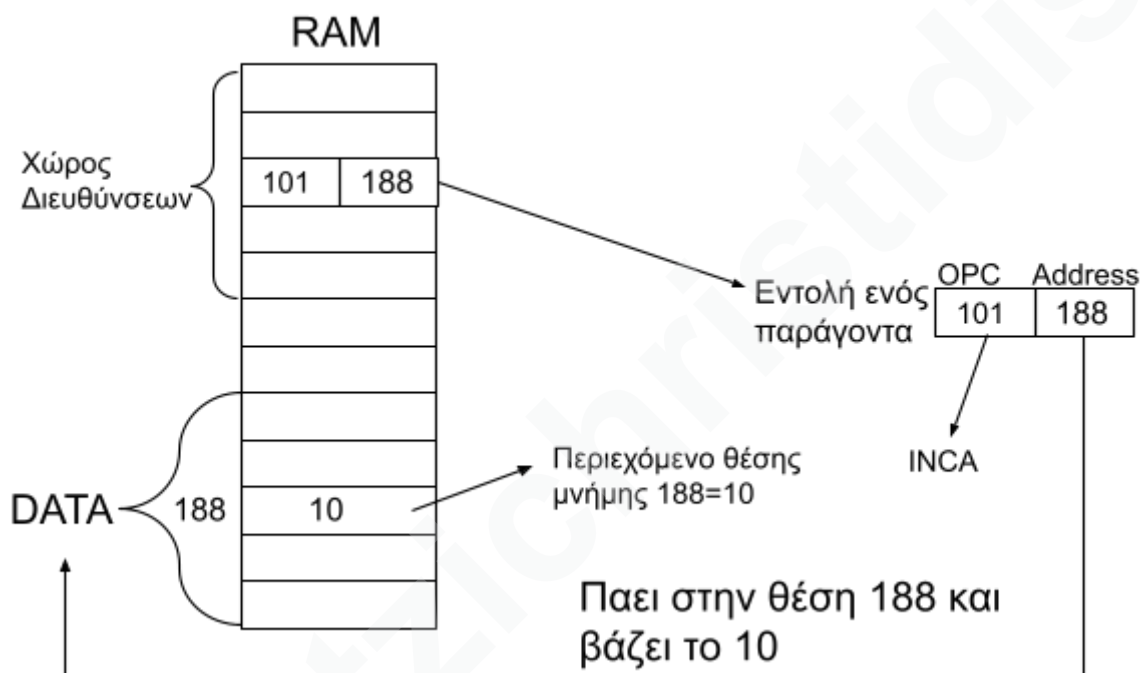
- Data Bus: μεταφέρουμε δεδομένα από την μνήμη είτε δίνουμε δεδομένα

- Address Bus: πρώτα ζητάμε την διεύθυνση από την μνήμη ώστε να πάρουμε τα δεδομένα

- 1) Εντολές ενός παράγοντα
- 2) Εντολές 2 παραγόντων
- 3) Εντολές 3 παραγόντων
- 4) Εντολές 0 παραγόντων
- 5) Εντολές σε αρχιτεκτονική με 2 διαύλους

8 εντολές CPU		LDA: Φορτώνει στον ACC τα περιεχόμενα της θέσης μνήμης A STA: Φορτώνει στη θέση μνήμης A τα περιεχόμενα τον ACC INCA: Αυξάνω τα περιεχόμενα της θέσης μνήμης A κατά 1 Εδώ έχουμε μόνο 6 Εντολές συνολικά είναι 8 με την τελική να έχει OPC 111
Εντολή	OPCODE	
LDA	000	
STA	001	
ADD	010	
AND	011	
ADD_A	100	
INCA	101	

Κώδικας	Γλώσσα Μηχανής
<pre>int A; A=A+1;</pre>	1) Όρισε θέση μνήμης για την μεταβλητή A 2) Έστω η θέση που γράφεται A είναι η 188 3) Διαβάζεται η θέση μνήμης A 4) Τα περιεχόμενα της στέλνονται στον ALU 5) Γίνεται η πράξη 6) Το αποτέλεσμα επιστρέφει στην A



Ανάκληση: ο PC δείχνει την διεύθυνση της επόμενης προς εκτέλεση εντολής

Έστω X η εντολή τότε : $\rightarrow PC=X$

Τι γίνεται στην ανάκληση ?

T0 $MAR \leftarrow PC, Z \leftarrow PC+1$

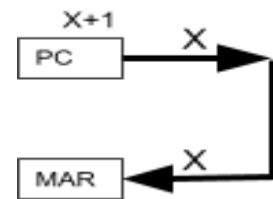
T1 $MDR \leftarrow M[MAR], PC \leftarrow Z$

T2 $IR \leftarrow MDR[OPCODE]$

- Ο Pc ξέρει ότι θα πρέπει να ξεκινήσει από το X
- Πρέπει η εντολή που βρίσκεται στην διεύθυνση X, να μεταφερθεί στην CPU, ώστε ο IR να πάρει το OPCODE

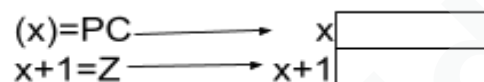
T0: $MAR \leftarrow PC, Z \leftarrow PC+1$

Στο T0 ο PC δίνει τη διεύθυνση X στον MAR για να ζητηθεί η ανάγνωση της, από το σήμα σελίδα 1 →



Όποτε θέλουμε να διαβάσουμε να γράψουμε από τη διεύθυνση μνήμης η διεύθυνση πρέπει να δοθεί από τον MAR

Μετά ο $MAR \leftarrow X, Z \leftarrow X+1$ → Αυτό γίνεται ώστε να ξέρω την επομενη εντολή που θα δείξει ο PC



T1: $MDR \leftarrow M[MAR], PC \leftarrow Z$

Ο MDR λαμβάνει μέσω του διαύλου δεδομένων την λέξη μνήμης

$MDR \leftarrow M[X]$

$PC \leftarrow Z$ ($Z=x+1$)

T1: MDR ←

101	188
-----	-----

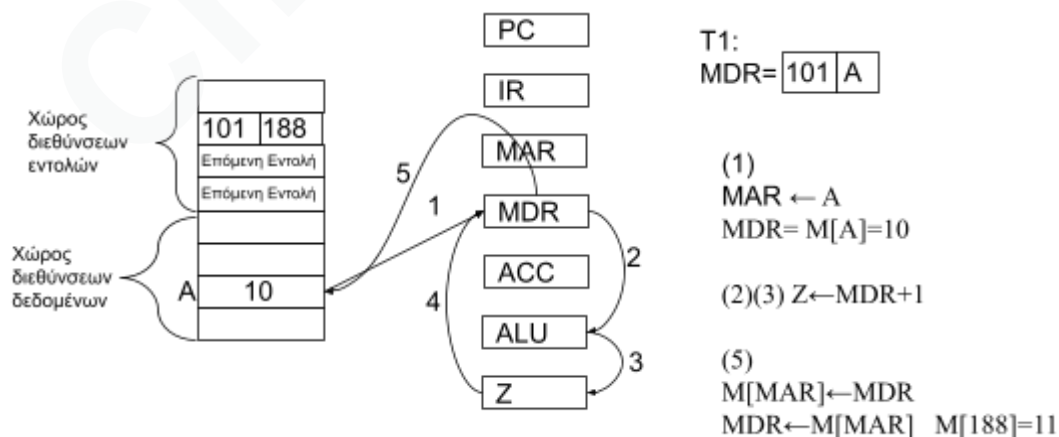
, $PC \leftarrow X+1$

T2:

Ο IR διαβάζει απο το MDR το OPCODE δηλαδή το 101 → IR

Εδώ πλέον ξέρει η CPU τι να κάνει

Η CPU λέει στην μονάδα ελέγχου ότι πρέπει να εκτελεστεί η INC



Εξήγηση

(1) ο MAR να πάρει την διεύθυνση A, για να ζητήσει απο την μνήμη (αποκωδικοποίηση) . Τα περιεχόμενα της θέσης μνήμης $A \rightarrow MDR$

Εκτέλεση

$MAR \leftarrow MDR(\text{Address})$

$MDR \leftarrow M[MAR]$

(2)(3) $Z \leftarrow MDR+1$

ΔΕΝ ΓΡΑΦΩ $ALU \leftarrow MDR$ ΓΙΑΤΙ ΔΕΝ ΕΙΝΑΙ ΚΑΤΑΧΩΡΗΤΗΣ Η ALU ΟΥΣΙΑΣΤΙΚΑ Η ALU ΕΙΝΑΙ ΤΟ “+” ΣΤΟ $Z \leftarrow MDR+1$

(5) Δίνουμε την διεύθυνση A στο MAR

$M[MAR] \leftarrow MDR :$

$MDR \leftarrow M[MAR] \quad M[188]=11$

Ανάγνωση από θέση μνήμης στην οποία δείχνει ο MAR

1) Ενημέρωση του MAR

2) $MDR \leftarrow M[MAR]$

Εγγραφή :

1) Ενημέρωση MAR

2) $M[MAR] \leftarrow MDR$

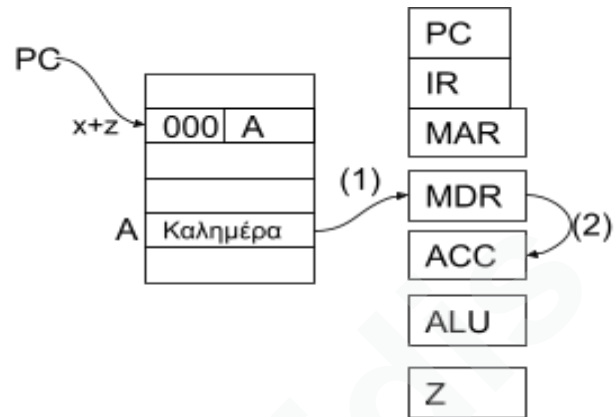
Εντολή INC
T0: $MAR \leftarrow PC, Z \leftarrow PC+1$
T1: $MDR \leftarrow M[MAR], PC \leftarrow Z$
T2: $IR \leftarrow MDR$ (OPCODE)
T3: $MAR \leftarrow MDR(\text{ADDRESS})$
T4: $MDR \leftarrow M[MAR]$
T5: $Z \leftarrow MDR+1$
T6: $MDR \leftarrow Z$
T7: $M[MAR] \leftarrow MDR$

} Ανάκληση

ΛΙΣΤΑ ΕΝΤΟΛΩΝ :

- **LDA/ LOAD:** OPCODE=000

Διαβάζει μια λέξη μνήμης A
και την φορτώνει στο
Accumulator(ACC)



Ανάκληση :

T0: $MAR \leftarrow X+Z, Z \leftarrow X+3$

T1: $MDR \leftarrow 000|A \quad PC \leftarrow X+3$

T2: $IR: 000$

Αφου $IR=000$ ξέρουμε οτι θα
εκτελεστεί η LDA

T3: $MAR \leftarrow MDR(\text{Address}) \mid MAR \leftarrow A$

T4: $MDR \leftarrow M[MAR] \mid MDR \leftarrow M[A]$

T5: $ACC \leftarrow MDR \mid ACC \leftarrow \text{"Καλημέρα"}$

- **STA/Store** OPCODE= 001

Φορτώνει στην A τα περιεχόμενα του ACC

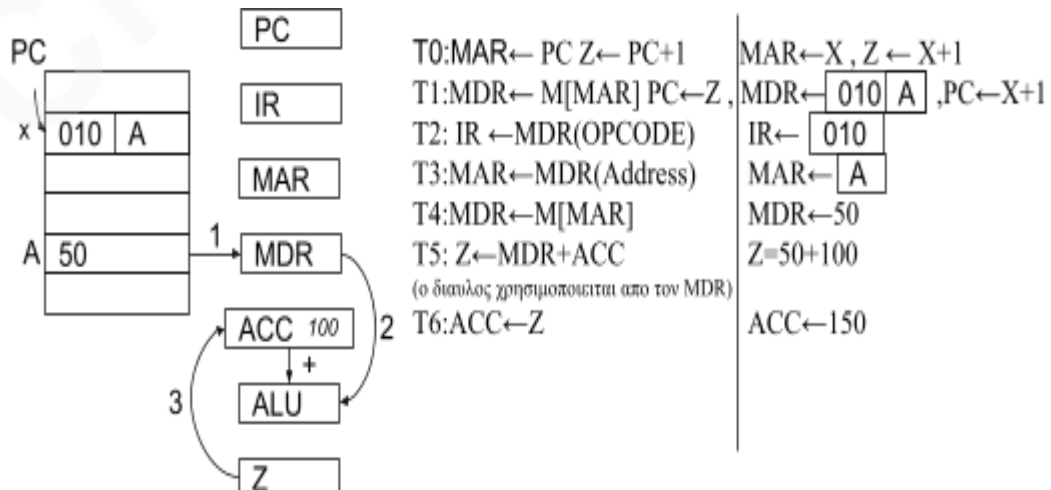
T3: $MAR \leftarrow MDR(\text{Address})$

T4: $MDR \leftarrow ACC$

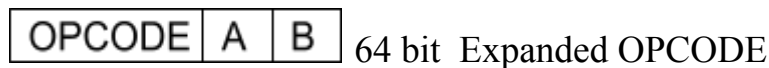
T5: $M[MAR] \leftarrow MDR$

- **ADD** OPCODE= 010

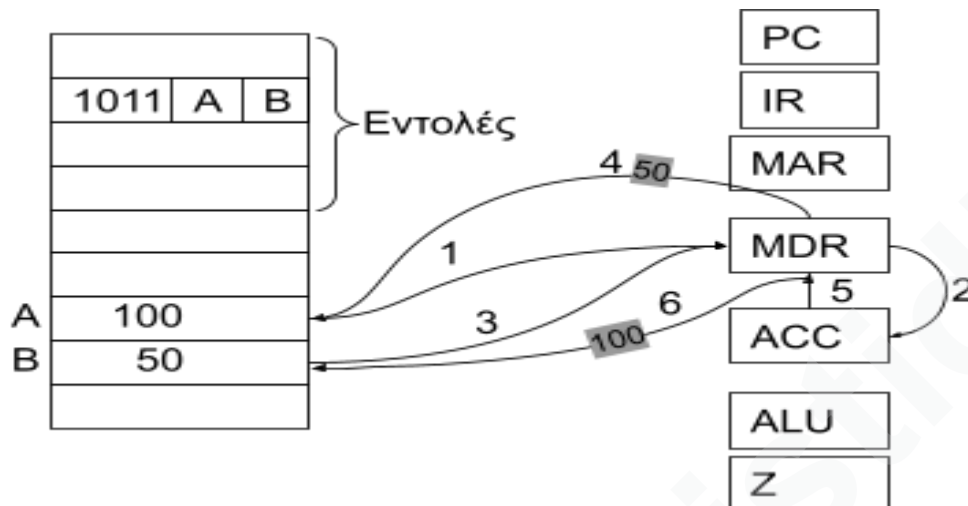
Προσθέτει τα περιεχόμενα της διεύθυνσης μνήμης A με τα περιεχόμενα
του ACC και αποθηκεύει τα αποτελέσματα στον ACC



Εντολές 2 παραγόντων :



Swar A,B : Αντιστρέφει τα περιεχόμενα των θέσεων μνήμης A και B, χρησιμοποιώντας ως ενδιάμεση μνήμη τον ACC.



- 1)Φέρνω την A στον MDR(χρήση MAR)
- 2)Αποθηκεύω προσωρινά στην ACC
- 3)Φέρνω την B στον MDR(χρήση MAR)
- 4)Στέλνω την B στην A →MAR
- 5)Φέρνω την A στον MDR
- 6)Στέλνω την A στην B →MAR

