

# Οργάνωση Υπολογιστών

Συστήματα Υπολογιστών

Παναγιώτης Παπαδημητρίου

[papadimitriou@uom.edu.gr](mailto:papadimitriou@uom.edu.gr)

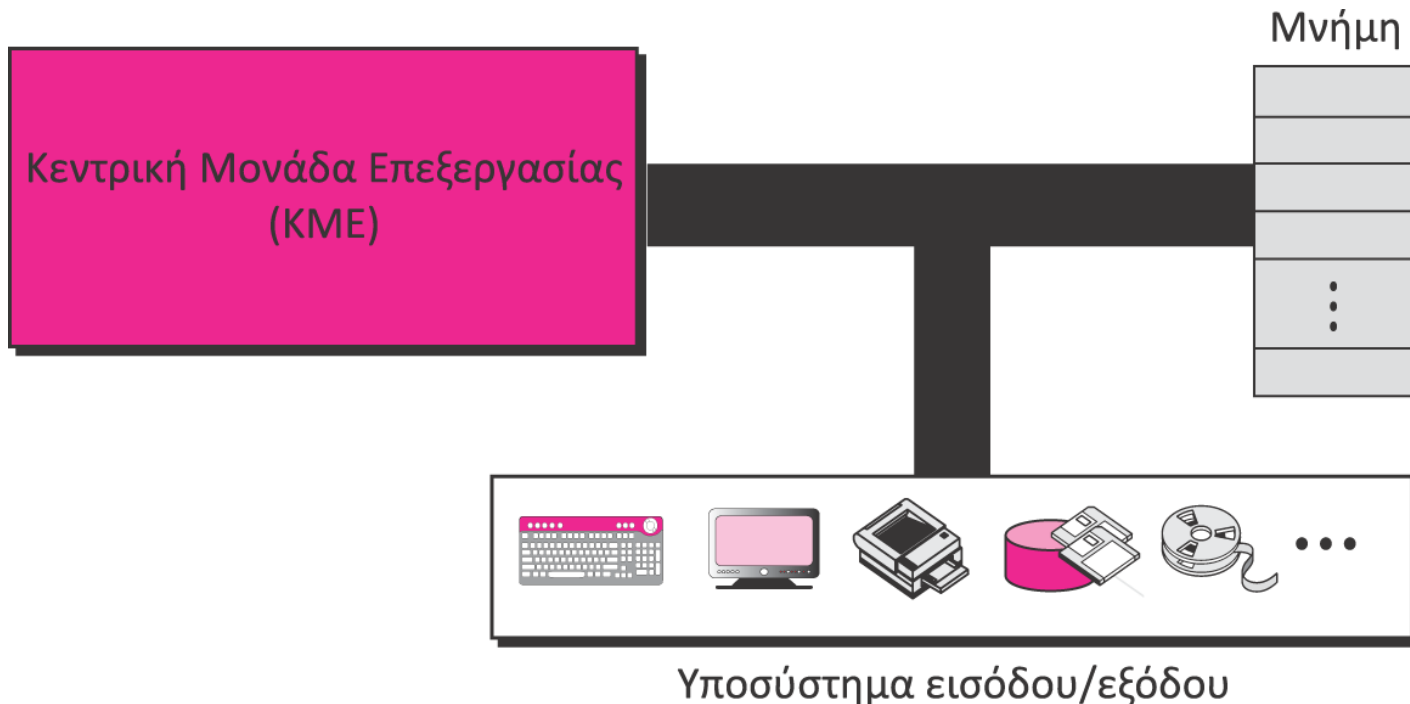
<https://sites.google.com/site/panagpapadimitriou/>

# Στοιχεία Συστημάτων Υπολογιστών

- **Υλικό:** παρέχει τους βασικούς υπολογιστικούς πόρους (επεξεργαστής, μνήμη, συσκευές εισόδου-εξόδου)
- **Λειτουργικό σύστημα:** ελέγχει και συντονίζει τη χρήση του υλικού μεταξύ των διαφόρων προγραμμάτων των χρηστών
- **Προγράμματα εφαρμογών:** καθορίζουν τους τρόπους με τους οποίους χρησιμοποιούνται οι πόροι για τη διεκπεραίωση εργασιών των χρηστών

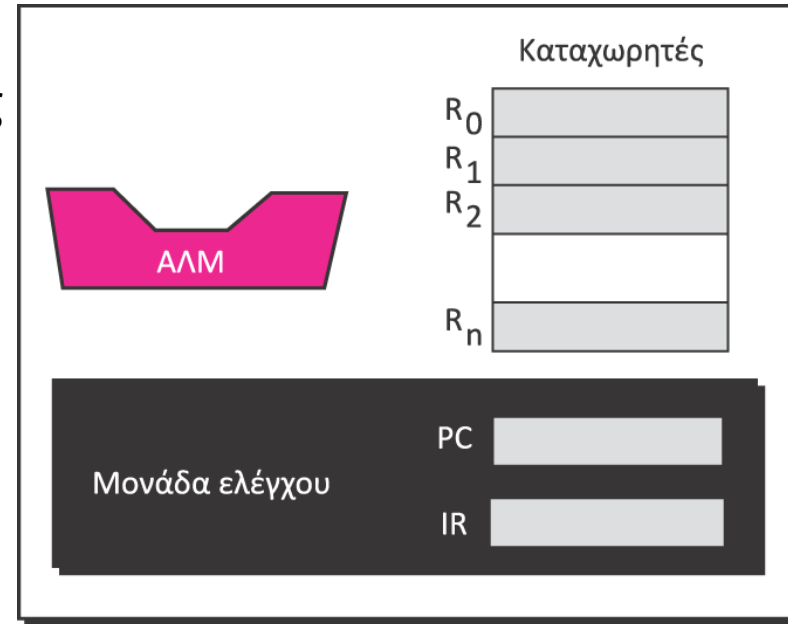
# Υλικό Υπολογιστών

- Βασικά τμήματα του υλικού του υπολογιστή:
  - Κεντρική Μονάδα Επεξεργασίας (ΚΜΕ)
  - Κεντρική Μνήμη
  - Συσκευές Εισόδου/Εξόδου (Ε/Ε)



# Κεντρική Μονάδα Επεξεργασίας (ΚΜΕ)

- Αριθμητική και Λογική Μονάδα (ΑΛΜ)
  - Εκτελεί αριθμητικές, λογικές και πράξεις μετατόπισης (ολίσθησης)
- Καταχωρητές
  - Θέσεις προσωρινής αποθήκευσης (π.χ. για την εκτέλεση πράξεων)
- Μονάδα Ελέγχου
  - Ελέγχει τη λειτουργία του κάθε υποσυστήματος της ΚΜΕ



Κεντρική Μονάδα Επεξεργασίας (ΚΜΕ)

# Καταχωρητές

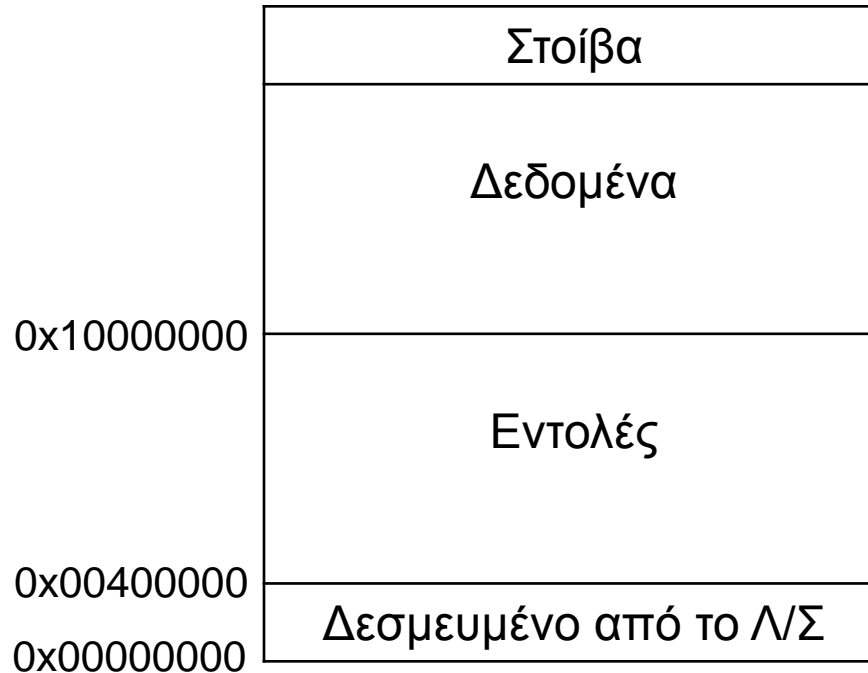
- Καταχωρητές δεδομένων:
  - Αποθηκεύουν δεδομένα εισόδου και αποτελέσματα πράξεων
- Καταχωρητής εντολών (Instruction Register - IR):
  - Αποθηκεύει την επόμενη εντολή που θα εκτελέσει η ΚΜΕ
- Μετρητής προγράμματος (Program Counter – PC):
  - Δείχνει στη διεύθυνση μνήμης όπου είναι αποθηκευμένη η επόμενη εντολή

# Κεντρική Μνήμη (RAM)

- Η κεντρική μνήμη είναι προσωρινός αποθηκευτικός χώρος για τις εντολές ενός προγράμματος και τα δεδομένα που αυτό διαχειρίζεται
  - Αποτελείται από ένα σύνολο θέσεων αποθήκευσης σε καθένα από το οποίο αντιστοιχεί μία μοναδική διεύθυνση

	Μήκος λέξης
	
0x1005	0 1 1 0 1 1 0 0 1 0 1 1 0 0 0 1
0x1004	1 1 0 0 0 1 1 1 1 0 1 0 0 1 1 1
0x1003	0 0 1 1 1 1 0 0 1 0 1 1 0 0 0 0
0x1002	1 1 0 0 1 1 1 0 1 0 0 0 0 0 0 1
0x1001	0 1 0 0 1 1 0 0 0 0 1 1 0 1 0 1
0x1000	1 0 1 0 1 1 1 1 1 0 1 1 0 0 0 0
Διεύθυνση	Περιεχόμενο

# Οργάνωση Κεντρικής Μνήμης



- Η στοίβα αποθηκεύει τη διεύθυνση επιστροφής (στο κύριο πρόγραμμα), όταν ολοκληρωθεί η εκτέλεση μίας ρουτίνας του προγράμματος

# Κρυφή Μνήμη (Cache)

- Η κρυφή μνήμη είναι ενσωματωμένη στην ΚΜΕ και επιτρέπει την αποθήκευση και ταχύτατη πρόσβαση σε ένα μικρό όγκο δεδομένων (πολύ πιο γρήγορα από στην κύρια μνήμη)
  - Τα δεδομένα αναζητούνται πρώτα στην κρυφή μνήμη και (αν δεν βρεθούν) μετά στην κεντρική μνήμη
  - Οι σύγχρονες ΚΜΕ έχουν κρυφή μνήμη 2 ή 3 επιπέδων (L1, L2, L3)

Πυρήνας 1	Πυρήνας 2	Πυρήνας 3	Πυρήνας 4
L1	L1	L1	L1
L2		L2	
L3			

Κρυφή μνήμη 3 επιπέδων σε 4-πύρηνο επεξεργαστή

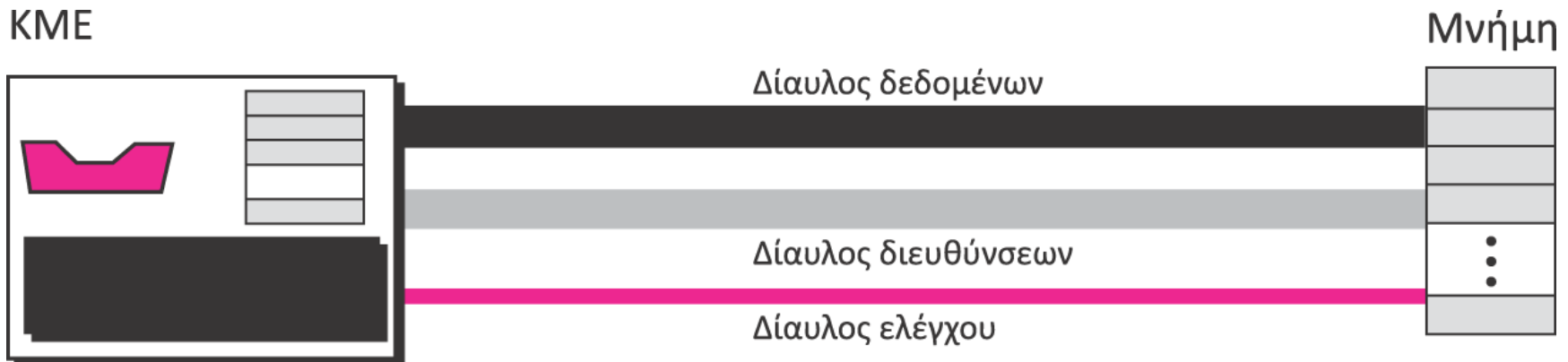


# Ιεραρχία Μνήμης



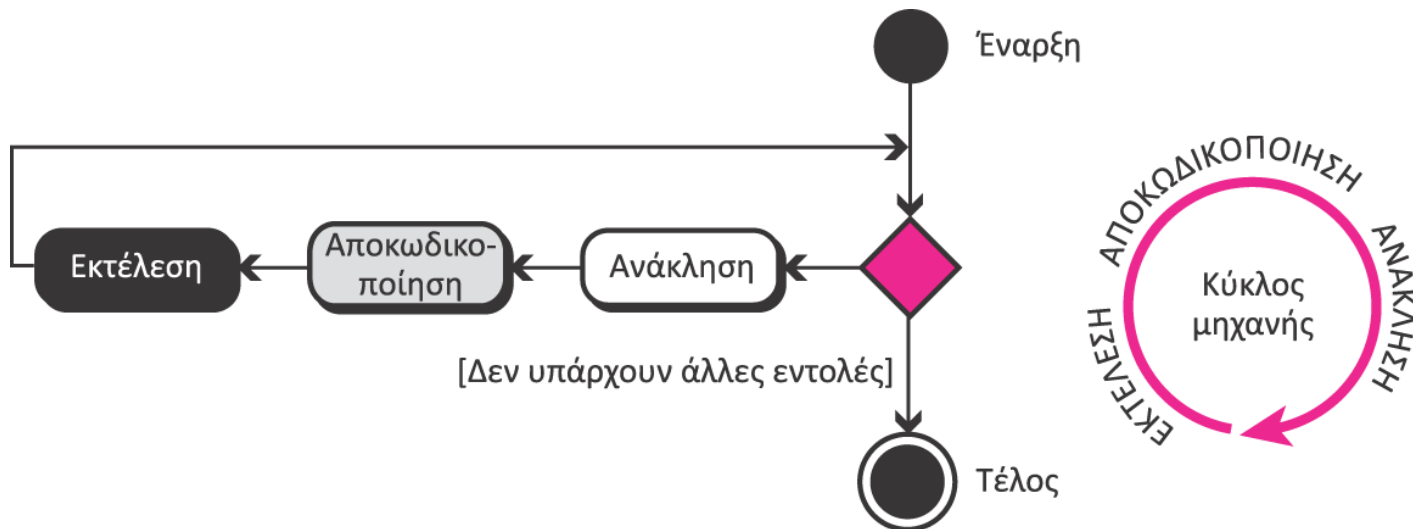
# ΕΠΙΚΟΙΝΩΝΙΑ ΚΜΕ με Μνήμη

- Δίαυλος δεδομένων:
  - Το εύρος του διαύλου καθορίζεται από το μήκος της λέξης (π.χ. 32 bits για 32-bit λέξη)
- Δίαυλος διευθύνσεων:
  - Το εύρος του διαύλου καθορίζεται από το σύνολο θέσεων της μνήμης ( $n$  bits για  $2^n$  θέσεις μνήμης)
- Δίαυλος ελέγχου:
  - Το εύρος του διαύλου καθορίζεται από το σύνολο των εντολών ελέγχου ( $m$  bits για  $2^m$  εντολές)

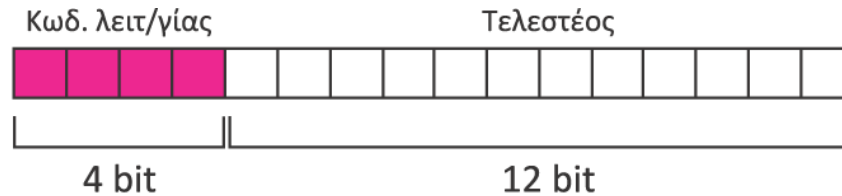


# Κύκλος Μηχανής

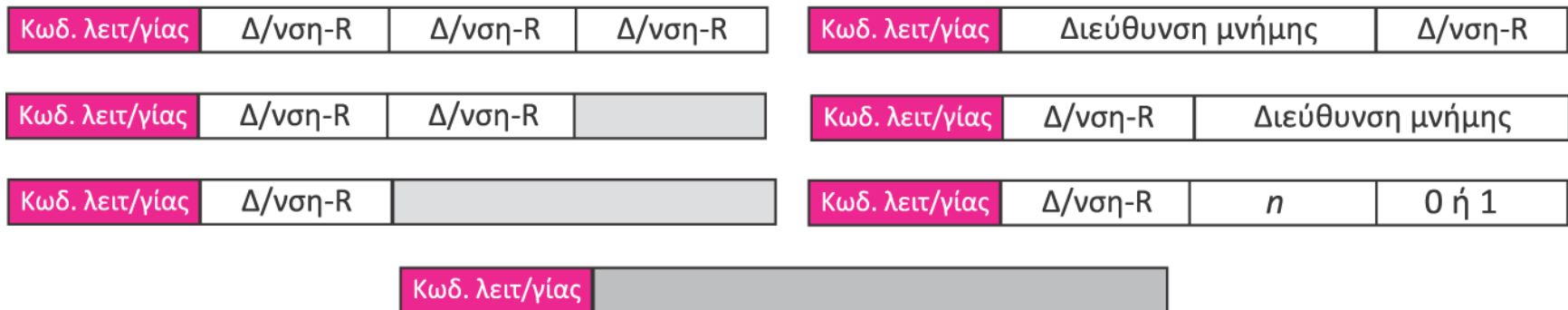
- Ανάκληση:
  - Αντιγράφεται η επόμενη εντολή (όπου δείχνει ο μετρητής προγράμματος) στον καταχωρητή εντολών
  - Αυξάνεται ο μετρητής προγράμματος κατά 1
- Αποκωδικοποίηση:
  - Αποκωδικοποιείται η εντολή (π.χ. LOAD, ADD) και οι τελεστές της (π.χ. καταχωρητές δεδομένων, διευθύνσεις μνήμης)



# Παράδειγμα Μορφής Εντολών



## α. Μορφή εντολής



## β. Τύποι εντολών

- Η μορφή και οι τύποι εντολών διαφέρουν μεταξύ αρχιτεκτονικών υπολογιστών (π.χ. x86, MIPS)

# Παράδειγμα Εντολών

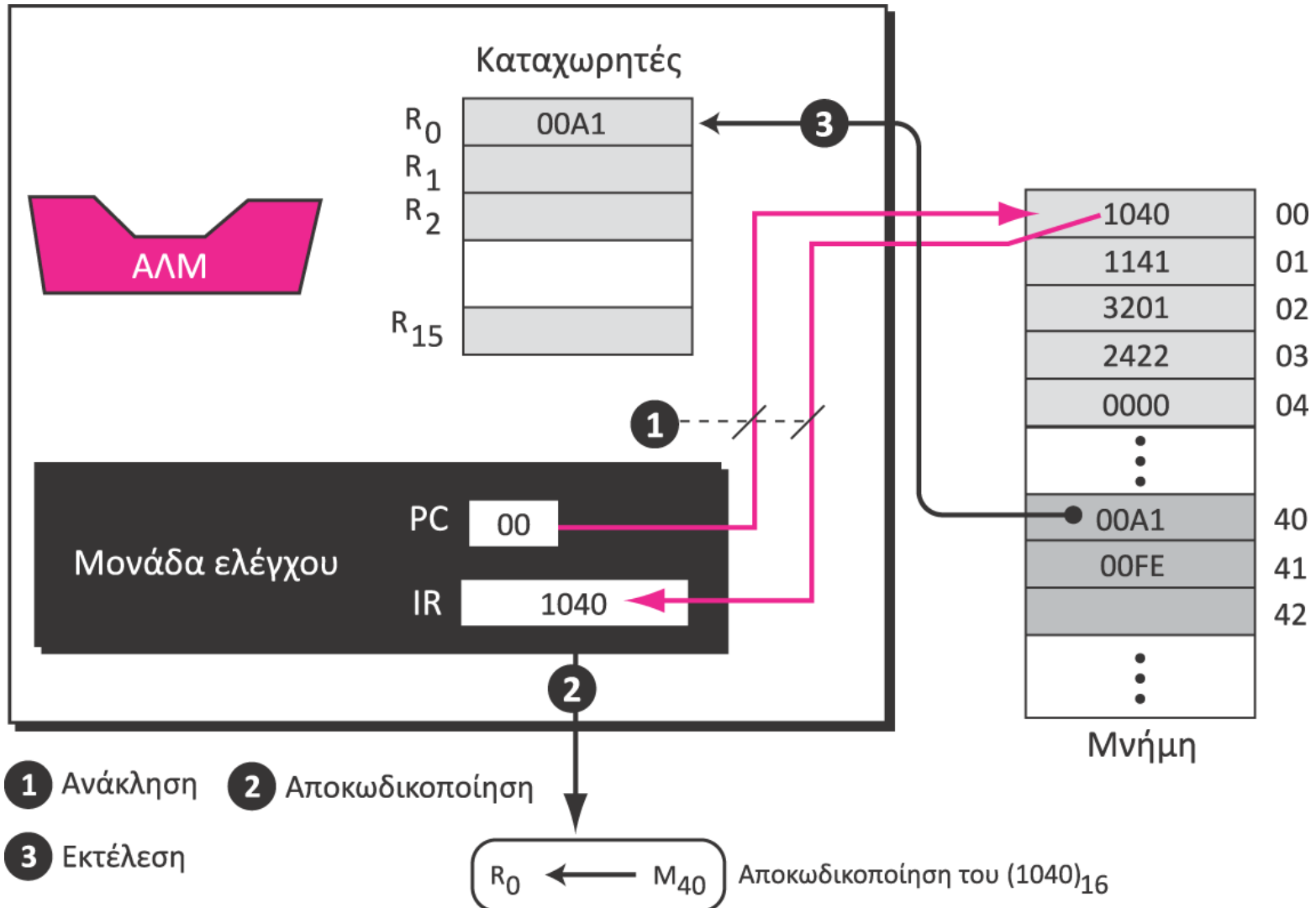
Εντολή	Κωδ.	Τελεστέοι			Ενέργεια
	$d_1$	$l_2$	$l_3$	$d_4$	
HALT	0				Διακόπτει την εκτέλεση του προγράμματος
LOAD	1	$R_D$	$M_S$		$R_D \leftarrow M_S$
STORE	2	$M_D$	$R_S$		$M_D \leftarrow R_S$
ADDI	3	$R_D$	$R_{S1}$	$R_{S2}$	$R_D \leftarrow R_{S1} + R_{S2}$
ADDF	4	$R_D$	$R_{S1}$	$R_{S2}$	$R_D \leftarrow R_{S1} + R_{S2}$
MOVE	5	$R_D$	$R_S$		$R_D \leftarrow R_S$
NOT	6	$R_D$	$R_S$		$R_D \leftarrow \overline{R_S}$
AND	7	$R_D$	$R_{S1}$	$R_{S2}$	$R_D \leftarrow R_{S1} \text{ AND } R_{S2}$
OR	8	$R_D$	$R_{S1}$	$R_{S2}$	$R_D \leftarrow R_{S1} \text{ OR } R_{S2}$
XOR	9	$R_D$	$R_{S1}$	$R_{S2}$	$R_D \leftarrow R_{S1} \text{ XOR } R_{S2}$
INC	A	R			$R \leftarrow R + 1$
DEC	B	R			$R \leftarrow R - 1$
ROTATE	C	R	n	0 ή 1	$\text{Rot}_n R$
JUMP	D	R	n		Αν $R_0 \neq R$ τότε $PC = n$ , ειδάλλως συνέχισε

**Υπόμνημα:**  $R_S, R_{S1}, R_{S2}$ : Δεκαεξαδική διεύθυνση των καταχωρητών προέλευσης  
 $R_D$ : Δεκαεξαδική διεύθυνση του καταχωρητή προορισμού  
 $M_S$ : Δεκαεξαδική διεύθυνση της θέσης μνήμης προέλευσης  
 $M_D$ : Δεκαεξαδική διεύθυνση της θέσης μνήμης προορισμού  
 $n$ : δεκαεξαδικός αριθμός  
 $d_1, d_2, d_3, d_4$ : 1ο, 2ο, 3ο, και 4ο δεκαεξαδικό ψηφίο

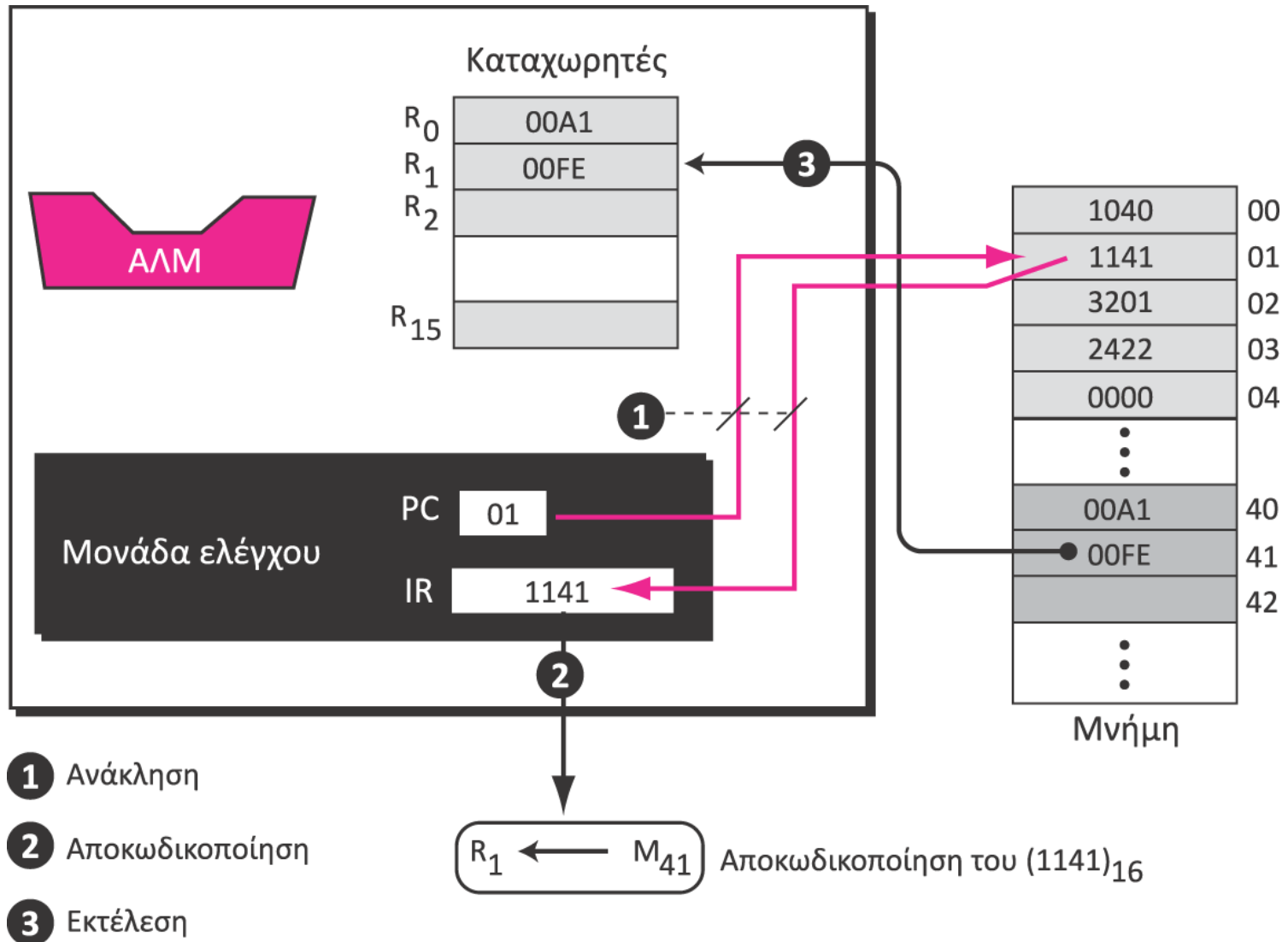
# Εκτέλεση Προγράμματος

- Πρόσθεση δύο αριθμών σε γλώσσα μηχανής:
  1. Φόρτωση 1<sup>ου</sup> αριθμού από τη μνήμη σε καταχωρητή ( $R_0$ )
  2. Φόρτωση 2<sup>ου</sup> αριθμού από τη μνήμη σε καταχωρητή ( $R_1$ )
  3. Πρόσθεση των τιμών των δύο καταχωρητών ( $R_0, R_1$ ) και αποθήκευση του αποτελέσματος σε τρίτο καταχωρητή ( $R_2$ )
  4. Αποθήκευση της τιμής του τρίτου καταχωρητή ( $R_2$ ) στη μνήμη
  5. Τερματισμός της εκτέλεσης

# Εκτέλεση Εντολής 1 (LOAD)

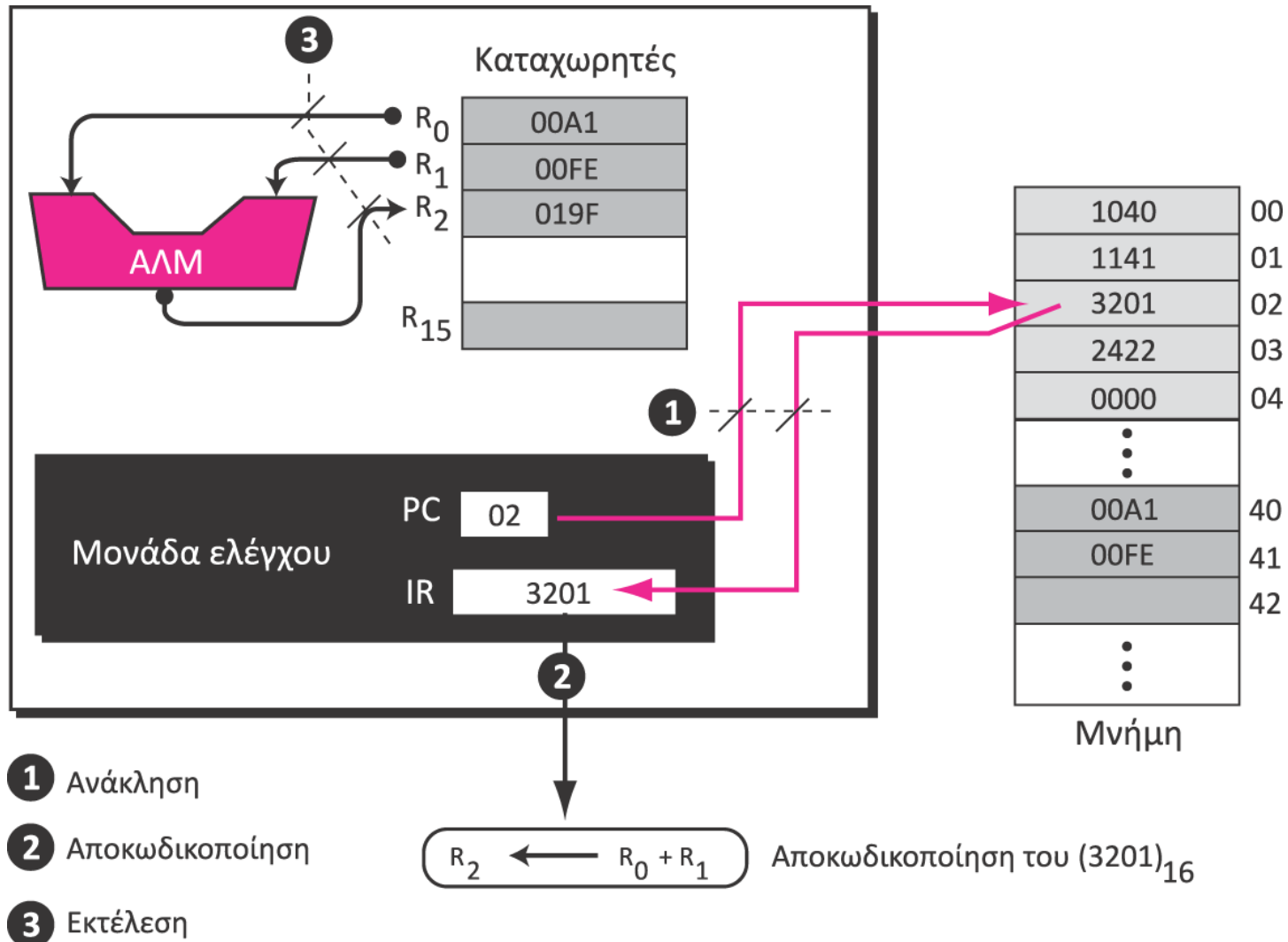


# Εκτέλεση Εντολής 2 (LOAD)

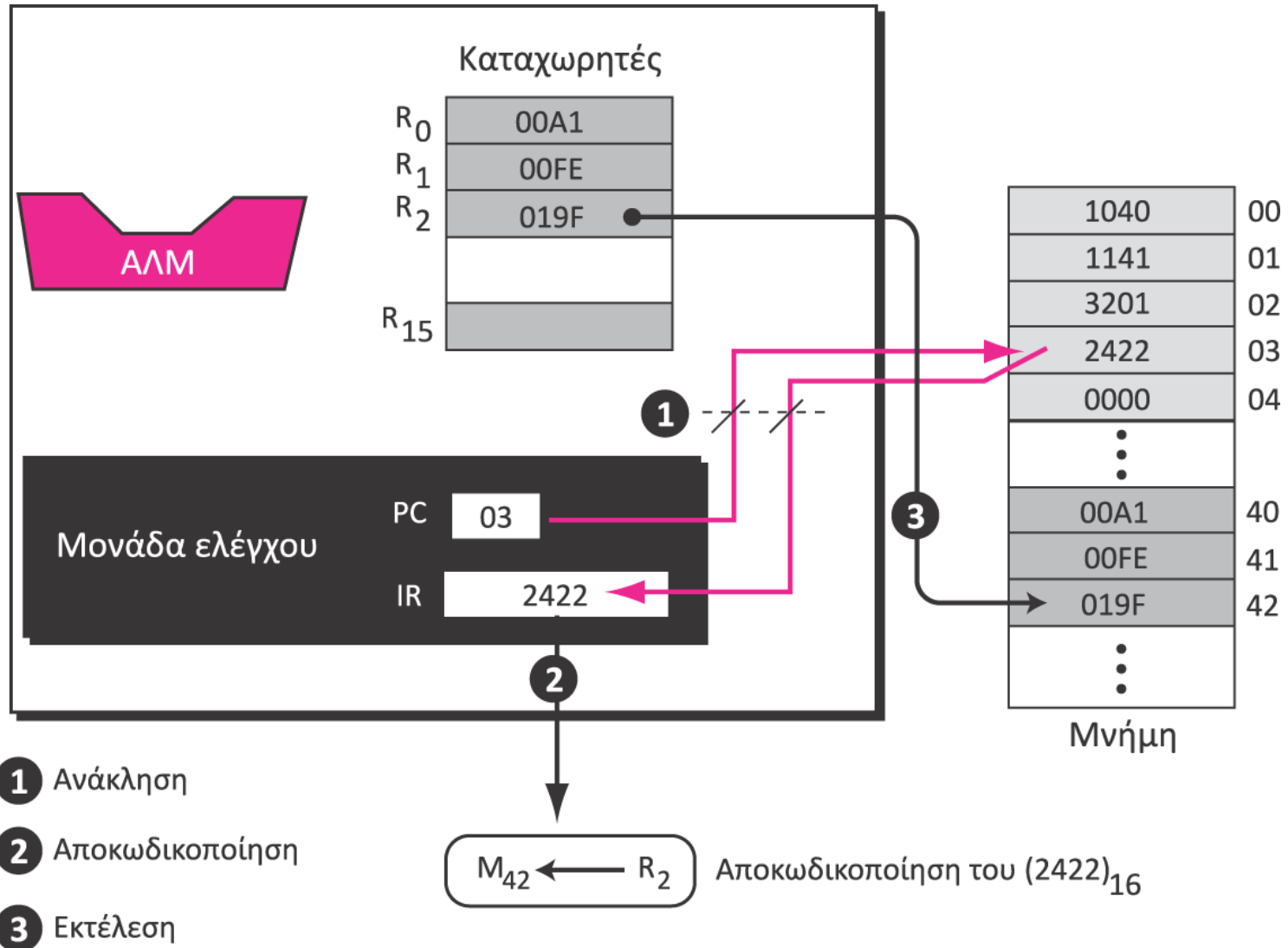




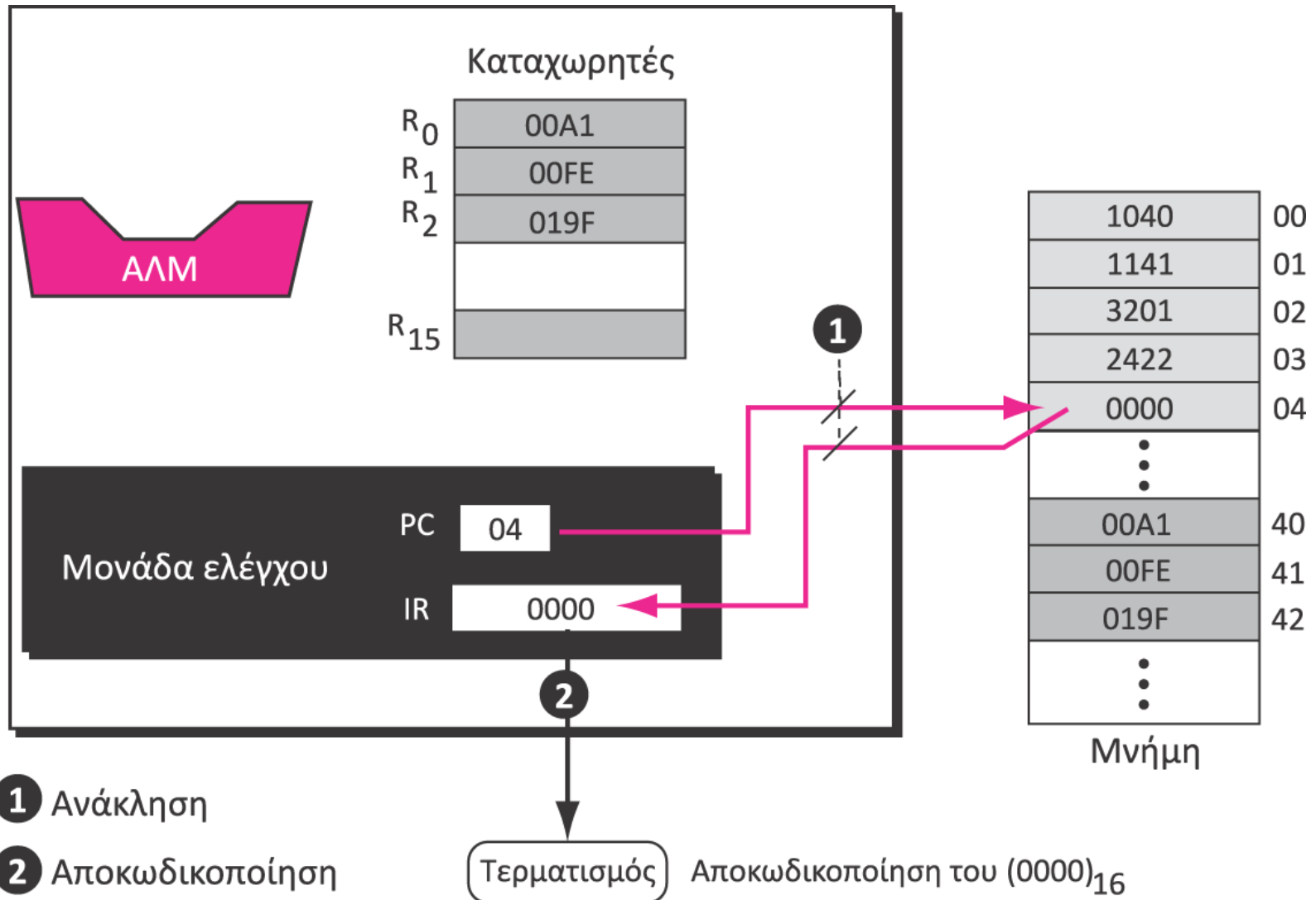
# Εκτέλεση Εντολής 3 (ADD)



# Εκτέλεση Εντολής 4 (STORE)



# Εκτέλεση Εντολής 5 (HALT)



Ερωτήσεις