



ΟΡΓΑΝΩΣΗ ΚΑΙ ΣΧΕΔΙΑΣΗ ΥΠΟΛΟΓΙΣΤΩΝ

Η διασύνδεση υλικού και λογισμικού



Κεφάλαιο 4

Ο επεξεργαστής

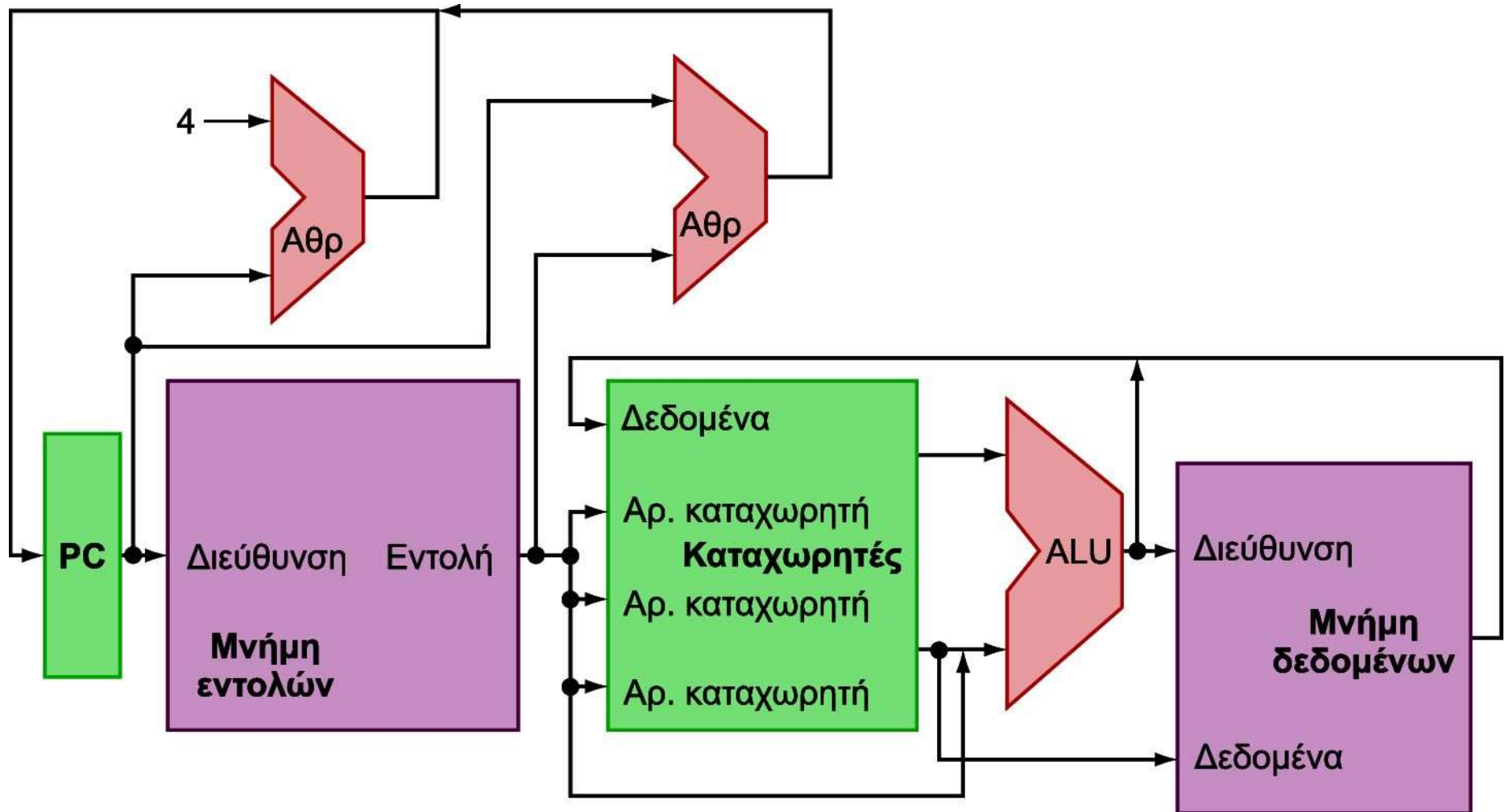
Εισαγωγή

- Παράγοντες που επηρεάζουν τις επιδόσεις της CPU
 - Πλήθος εντολών
 - Καθορίζεται από την ISA και τον μεταγλωττιστή
 - Κύκλοι ανά εντολή (clocks per instruction, CPI) και χρόνος κύκλου ρολογιού
 - Καθορίζεται από το υλικό της CPU
- Θα εξετάσουμε δύο υλοποιήσεις RISC-V
 - Μια απλουστευμένη έκδοση
 - Μια πιο ρεαλιστική υλοποίηση με διοχέτευση
- Απλό υποσύνολο, δείχνει τις περισσότερες πτυχές
 - Εντολές αναφοράς μνήμης: lw, sw
 - Αριθμητικές-λογικές εντολές: add, sub, and, or
 - Εντολές μεταφοράς του ελέγχου: beq

Εκτέλεση εντολών

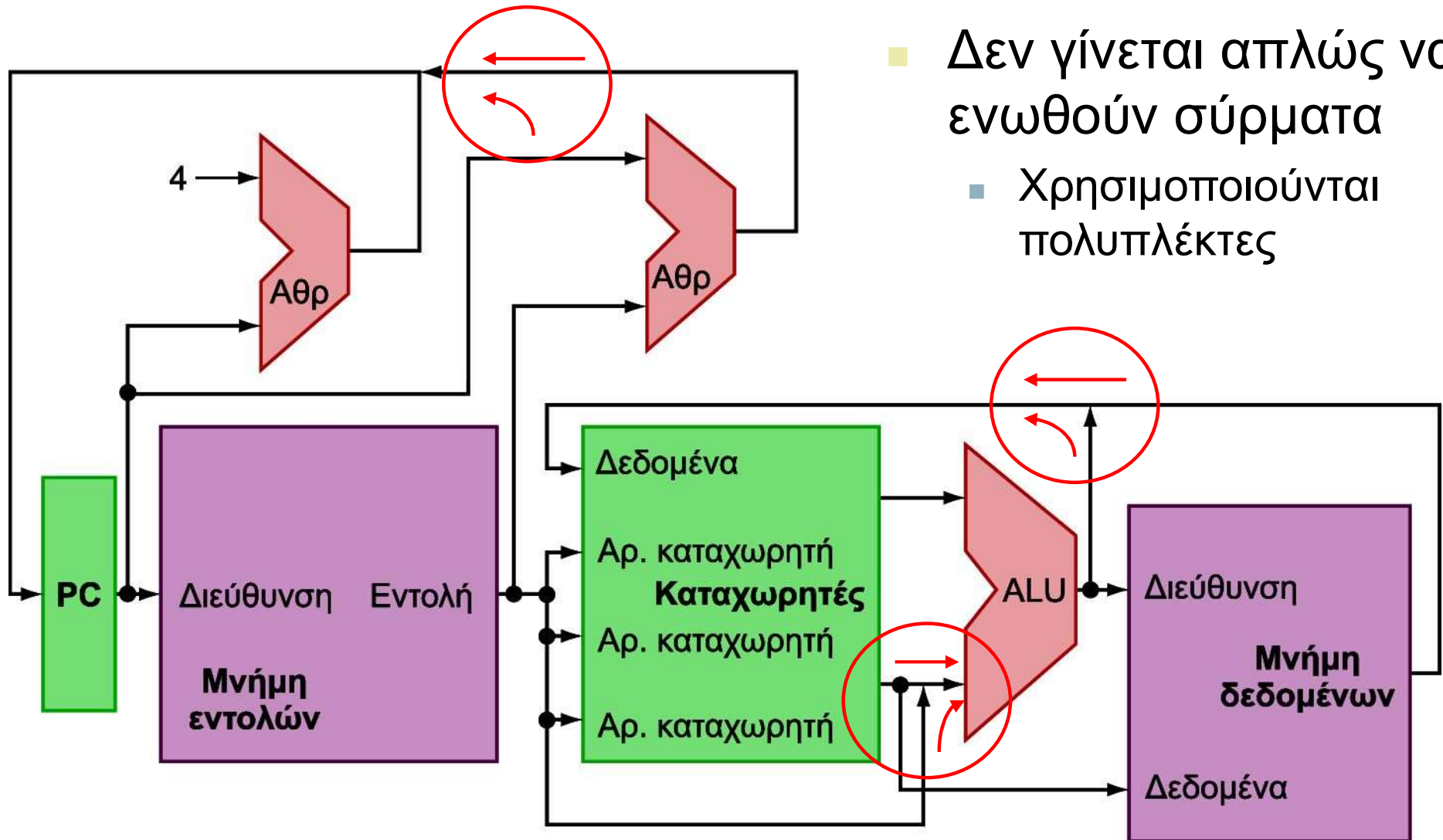
- Μετρητής προγράμματος (PC) → μνήμη εντολών, προσκόμιση εντολής
- Αριθμοί καταχωρητών → αρχείο καταχωρητών, ανάγνωση καταχωρητών
- Ανάλογα με την κατηγορία της εντολής
 - Χρήση της ALU για να υπολογιστεί
 - Το αριθμητικό αποτέλεσμα
 - Η διεύθυνση μνήμης για φόρτωση/αποθήκευση
 - Σύγκριση διακλάδωσης
 - Προσπέλαση της μνήμης δεδομένων για φόρτωση/αποθήκευση
 - $PC \leftarrow \text{διεύθυνση προορισμού ή } PC + 4$

Επισκόπηση της CPU

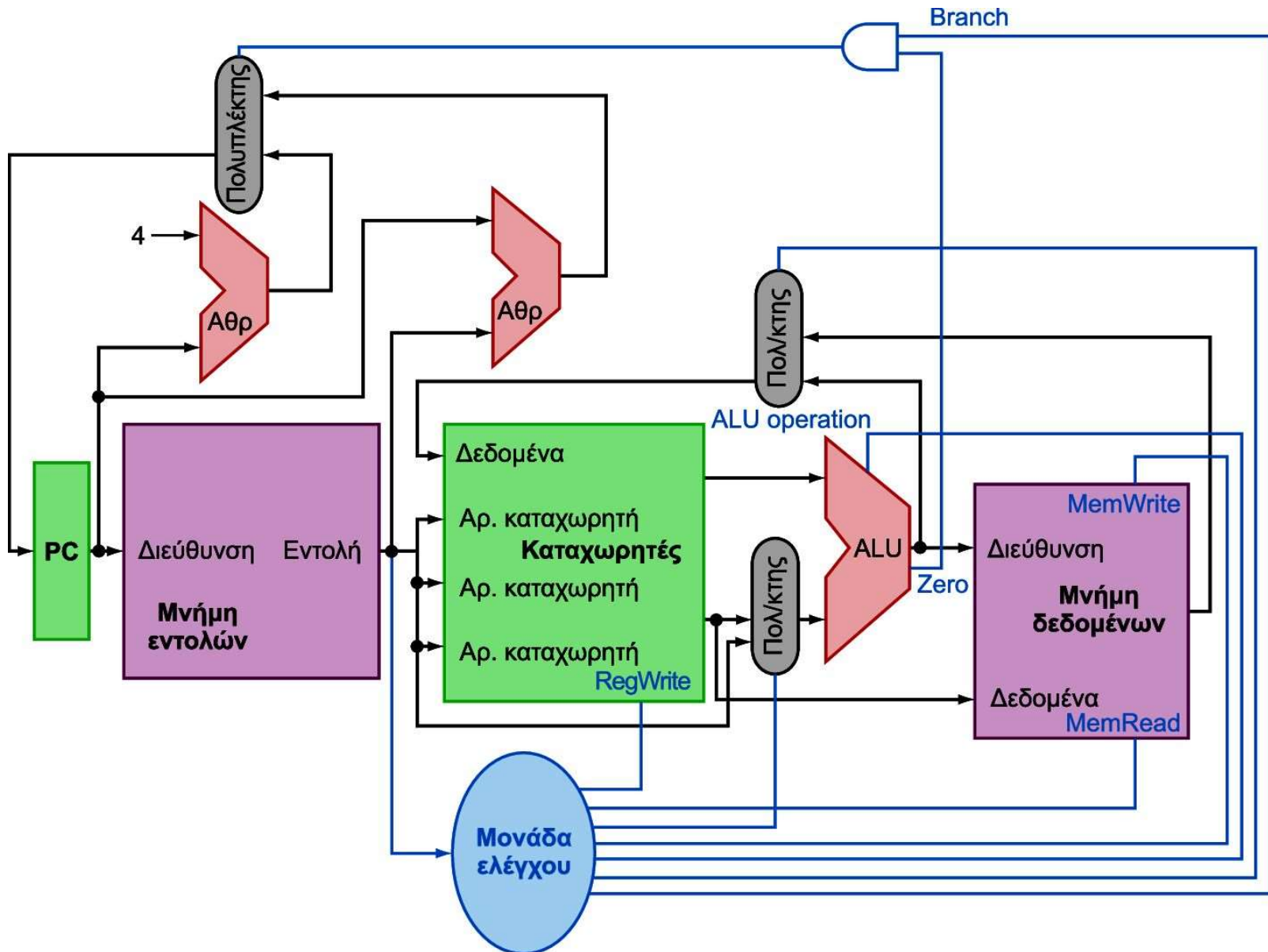


Πολυπλέκτες

- Δεν γίνεται απλώς να ενωθούν σύρματα
 - Χρησιμοποιούνται πολυπλέκτες



Μονάδα ελέγχου



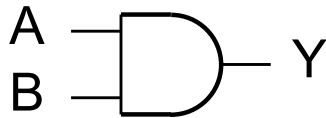
Βασικές αρχές της λογικής σχεδίασης

- Οι πληροφορίες κωδικοποιούνται σε δυαδική μορφή
 - Χαμηλή τάση = 0, υψηλή τάση = 1
 - Ένα σύρμα για κάθε bit
 - Δεδομένα πολλών bit κωδικοποιούνται σε διαύλους με πολλά σύρματα
- Συνδυαστικά στοιχεία
 - Χειρίζονται δεδομένα
 - Η έξοδος εξαρτάται από την είσοδο
- Στοιχεία κατάστασης (ακολουθιακά)
 - Αποθηκεύονται πληροφορίες σε αυτά

Συνδυαστικά στοιχεία

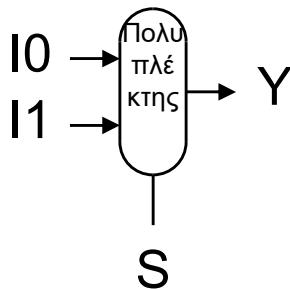
- Πύλη AND

- $Y = A \& B$



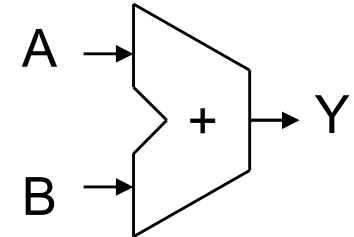
- Πολυπλέκτης

- $Y = S \wedge I1 : I0$



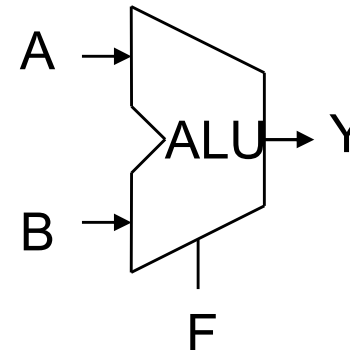
- Αθροιστής

- $Y = A + B$



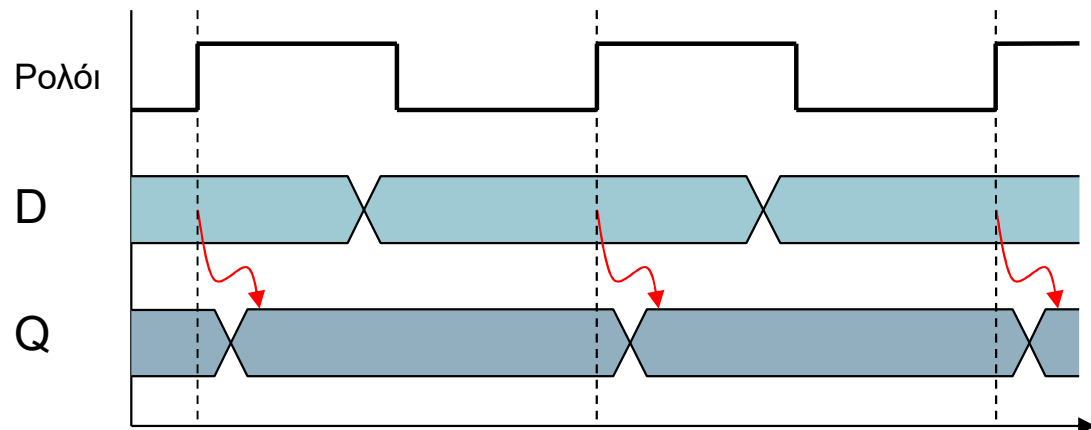
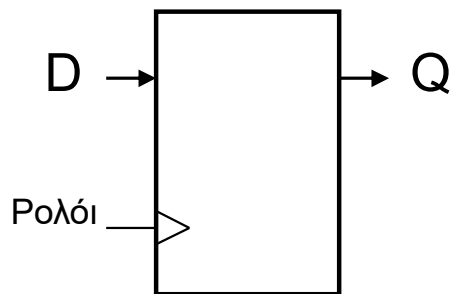
- Αριθμητική/Λογική Μονάδα (ALU)

- $Y = F(A, B)$



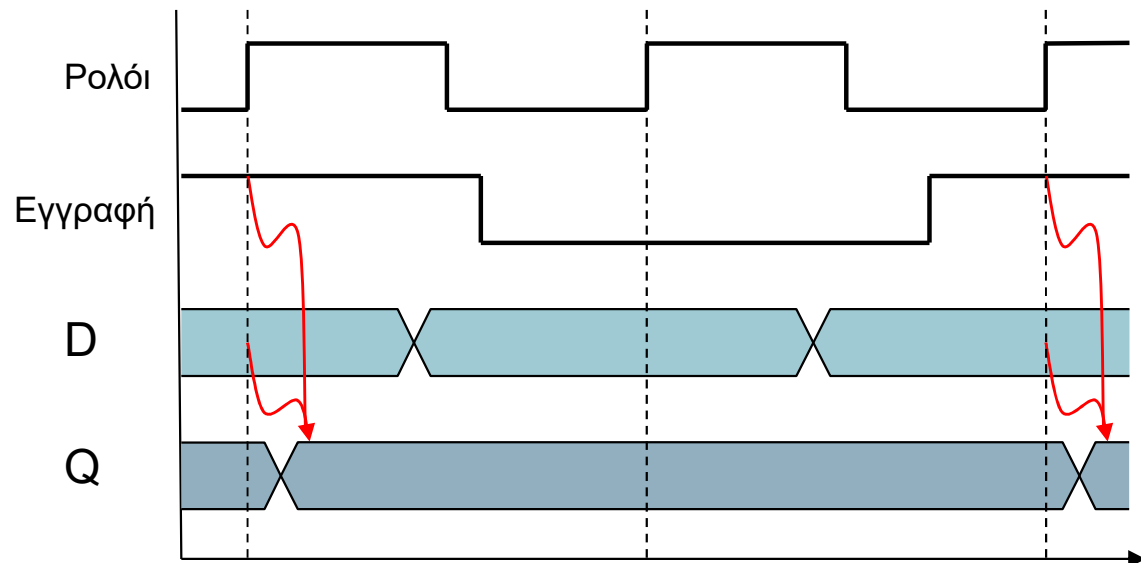
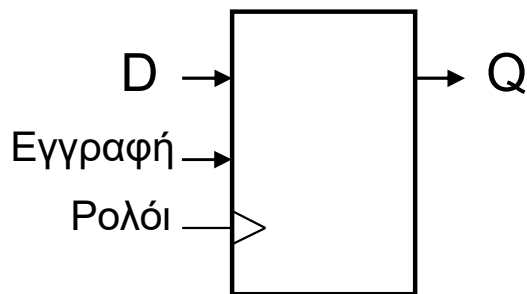
Ακολουθιακά στοιχεία

- Καταχωρητής: αποθηκεύει δεδομένα σε ένα κύκλωμα
 - Χρησιμοποιεί το σήμα του ρολογιού για να προσδιορίσει πότε πρέπει να ενημερωθεί η αποθηκευμένη τιμή
 - Αυτοπυροδοτούμενη: γίνεται ενημέρωση όταν το ρολόι αλλάζει από 0 σε 1



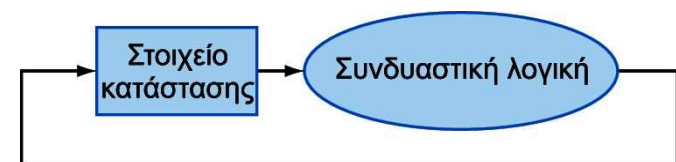
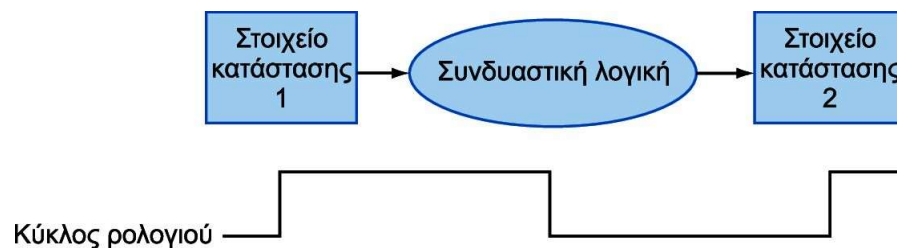
Ακολουθιακά στοιχεία

- Καταχωρητής με έλεγχο εγγραφής
 - Κάνει ενημέρωση σε ακμή του ρολογιού όταν το σήμα ελέγχου εγγραφής είναι ενεργοποιημένο (έχει τιμή 1)
 - Χρησιμοποιείται όταν η αποθηκευμένη τιμή απαιτείται αργότερα



Μεθοδολογία χρονισμού

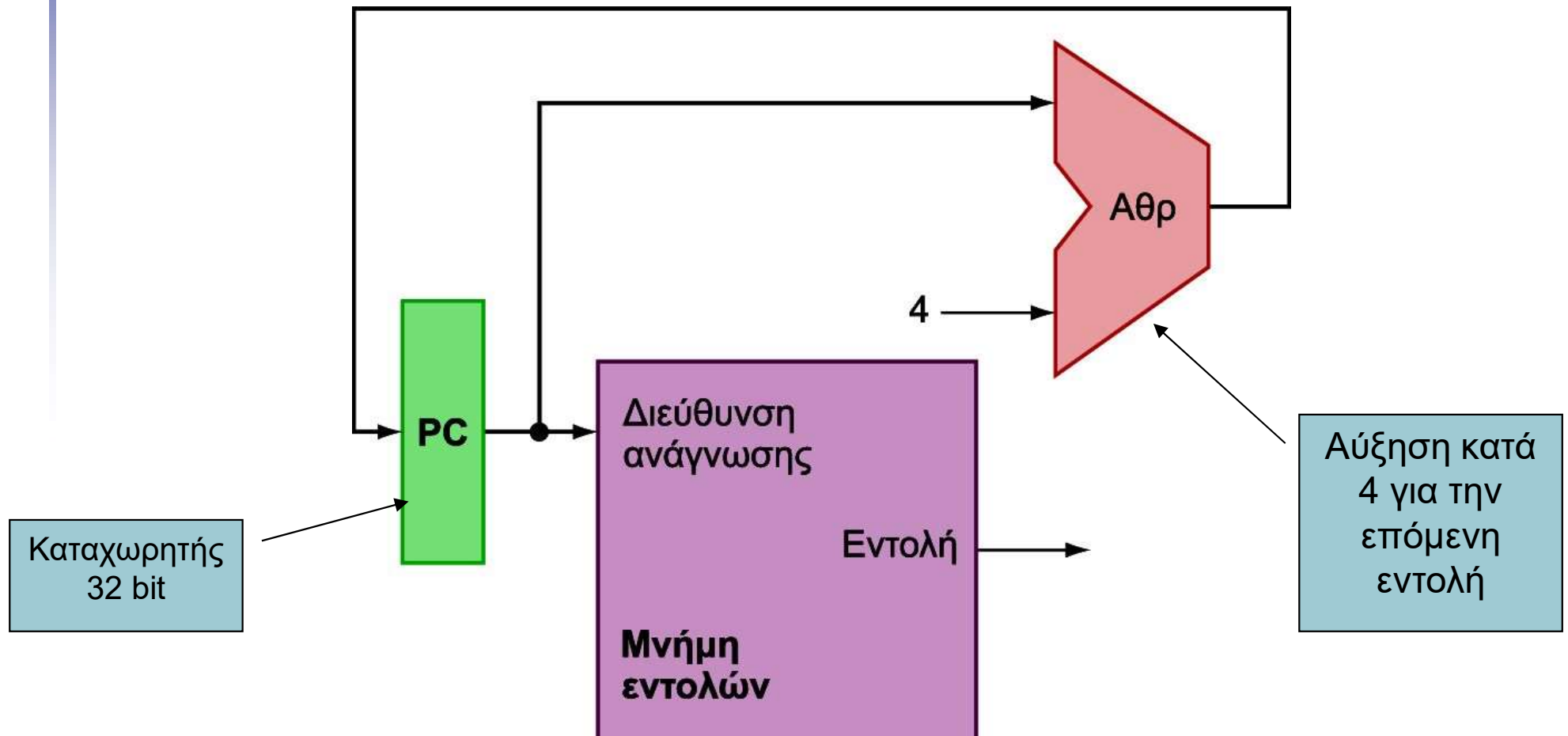
- Τα δεδομένα μετασχηματίζονται με συνδυαστική λογική κατά τη διάρκεια των κύκλων του ρολογιού
 - Μεταξύ ακμών του ρολογιού
 - Είσοδος από στοιχεία κατάστασης, έξοδος σε στοιχείο κατάστασης
 - Η μεγαλύτερη καθυστέρηση καθορίζει την περίοδο του ρολογιού



Κατασκευή μιας διαδρομής δεδομένων

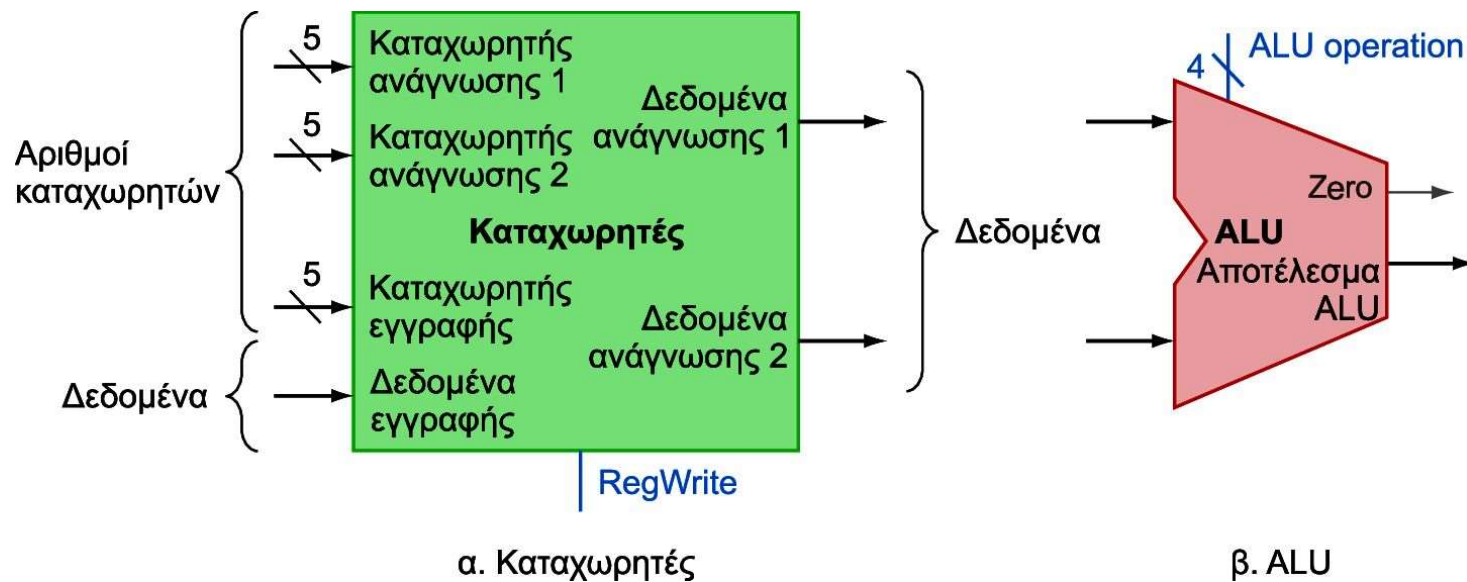
- Διαδρομή δεδομένων
 - Στοιχεία που επεξεργάζονται δεδομένα και διευθύνσεις στη CPU
 - Καταχωρητές, μονάδες ALU, πολυπλέκτες, μνήμες, ...
- Θα κατασκευάσουμε τη διαδρομή δεδομένων ενός επεξεργαστή RISC-V βήμα-βήμα
 - Βελτίωση της γενικότερης σχεδίασης

Προσκόμιση εντολής



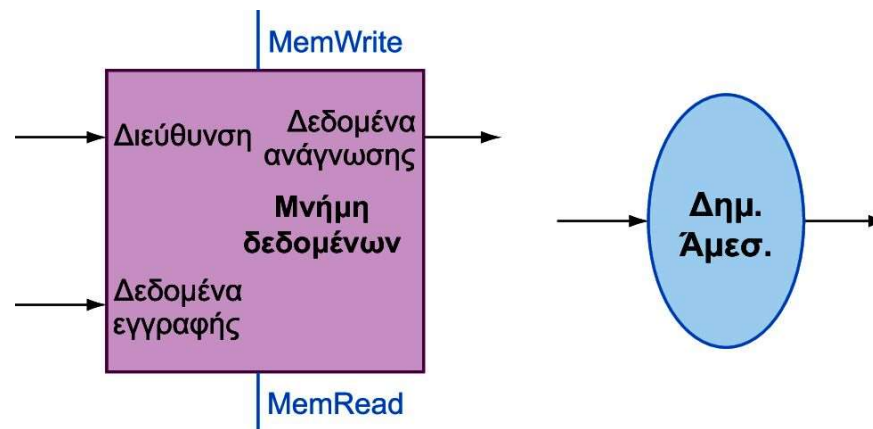
Εντολές μορφής R

- Ανάγνωση δύο τελεστών καταχωρητών
- Εκτέλεση αριθμητικής/λογικής πράξης
- Εγγραφή του αποτελέσματος στον καταχωρητή



Εντολή φόρτωσης/αποθήκευσης

- Ανάγνωση τελεστών καταχωρητών
- Υπολογισμός της διεύθυνσης με σχετική απόσταση 12 bit
 - Χρήση της ALU, αλλά με επέκταση προσήμου (σχετική απόσταση)
- Φόρτωση: Ανάγνωση από μνήμη, και ενημέρωση καταχωρητή
- Αποθήκευση: Εγγραφή τιμής καταχωρητή στη μνήμη



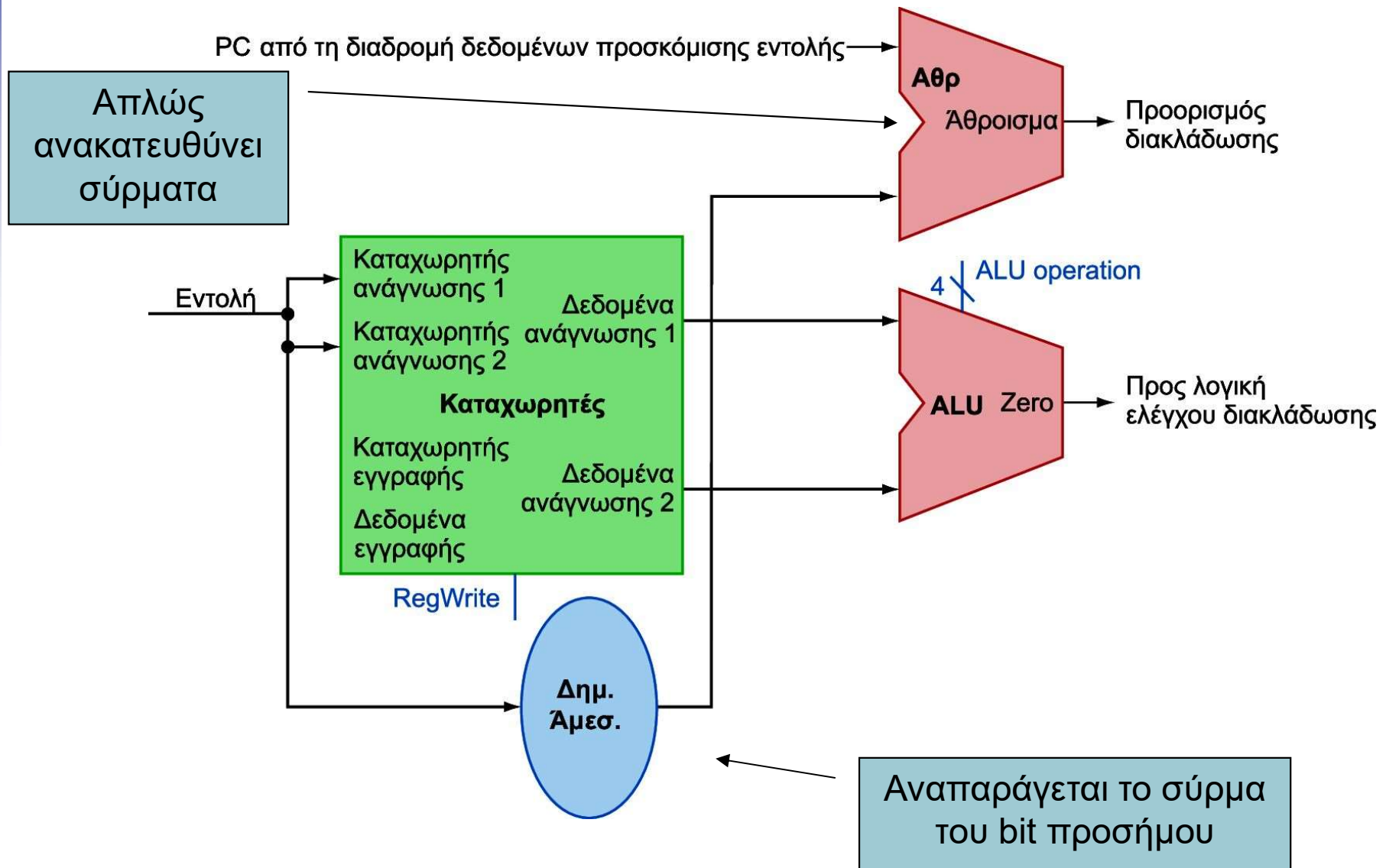
α. Μονάδα μνήμης δεδομένων

β. Μονάδα δημιουργίας άμεσου

Εντολές διακλάδωσης

- Ανάγνωση τελεστών καταχωρητών
- Σύγκριση τελεστών
 - Χρήση της ALU, αφαίρεση και έλεγχος εξόδου Zero
- Υπολογισμός διεύθυνσης προορισμού
 - Επέκταση προσήμου στη μετατόπιση
 - Αριστερή ολίσθηση κατά 1 θέση (μετατόπιση ημιλέξης)
 - Πρόσθεση στην τιμή του PC

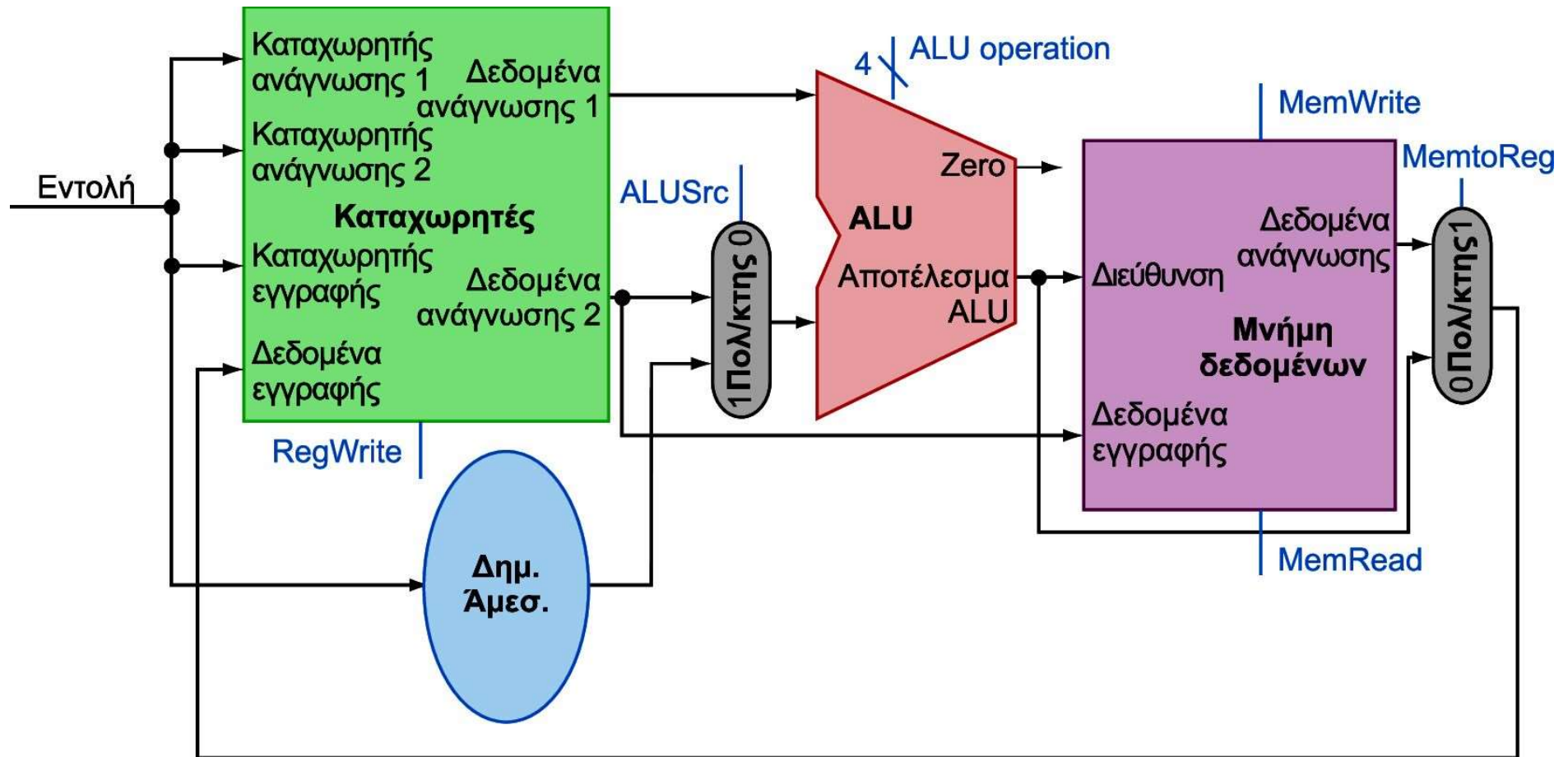
Εντολές διακλάδωσης



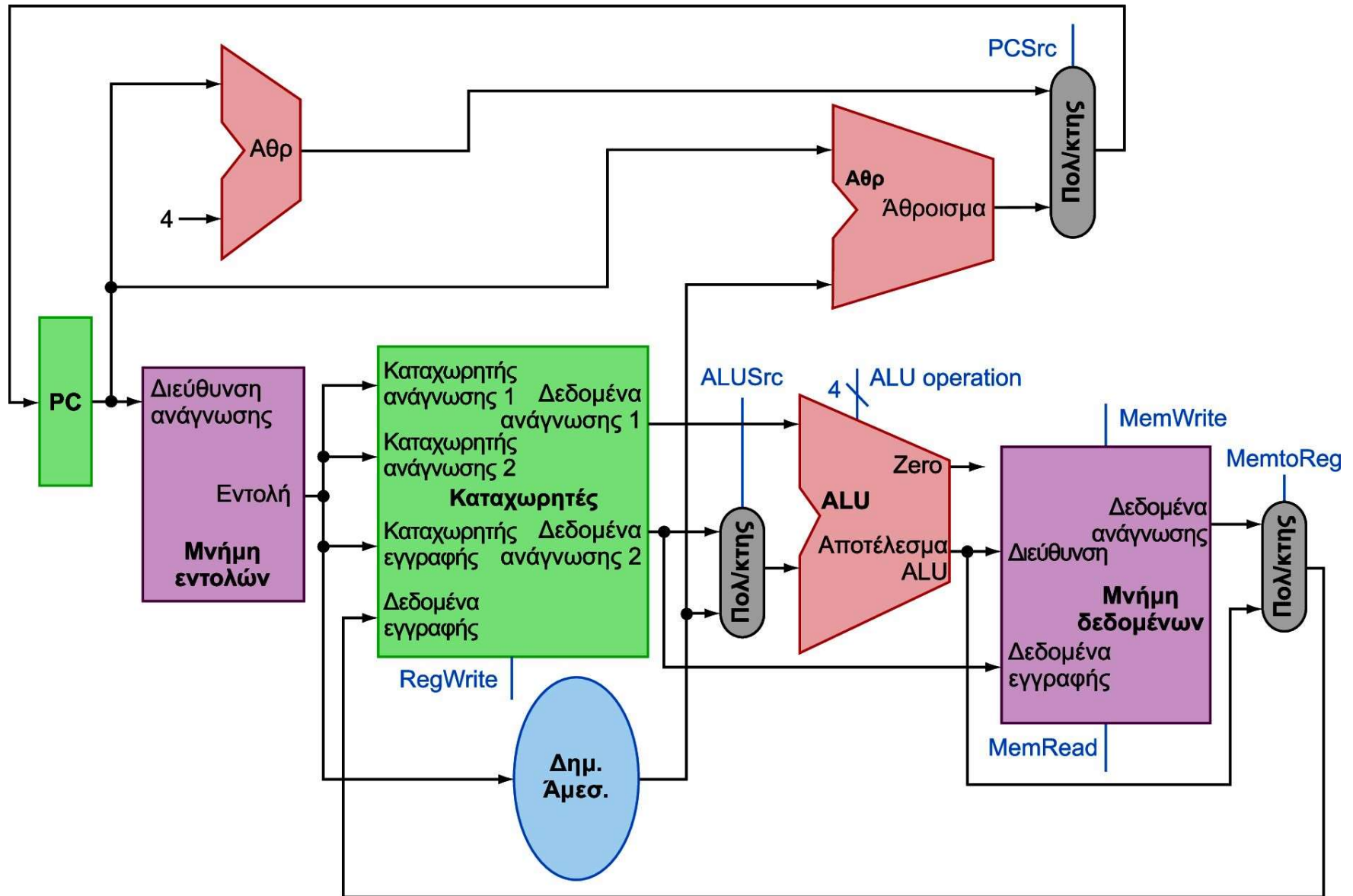
Σύνθεση των στοιχείων

- Η πρωτόλεια διαδρομή δεδομένων εκτελεί μία εντολή σε έναν κύκλο του ρολογιού
 - Κάθε στοιχείο της διαδρομής δεδομένων μπορεί να κάνει μόνο ένα πράγμα κάθε φορά
 - Γι' αυτό χρειάζονται ξεχωριστές μνήμες για εντολές και δεδομένα
- Πρέπει να χρησιμοποιούνται πολυπλέκτες όπου υπάρχουν εναλλακτικές πηγές προέλευσης δεδομένων για κάθε εντολή

Διαδρομή δεδομένων εντολών μορφής R/φόρτωσης/αποθήκευσης



Η πλήρης διαδρομή δεδομένων



Έλεγχος ALU

- Η ALU χρησιμοποιείται για
 - Φόρτωση/αποθήκευση: λειτουργία = add
 - Διακλάδωση: λειτουργία = subtract
 - Εντολή μορφής R: Η F εξαρτάται από τον κωδικό λειτουργίας (opcode)

Γραμμή ελέγχου ALU	Λειτουργία
0000	AND
0001	OR
0010	add
0110	subtract

Έλεγχος ALU

- Από τον κωδικό λειτουργίας θεωρούμε ένα πεδίο ελέγχου 2 bit με όνομα ALUOp
 - Από τη συνδυαστική λογική προκύπτουν οι γραμμές ελέγχου της ALU

opcode	ALUOp	Λειτουργία	Πεδίο κωδ. λειτουργίας	Λειτουργία ALU	Γραμμή ελέγχου ALU
lw	00	load register	XXXXXXXXXXXX	add	0010
sw	00	store register	XXXXXXXXXXXX	add	0010
beq	01	branch on equal	XXXXXXXXXXXX	subtract	0110
Μορφής R	10	add	100000	add	0010
		subtract	100010	subtract	0110
		AND	100100	AND	0000
		OR	100101	OR	0001

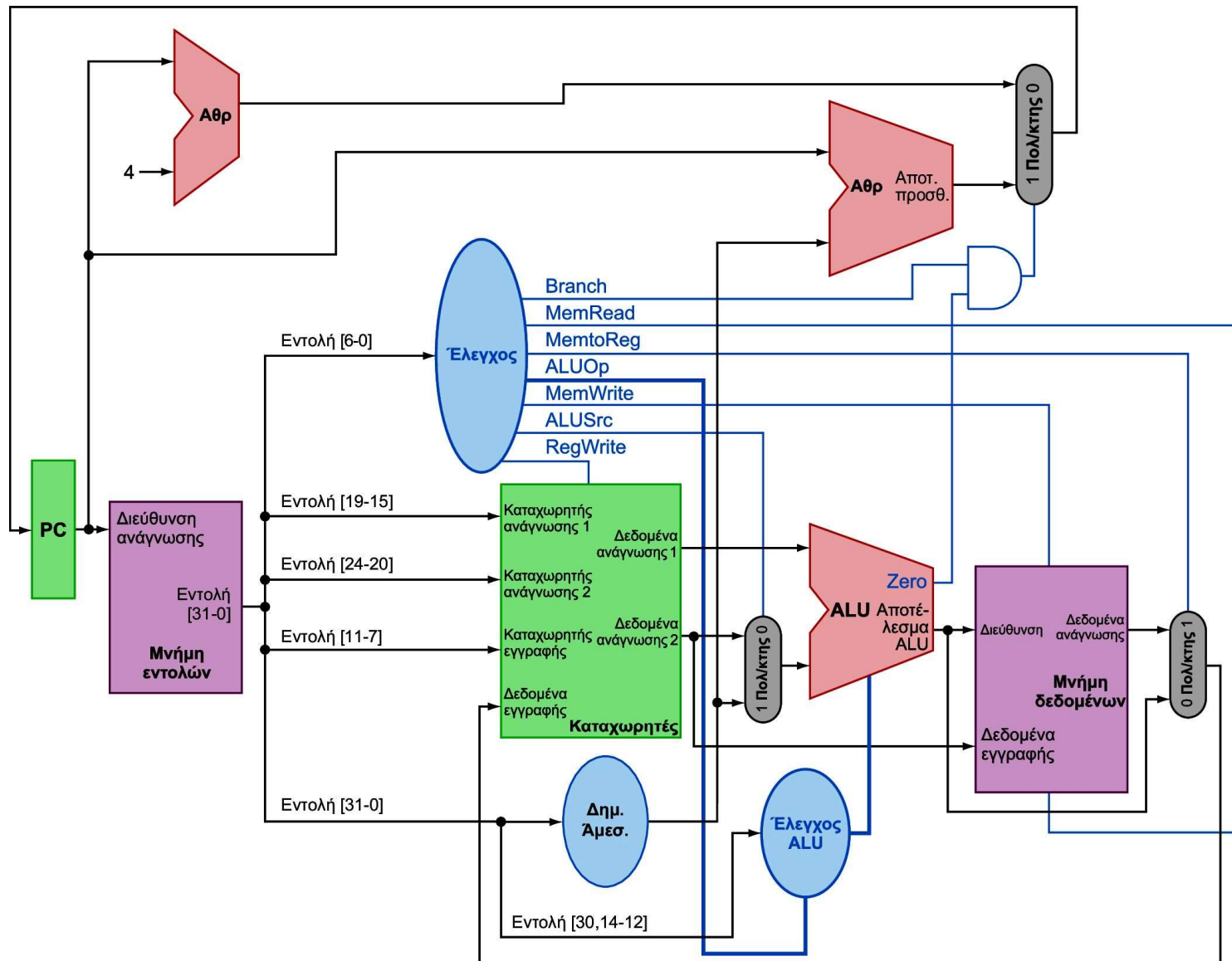
Η κύρια μονάδα ελέγχου

- Τα σήματα ελέγχου συνάγονται από την εντολή

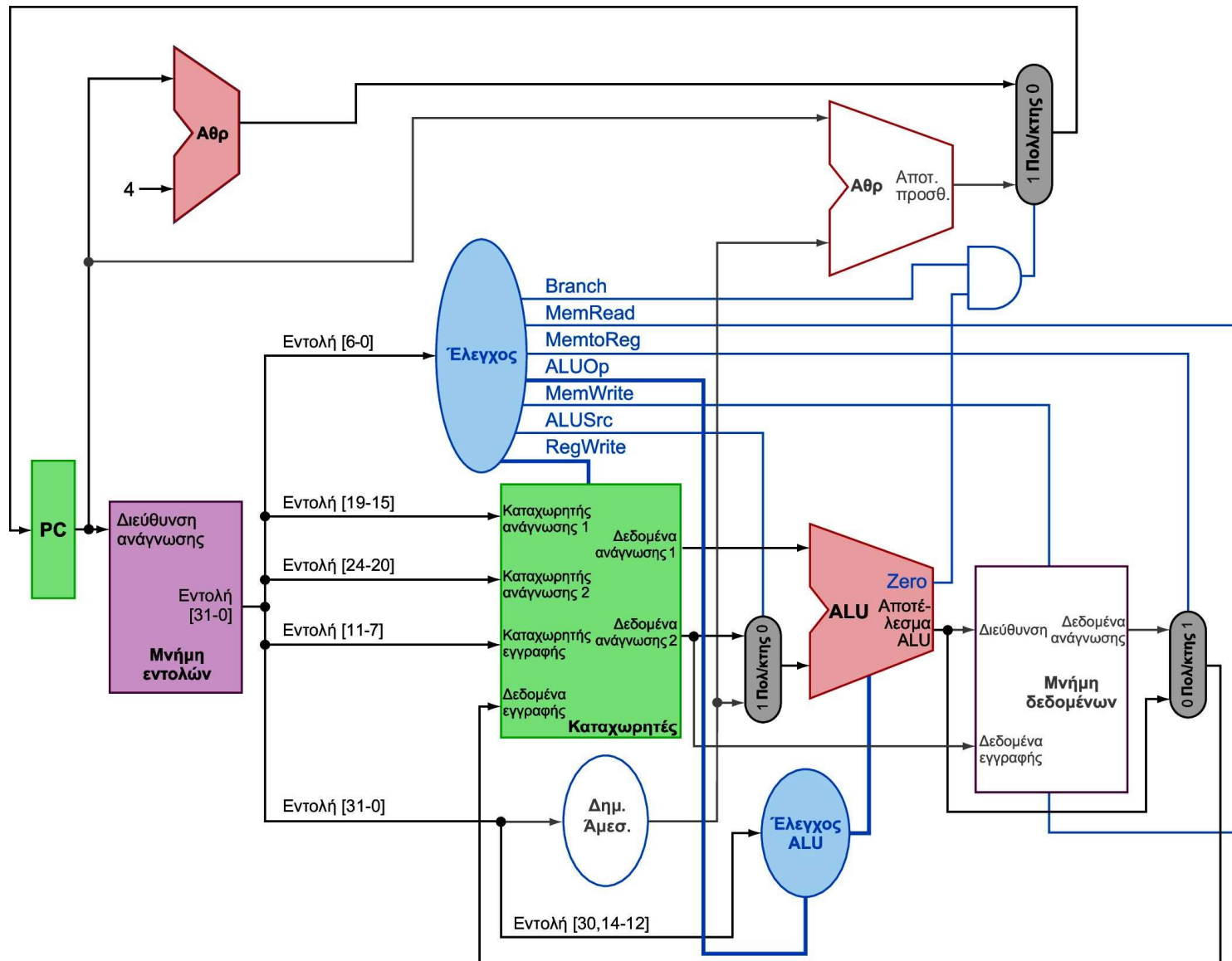
Όνομα (Θέση bit)	Πεδία					
	31:25	24:20	19:15	14:12	11:7	6:0
(α) Τύπου R	funct7	rs2	rs1	funct3	rd	opcode
(β) Τύπου I	immediate[11:0]		rs1	funct3	rd	opcode
(γ) Τύπου S	immed[11:5]	rs2	rs1	funct3	immed[4:0]	opcode
(δ) Τύπου SB	immed[12,10:5]	rs2	rs1	funct3	immed[4:1,11]	opcode

ALUOp		Πεδίο funct7							Πεδίο funct3			Λειτουργία
ALUOp1	ALUOp1	I[31]	I[30]	I[29]	I[28]	I[27]	I[26]	I[25]	I[14]	I[13]	I[12]	
0	0	X	X	X	X	X	X	X	X	X	X	0010
X	1	X	X	X	X	X	X	X	X	X	X	0110
1	X	0	0	0	0	0	0	0	0	0	0	0010
1	X	0	1	0	0	0	0	0	0	0	0	0110
1	X	0	0	0	0	0	0	0	1	1	1	0000
1	X	0	0	0	0	0	0	0	1	1	0	0001

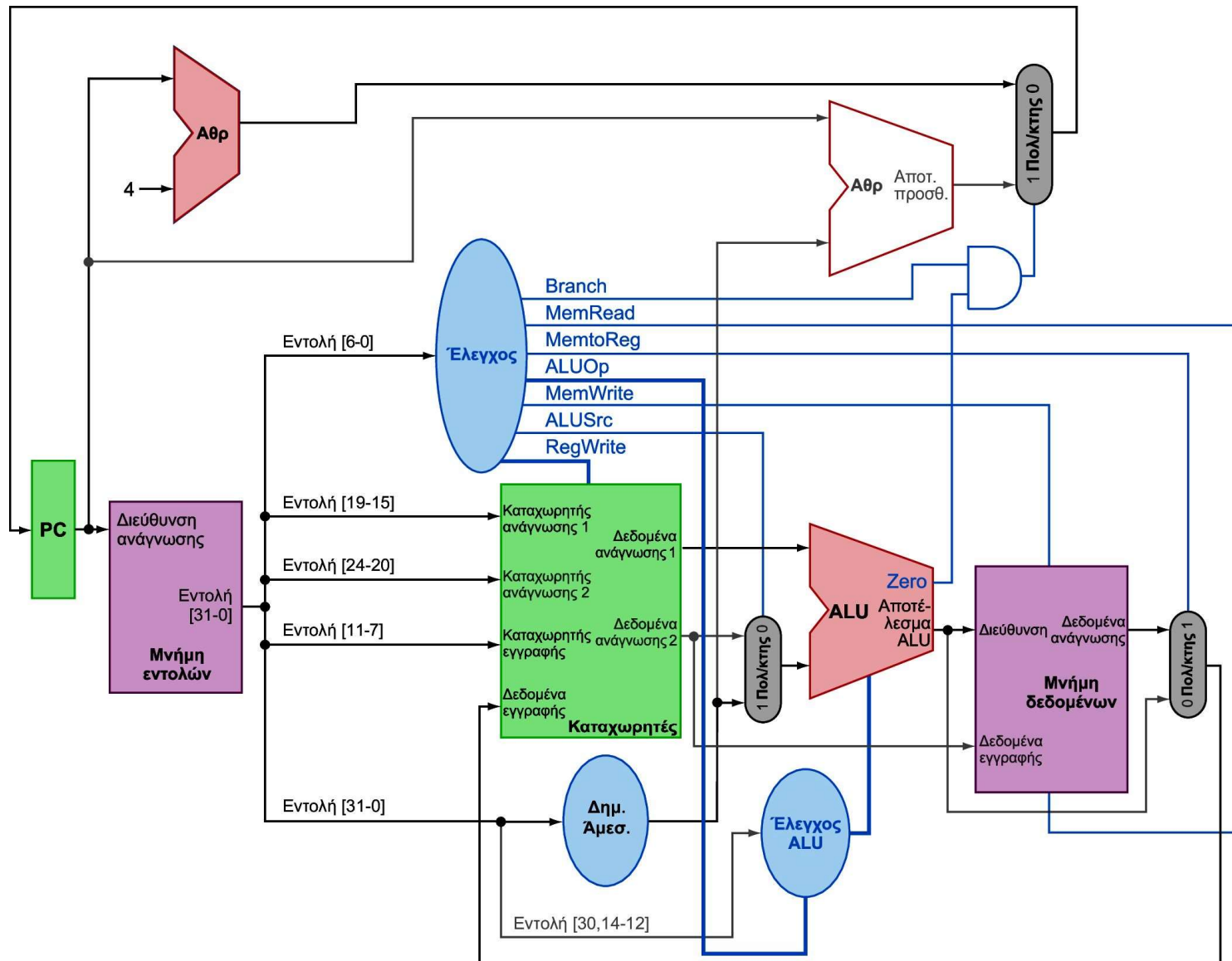
Η διαδρομή δεδομένων με τη μονάδα ελέγχου

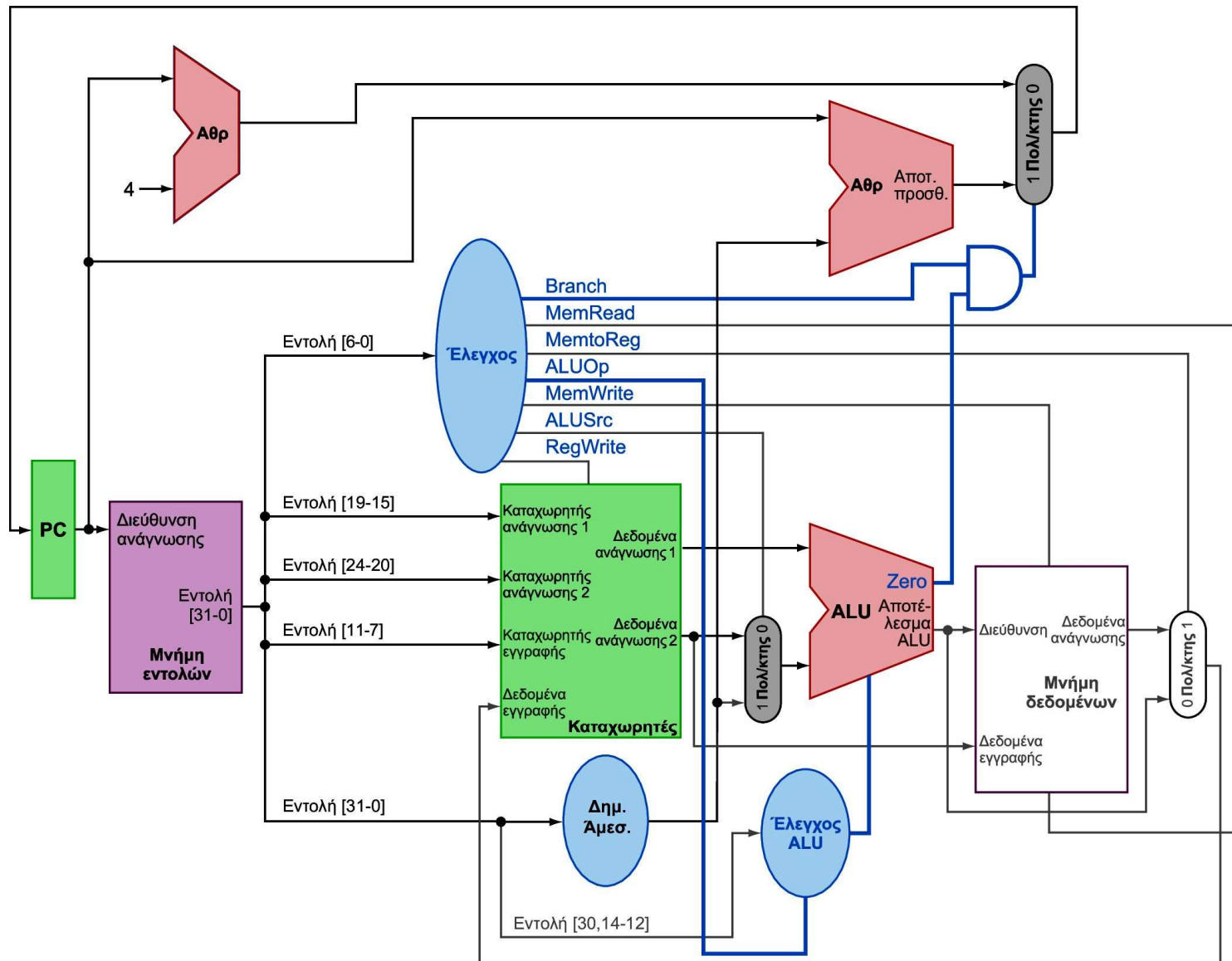


Εντολή μορφής R



Εντολή load



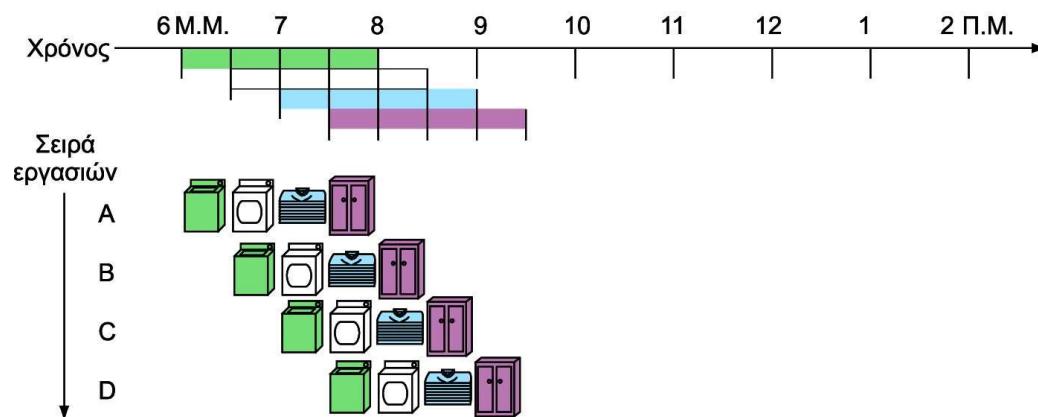
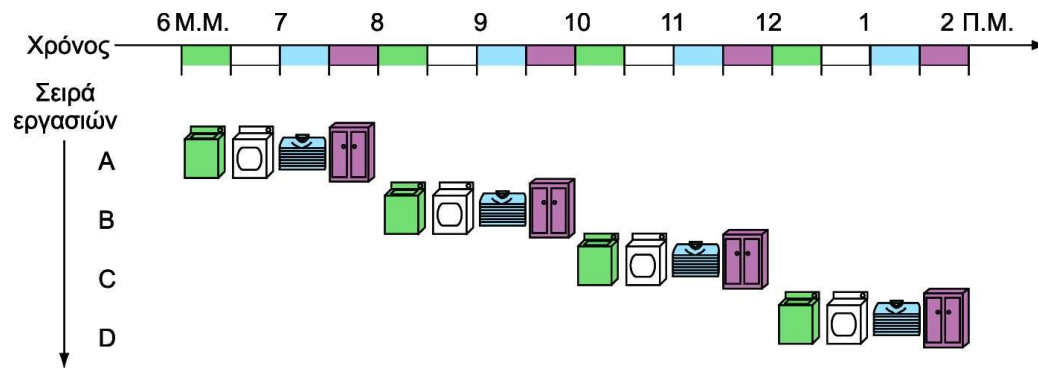


Ζητήματα απόδοσης

- Η μεγαλύτερη καθυστέρηση καθορίζει την περίοδο του ρολογιού
 - Κρίσιμη διαδρομή: εντολή load
 - Μνήμη εντολών → αρχείο καταχωρητών → ALU → μνήμη δεδομένων → αρχείο καταχωρητών
- Δεν μπορεί να διαφοροποιηθεί η περίοδος για κάθε εντολή
- Παραβιάζεται η σχεδιαστική αρχή
 - Επιτάχυνση της πιο κοινής (συνηθισμένης) περίπτωσης
- Βελτίωση της απόδοσης με διοχέτευση

Αναλογία για τη διοχέτευση

- Μπουγάδα με διοχέτευση: Επικάλυψη της εκτέλεσης
 - Η παραλληλία βελτιώνει την απόδοση



- Τέσσερις εντολές φόρτωσης:

- Επιτάχυνση
= $8/3.5 = 2.3$

- Χωρίς σταμάτημα:

- Επιτάχυνση
= $2n/0.5n + 1.5 \approx 4$
= αριθμός σταδίων

Διοχέτευση RISC-V

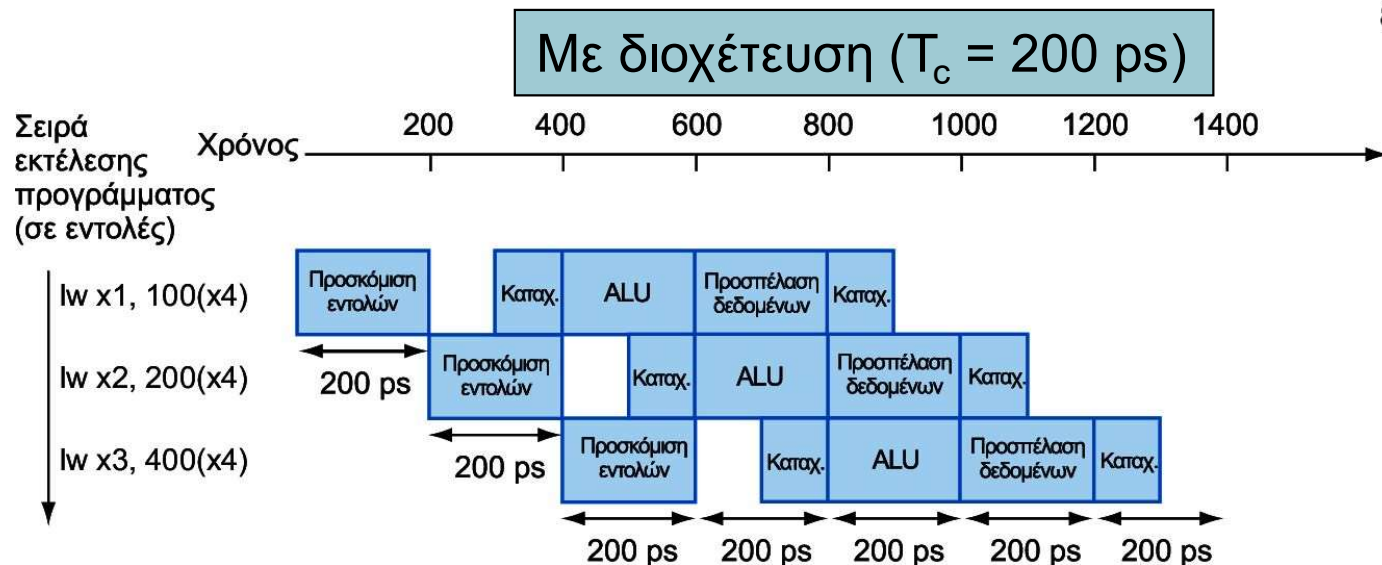
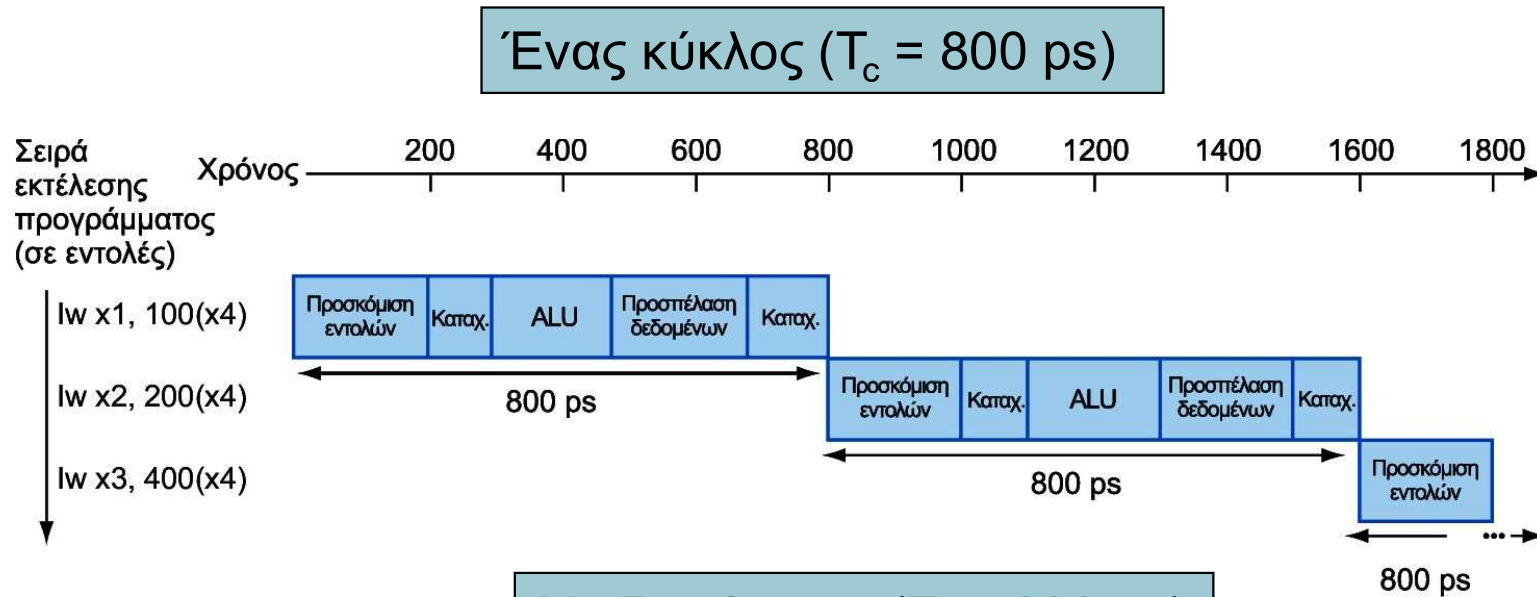
- Πέντε στάδια, ένα βήμα σε κάθε στάδιο
 1. IF: Προσκόμιση εντολής από τη μνήμη
 2. ID: Αποκωδικοποίηση εντολής και ανάγνωση αρχείου καταχωρητών
 3. EX: Εκτέλεση λειτουργίας (πράξης) ή υπολογισμός διεύθυνσης
 4. MEM: Προσπέλαση τελεστέου μνήμης
 5. WB: Εγγραφή του αποτελέσματος ξανά στον καταχωρητή

Απόδοση της διοχέτευσης

- Θεωρούμε ότι ο χρόνος των σταδίων είναι
 - 100 ps για ανάγνωση/εγγραφή καταχωρητών
 - 200 ps για τα άλλα στάδια
- Σύγκριση διαδρομής δεδομένων με διοχέτευση και με έναν κύκλο

Εντολή	Προσκόμισ η εντολής	Ανάγνωση καταχωρητών	Λειτουργία ALU	Προσπέλαση μνήμης	Εγγραφή καταχωρητών	Συνολικός χρόνος
lw	200 ps	100 ps	200 ps	200 ps	100 ps	800 ps
sw	200 ps	100 ps	200 ps	200 ps		700 ps
μορφής R	200 ps	100 ps	200 ps		100 ps	600 ps
beq	200 ps	100 ps	200 ps			500 ps

Απόδοση της διοχέτευσης



Επιτάχυνση που επιτυγχάνεται με τη διοχέτευση

- Αν όλα τα στάδια είναι ισορροπημένα
 - δηλαδή όλα διαρκούν τον ίδιο χρόνο
 - Χρόνος μεταξύ εντολών_{με διοχέτευση} = Χρόνος μεταξύ εντολών_{χωρίς διοχέτευση} / Αριθμός σταδίων
- Αν δεν είναι ισορροπημένα, μικρότερη επιτάχυνση
- Η επιτάχυνση είναι απόρροια της αυξημένης διεκπεραιωτικής ικανότητας
 - Δεν αυξάνεται ο λανθάνων χρόνος (ο χρόνος για κάθε εντολή)

Διοχέτευση και σχεδίαση ISA

- Η αρχιτεκτονική συνόλου εντολών (ISA) του RISC-V είναι σχεδιασμένη για εκτέλεση με διοχέτευση
 - Όλες οι εντολές είναι των 32 bit
 - Ευκολότερη η προσκόμιση και η αποκωδικοποίηση εντολών σε έναν κύκλο
 - πρβλ. x86: εντολές 1 έως 17 byte
 - Λίγες κανονικές μορφές εντολών
 - Μπορεί να κάνει αποκωδικοποίηση και ανάγνωση των καταχωρητών σε ένα βήμα
 - Διευθυνσιοδότηση φόρτωσης/αποθήκευσης
 - Μπορεί να υπολογίζει τη διεύθυνση στο τρίτο στάδιο, να προσπελάζει τη μνήμη στο τέταρτο στάδιο

Κίνδυνοι

- Περιπτώσεις στις οποίες η επόμενη εντολή δεν μπορεί να εκτελεστεί στον επόμενο κύκλο ρολογιού
- Δομικοί κίνδυνοι
 - Ένας πόρος που απαιτείται δεν είναι διαθέσιμος
- Κίνδυνοι δεδομένων
 - Απαιτείται αναμονή μέχρι η προηγούμενη εντολή να ολοκληρώσει την ανάγνωση/εγγραφή δεδομένων
- Κίνδυνοι ελέγχου
 - Η επιλογή ενέργειας ελέγχου εξαρτάται από την προηγούμενη εντολή

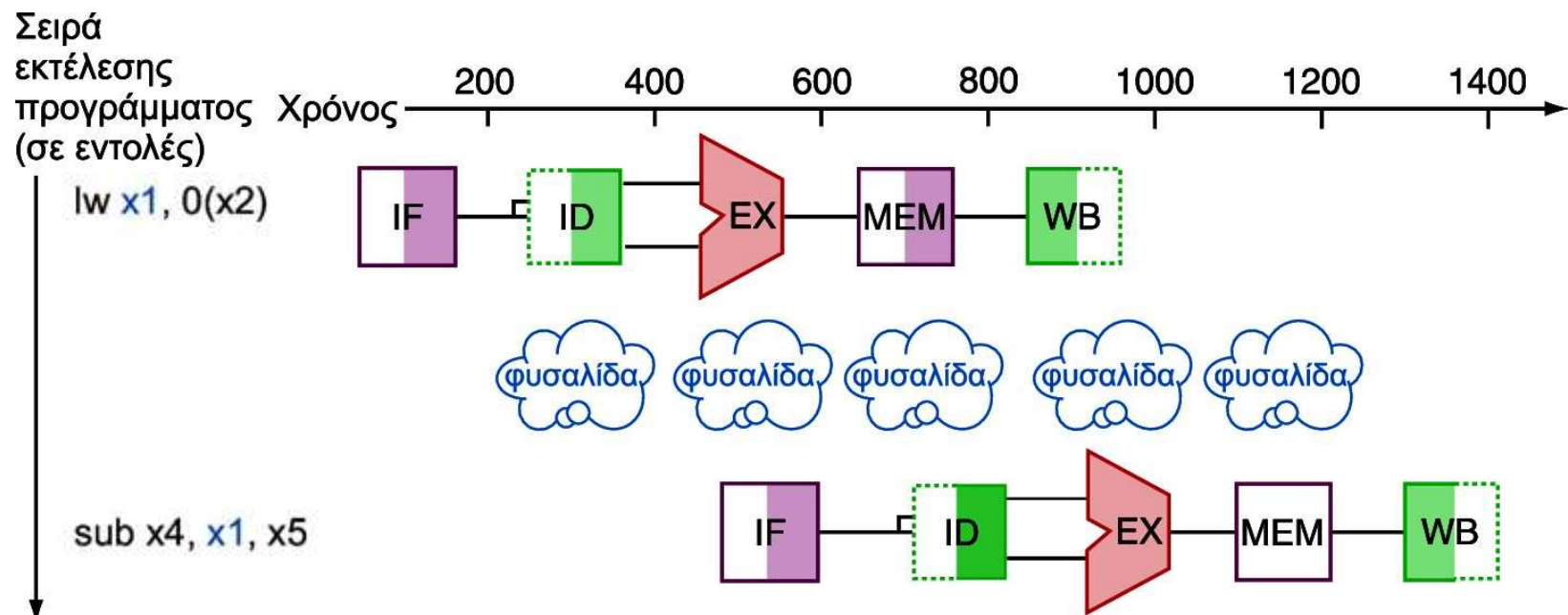
Δομικοί κίνδυνοι

- Διένεξη για τη χρήση ενός πόρου
- Στη διοχέτευση στον RISC-V με μία μνήμη
 - Για φόρτωση/αποθήκευση απαιτείται προσπέλαση δεδομένων
 - Η προσκόμιση της εντολής θα *καθυστερούσε* σε αυτόν τον κύκλο
 - Θα προκαλούσε μια «φουσαλίδα» στη διοχέτευση
- Γι' αυτό, στις διαδρομές δεδομένων με διοχέτευση απαιτούνται ξεχωριστές μνήμες για εντολές/δεδομένα
 - Ή ξεχωριστές κρυφές μνήμες εντολών/δεδομένων

Κίνδυνοι δεδομένων

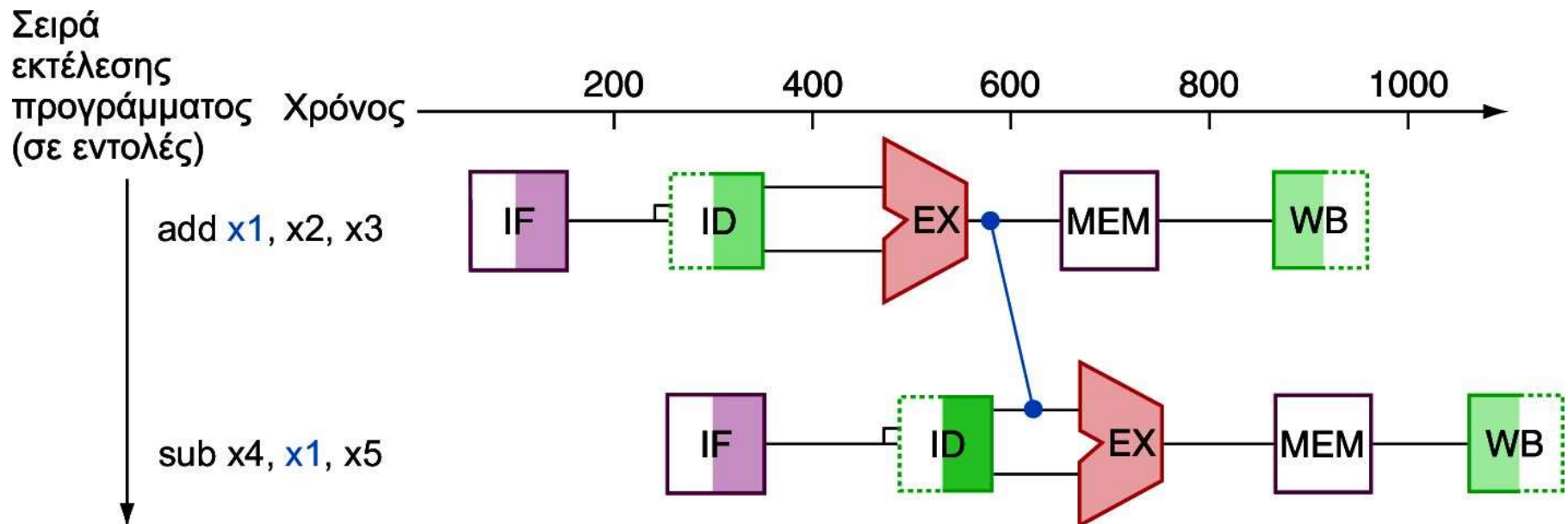
- Μια εντολή εξαρτάται από την ολοκλήρωση της προσπέλασης δεδομένων από μια προηγούμενη εντολή

- add x19, x0, x1
 sub x2, x19, x3



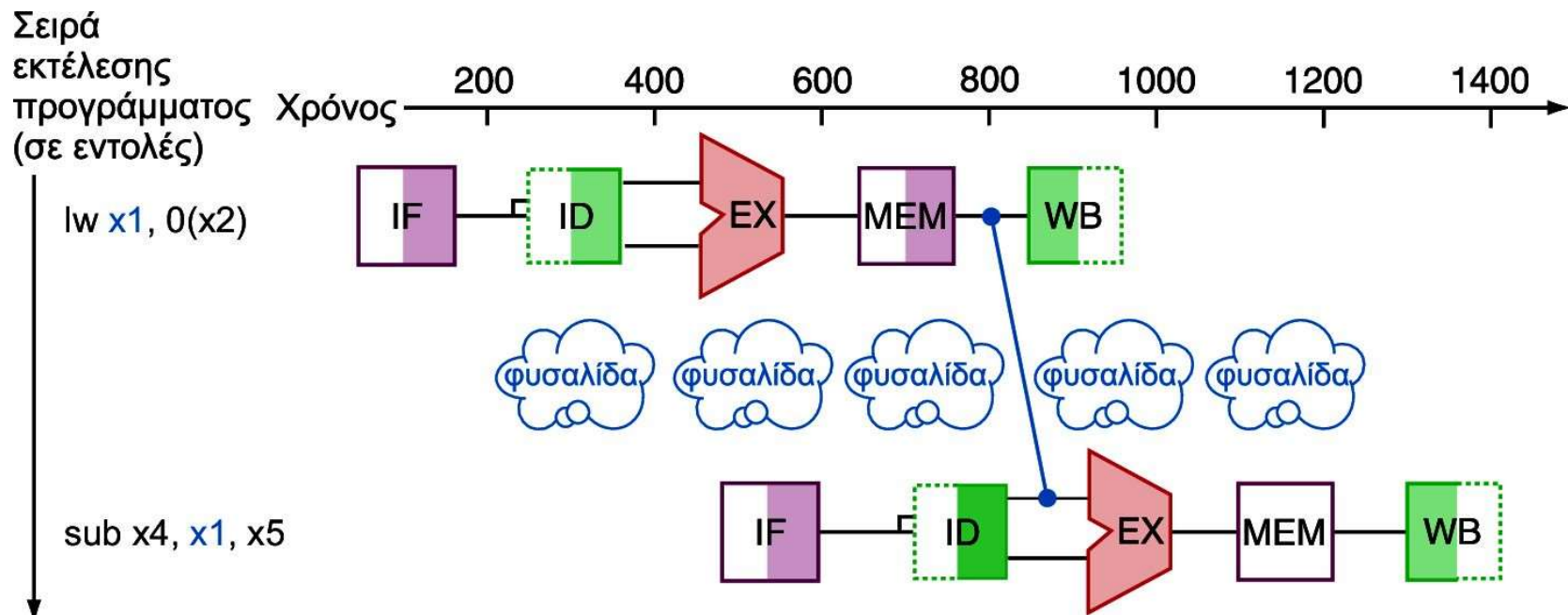
Πρώθηση (forwarding) ή παράκαμψη (bypassing)

- Το αποτέλεσμα χρησιμοποιείται αμέσως μετά από τον υπολογισμό του
 - Χωρίς αναμονή μέχρι να αποθηκευτεί σε καταχωρητή
 - Απαιτούνται επιπλέον συνδέσεις στη διαδρομή δεδομένων



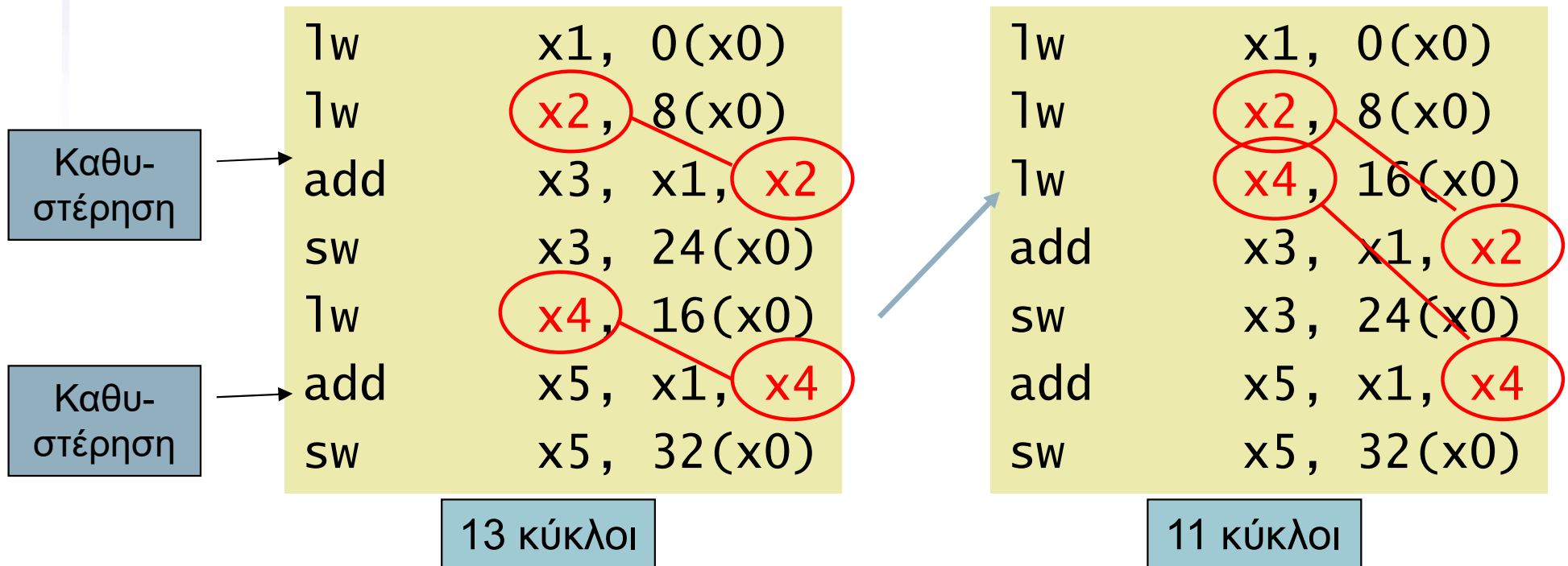
Κίνδυνοι δεδομένων φόρτωσης/χρήσης

- Δεν αποφεύγονται πάντα οι καθυστερήσεις με προώθηση
 - Αν η τιμή δεν έχει υπολογιστεί όταν είναι απαραίτητη για τη συνέχεια
 - Δεν γίνεται προώθηση προς τα πίσω στον χρόνο!



Αναδιάταξη κώδικα για την αποφυγή καθυστερήσεων

- Ο κώδικας πρέπει να αναδιαταχθεί ώστε να αποφευχθεί η χρήση του αποτελέσματος της load στην επόμενη εντολή
- Κώδικας C για $a = b + e; c = b + f;$

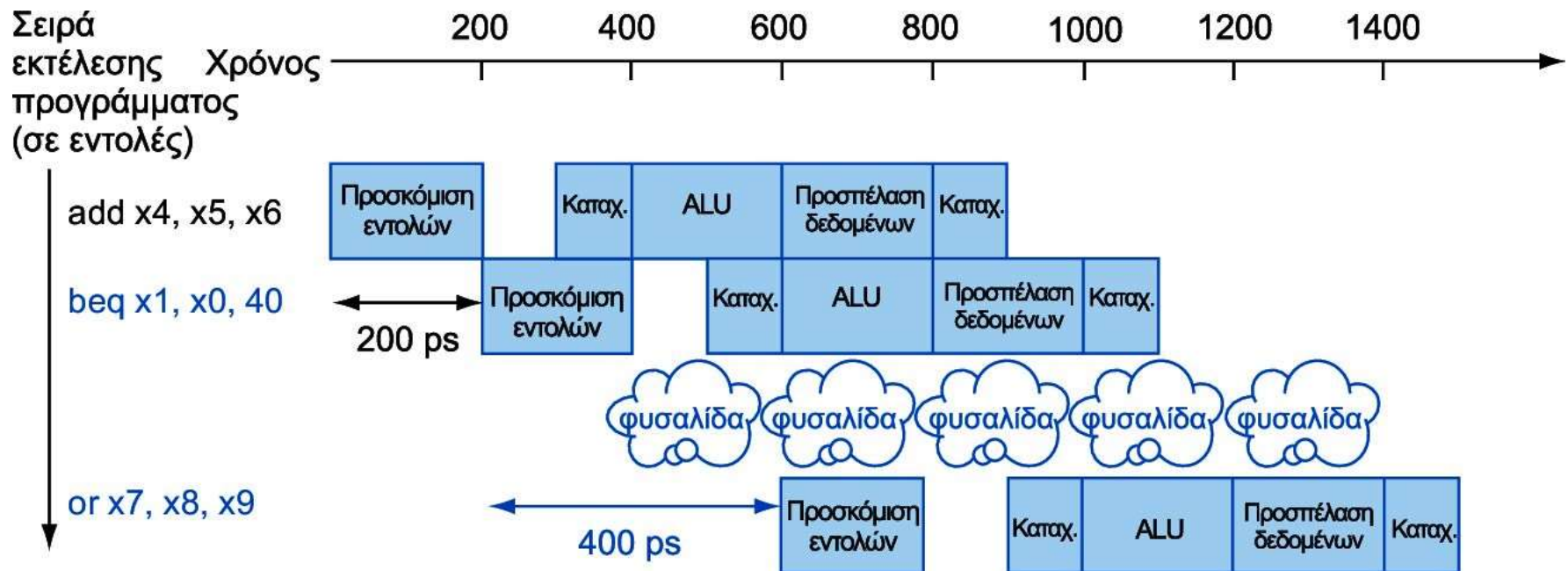


Κίνδυνοι ελέγχου

- Η διακλάδωση καθορίζει τη ροή ελέγχου
 - Η προσκόμιση της επόμενης εντολής εξαρτάται από το αποτέλεσμα της διακλάδωσης
 - Η διοχέτευση δεν προσκομίζει πάντα τη σωστή εντολή
 - Απασχολείται ακόμη στο στάδιο ID της διακλάδωσης
- Στη διοχέτευση RISC-V
 - Χρειάζεται να γίνεται σύγκριση των καταχωρητών και υπολογισμός του προορισμού νωρίς στη διοχέτευση
 - Προσθήκη υλικού για να γίνεται στο στάδιο ID

Καθυστερήση σε διακλάδωση

- Αναμονή ώσπου να προσδιοριστεί το αποτέλεσμα της διακλάδωσης, και προσκόμιση της επόμενης εντολής μετά



Πρόβλεψη διακλάδωσης

- Οι μεγαλύτερες διοχετεύσεις δεν μπορούν να προσδιορίσουν εύκολα το αποτέλεσμα μιας διακλάδωσης νωρίς
 - Η επιβάρυνση λόγω της καθυστέρησης γίνεται απαράδεκτη
- Πρόβλεψη του αποτελέσματος της διακλάδωσης
 - Καθυστέρηση μόνο αν η πρόβλεψη είναι λανθασμένη
- Στη διοχέτευση RISC-V
 - Μπορεί να γίνει πρόβλεψη των διακλαδώσεων που δεν ακολουθήθηκαν
 - Προσκόμιση της εντολής μετά τη διακλάδωση, χωρίς καθυστέρηση

Πιο ρεαλιστική πρόβλεψη διακλάδωσης

- Στατική πρόβλεψη διακλάδωσης
 - Με βάση τη συνήθη συμπεριφορά μιας διακλάδωσης
 - Παράδειγμα: διακλαδώσεις βρόχων και εντολών if
 - Πρόβλεψη της ληφθείσας διακλάδωσης προς τα πίσω
 - Πρόβλεψη της μη ληφθείσας διακλάδωσης προς τα εμπρός
- Δυναμική πρόβλεψη διακλάδωσης
 - Η πραγματική συμπεριφορά της διακλάδωσης μετριέται μέσω υλικού
 - π.χ. με καταγραφή του πρόσφατου παρελθόντος-ιστορικού κάθε διακλάδωσης
 - Υπόθεση για συνέχιση της ίδιας τάσης στο άμεσο μέλλον
 - Σε περίπτωση λάθους: καθυστέρηση για εκ νέου προσκόμιση, και ενημέρωση ιστορικού

Περίληψη της διοχέτευσης

η ΓΕΝΙΚΗ εικόνα

- Η διοχέτευση βελτιώνει την απόδοση αυξάνοντας τη διεκπεραιωτική ικανότητα των εντολών
 - Εκτελεί περισσότερες από μία εντολές ταυτόχρονα
 - Κάθε εντολή έχει τον ίδιο λανθάνοντα χρόνο
- Υπόκειται σε κινδύνους
 - Δομικούς, δεδομένων, ελέγχου
- Η σχεδίαση του συνόλου εντολών επηρεάζει την πολυπλοκότητα της υλοποίησης της διοχέτευσης