

Αρχιτεκτονική 2 Σημειώσεις για εξετάσεις

Κωνσταντίνος Ζουριδάκης

1 Δυναμική ενέργεια και ισχύς

$$\Delta\text{υναμική Ενέργεια} = \frac{1}{2}CV^2$$

$$\Delta\text{υναμική Ισχύς} = \frac{1}{2}CV^2 \underbrace{\text{Συχνότητα μεταβάσεων}}_{f_v}$$

2 Κόστος ολοκληρωμένων κυκλωμάτων

$$\text{Cost} = \frac{C_3 + CT_3 + C_{PT}}{Y}$$

Όπου:

Cost = Κόστος ολοκληρωμένου κυκλώματος

C_3 = Κόστος κύβου

CT_3 = Κόστος δοκιμής κύβου

C_{PT} = Κόστος συσκευασίας και τελικής δοκιμής

Y = Εσοδεία τελικής δοκιμής

$$C_3 = \frac{C_{PL}}{CPL_3 \times Y_3}$$

Όπου:

C_{PL} = Κόστος πλακιδίου

CPL_3 = Κύβοι ανά πλακίδιο

Y_3 = Εσοδεία κύβων

$$CPL_3 = \frac{\pi \times \left(\frac{d_{PL}}{2}\right)^2}{S_3} - \frac{\pi \times d_{PL}}{\sqrt{2} \times S_3}$$

Όπου;

d_{PL} = Διάμετρος πλακιδίου

S_3 = Επιφάνεια κύβου

Bose-Einstein τύπος:

$$Y_3 = Y_{PL} \frac{1}{(1 + \text{imp} \times S_3)^N}$$

Όπου:

Y_{PL} = Εσοδεία πλακιδίων

imp = Ατέλειες ανά μονάδα επιφανείας

N = Συντελεστής πολυπλοκότητας διαδικασίας

3 Φερεγγυότητα

$$MTBF = MTTF + MTTR$$

$$\text{Availability} = \frac{MTTF}{MTBF}$$

Όπου:

$MTTF$ = Mean time to failure

$MTTR$ = Mean time to repair

$MTBF$ = Mean time between failures

4 Μέτρηση της απόδοσης

Επιτάχυνση του X σε σχέση με τον Y :

$$\text{Speedup}_{XY} = \frac{\text{Χρόνος Εκτέλεσης}_Y}{\text{Χρόνος Εκτέλεσης}_X}$$

5 Νόμος του Amdahl

$$\text{Χρόνος Εκτέλεσης}_{\text{νέος}} = \text{Χρόνος εκτέλεσης}_{\text{παλιός}} \times \left((1 - \text{Κλάσμα}_{\text{βελτίωσης}}) + \frac{\text{Κλάσμα}_{\text{βελτίωσης}}}{\text{Επιτάχυνση}_{\text{βελτίωσης}}} \right)$$

$$\text{Επιτάχυνση}_{\text{συνολική}} = \frac{\text{Χρόνος εκτέλεσης}_{\text{παλιός}}}{\text{Χρόνος εκτέλεσης}_{\text{νέος}}} = \frac{1}{(1 - \text{Κλάσμα}_{\text{βελτίωσης}}) + \frac{\text{Κλάσμα}_{\text{βελτίωσης}}}{\text{Επιτάχυνση}_{\text{βελτίωσης}}}}$$

6 Εξίσωση απόδοσης επεξεργαστή

Χρόνος CPU = Κύκλοι ρολογιού CPU για ένα πρόγραμμα \times Χρόνος κύκλου ρολογιού

$$CPI = \frac{\text{Κύκλοι ρολογιού CPU για ένα πρόγραμμα}}{\text{Πλήθος εντολών}}$$

Χρόνος CPU = Πλήθος εντολών \times Κύκλοι ανά εντολή \times Χρόνος κύκλου ρολογιού

$$\frac{\text{Εντολές}}{\text{Πρόγραμμα}} \times \frac{\text{Κύκλοι ρολογιού}}{\text{Εντολή}} \times \frac{\text{Δευτερόλεπτα}}{\text{Χρόνος κύκλου}} = \frac{\text{Δευτερόλεπτα}}{\text{Πρόγραμμα}} = \text{Χρόνος CPU}$$

$$\text{Κύκλοι ρολογιού CPU} = \sum_{i=1}^n \text{Πλήθος εντολών}_i \times CPI_i$$

$$\text{Χρόνος CPU} = \left(\sum_{i=1}^n \text{Πλήθος εντολών}_i \times CPI_i \right) \times \text{Χρόνος κύκλου ρολογιού}$$

7 Αστοχίες

$$\frac{\text{Αστοχίες}}{\text{Εντολή}} = \frac{\text{Ρυθμός αστοχίας} \times \text{Προσπελάσεις μνήμης}}{\text{Πλήθος εντολών}} = \text{Ρυθμός αστοχίας} \times \frac{\text{Προσπελάσεις μνήμης}}{\text{Εντολή}}$$

Μέσος χρόνος προσπέλασης μνήμης = Χρόνος ευστοχίας + Ρυθμός αστοχίας \times Ποινή αστοχίας

8 Διοχέτευση

$$CPI_{\text{διοχέτευσης}} = CPI_I + \mathcal{S} + \mathcal{D} + \mathcal{C}$$

Όπου:

CPI_I = CPI ιδανικής διοχέτευσης

\mathcal{S} = Καθυστερήσεις δομής

\mathcal{D} = Καθυστερήσεις κινδύνων δεδομένων

\mathcal{C} = Καθυστερήσεις ελέγχου

8.1 Κίνδυνοι δεδομένων

- Ανάγνωση μετά την εγγραφή.
- Εγγραφή μετά την εγγραφή.
- Εγγραφή μετά την ανάγνωση.

Μια εντολή εξαρτάται από την ολοκλήρωση μιας προσπέλασης δεδομένων μιας προηγούμενης εντολής

```
add $s0, $t0, $t1  
sub $t2, $s0, $t3
```

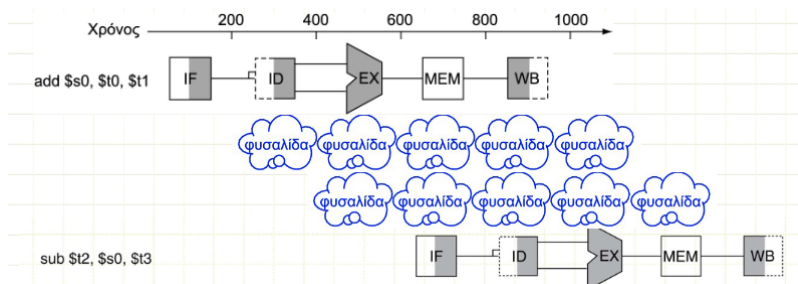


Figure 1: Κίνδυνος δεδομένων.

Το πρόβλημα μπορεί να αντιμετωπιστεί με **προώθηση**:

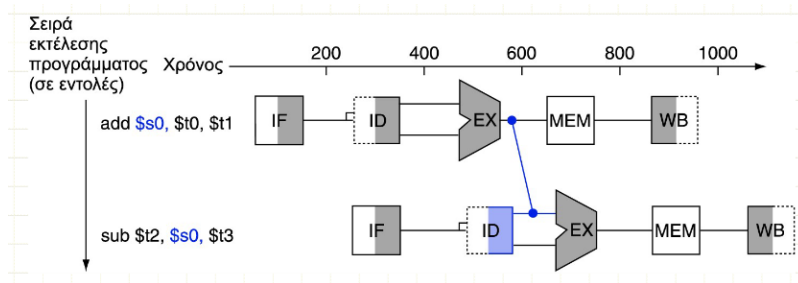


Figure 2: Προώθηση

Το πρόβλημα **Φόρτωσης/Χρήσης** μπορεί να λυθεί επίσης:

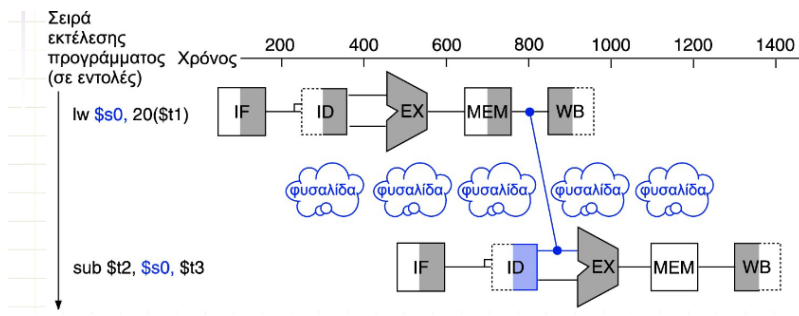


Figure 3: Φόρτωση/Χρήση

Μπορούμε να αποφύγουμε τις καθυστερήσεις με κατάλληλο **χρονοπρογραμματισμό**:

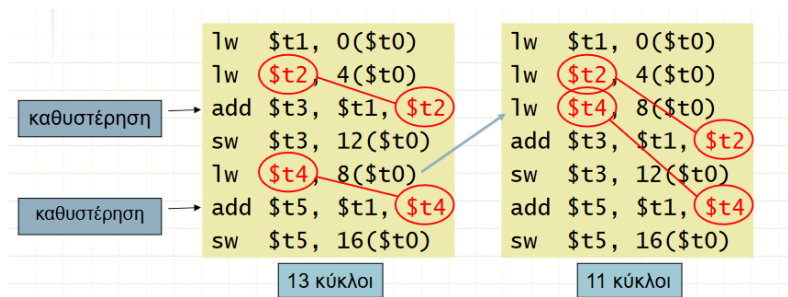


Figure 4: Χρονοπρογραμματισμός

8.2 Κίνδυνοι δομής

Συμβαίνει στη διοχέτευση MIPS με μία μοναδική μνήμη. Οι εντολές load/store πραγματοποιούν προσπέλαση μνήμης και απαιτείται καθυστέρηση σε περίπτωση εκτέλεσης της μίας μετά την άλλη. Αυτό μπορεί να αποφευχθεί με τη χρήση ξεχωριστών μνημών.

8.3 Κίνδυνοι ελέγχου

Η προσκόμιση της επόμενης εντολής εξαρτάται από το αποτέλεσμα της διακλάδωσης. Η διοχέτευση δεν μπορεί να προσκομίσει πάντα τη σωστή εντολή, καθώς ακόμη δουλεύει στο στάδιο ID.

Ένας τρόπος αντιμετώπισης είναι το **stall on branch**.

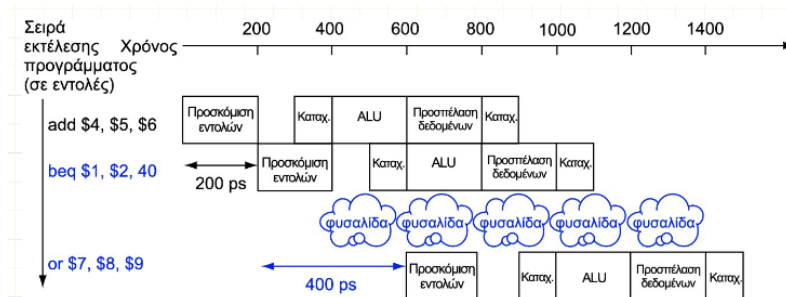


Figure 5:

Ένας άλλος τρόπος αντιμετώπισης είναι με **πρόβλεψη διακλάδωσης**.

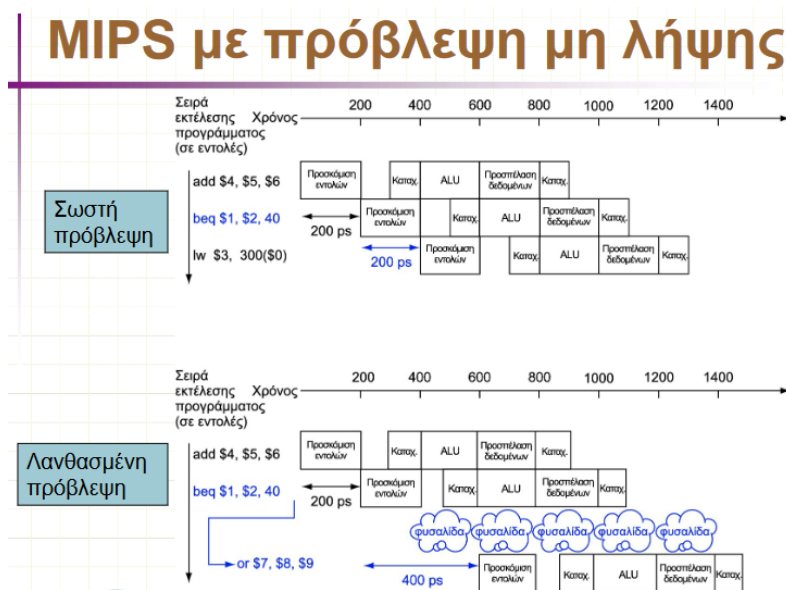


Figure 6:

Πιο ρεαλιστική πρόβλεψη διακλάδωσης

- Στατική πρόβλεψη διακλάδωσης
 - Βασίζεται στην τυπική συμπεριφορά των διακλαδώσεων
 - Παράδειγμα: διακλαδώσεις σε βρόχους και εντολές if
 - Πρόβλεψη διακλαδώσεων προς τα πίσω (backward branches) ως λαμβανόμενες
 - Πρόβλεψη διακλαδώσεων προς τα εμπρός (forward branches) ως μη λαμβανόμενες
- Δυναμική πρόβλεψη διακλάδωσης
 - Το υλικό μετράει τη πραγματική συμπεριφορά διακλαδώσεων
 - π.χ., καταγράφει την πρόσφατη ιστορία κάθε διακλάδωσης
 - Υποθέτει ότι η μελλοντική συμπεριφορά θα συνεχίσει την τάση
 - Σε περίπτωση λάθους, γίνεται καθυστέρηση κατά την επαναπροσκόμιση, και ενημέρωση του ιστορικού

Figure 7:

«Καθυστερημένες» διακλαδώσεις

- Καθυστερημένες διακλαδώσεις [delayed branches]
- **Ιδέα:** καθυστέρησε να πάρεις απόφαση για μια διακλάδωση
 - Αυτό υλοποιεί ο MIPS
- **Μια «καθυστερημένη» διακλάδωση εκτελεί πάντα την επόμενη της εντολή**
 - Δεν ελέγχει την «ορθότητα» αυτής της επιλογής
 - Πρέπει η επόμενη εντολή να κάνει «άσχετο» με τη διακλάδωση έργο
 - Στο προηγούμενο παράδειγμα, η add \$4, \$5, \$6 μπορεί να μεταφερθεί μετά τη beq
 - Δεν θα αλλάξει τίποτε και θα αποφευχθεί ο 1 κύκλος καθυστέρησης
 - Αυτό το κάνουν οι μεταγλωττιστές

Figure 8:

8.4 Επιτάχυνση λόγω διοχέτευσης

Με την προϋπόθεση ότι όλα τα στάδια διαρκούν τον ίδιο χρόνο:

$$\text{Χρόνος μεταξύ εντολών}_{\text{Με διοχέτευση}} = \frac{\text{Χρόνος μεταξύ εντολών}_{\text{Χωρίς διοχέτευση}}}{\text{Αριθμός σταδίων}}$$

8.5 Στάδια διοχέτευσης

1. **IF**: Instruction fetch from memory
(προσκόμιση εντολής από τη μνήμη)
2. **ID**: Instruction decode & register read
(αποκωδικοποίηση εντολής & ανάγνωση καταχωρητών)
3. **EX**: Execute operation or calculate address
(εκτέλεση λειτουργίας ή υπολογισμός δ/νσης)
4. **MEM**: Access memory operand (προσπέλαση τελεστέου μνήμης)
5. **WB**: Write result back to register
(επανεγγραφή αποτελέσματος σε καταχωρητή)

Figure 9:

Διαδρομή δεδομένων MIPS με διοχέτευση

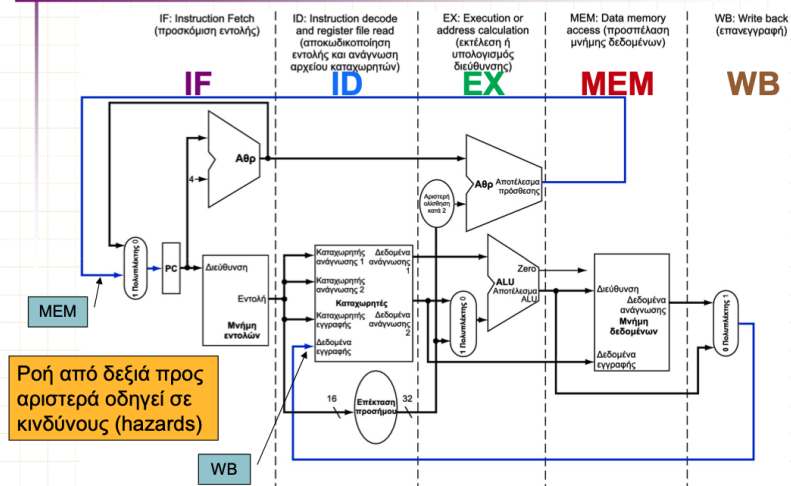


Figure 10: