



ΕΘΝΙΚΟ ΜΕΤΣΟΒΙΟ ΠΟΛΥΤΕΧΝΕΙΟ
 ΣΧΟΛΗ ΗΛΕΚΤΡΟΛΟΓΩΝ ΜΗΧΑΝΙΚΩΝ ΚΑΙ ΜΗΧΑΝΙΚΩΝ ΥΠΟΛΟΓΙΣΤΩΝ
 ΤΟΜΕΑΣ ΤΕΧΝΟΛΟΓΙΑΣ ΠΛΗΡΟΦΟΡΙΚΗΣ ΚΑΙ ΥΠΟΛΟΓΙΣΤΩΝ
 ΕΡΓΑΣΤΗΡΙΟ ΥΠΟΛΟΓΙΣΤΙΚΩΝ ΣΥΣΤΗΜΑΤΩΝ
 www.cslab.ece.ntua.gr

2η ΑΣΚΗΣΗ ΣΤΗΝ ΑΡΧΙΤΕΚΤΟΝΙΚΗ ΥΠΟΛΟΓΙΣΤΩΝ
Ακ. έτος 2010-2011, 5ο Εξάμηνο Σχολή ΗΜ&ΜΥ

Δίνεται ο παρακάτω κώδικας :

```

Loop:   lw    $1, 0($2)
        addi $1, $1, 1
        sw   $1, 0($2)
        addi $2, $2, 4
        sub  $4, $3, $2
        bne  $4, $0, Loop
  
```

Exit:

Αφού η αρχική τιμή του \$3 είναι \$2+320 και στο τέλος κάθε iteration η τιμή του μειώνεται κατά 4, το loop θα εκτελεστεί συνολικά $320/4 = 80$ φορές.

A) Στο loop αυτό έχουμε τις εξής εξαρτήσεις δεδομένων:

1. RAW : 2-1(\$1), 3-2(\$1), 5-4(\$2), 6-5(\$4)
2. WAR : 4-3(\$2)
3. WAW: 2-1(\$1)

Στο pipeline που εξετάζουμε, hazards μπορούν να δημιουργήσουν μόνο οι RAW εξαρτήσεις. Τα hazards αντιμετωπίζονται με stalls εφόσον δεν υπάρχει σχήμα προώθησης, όπως φαίνεται και στο διάγραμμα χρονισμού.

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22
lw \$1,0(\$2)	IF	ID	EX	M	WB																	
addi \$1,\$1,1		IF	ID	-	-	EX	M	WB														
sw \$1,0(\$2)			IF	-	-	ID	-	-	EX	M	WB											
addi \$2,\$2,4						IF	-	-	ID	EX	M	WB										
sub \$4,\$3,\$2									IF	ID	-	-	EX	M	WB							
bne \$4,\$0,Loop										IF	-	-	ID	-	-	EX	M	WB				
lw \$1,0(\$2)																		IF	ID	EX	M	WB

Συνολικά, για την εκτέλεση του βρόχου απαιτούνται $79*17+18 = 1361$ κύκλοι.

B)

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22
lw \$1, 0 (\$2)	IF	ID	EX	M	WB																	
addi \$1, \$1, 1		IF	ID	-	EX	M	WB															
sw \$1, 0 (\$2)			IF	-	ID	EX	M	WB														
addi \$2, \$2, 4					IF	ID	EX	M	WB													
sub \$4, \$3, \$2						IF	ID	EX	M	WB												
bne \$4, \$0, Loop							IF	ID	EX	M	WB											
lw \$1, 0 (\$2)											IF	ID	EX	M	WB							

Συνολικά, για την εκτέλεση του βρόχου απαιτούνται $79 \cdot 10 + 11 = 801$ κύκλοι.

Γ) Η χρήση της προώθησης αντιμετώπισε σχεδόν όλες τις καθυστερήσεις με εξαίρεση το 1 stall που χρειάζεται μετά το load της 1^{ης} εντολής. Το πρόβλημα αυτό θα μπορούσε να επιλυθεί με κατάλληλη αναδιάταξη του κώδικα, έτσι ώστε ο επεξεργαστής να κάνει κάτι “χρήσιμο” μεταξύ του load και του addi. Πιο συγκεκριμένα μεταφέρουμε την εντολή addi \$2,\$2, 4 στη δεύτερη θέση αλλάζοντας βέβαια κατάλληλα και το offset της εντολής sw. Το διάγραμμα χρονισμού τώρα θα είναι ως εξής:

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22
lw \$1, 0 (\$2)	IF	ID	EX	M	WB																	
addi \$2, \$2, 4		IF	ID	EX	M	WB																
addi \$1, \$1, 1			IF	ID	EX	M	WB															
sw \$1, -4 (\$2)				IF	ID	EX	M	WB														
sub \$4, \$3, \$2					IF	ID	EX	M	WB													
bne \$4, \$0, Loop						IF	ID	EX	M	WB												
lw \$1, 0 (\$2)											IF	ID	EX	M	WB							

Συνολικά, για την εκτέλεση του βρόχου απαιτούνται $79 \cdot 9 + 10 = 721$ κύκλοι.