

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

2η ΑΣΚΗΣΗ ΣΤΗΝ ΑΡΧΙΤΕΚΤΟΝΙΚΗ ΥΠΟΛΟΓΙΣΤΩΝ Ακ. έτος
 2021-2022, 5ο Εξάμηνο, Σχολή ΗΜ&ΜΥ

Τσίπη Αργυρώ
031 19950

1. Δε διαθέτει σχήμα προώθησης - without forwarding

Αρχικά, βρίσκουμε τις εξαρτήσεις όπως φαίνονται με τα αντίστοιχα χρώματα στον παρακάτω κώδικα:

```

LOOP:    LW $t1, 0($t0)
          ADD $t3, $t1, $t2
          SW $t3, 0($t1)
          ADD $t4, $t4, $t3
          LW $t2, 0($t4)
          SUB $t2, $t2, $t3
          ADDI $t6, $t6, 4
          LW $t4, 0($t6)
          ADD $t0, $t2, $t4
          ADDI $t9, $t9, -1
          BNEZ $t9, LOOP
  
```

Το διάγραμμα είναι το εξής:

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32
1	F	D	X	M	W																											
2		F	D	-	-	X	M	W																								
3			F	-	-	D	-	-	X	M	W																					
4						F	-	-	D	X	M	W																				
5									F	D	-	-	X	M	W																	
6									F	-	-	D	-	-	X	M	W															
7											F	-	-	D	X	M	W															
8															F	D	-	-	X	M	W											
9																F	-	-	D	-	-	X	M	W								
10																			F	-	-	D	X	M	W							
11																						F	D	-	-	X	M	W				
1																												F	D	X	M	W

$$200 \times 16 / 1 \times 10 = 512$$

$$511 \times 27 + 29 = 13.826 \text{ ΚΥΚΛΟΙ}$$

Οι κίνδυνοι που προκύπτουν είναι οι εξής:

Stall σε κύκλους 4&5, 7&8, 11&12, 14&15, 18&19, 21&22 λόγω εξαρτήσεων των εντολών.

Οι παραπάνω κίνδυνοι μπορούν να αποφευχθούν είτε αναδιατάσσοντας τις εντολές κατάλληλα, είτε χρησιμοποιώντας σχήμα προώθησης.

2. ΔΙΑΘΕΤΕΙ ΣΧΗΜΑ ΠΡΟΩΘΗΣΗΣ - WITH FORWARDING

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32
1	F	D	X	M	W																											
2		F	D	-	X	M	W																									
3			F	-	D	X	M	W																								
4				F	D	X	M	W																								
5					F	D	X	M	W																							
6						F	D	-	X	M	W																					
7							F	-	D	X	M	W																				
8								F	D	X	M	W																				
9									F	D	-	X	M	W																		
10										F	-	D	X	M	W																	
11											F	D	X	M	W																	
1																	F	D	X	M	W											

$$511 \cdot 16 + 18 = 8.194 \text{ ΚΥΚΛΟΙ}$$

Οι προωθήσεις που γίνονται φαίνονται στο διάγραμμα ως βελάκια.

3. ΔΙΑΡΚΕΙΑ ΚΥΚΛΟΥ ΣΩΛΗΝΩΣΗΣ

Η διάρκεια του κύκλου υπολογίζεται με βάση το στάδιο που καταλαμβάνει τον περισσότερο χρόνο, δηλαδή εδώ το στάδιο MEM 500ps.

$$\text{Αρα, } CT = 500\text{ps} + 20\text{ps} \text{ (λόγω ανανέωσης καταχωρητή μεταξύ σταδίων)} = 520 \text{ ps}$$

$$\text{Latency} = 5 \cdot CT = 5 \cdot 520 = 2.600$$

4.ΒΕΛΤΙΩΣΗ ΕΠΙΔΟΣΗΣ ΕΠΕΞΕΡΓΑΣΤΗ

Θα σπάσουμε το στάδιο Memory Access σε δύο στάδια των 250ps. Το νέο χρονοβόρο στάδιο είναι το ID 400ps. Η διάρκεια του νέου κύκλου ρολογιού είναι $CT = 400ps + 20ps$ (λόγω ανανέωσης καταχωρητή μεταξύ σταδίων). Άρα, σύνολο 420ps. Και $latency = 6 * CT = 6 * 420 = 2520ps$.

Το διάγραμμα είναι το εξής:

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32
1	F	D	X	M	M	W																										
2		F	D	-	X	M	M	W																								
3			F	-	D	X	M	M	W																							
4				F	D	X	M	M	W																							
5					F	D	X	M	M	W																						
6						F	D	-	X	M	M	W																				
7							F	-	D	X	M	M	W																			
8								F	D	X	M	M	W																			
9									F	D	-	X	M	W																		
10										F	-	D	X	M	M	W																
11											F	D	X	M	M	W																
1																		F	D	X	M	M	W									

$$511 * 17 + 19 = 8.706 \text{ ΚΥΚΛΟΙ}$$

Οι προωθήσεις που γίνονται φαίνονται στο διάγραμμα ως βελάκια.

5. ΜΕ ΑΝΑΔΙΑΤΑΞΗ & ΠΡΟΩΘΗΣΗ ο κώδικας γίνεται:

LOOP:

```

1. LW $t1, 0($t0)
2. ADDI $t6, $t6, 4
3. ADD $t3, $t1, $t2
4. SW $t3, 0($t1)
5. ADD $t4, $t4, $t3
6. LW $t2, 0($t4)
7. ADDI $t9, $t9, -1
8. SUB $t2, $t2, $t3
9. ADD $t0, $t2, $t4
10. LW $t4, 0($t6)
11. BNEZ $t9, LOOP

```

Το διάγραμμα είναι το εξής:

[illegible]
$$511 \cdot 13 + 15 = 6.658 \text{ ΚΥΚΛΟΙ}$$

Οι προωθήσεις που γίνονται φαίνονται στο διάγραμμα ως βελάκια.

6. ΜΕ ΑΝΑΔΙΑΤΑΞΗ & ΠΡΟΩΘΗΣΗ ο κώδικας γίνεται:

LOOP:

```

1. LW $t1, 0($t0)
2. ADDI $t6, $t6, 4
3. ADD $t3, $t1, $t2
4. SW $t3, 0($t1)
5. ADD $t4, $t4, $t3
6. LW $t2, 0($t4)
7. ADDI $t9, $t9, -1
8. SUB $t2, $t2, $t3
9. ADD $t0, $t2, $t4
10. LW $t4, 0($t6)
11. BNEZ $t9, LOOP

```

[illegible]

$511 \cdot 14 + 16 = 7.170$ ΚΥΚΛΟΙ

Οι προωθήσεις που γίνονται φαίνονται στο διάγραμμα ως βελάκια.

7. ΚΑΛΥΤΕΡΗ ΕΠΙΔΟΣΗ

Βέλτιστη επίδοση παρατηρούμε στην περίπτωση όπου χρησιμοποιήσαμε και προώθηση και αναδιάρθρωση κώδικα. Όχι μόνο αποφύγαμε τα stalls αλλά μειώσαμε και τη διάρκεια κύκλου.