

**HY425****Θεωρητική Άσκηση 3****Ανάθεση: 14/11/2008****Προθεσμία: 24/11/2008**

Οδηγίες: Απαντήστε στα ερωτήματα χρησιμοποιώντας κειμενογράφο της αρεσκείας σας και υποβάλετε τις απαντήσεις σε ένα αρχείο μορφής .pdf, με e-mail στο βοηθό του μαθήματος ([kavadias@ics.forth.gr](mailto:kavadias@ics.forth.gr)). Χρησιμοποιήστε την επικεφαλίδα: HY425: HW3 στο μήνυμά σας.

**Άσκηση 1 (5 ισοδύναμα ερωτήματα)**

Δίνεται επεξεργαστής με 4 βαθμίδες (IF, ID, EX, WB). Η ανάγνωση των καταχωρητών να γίνεται στο ID και η ανάγνωση/εγγραφή μνήμης γίνεται στο EX για τις εντολές LD, SD. Η εγγραφή των αποτελεσμάτων στους καταχωρητές γίνεται στο WB. Υποθέτουμε ότι όλες οι βαθμίδες έχουν καθυστέρηση ενός κύκλου, εκτός της βαθμίδας EX, λόγω των διαφορετικών καθυστερήσεων των μονάδων εκτέλεσης. Οι διαθέσιμες μονάδες εκτέλεσης του επεξεργαστή έχουν τις εξής καθυστερήσεις:

| <b>Καθυστερήσεις μονάδων εκτέλεσης</b> |          |
|----------------------------------------|----------|
| Memory LD                              | 4 κύκλοι |
| Memory SD                              | 2 κύκλοι |
| Integer ADD, SUB                       | 1 κύκλος |
| Branches                               | 2 κύκλοι |
| ADDD (floating point)                  | 3 κύκλοι |
| MULTD (floating point)                 | 5 κύκλοι |

Παρατηρήστε ότι υπάρχουν συνολικά 6 μονάδες εκτέλεσης (load, store, add/sub, branch, addd, multd).

Δίνεται ο παρακάτω κώδικας:

```
Loop:      LD      F2, 0(Rx)
I0:        LD      F4, 0(Ry)
I1:        MULTD   F6, F2, F4
I2:        ADDD    F8, F6, F8
I3:        ADDI    Rx, Rx, #8
I4:        SUBI    Ry, Ry, #8
I5:        SUB     R20, R4, Rx
I6:        BNZ     R20, Loop
```

- a. Υποθέστε ότι i) ο επεξεργαστής μπορεί να κάνει issue μία εντολή ανά κύκλο; ii) μετράμε χρόνο από τη στιγμή που η πρώτη εντολή LD ξεκινάει την εκτέλεσή της στη βαθμίδα EX; iii) ο επεξεργαστής δεν υποστηρίζει forwarding, κατά συνέπεια εάν μία εντολή καταναλώνει το αποτέλεσμα το οποίο παράγει μία προηγούμενη εντολή, πρέπει υποχρεωτικά να περιμένει στο ID μέχρι να γίνει writeback του αποτελέσματος; iv) μία

εντολή μπορεί να κάνει stall λόγω μη διαθεσιμότητας μονάδας εκτέλεσης και σε αυτή την περίπτωση η εντολή περιμένει στο ID; v) το branch έχει ένα delay slot δηλ. στον κύκλο που ακολουθεί το IF του branch μπορούμε να δρομολογήσουμε μία ανεξάρτητη εντολή και να επικαλύψουμε την καθυστέρηση ενός κύκλου λόγω του branch; vi) το branch είναι πάντα taken. Πόσους κύκλους ανά επανάληψη απαιτεί ο κώδικας του βρόχου που σας δίνεται;

*Σημείωση: Απαντήστε αυτή και τις επόμενες ερωτήσεις σχεδιάζοντας λεπτομερώς την εκτέλεση των εντολών στο pipeline του επεξεργαστή.*

- b. Υποθέστε ότι ο επεξεργαστής υποστηρίζει forwarding από το τέλος του EX stage σε οποιαδήποτε εντολή τελειώνει το ή περιμένει στο ID stage. Πόσους κύκλους ανά επανάληψη απαιτεί ο κώδικας που σας δίνεται; Ισχύουν όλες οι άλλες υποθέσεις του ερωτήματος α.
- c. Υποθέστε ότι στον επεξεργαστή υλοποιείται multiple issue logic, που σας επιτρέπει να κάνετε issue 2 εντολές ανά κύκλο. Υποθέστε επίσης ότι στον επεξεργαστή διπλασιάζουμε τον αριθμό των μονάδων εκτέλεσης, προσθέτοντας μία επιπλέον μονάδα του κάθε είδους που ήταν διαθέσιμο στον επεξεργαστή των ερωτημάτων α. και b. Ισχύουν όλες οι άλλες υποθέσεις που κάναμε στο ερώτημα b., πλην της υπόθεσης ότι ο επεξεργαστής κάνει issue μία εντολή ανά κύκλο. Πόσους κύκλους ανά επανάληψη απαιτεί ο κώδικας που σας δίνεται;
- d. Εφαρμόστε την τεχνική loop unrolling στον κώδικα που σας δίνεται για να αυξήσετε τον παραλληλισμό μεταξύ εντολών. Συγκεκριμένα κάνετε unroll το βρόχο 2 φορές (δηλ. αλλάξτε τον κώδικα ώστε να εκτελεί 3 επαναλήψεις του αρχικού βρόχου σε κάθε επανάληψη του unrolled βρόχου). Κάντε τις ίδιες υποθέσεις με το ερώτημα b. (single issue επεξεργαστής). Θεωρήστε ότι υπάρχουν πάντα διαθέσιμοι επιπλέον καταχωρητές για να γίνει το unrolling. Πόσους κύκλους ανά επανάληψη απαιτεί ο unrolled κώδικας;
- e. Επαναλάβετε το ερώτημα d., υποθέτωντας τώρα ότι έχετε τον multiple issue επεξεργαστή του ερωτήματος c. και ότι κάνετε unroll το βρόχο 3, αντί για 2 φορές. Πόσους κύκλους ανά επανάληψη απαιτεί ο unrolled κώδικας;