

Άσκηση 9: Εισαγωγή στην Ομοχειρία (Pipelining - Διοχέτευση)

Προθεσμία έως Δευτέρα 16 Απριλίου 2018, ώρα 23:59 (βδ. 9.1)

Βιβλίο (4η έκδοση): Διαβάστε τις ενότητες 4.5 έως και 4.8, σελίδες 389 - 448.

9.1 Εισαγωγικά, Παροχή, Καθυστέρηση, το Παράδειγμα της Μπουγάδας:

Διαβάστε τις σελίδες 389 - 392, και δείτε το video της διάλεξης στο elearn.uoc.gr/mod/page/view.php?id=7192

9.2 Η Βασική Ιδέα του Pipelined Datapath:

Δείτε το video της διάλεξης στο elearn.uoc.gr/mod/page/view.php?id=7193, και διαβάστε από το βιβλίο τις σελίδες 405 - 409.

9.3 Παροχή και Καθυστέρηση, ξανά:

Διαβάστε τις σελίδες 393 - 394, και δείτε το video της διάλεξης στο elearn.uoc.gr/mod/page/view.php?id=7194

9.4 Λεπτομερής Λειτουργία του Pipelined Datapath χωρίς Αλληλεξαρτήσεις ή Διακλαδώσεις:

Δείτε το video της διάλεξης στο elearn.uoc.gr/mod/page/view.php?id=7236, και διαβάστε από το βιβλίο τις σελίδες 409 - 417.

9.5 Γραφική Αναπαράσταση της Ομοχειρίας:

Δείτε το video της διάλεξης στο elearn.uoc.gr/mod/page/view.php?id=7237, και διαβάστε από το βιβλίο τις σελίδες 417 - 420.

9.6 Μονάδα Ελέγχου για την Ομοχειρία:

Δείτε το video της διάλεξης στο elearn.uoc.gr/mod/page/view.php?id=7238, και διαβάστε από το βιβλίο τις σελίδες 420 - 425.

9.7 Εντολές ALU: 4 ή 5 Βαθμίδες;

Δείτε το video της διάλεξης στο elearn.uoc.gr/mod/page/view.php?id=7239

9.8 Αλληλεξαρτήσεις μεταξύ Εντολών ALU: Προσπέρασμα

Δείτε το video της διάλεξης στο elearn.uoc.gr/mod/page/view.php?id=7289, και διαβάστε από το βιβλίο τις σελίδες 395 - 397, καθώς και τις 425 - 434 οι οποίες καλύπτουν επίσης και τα παρακάτω θέματα 9.9 και 9.10. Στις σελίδες 430-434, λάβετε υπ όψη σας ότι εμείς στις διαλέξεις βάλαμε το κύκλωμα ελέγχου / ανίχνευσης για την ύπαρξη αλληλεξαρτήσεων και την απόφαση προσπεράσματος στη δεύτερη βαθμίδα της pipeline, όπως κάνουν οι κανονικοί επεξεργαστές, και όχι στην τρίτη όπως κάνει το βιβλίο για λόγους απλότητας. Ο λόγος να βάλει κανείς το κύκλωμα αυτό στη δεύτερη βαθμίδα είναι ότι εκεί υπάρχει "ελεύθερος" χρόνος

να γίνει αυτή η δουλειά εν παραλλήλω με την ανάγνωση καταχωρητών (δηλαδή χωρίς να χάνεται επιπλέον χρόνος ειδικά για αυτή τη δουλειά), ενώ εάν αυτό τοποθετηθεί στην τρίτη βαθμίδα, τότε στην αρχή του κύκλου ρολογιού, στην τρίτη βαθμίδα, το datapath δεν κάνει τίποτα χρήσιμο περιμένοντας να αποφασίσει το κύκλωμα αυτό εάν πρέπει ή δεν πρέπει να γίνει προσπέρασμα, και από πού.

9.9 Datapath για Προσπεράσματα:

Δείτε το video της διάλεξης στο elearn.uoc.gr/mod/page/view.php?id=7290, και διαβάστε αυτά που έγραφε το θέμα 9.8 παραπάνω.

9.10 Έλεγχος για Προσπεράσματα:

Δείτε το video της διάλεξης στο elearn.uoc.gr/mod/page/view.php?id=7291, και διαβάστε αυτά που έγραφε το θέμα 9.8 παραπάνω.

9.11 Εξάρτηση από προηγούμενη Εντολή Load: Αναμονή

Δείτε το video της διάλεξης στο elearn.uoc.gr/mod/page/view.php?id=7292, και διαβάστε από το βιβλίο τις σελίδες 397 - 398, καθώς και τις 434 - 438 οι οποίες καλύπτουν επίσης και το παρακάτω θέμα 9.12.

9.12 Κύκλωμα Ελέγχου για Αναμονή:

Δείτε το video της διάλεξης στο elearn.uoc.gr/mod/page/view.php?id=7293, και διαβάστε αυτά που έγραφε το θέμα 9.11 παραπάνω.

9.13 Στατική Αναδιάταξη Εντολών για την Αποφυγή Αναμονών:

Δείτε το video της διάλεξης στο elearn.uoc.gr/mod/page/view.php?id=7294, και διαβάστε από το βιβλίο τις σελίδες 398 - 399.

9.14 Αλληλεξαρτήσεις RAW, WAW, WAR:

Δείτε το video της διάλεξης στο elearn.uoc.gr/mod/page/view.php?id=7295

9.15 Οι Αλληλεξαρτήσεις στη δική μας Pipeline, Συνολικά:

Δείτε το video της διάλεξης στο elearn.uoc.gr/mod/page/view.php?id=7311

9.16 Διακλαδώσεις και Άλματα, Υπόθεση Αποτυχίας της Διακλάδωσης και Ακύρωση Εντολών εάν αυτή Επιτύχει:

Δείτε το video της διάλεξης στο elearn.uoc.gr/mod/page/view.php?id=7312, και διαβάστε από το βιβλίο τις σελίδες 399 - 405 και 438 - 442.

9.17 Πρόβλεψη Διακλαδώσεων:

Δείτε το video της διάλεξης στο elearn.uoc.gr/mod/page/view.php?id=7313, και διαβάστε από το βιβλίο τις σελίδες 442 - 448.

Άσκηση 9: Οπτικοποίηση του Pipelining:

Στην άσκηση αυτή θα χρησιμοποιήσετε έναν προσομοιωτή για την οπτικοποίηση της λειτουργίας της απλής pipeline του MIPS που είδαμε στο μάθημα. Πρόκειται για το εργαλείο

"Hase" που έχει αναπτυχθεί στο Πανεπιστήμιο του Εδιμβούργου, που μπορεί να χρησιμοποιηθεί για πλήθος σκοπών, μεταξύ των οποίων και για την οπτικοποίηση της δικής μας pipeline (ή τέλος-πάντων κάποιας παραλλαγής της), και που είναι διαθέσιμο δωρεάν από το Διαδίκτυο: www.icsa.inf.ed.ac.uk/research/groups/hase/ –αλλά που σας προτείνουμε να το πάρετε και εγκαταστήσετε από την περιοχή του μαθήματός μας, σύμφωνα με τις παρακάτω οδηγίες. Δεδομένου ότι χρειάζεται γραφικό περιβάλλον για να παίξει, μάλλον θα χρειαστεί να το εγκαταστήσετε στον προσωπικό σας υπολογιστή –μάλλον δεν θα καταφέρετε να το τρέξετε remotely από τους υπολογιστές του CSD. Διαβάστε και ακολουθήστε τις Οδηγίες Εγκατάστασης και Χρήσης του Hase από τη διεύθυνση:

www.csd.uoc.gr/~hy225/18a/ex09_hase_man.pdf

(1) Απαριθμήστε τις βαθμίδες (στάδια) της pipeline του επεξεργαστή και για την κάθε μία περιγράψτε περιληπτικά (2-3 προτάσεις) την λειτουργία της.

(2) Ποιές βαθμίδες είναι απαραίτητες για όλες τις εντολές; Δηλαδή, ποιά στάδια της pipeline εκτελούν κάποια χρήσιμη λειτουργία για όλες τις εντολές, ανεξαρτήτως τύπου;

(3) Στην Pipeline υπάρχουν κάποια στάδια τα οποία δεν εκτελούν κάποια χρήσιμη λειτουργία για κάποιους τύπους εντολών. Ποιά στάδια είναι αυτά; Ποιο στάδιο δεν κάνει κάποια χρήσιμη λειτουργία για τις εντολές αριθμητικής; Ποιο στάδιο δεν κάνει κάποια χρήσιμη λειτουργία για τις εντολές store;

(4) Ακολουθήστε τις παραπάνω οδηγίες ώστε να εγκαταστήσετε στην περιοχή σας τον προσομοιωτή Hase. Παρατηρήστε την εκτέλεση των εντολών και πώς αυτές μεταβαίνουν από το ένα στάδιο της pipeline στο επόμενο μέχρι την ολοκλήρωσή τους. Γράψτε μερικές δικές σας αλληλουχίες εντολών και παρατηρήστε την εκτέλεσή τους. Συγκεκριμένα, γράψτε μερικές εντολές αριθμητικών πράξεων μεταξύ καταχωρητών, μερικές μεταξύ καταχωρητών και σταθερών, καθώς και μερικές εντολές load και store. Για κάθε μία από αυτές τις (τρεις) περιπτώσεις γράψτε στην αναφορά σας τις εντολές που εκτελέσατε (τα περιεχόμενα του αρχείου "MEMORY.instr_mem.mem") συνοδευόμενα από ένα screenshot από την προσομοίωση.

(5) τρέξτε τις εξής δυο ακολουθίες εντολών:

ADD R7 R8 R9	ADD R7 R8 R9
ADD R4 R5 R6	ADD R4 R7 R6
BREAK	BREAK

Υπάρχει διαφορά ανάμεσα στον χρόνο εκτέλεσης των δυο ακολουθιών; Πόση είναι και πού οφείλεται; Στο σχηματικό Pipeline (κατω αριστερά στο σχήμα του datapath) μπορείτε να δείτε ποιά εντολή βρίσκεται σε κάθε στάδιο του Pipeline, τι διαφορετικό παρατηρείτε στην εκτέλεση των δυο ακολουθιών; Ποιά εντολή περιμένει στη δεύτερη ακολουθία; Σε ποιο στάδιο και γιατί; (Παρατηρήστε ότι ο Hase υποθέτει μία pipeline όχι ακριβώς ίδια με αυτήν που εμείς περιγράψαμε στο μάθημα, ή τέλος-πάντων όχι με όλα όσα είπαμε εμείς στο μάθημα...).

(6) Βασισμένοι στις διαπιστώσεις της προηγούμενης ερώτησης μπορείτε να πείτε αν υπάρχει Forwarding στο datapath που προσομοιώσατε; Τι θα άλλαζε αν ήταν διαφορετικά απ' ότι είναι; Θα προχωρούσε στην εκτέλεση η εντολή που περιμένει για το αποτέλεσμα της προηγούμενης νωρίτερα/αργότερα αν ήταν διαφορετικά;

Τρόπος Παράδοσης:

Παραδώστε την άσκηση αυτή on-line, σε μορφή PDF (μόνον) (μπορεί να είναι κείμενο μηχανογραφημένο ή/και "σκαναρισμένο" χειρόγραφο, αλλά *μόνον* σε μορφή PDF). Παραδώστε μέσω **turnin ex09@hy225 [directoryName]** ένα αρχείο ονόματι **ex09.pdf** που θα περιέχει τις απαντήσεις σας σε όλες τις παραπάνω ερωτήσεις (1) έως και (6).

Θα εξεταστείτε **και προφορικά** για τις ασκήσεις 8 και 9 (μαζί), από βοηθό του μαθήματος, με διαδικασία για την οποία θα ενημερωθείτε μέσω ηλτά (email) στη λίστα του μαθήματος.