

Πανεπιστήμιο Θεσσαλίας
Τμήμα Πληροφορικής

Αρχιτεκτονική Υπολογιστών

Ασκήσεις Χειμερινού Εξαμήνου 2018-2019
(μέρος Α')

Άσκηση 1:

Θεωρήστε κάποιον επεξεργαστή γενικού σκοπού που λειτουργεί σε συχνότητα 2,22GHz και εκτελεί μόνο εντολές σταθερής υποδιαστολής. Στις εντολές αυτές συμπεριλαμβάνονται εντολές που υλοποιούν δύο σύνθετες λειτουργίες, έστω Λ_A και Λ_B , σε 12 και 48 φορές τον αριθμό κύκλων των απλών εντολών, αντίστοιχα. Επιπλέον, ο επεξεργαστής αυτός εξομοιώνει την εκτέλεση εντολών κινητής με εντολές σταθερής υποδιαστολής μέσα από κατάλληλη βιβλιοθήκη, έτσι ώστε τρεις διαφορετικές κατηγορίες λειτουργιών κινητής υποδιαστολής, έστω Λ_Γ , Λ_Δ και Λ_E , να απαιτούν κατά μέσο όρο 24, 46 και 280 εντολές σταθερής υποδιαστολής, αντίστοιχα, στις οποίες η Λ_A εμφανίζεται 3, 4 και 27 φορές, και η Λ_B εμφανίζεται 0, 1 και 9 φορές, αντίστοιχα.

Σε μια νέα έκδοση του επεξεργαστή, καταφέραμε να επιταχύνουμε τη λειτουργία Λ_A κατά έναν παράγοντα 3,6 πάνω στον αριθμό κύκλων ανά εντολή. Για να επιτύχουμε κάτι τέτοιο όμως, αναγκαστήκαμε να περιορίσουμε το υλικό της λειτουργίας Λ_B με αποτέλεσμα την επιβράδυνσή της 2,1 φορές πάνω στον αριθμό κύκλων ανά εντολή. Επίσης προσθέσαμε υλικό κινητής υποδιαστολής, που υλοποιεί σε εντολές τις παραπάνω λειτουργίες Λ_Γ , Λ_Δ και Λ_E σε 8, 15 και 64 φορές τον αριθμό κύκλων μηχανής των απλών εντολών σταθερής υποδιαστολής, αντίστοιχα. Ο νέος επεξεργαστής λειτουργεί σε συχνότητα 3,33GHz.

Θεωρήστε κάποια εφαρμογή, η οποία μεταφράζεται και για τους δύο επεξεργαστές από τον ίδιο μεταγλωττιστή με τις ίδιες επιλογές μετάφρασης, εκτός από την επιλογή που καθορίζει τον τρόπο μετάφρασης των πράξεων κινητής υποδιαστολής. Όταν η εφαρμογή εκτελείται στο βελτιωμένο επεξεργαστή, χρησιμοποιεί τις πέντε λειτουργίες $\Lambda_A - \Lambda_E$ σε ποσοστά 31%, 7%, 4%, 2% και 1% του χρόνου εκτέλεσης.

Για τις παραπάνω τροποποιήσεις:

A. Ποια είναι η επιτάχυνση απόδοσης του επεξεργαστή για ολόκληρη την εφαρμογή;

B. Ποιο ποσοστό του χρόνου εκτέλεσης της εφαρμογής στον αρχικό επεξεργαστή αντιστοιχεί στη λειτουργία Λ_A , ποιο στη λειτουργία Λ_B και ποιο στην εξομοίωση καθεμιάς από τις τρεις λειτουργίες κινητής υποδιαστολής Λ_Γ , Λ_Δ και Λ_E ; Για τον υπολογισμό των δύο πρώτων να μη συμπεριλάβετε τις Λ_A και Λ_B που εμπεριέχονται στις Λ_Γ , Λ_Δ και Λ_E .

Γ. Έστω τώρα ότι ένας νέος μεταγλωττιστής μεταφράζει ξανά τον κώδικα της εφαρμογής με τέτοιο τρόπο, ώστε να μη χρησιμοποιεί καθόλου τη λειτουργία Λ_B , και οι εντολές που προηγουμένως εκτελούσαν τη λειτουργία αυτή τώρα να εκτελούν απλές λειτουργίες σταθερής υποδιαστολής. Όμως κάθε εντολή λειτουργίας Λ_B μετατρέπεται σε a εντολές απλών λειτουργιών κατά μέσο όρο, κι επομένως έχουμε αύξηση του πλήθους των εντολών της εφαρμογής. Να βρείτε πόσο το πολύ πρέπει να είναι το a , ώστε να μη συμβεί επιβράδυνση (i) στον αρχικό επεξεργαστή έναντι του παλαιού μεταγλωττιστή, (ii) στο βελτιωμένο επεξεργαστή έναντι του παλαιού μεταγλωττιστή, και (iii) στο βελτιωμένο έναντι του αρχικού επεξεργαστή για το νέο μεταγλωττιστή.

Δ. Σε επόμενη έκδοση, επιπλέον της απαλοιφής της λειτουργίας Λ_B από την εφαρμογή, ο νέος μεταγλωττιστής συνοδεύεται από νέα βελτιωμένη βιβλιοθήκη κινητής υποδιαστολής, η οποία εξομοιώνει τις λειτουργίες Λ_Γ , Λ_Δ και Λ_E , έτσι ώστε τώρα να απαιτούν κατά μέσο όρο 12, 38 και 120 εντολές σταθερής υποδιαστολής, αντίστοιχα, στις οποίες η Λ_A εμφανίζεται 3, 5 και 16 φορές, αντίστοιχα, ενώ η Λ_B δεν εμφανίζεται καθόλου. Αν $a = 8,6$, να βρείτε την επιτάχυνση

στην εκτέλεση της εφαρμογής (i) στον αρχικό επεξεργαστή έναντι του αρχικού μεταγλωττιστή, (ii) στον αρχικό επεξεργαστή έναντι του μεταγλωττιστή του προηγούμενου ερωτήματος, και (iii) στο βελτιωμένο έναντι του αρχικού επεξεργαστή για το νέο μεταγλωττιστή.

Άσκηση 2:

Ένα μετροπρόγραμμα εκτελεί 195578 πράξεις κινητής υποδιαστολής, σύμφωνα με τον παρακάτω πίνακα:

Τύπος πράξεων	Αριθμός πράξεων
Add	82014
Subtract	8229
Multiply	73220
Divide	21399
Convert	6006
Compare	4710
Σύνολο	195578

Υποθέστε ότι το μετροπρόγραμμα μεταφράζεται και εκτελείται σε έναν επεξεργαστή RISC και στη συνέχεια ξαναμεταφράζεται και εκτελείται σε έναν επεξεργαστή ειδικού σκοπού, που αποτελεί παραλλαγή του πρώτου. Ο πρώτος επεξεργαστής εκτελεί όλες τις πιο πάνω πράξεις με αντίστοιχες εντολές κινητής υποδιαστολής. Ο δεύτερος επεξεργαστής υλοποιεί το ίδιο σύνολο εντολών σταθερής υποδιαστολής με τον πρώτο, αλλά δεν περιέχει υπομονάδες κινητής υποδιαστολής, για λόγους κόστους, κατανάλωσης ενέργειας, καθώς και έλλειψης αναγκαιότητας υποστήριξής τους για το εύρος εφαρμογών του επεξεργαστή. Οι όποιες πράξεις κινητής υποδιαστολής εξομοιώνονται στο δεύτερο επεξεργαστή μέσω βιβλιοθήκης υποπρογραμμάτων σταθερής υποδιαστολής. Οι δύο μεταφράσεις του μετροπρογράμματος γίνονται από τον ίδιο μεταγλωττιστή με τις ίδιες επιλογές μετάφρασης, εκτός από την επιλογή που καθορίζει τον τρόπο μετάφρασης των πράξεων κινητής υποδιαστολής.

Έστω λοιπόν ότι το μετροπρόγραμμα αυτό ολοκληρώνεται σε 60,8 msec στον πρώτο, και σε 106,4 msec στο δεύτερο επεξεργαστή. Ο μέσος αριθμός CPI για το μετροπρόγραμμα στους δύο επεξεργαστές είναι αντίστοιχα 10,24 και 8,93. Η συχνότητα των δύο επεξεργαστών είναι αντίστοιχα 2,1 και 3,2 GHz.

A. Ποιος είναι ο συνολικός αριθμός εντολών που εκτελούνται στις δύο εκτελέσεις του μετροπρογράμματος;

B. Ποιο είναι το μέτρο MIPS και ποιο είναι το μέτρο MFLOPS για τις δύο αυτές εκτελέσεις;

Γ. Πόσες εντολές σταθερής υποδιαστολής απαιτούνται κατά μέσο όρο για την εκτέλεση μιας πράξης κινητής υποδιαστολής στο δεύτερο επεξεργαστή, για καθεμία από τις πράξεις που δίνονται στον πίνακα; Υποθέστε ότι οι πράξεις multiply και divide είναι 3 και 9 φορές πιο πολύπλοκες από τις υπόλοιπες, αντίστοιχα.

Άσκηση 3:

Έστω μια αρχιτεκτονική καταχωρητή-μνήμης, την οποία σχεδιάζουμε με τεχνική μερικής επικάλυψης. Η ΜΕΔ της αρχιτεκτονικής περιλαμβάνει ένα ΦΚ γενικού σκοπού, μια κρυφή μνήμη εντολών, μια κρυφή μνήμη δεδομένων και όποιον αριθμό ΑΛΜ απαιτούνται για τον υπολογισμό τελικής διεύθυνσης προσπέλασης μνήμης ή για την εκτέλεση αριθμητικών/λογικών πράξεων. Ο ΦΚ διαθέτει τρεις θύρες ανάγνωσης και μία θύρα εγγραφής. Η προσπέλαση μνήμης στην αρχιτεκτονική γίνεται με σχετική έμμεση (μέσω καταχωρητή) και δεικτοδοτούμενη διεύθυνσιοδότηση, που υλοποιείται με τη βοήθεια ενός καταχωρητή βάσης, ενός καταχωρητή δείκτη και μιας σταθεράς μετατόπισης.

Οι αριθμητικές/λογικές εντολές μπορούν να εκτελούν πράξεις με δύο ή τρία τελούμενα εισόδου. Για την υποστήριξη τριών τελούμενων εισόδου στη φάση εκτέλεσης, απαιτείται διάταξη δύο ΑΛΜ, όπου η δεύτερη ΑΛΜ συνδέεται στην έξοδο της πρώτης. Κάθε λειτουργία ΑΛΜ δύο εισόδων απαιτεί μισό κύκλο μηχανής, έτσι ώστε η διπλή ΑΛΜ τριών εισόδων να λειτουργεί σε έναν κύκλο μηχανής. Πολυπλέκτες μπορούν να επιτρέπουν χρήση μόνο της μιας από τις δύο ΑΛΜ σε μια εντολή.

Η ΜΕΔ αυτή θέλουμε να υποστηρίξει:

(α) Εντολές ΑΛΜ με τη μορφή:

`op Rdest, Rsrc1, Rsrc2`

όπου όλα τα τελούμενα είναι καταχωρητές, με τον Rdest να αποθηκεύει το αποτέλεσμα της πράξης op μεταξύ των περιεχομένων των Rsrc1 και Rsrc2.

(β) Εντολές ΑΛΜ με τη μορφή:

`oplop2 Rdest, Rsrc1, Rsrc2, Rsrc3`

όπου όλα τα τελούμενα είναι καταχωρητές, με τον Rdest να αποθηκεύει το αποτέλεσμα της πράξης op2 μεταξύ (i) του αποτελέσματος της πράξης op1 μεταξύ των περιεχομένων των Rsrc1 και Rsrc2 και (ii) του περιεχομένου του Rsrc3.

(γ) Εντολές ΑΛΜ με τη μορφή:

`op Rdest, Rsrc, Mem`

όπου το τελούμενο Rsrc είναι καταχωρητής γενικού σκοπού, ενώ το τελούμενο Mem είναι θέση μνήμης, εκφρασμένη είτε με έναν καταχωρητή Rbase και μια σταθερά offset, ως `offset(Rbase)`, είτε με δύο καταχωρητές Rbase και Rindex, ως `(Rbase, Rindex)`. Το τελούμενο Rdest είναι καταχωρητής, όπου αποθηκεύεται το αποτέλεσμα της πράξης op μεταξύ του περιεχομένου του Rsrc και του περιεχομένου της θέσης μνήμης Mem.

(δ) Εντολές φόρτωσης και αποθήκευσης με τη μορφή:

`load Rdest, Mem`

και

`store Rsrc, Mem`

όπου τα τελούμενα Rdest και Rsrc είναι καταχωρητές γενικού σκοπού, ενώ το τελούμενο Mem είναι θέση μνήμης, εκφρασμένη με δύο καταχωρητές γενικού σκοπού Rbase και Rind και μια σταθερά μετατόπισης offset, ως `offset(Rbase, Rind)`.

(ε) Διακλαδώσεις με τη μορφή:

`bcond Rsrc1, Rsrc2, offset`

που εκτελούνται με σύγκριση των περιεχομένων δύο καταχωρητών Rsrc1 και Rsrc2 για συνθήκη cond, και πιθανό άλμα σε διεύθυνση σχετική με τον PC.

(στ) Διακλαδώσεις με τη μορφή:

`hopcond Rsrc1, Rsrc2, Rsrc3, offset`

που εκτελούνται με λειτουργία ΑΛΜ μεταξύ των περιεχομένων των Rsrc1 και Rsrc2, σύγκριση του αποτελέσματος με το περιεχόμενο του καταχωρητή Rsrc3 για συνθήκη cond, και πιθανό άλμα σε διεύθυνση σχετική με τον PC.

(ζ) Διακλαδώσεις με τη μορφή:

`bcond Rsrc1, Rsrc2, Mem`

που εκτελούνται με σύγκριση των περιεχομένων δύο καταχωρητών Rsrc1 και Rsrc2 για συνθήκη cond, και πιθανό άλμα σε διεύθυνση που διαβάζεται από τη θέση μνήμης Mem, εκφρασμένη με έναν καταχωρητή βάσης Rbase και μια σταθερά offset, ως `offset(Rbase)`.

Σε όλες τις εντολές, op/op1/op2 μπορεί να είναι add, sub, or, and, nor ή xor, ενώ cond μπορεί να είναι eq ή ne.

Όλες οι παραπάνω εντολές κωδικοποιούνται στο ίδιο μέγεθος λέξης εντολής, έτσι ώστε ο ρυθμός προσκόμισης εντολών να είναι μία εντολή ανά κύκλο μηχανής. Ο κύκλος εντολής χωρίζεται σε φάσεις που μοιάζουν με εκείνες της αρχιτεκτονικής MIPS, με σειρά όμως που πρέπει να είναι προσαρμοσμένη σε αρχιτεκτονική καταχωρητή-μνήμης, και με πιθανή προσθήκη νέων

φάσεων που εξαρτάται από τη μορφή των εντολών. Για απλούστευση στη σχεδίαση του συστήματος μνήμης, ο κύκλος εντολής μπορεί να περιλαμβάνει μόνο μία φάση προσπέλασης της μνήμης δεδομένων.

A. Με λέξη εντολής εύρους 32 bits, κωδικό λειτουργίας 6 bits, και ΦΚ με 32 καταχωρητές γενικού σκοπού, πόσο εύρος μπορεί να έχει το πεδίο μετατόπισης σε εντολές με μετατόπιση;

B. Δώστε μια διαδοχή φάσεων του κύκλου εντολής που να καλύπτει όλες τις εντολές που υποστηρίζονται από την αρχιτεκτονική. Ονομάστε κατάλληλα τις νέες φάσεις που πιθανά προσθέσατε.

Γ. Προσδιορίστε τις γραμμές παροχέτευσης που απαιτούνται στην υλοποίηση της παραπάνω αρχιτεκτονικής, ώστε να αντιμετωπίζονται επιτυχώς κίνδυνοι εξαρτήσεων από δεδομένα. Δώστε την απάντησή σας σε έναν πίνακα, ο οποίος για καθένα συνδυασμό φάσεων που μπορούν να εμφανίσουν τέτοιους κινδύνους, να δίνει δύο πιθανές εντολές που να βρίσκονται σε αυτές τις φάσεις. Θεωρήστε ότι η ανάγνωση και η αποθήκευση στο ΦΚ μπορούν να γίνουν στον ίδιο κύκλο μηχανής, με την δεύτερη να προηγείται της πρώτης, ώστε κάποιο δεδομένο που γράφεται στο ΦΚ να μπορεί να διαβαστεί στον ίδιο κύκλο μηχανής.

Δ. Αναφέρετε όλους τους κινδύνους εξαρτήσεων από δεδομένα που δε μπορούν να αντιμετωπιστούν με επιτυχία με τις πιο πάνω γραμμές παροχέτευσης, δίνοντας τους συνδυασμούς φάσεων που μπορούν να εμφανίσουν αυτούς τους κινδύνους, καθώς και δύο πιθανές εντολές που να βρίσκονται σε αυτές τις φάσεις για καθένα συνδυασμό.

Ε. Δείξτε όλους τους δυνατούς κινδύνους από διαδικασιακές εξαρτήσεις που εμφανίζονται στην αρχιτεκτονική, δίνοντας, όπως και προηγουμένως, τους συνδυασμούς φάσεων που μπορούν να εμφανίσουν αυτούς τους κινδύνους, καθώς και δύο πιθανές εντολές που να βρίσκονται σε αυτές τις φάσεις για καθένα συνδυασμό. Υποθέστε ότι οι διακλαδώσεις εκτελούνται μετά τη φάση αποκωδικοποίησης χωρίς θέσεις καθυστέρησης. Πόσο θα είναι το κόστος αποτυχίας κάποιου μηχανισμού πρόβλεψης διακλαδώσεων σε αυτή την αρχιτεκτονική;

Άσκηση 4 (εργαστηριακή):

Να γράψετε ένα πρόγραμμα προσομοίωσης μιας ΜΕΔ μερικά επικαλυπτόμενων εντολών MIPS, στην οποία έχει ενσωματωθεί υλικό κινητής υποδιαστολής. Το πρόγραμμα θα πρέπει να δέχεται ως είσοδο κατάλληλα κωδικοποιημένο κώδικα MIPS, και να βγάζει ως έξοδο έναν πίνακα που να απεικονίζει το διάγραμμα χρονισμού από την εκτέλεση του κώδικα, διαφορώντας για τις τιμές δεδομένων που παράγονται από την εκτέλεση αυτή.

Ειδικότερα, εκτός από την υπομονάδα εκτέλεσης πράξεων σταθερής υποδιαστολής, η ΜΕΔ διαθέτει και τρεις υπομονάδες εκτέλεσης πράξεων κινητής υποδιαστολής, οι οποίες είναι συνδεδεμένες με ένα ΦΚ κινητής υποδιαστολής, και περιγράφονται ως εξής:

- (i) Η υπομονάδα πρόσθεσης αριθμών κινητής υποδιαστολής απαιτεί 3 κύκλους μηχανής για την εκτέλεση μιας πρόσθεσης, που αντιστοιχούν σε 3 μερικά επικαλυπτόμενες διαδοχικές φάσεις εκτέλεσης A1-A3.
- (ii) Η υπομονάδα πολλαπλασιασμού αριθμών κινητής υποδιαστολής απαιτεί 5 κύκλους μηχανής για την εκτέλεση ενός πολλαπλασιασμού, που αντιστοιχούν σε 5 μερικά επικαλυπτόμενες διαδοχικές φάσεις εκτέλεσης M1-M5.
- (iii) Η υπομονάδα διαίρεσης αριθμών κινητής υποδιαστολής απαιτεί 14 κύκλους μηχανής για την εκτέλεση μιας διαίρεσης, που αντιστοιχούν σε 3 μερικά επικαλυπτόμενες διαδοχικές φάσεις εκτέλεσης D1-D3, οι οποίες απαιτούν 4, 6 και 4 κύκλους μηχανής, αντίστοιχα, με τους διαδοχικούς κύκλους των τριών φάσεων να μη δέχονται επικάλυψη.

Εντολές MIPS εκτελούνται στην πιο πάνω ΜΕΔ με κάποιες παραλλαγές από την κλασική MIPS, οι οποίες περιγράφονται στη συνέχεια:

1. Η φάση ανάκλησης είναι κοινή για όλες τις εντολές (IF).
2. Η φάση αποκωδικοποίησης χωρίζεται σε δύο επικαλυπτόμενες φάσεις, από τις οποίες η πρώτη είναι κοινή για όλες τις εντολές (ID), ενώ η δεύτερη είναι διαφορετική για εντολές

σταθερής από εντολές κινητής υποδιαστολής (IDI και IDF). Η ανάγνωση τελούμενων γίνεται στη δεύτερη από τις δύο φάσεις, από τον κατάλληλο ΦΚ.

3. Οι υπόλοιπες φάσεις είναι διαφορετικές για κάθε υπομονάδα εκτέλεσης. Στην υπομονάδα πράξεων σταθερής υποδιαστολής αυτές είναι οι κλασικές φάσεις EX, MEM και WB της ΜΕΔ MIPS. Στις υπομονάδες πράξεων κινητής υποδιαστολής αυτές είναι οι φάσεις που αναφέρθηκαν πιο πάνω, και η κοινή τελευταία φάση WB, η οποία όμως αναφέρεται στο ΦΚ κινητής υποδιαστολής.
4. Μετά τις φάσεις IF και ID, οι φάσεις που ακολουθούν σε διαφορετικές υπομονάδες επικαλύπτονται πλήρως. Έτσι, μια πράξη πρόσθεσης κινητής υποδιαστολής μπορεί να εκτελείται ταυτόχρονα με μια λογική πράξη, αλλά και με μια πράξη πολλαπλασιασμού κινητής υποδιαστολής. Φυσικά οι κοινές φάσεις IDF και WB επικαλύπτονται μόνο μερικά.
5. Ο μηχανισμός παροχέτευσης μπορεί να προωθεί δεδομένα είτε μέσα σε κάθε υπομονάδα εκτέλεσης, είτε μεταξύ των υπομονάδων εκτέλεσης, και πάντα από τη φάση που παράγει κάποιο δεδομένο προς τη φάση που το χρησιμοποιεί. Σε περίπτωση πολλαπλών κύκλων εκτέλεσης τα δεδομένα εισόδου χρησιμοποιούνται από την πρώτη φάση, και το δεδομένο εξόδου παράγεται από την τελευταία φάση εκτέλεσης.
6. Όλες οι εντολές προσπέλασης στη μνήμη, ακόμα κι αν αφορούν καταχωρητές κινητής υποδιαστολής, εκτελούνται στην υπομονάδα σταθερής υποδιαστολής, με τον τρόπο που γνωρίζουμε. Για παράδειγμα, η εντολή `lw`, αν και αποθηκεύει το αποτέλεσμα της σε κάποιον καταχωρητή κινητής υποδιαστολής, εκτελείται σαν εντολή σταθερής υποδιαστολής. Σαν αποτέλεσμα των παραπάνω, ο κύκλος εντολής κινητής υποδιαστολής δεν περιλαμβάνει φάση προσπέλασης μνήμης.
7. Οι εντολές μεταφοράς δεδομένων μεταξύ των δύο ΦΚ εκτελούνται στην υπομονάδα σταθερής υποδιαστολής. Έτσι, η εντολή `mfhc` διαβάζει το τελούμενο εισόδου από έναν καταχωρητή κινητής υποδιαστολής κατά τη φάση IDI, και αν δεν υπάρξει παροχέτευση από κάποια από τις δύο υπομονάδες κινητής υποδιαστολής, το αποθηκεύει κατά τη φάση WB σε έναν καταχωρητή σταθερής υποδιαστολής. Την αντίστροφη μεταφορά υλοποιεί η εντολή `mtc`.
8. Δύο εγγραφές σε διαφορετικό ΦΚ μπορούν να γίνουν ταυτόχρονα. Μ' άλλα λόγια, οι φάσεις WB διαφορετικών εντολών μπορούν να επικαλύπτονται, αν αφορούν διαφορετικούς ΦΚ.
9. Πάγωμα του μηχανισμού μερικής επικάλυψης σε κάποια υπομονάδα εκτέλεσης μπορεί να οφείλεται σε κοντινές διαιρέσεις που οδηγούν σε κίνδυνο δομικής εξάρτησης στις φάσεις D3 και D4 της υπομονάδας διαίρεσης κινητής υποδιαστολής, ή σε ταυτόχρονη προσπάθεια εγγραφής του ΦΚ κινητής υποδιαστολής από διαφορετικές υπομονάδες που οδηγεί σε κίνδυνο δομικής εξάρτησης στο ΦΚ κινητής υποδιαστολής. Τις περισσότερες φορές τέτοιο πάγωμα δεν επηρεάζει το μηχανισμό μερικής επικάλυψης σε άλλες υπομονάδες, οπότε μπορούμε να προχωρούμε σε ολοκλήρωση μιας εντολής, ακόμα κι αν μια προηγούμενη εντολή δεν έχει ολοκληρωθεί, φτάνει οι δύο εντολές να εκτελούνται σε διαφορετικές υπομονάδες, και να μην υπάρχει μεταξύ τους κάποια εξάρτηση από δεδομένα.
10. Οι διακλαδώσεις εκτελούνται στη φάση EX χωρίς καθυστέρηση.

A. Θεωρήστε τον κώδικα MIPS που βρίσκεται στο αρχείο `frcode.txt`. Χρησιμοποιήστε το πρόγραμμα που γράψατε για να βγάλετε το αντίστοιχο διάγραμμα χρονισμού, εκτελώντας τέσσερις επαναλήψεις στο βρόχο που υπάρχει στον κώδικα. Για απλούστευση θεωρήστε ότι οι διακλαδώσεις προβλέπονται ιδανικά με χρήση BTB. Σχολιάστε την επίδραση των κινδύνων από δομικές εξαρτήσεις και από εξαρτήσεις δεδομένων στην εκτέλεση του κώδικα.

B. Στη συνέχεια, προσπαθήστε να βρείτε μια αναδιάταξη των εντολών του κώδικα ώστε να ελαχιστοποιούνται τα παγώματα που εμφανίστηκαν. Κατασκευάστε ξανά το διάγραμμα χρονισμού με βάση τη νέα διάταξη εντολών που βρήκατε. Εξαλείψατε κάποιους από τους κινδύνους που είχατε νωρίτερα και ποιους;

Γ. Η παραπάνω ΜΕΔ έχει το μειονέκτημα ότι τυχόν καθυστέρηση στην παραγωγή μιας τιμής από υπομονάδα κινητής υποδιαστολής, η οποία πρόκειται να αποθηκευτεί στη μνήμη, οδηγεί

σε πάγωμα του μηχανισμού επικάλυψης λόγω παγώματος της εντολής αποθήκευσης. Σκεφτείτε, περιγράψτε και προσομοιώστε κάποιο μηχανισμό, με τον οποίο θα μπορέσετε να αποφύγετε αυτό το πάγωμα.

Ο μηχανισμός αυτός θα πρέπει να ανιχνεύει αυτό το είδος κινδύνου και να κρατάει την εκκρεμή εντολή αποθήκευσης σε κατάλληλη λίστα, απελευθερώνοντας ταυτόχρονα τη ΜΕΔ, ολοκληρώνοντας την αποθήκευση μόλις η τιμή αποθήκευσης γίνει διαθέσιμη. Όμως κάτι τέτοιο ενεργοποιεί πολλούς διαφορετικούς κινδύνους, δομικούς στη μνήμη, κινδύνους EME μνήμης από αποθήκευση εκτός σειράς σε επικαλυπτόμενες διευθύνσεις, αλλά και κινδύνους AME μνήμης από φόρτωση εκτός σειράς σε σχέση με αποθηκεύσεις σε επικαλυπτόμενες διευθύνσεις. Ο μηχανισμός σας θα πρέπει να αντιμετωπίζει με επιτυχία όσες τέτοιες περιπτώσεις κινδύνου εμφανίζονται.

Για να μπορέσετε να προσομοιώσετε αυτό το μηχανισμό, θεωρήστε ότι κάθε εντολή προσπέλασης μνήμης παράγει στη φάση EX κάποια τυχαία διεύθυνση από 0 μέχρι N-1, για κάποια τιμή του N. Υλοποιώντας μια τέτοια παραγωγή τυχαίων διευθύνσεων, θα μπορείτε να ελέγχετε για κινδύνους μνήμης. Δοκιμάστε το μηχανισμό για διάφορες τιμές του N, τόσο για τον αρχικό κώδικα, όσο και για τον κώδικα του ερωτήματος B.

Άσκηση 5:

Αντικείμενο της άσκησης αυτής είναι η σχεδίαση μονάδων εκτέλεσης πράξεων με μερική επικάλυψη. Θεωρήστε λοιπόν μια μονάδα κινητής υποδιαστολής που εκτελεί κάποια σύνθετη πράξη DSP μεταξύ τεσσάρων αριθμών a, b, c και d, η οποία περιλαμβάνει δύο προσθέσεις, έναν πολλαπλασιασμό και μια διαίρεση, παράγοντας ως αποτέλεσμα τον αριθμό:

$$\frac{(a+c) \times (b+c)}{d}$$

Η μονάδα ακολουθεί την τυποποίηση IEEE 754, και η πράξη υλοποιείται σε συνδυασμένα βήματα χειρισμού εκθετών, πράξης συντελεστών, κανονικοποίησης και στρογγυλοποίησης. Ειδικότερα, για την υλοποίηση της πράξης διαθέτουμε τις ακόλουθες υπομονάδες που πρέπει να αντιστοιχίσουμε στα βήματα που υλοποιούν την πράξη:

1. Υπομονάδα πρόσθεσης/αφαίρεσης εκθετών.
2. Υπομονάδα δεξιάς ολίσθησης ενός συντελεστή για ευθυγράμμιση κατά τόσα ψηφία όσα δίνει η προηγούμενη υπομονάδα.
3. Υπομονάδα πρόσθεσης σταθερής υποδιαστολής μεταξύ συντελεστών που μπορεί να είναι έξοδοι άλλων υπομονάδων.
4. Υπομονάδα πρόσθεσης/αφαίρεσης εκθετών με αφαίρεση/πρόσθεση πώλωσης.
5. Υπομονάδα πολλαπλασιασμού σταθερής υποδιαστολής μεταξύ συντελεστών που μπορεί να είναι έξοδοι άλλων υπομονάδων.
6. Υπομονάδα ROM.
7. Υπομονάδα συσσώρευσης σταθερής υποδιαστολής.
8. Υπομονάδα κανονικοποίησης ή ρύθμισης συντελεστή και εκθέτη για μη κανονικοποιημένους αριθμούς.
9. Υπομονάδα στρογγυλοποίησης.

Οι παραπάνω υπομονάδες δε δίνονται απαραίτητα με τη σειρά που υλοποιούν τα βήματα της πράξης, ενώ κάποιες από αυτές χρησιμοποιούνται πολλαπλές φορές και πιθανά με διαφορετικούς τρόπους. Πιο συγκεκριμένα:

- Η υπομονάδα 1 χρησιμοποιείται τρεις φορές (για τις δύο προσθέσεις της πράξης και την τελική κανονικοποίηση/ρύθμιση εκθέτη) και απαιτεί έναν κύκλο μηχανής.
- Η υπομονάδα 2 χρησιμοποιείται δύο φορές (για τις δύο προσθέσεις της πράξης), απαιτεί έναν κύκλο μηχανής και λαμβάνει ως μία είσοδο την έξοδο της προηγούμενης.
- Η υπομονάδα 3 χρησιμοποιείται δύο φορές για απλή πρόσθεση σταθερής υποδιαστολής (για τις δύο προσθέσεις της πράξης), απαιτεί έναν κύκλο μηχανής και λαμβάνει ως είσοδο την έξοδο της προηγούμενης.

- Η υπομονάδα 4 χρησιμοποιείται δύο φορές (για τον πολλαπλασιασμό και τη διαίρεση της πράξης) και απαιτεί έναν κύκλο μηχανής.
- Η υπομονάδα 5 χρησιμοποιείται μία φορά για απλό πολλαπλασιασμό σταθερής υποδιαστολής (για τον πολλαπλασιασμό της πράξης), οπότε απαιτεί δύο κύκλους μηχανής και λαμβάνει ως μία είσοδο την έξοδο της υπομονάδας 3 ή της 7 – ανάλογα με τον αλγόριθμο που χρησιμοποιείται, και άλλες δύο φορές σε συνδυασμό με την υπομονάδα 7 (για τη διαίρεση της πράξης), όπως εξηγείται παρακάτω.
- Η υπομονάδα 6 χρησιμοποιείται μία φορά για εύρεση της αρχικής προσέγγισης του αντιστρόφου που απαιτεί η διαίρεση και απαιτεί έναν κύκλο μηχανής. Η αρχική προσέγγιση παρέχεται από πίνακα κωδικοποιημένο σε κατάλληλη ROM, η οποία διευθυνσιοδοτείται από τα πιο σημαντικά ψηφία του συντελεστή του διαιρέτη.
- Η υπομονάδα 7 χρησιμοποιείται σε συνδυασμό με την υπομονάδα 5 για βελτίωση της ακρίβειας του αντιστρόφου με ταυτόχρονο πολλαπλασιασμό με το διαιρετέο (για υλοποίηση της διαίρεσης). Οι δύο υπομονάδες 5 και 7 απαιτούν δύο κύκλους μηχανής και χρησιμοποιούνται από δύο φορές, με μερική επικάλυψη μεταξύ τους, καθώς η μία τροφοδοτεί την άλλη με το πρώτο μισό του αποτελέσματός της. Για να υλοποιηθεί η διαίρεση, πρέπει την πρώτη φορά η υπομονάδα 5 να λάβει ως εισόδους την έξοδο της υπομονάδας 6 και την έξοδο της υπομονάδας 3 ή της 5 – ανάλογα με τον αλγόριθμο που χρησιμοποιείται.
- Η υπομονάδα 8 χρησιμοποιείται δύο φορές, μία φορά πριν, και μία φορά μετά τη στρογγυλοποίηση του αποτελέσματος της πράξης, και απαιτεί έναν κύκλο μηχανής. Την πρώτη φορά πρέπει φυσικά να έχουν ολοκληρωθεί οι προηγούμενες φάσεις της πράξης. Τη δεύτερη φορά η υπομονάδα 8 χρησιμοποιείται παράλληλα με την υπομονάδα 1.
- Η υπομονάδα 9 χρησιμοποιείται μία φορά και απαιτεί δύο κύκλους μηχανής. Λαμβάνει είσοδο από την υπομονάδα 8, ενώ στέλνει την έξοδό της προς τις υπομονάδες 8 και 1.

Για την υλοποίηση της πράξης μπορούμε να χρησιμοποιήσουμε έναν από δύο αλγόριθμους, είτε πρώτα διαίρεση και μετά πολλαπλασιασμό, είτε αντίστροφα. Επιπλέον, παρατηρήστε ότι ο σχετικός χρονισμός μεταξύ των βημάτων της πράξης δεν είναι πάντα μονοσήμαντα ορισμένος, και περιορίζεται μόνο από τις μεταξύ τους εξαρτήσεις. Επιδίωξή μας θα είναι να επιλέξουμε την υλοποίηση που οδηγεί στη βέλτιστη δυνατή πράξη.

A. Για καθέναν από τους δύο διαφορετικούς αλγόριθμους υλοποίησης, να κατασκευάσετε ένα σχηματικό διάγραμμα της μονάδας, στο οποίο να φαίνονται οι υπομονάδες που εκτελούν κάθε βήμα της πράξης, και οι συνδέσεις δεδομένων μεταξύ των υπομονάδων. Δείξτε τα σημεία στα οποία υπάρχει ελευθερία αντιστοίχισης υπομονάδων και βημάτων της πράξης.

B. Υπολογίστε ένα άνω φράγμα στο ρυθμό ολοκλήρωσης πράξεων, με βάση τον αριθμό κύκλων μηχανής που καταλαμβάνει κάθε υπομονάδα στη διάρκεια μιας πράξης, όπως αυτός προκύπτει από την παραπάνω περιγραφή.

Γ. Για καθέναν από τους δύο αλγόριθμους, χρησιμοποιήστε την τεχνική των απαγορευμένων κελιών, για να κατασκευάσετε τον πίνακα δέσμευσης της μονάδας που δίνει ρυθμό ολοκλήρωσης πράξεων ίσο με το άνω φράγμα. Επιλέξτε κατάλληλο χρονισμό μεταξύ των βημάτων, για τα οποία αυτός δεν ορίζεται μονοσήμαντα, ώστε να επιτύχετε τον ελάχιστο χρόνο ολοκλήρωσης μιας πράξης. Για το σκοπό αυτό, βρείτε τα απαγορευμένα κελιά πρώτα για τα βήματα που έχουν ορισμένο χρονισμό, και μετά για τα υπόλοιπα, αντιστοιχίζοντάς τα σε ελεύθερα κελιά μέσα στο εύρος χρονισμού που τους διατίθεται. Εισάγετε υπομονάδες καθυστέρησης, μόνο αν αυτό είναι απαραίτητο.

Δ. Επιλέξτε τώρα τον αλγόριθμο που οδηγεί στη συντομότερη πράξη. Σε περίπτωση ισοπαλίας επιλέξτε όποιον αλγόριθμο θέλετε. Επαληθεύστε το αποτέλεσμα του προηγούμενου ερωτήματος για τον αλγόριθμο που επιλέξατε, δίνοντας το διάνυσμα σύγκρουσης που προκύπτει, και κατασκευάζοντας το ελαχιστοποιημένο διάγραμμα καταστάσεων, όπου να φαίνεται ο κύκλος με ρυθμό ολοκλήρωσης ίσο με το άνω φράγμα.