

Ατομική Διπλωματική Εργασία

**ΑΝΑΛΥΣΗ ΚΑΙ ΜΕΛΕΤΗ ΜΗΧΑΝΙΣΜΟΥ ΠΡΟΒΛΕΨΗΣ
ΔΙΑΚΛΑΔΩΣΕΩΝ TAGE**

Γιάννης Δημητριάδης

ΠΑΝΕΠΙΣΤΗΜΙΟ ΚΥΠΡΟΥ



ΤΜΗΜΑ ΠΛΗΡΟΦΟΡΙΚΗΣ

Μάιος 2018

ΠΑΝΕΠΙΣΤΗΜΙΟ ΚΥΠΡΟΥ

ΤΜΗΜΑ ΠΛΗΡΟΦΟΡΙΚΗΣ

**ΑΝΑΛΥΣΗ ΚΑΙ ΜΕΛΕΤΗ ΜΗΧΑΝΙΣΜΟΥ ΠΡΟΒΛΕΨΗΣ
ΔΙΑΚΛΑΔΩΣΕΩΝ TAGE**

Γιάννης Δημητριάδης

Επιβλέπων Καθηγητής

Γιαννάκης Σαζεΐδης

Η Ατομική Διπλωματική Εργασία υποβλήθηκε προς μερική εκπλήρωση των
απαιτήσεων απόκτησης του πτυχίου Πληροφορικής του Τμήματος Πληροφορικής του
Πανεπιστημίου Κύπρου

Μάιος 2018

Ευχαριστίες

Στο σημείο αυτό θα ήθελα να αφιερώσω μία σελίδα σε όλους τους ανθρώπους, που ο καθένας με τον δικό του τρόπο, βοήθησαν στην ολοκλήρωση της εργασίας μου. Η βοήθεια και η στήριξη αυτών των ανθρώπων ήταν ιδιαίτερα σημαντική για εμένα και δεν θα μπορούσα να τους παραλείψω.

Πρωτίστως, θα ήθελα να ευχαριστήσω τον επιβλέποντα καθηγητή μου, Δρ. Γιαννάκη Σαζεΐδη, για την ευκαιρία που μου έδωσε να εκπονήσω μαζί του την διπλωματική μου εργασία, σε ένα τόσο ενδιαφέρον θέμα. Επίσης θα ήθελα να τον ευχαριστήσω για την εμπιστοσύνη που έδειξε στο πρόσωπο μου και με ένταξε στην ομάδα του. Η βοήθεια, ο τρόπος σκέψης και οι γνώσεις που μου έχει μεταφέρει ήταν πολύ σημαντικές και χρήσιμες για την ολοκλήρωση της εργασίας μου. Τον ευχαριστώ επίσης για τις ατελείωτες συμβουλές του, τόσο σε ακαδημαϊκό επίπεδο όσο και σε προσωπικό. Ακόμη, η απέραντη υπομονή, η κατανόηση και ευχάριστη διάθεση ήταν πάντα κάτι που τον χαρακτήριζε σε κάθε συνάντηση μαζί του. Η καθοδήγηση και η στήριξή του ήταν πολύ σημαντική για εμένα. Η φιλική ατμόσφαιρα που δημιουργούσε σε κάθε συνάντηση μας ήταν αυτή που μου έδινε κίνητρο να συνεχίζω με τον ίδιο ζήλο και να μαθαίνω ακόμη περισσότερα.

Τέλος φυσικά, θα ήθελα να ευχαριστήσω τους γονείς μου, Παντελή και Ρένα, καθώς επίσης και την αδελφή μου Χριστίνα, στους οποίους είμαι ευγνώμων γιατί είναι πάντα δίπλα μου, στηρίζοντας τις επιλογές μου και δίνοντάς μου σημαντικές συμβουλές. Θα ήθελα να τους εκφράσω την ευγνωμοσύνη μου για όλα αυτά που έχουν κάνει για μένα και για την απέραντη τους υπομονή. Δεν θα μπορούσα να παραλείψω τις θερμές μου ευχαριστίες στον άνθρωπο ο οποίος είναι πάντα δίπλα μου, στην Θεοδώρα. Με την απέραντη υπομονή και κατανόηση της κατάφερνε πάντα να με κάνει να γελώ με τον δικό της αυθεντικό πάντα τρόπο, να ηρεμώ και να συνεχίζω σε κάθε σημείο που τα πράγματα ήταν σκούρα. Είναι αυτή που με καθοδηγεί σε όλες τις δύσκολες στιγμές και αποφάσεις. Τέλος ευχαριστώ όλους μου τους φίλους για τις ατελείωτες στιγμές που ζήσαμε μαζί.

Αφιερώνω την διπλωματική μου εργασία σε όλους αυτούς.

Περίληψη

Στην παρούσα Διπλωματική Εργασία θα μελετήσουμε, εξετάσουμε και αναλύσουμε έναν μοντέρνο Μηχανισμό Πρόβλεψης Διακλαδώσεων. Ο μηχανισμός ο οποίος θα εξετάσουμε είναι TAGE, συγκεκριμένη έκδοση του έχει κερδίσει σε διαγωνισμό και έχει θεωρηθεί ως ένας από τους πιο επιτυχημένους μηχανισμούς πρόβλεψης.

Συγκεκριμένα, θα μελετήσουμε κάθε συστατικό που αποτελεί τον μηχανισμό αυτό, τον τρόπο λειτουργίας του και θα τα συνδέσουμε έτσι ώστε να εξηγήσουμε τους τρόπους Πρόβλεψης, Δέσμευσης και Ενημέρωσης οι οποίοι αποτελούν τις τρεις βασικές λειτουργίες που εκτελεί για να φέρει εις πέρας το έργο που έχει να επιτελέσει.

Στην συνέχεια, θα προσομοιώσουμε τον μηχανισμό αυτό με την χρήση ειδικού προσομοιωτή μηχανισμών πρόβλεψης διακλαδώσεων και θα αναλύσουμε τη συμπεριφορά του. Επίσης, θα εισάγουμε κάποιες καινούριες έννοιες οι οποίες αποτελούν μέρος της προσπάθειας μας να βελτιώσουμε την ακρίβεια του εκμεταλλευόμενος κάποια ευπάθεια του, το Aliasing. Θα προσομοιώσουμε και θα αναλύσουμε τον καινούριο μηχανισμό, μέρος του οποίου αποτελείται από κενά κελιά, και που θα εκμεταλλεύεται αυτή την ευπάθεια.

Τέλος θα εξάγουμε τα συμπεράσματα τα οποία προκύπτουν από όλη την έκταση της Διπλωματικής Εργασίας και θα προτείνουμε την μελλοντική εργασία που πρέπει να επιτελεσθεί.

ΠΕΡΙΕΧΟΜΕΝΑ

Εισαγωγή	1
1.1 Γενική Εισαγωγή.....	1
1.2 Στόχος Διπλωματικής Εργασίας	2
Τεχνολογίες – Εργαλεία	4
2.1 Προσομοιωτής.....	4
2.2 Traces	4
Απαιτούμενη Γνώση	5
3.1 Εισαγωγή Κεφαλαίου.....	5
3.2 Διασολήνωση.....	6
3.2.1 Κίνδυνοι Διασολήνωσης	7
3.2.2 Τρόποι Αντιμετώπισης των Κινδύνων Διασολήνωσης.....	10
3.3 Μηχανισμοί Πρόβλεψης Διακλαδώσεων.....	13
3.3.1 Bimodal.....	14
3.3.2 Global History Bimodal.....	16
3.3.3 Gshare	17
3.3.4 TAGE.....	18
3.3.5 Προβλήματα στους Μηχανισμούς Πρόβλεψης Διακλαδώσεων.....	18
Μηχανισμός Πρόβλεψης Διακλαδώσεων TAGE	21
4.1 Γενική Εισαγωγή στην Μηχανισμό Πρόβλεψης Διακλαδώσεων TAGE	21
4.2 Μηχανισμοί και Συστατικά.....	22
4.3 Tag.....	23
4.4 Prediction Counter.....	24
4.5 Useful Counter	25
4.6 Μέγεθος Ιστοριών και ο Ρόλος τους.....	25
4.7 Υπολογισμός Index και Tag.....	26
4.7.1 Global History.....	27
4.7.2 Folded Histories.....	28
4.7.3 Path History	29
4.7.4 Υπολογισμός Index.....	29
4.7.5 Υπολογισμός Tag.....	30
4.8 Πρόβλεψη Διακλάδωσης.....	30
4.9 Δέσμευση	32
4.10 Ενημέρωση.....	32
4.11 Διαφορές μεταξύ της έκδοσης 8KB και 64KB	34
TAGE Characterization	35
5.1 TAGE-SC-L	35
5.1.1 Statistical Corrector	36
5.1.2 Loop Predictor	36

5.1.3	Σύγκριση μεταξύ TAGE-SC-L και TAGE	37
5.2	Σύγκριση μεταξύ TAGE 8KB και 64KB	38
5.3	Ανάλυση Heat Maps στον TAGE	42
Έννοια των Κενών Κελιών.....		46
6.1	Εισαγωγή Κεφαλαίου	46
6.2	Ανάλυση αποτελεσμάτων της προσομοίωσης TAGE με κενά κελιά	47
Συμπεράσματα και Μελλοντική Εργασία		50
7.1	Συμπεράσματα	50
7.2	Μελλοντική Εργασία	51
Βιβλιογραφία		52

Κεφάλαιο 1

Εισαγωγή

1.1 Γενική Εισαγωγή

1.2 Στόχος Διπλωματικής Εργασίας

1.1 Γενική Εισαγωγή

Η σχεδίαση σύγχρονων επεξεργαστών απαιτεί την συνεχή αύξηση της επίδοσης, την μείωση της κατανάλωσης ενέργειας και την προσθήκη νέων μηχανισμών. Αυτό απαιτεί από τους σχεδιαστές την συνεχή έρευνα για την ανάπτυξη νέων μηχανισμών.

Οι επεξεργαστές βρίσκονται παντού, από τα κινητά μας τηλέφωνα, στις τηλεοράσεις και κάθε είδους ηλεκτρικής συσκευής, στα αυτοκίνητα ακόμα και στις κάρτες τηλεφωνίας. Είναι αυτοί οι οποίοι συντονίζουν και εκτελούν όλες εκείνες τις πράξεις οι οποίες δίνουν την δυνατότητα στην κάθε συσκευή να εκτελέσει το δικό της έργο. Η επίδοση τους όμως αποτελεί ένα από τα πιο υψηλά κριτήρια τόσο κατά την σχεδίαση όσο και κατά την αγορά μιας ηλεκτρονικής συσκευής.

Οι σύγχρονοι επεξεργαστές ενσωματώνουν περισσότερους από ένα πυρήνες επιτρέποντας έτσι στον ίδιο τον επεξεργαστή να εκτελεί περισσότερα από ένα προγράμματα παράλληλα. Κάθε πυρήνας ενός μοντέρνου επεξεργαστή υψηλής απόδοσης, διαθέτει διασωλήνωση και εκτέλεση εντολών εκτός σειράς (Out Of Order Execution)[9]. Επίσης χρησιμοποιούν τακτικές και μηχανισμούς οι οποίοι τους επιτρέπουν να εκτελούν εντολές κερδοσκοπικά (speculative), δηλαδή η εκτέλεση εντολών οι οποίες δεν είναι απόλυτα σίγουρος ο επεξεργαστής ότι βρίσκονται στο μονοπάτι εκτέλεσης του προγράμματος. Ο σημαντικότερος μηχανισμός για την επιτυχία τη κερδοσκοπικής εκτέλεσης αποτελεί ο Μηχανισμός Πρόβλεψης Διακλαδώσεων, ο οποίος είναι και το κύριο θέμα το οποίο θα αναλύσουμε στην παρούσα διπλωματική εργασία. Ο κύριος σκοπός των μηχανισμών αυτών είναι η πρόβλεψη του αποτελέσματος μιας διακλάδωσης. Αυτό είναι πολύ σημαντικό διότι, όταν εισέλθει μια εντολή διακλάδωσης στον επεξεργαστή δεν μπορούμε να γνωρίζουμε τη διεύθυνση της επόμενης εντολής που θα

εκτελεστεί μέχρι να τελειώσει η εκτέλεση της διακλάδωσης. Αυτό μπορεί να κοστίσει αρκετούς κύκλους μηχανής. Με σκοπό την αύξηση της επίδοσης ο μηχανισμός προσπαθεί να προβλέψει ποια θα είναι η επόμενη εντολή έτσι ώστε να επωφεληθεί από τους χαμένους κύκλους. Σε περίπτωση όμως λάθους πρόβλεψης, στους μοντέρνους επεξεργαστές οι οποίοι διαθέτουν πολύπλοκες διασωληνώσεις και εκτέλεση εντολών εκτός σειράς η διαδικασία του Flush, δηλαδή, της αφαίρεσης των εντολών που βρίσκονται λανθασμένα στον επεξεργαστή και η εισαγωγή των εντολών που βρίσκονται στο πραγματικό μονοπάτι εκτέλεσης, κοστίζει αρκετά. Το κόστος αυτό το ονομάζουμε ως Κόστος Λανθασμένης Πρόβλεψης (Misprediction Penalty). Το μετρικό αξιολόγησης των μηχανισμών αυτών είναι τα Mispredictions Per Kilo Instructions (MPKI), δηλαδή πόσες λανθασμένες προβλέψεις έχουν παρθεί ανά 1000 εντολές, ή διαφορετικά πόσες φορές ο επεξεργαστής θα πρέπει να κάνει Flush στο διάστημα 1000 εντολών.

1.2 Στόχος Διπλωματικής Εργασίας

Ο στόχοι της παρούσας Διπλωματικής Εργασίας χωρίζονται σε δύο μεγάλα θέματα τα οποία είναι άμεσα συνδεδεμένα μεταξύ τους.

1. Ο πρώτος στόχος είναι η μελέτη και κατανόηση της λειτουργίας και των συστατικών του Μηχανισμού Πρόβλεψης Διακλαδώσεων TAGE, ο οποίος προτάθηκε από τον Seznec^[1-7].
2. Ο δεύτερος στόχος, ο οποίος αποτελεί την πλήρη κατανόηση όλων των συστατικών καθώς επίσης και τον τρόπο λειτουργίας του συγκεκριμένου μηχανισμού, είναι η μελέτη της ευαισθησίας του μηχανισμού αυτού σε διάφορες παραμέτρους και η εξερεύνηση μηχανισμών ή γενικών προσθηκών οι οποίες θα επιτρέψουν της βελτίωση της ακρίβειας του.

Τα κυριότερα συμπεράσματα της διπλωματικής είναι:

1. Ο TAGE, αναλαμβάνει το μεγαλύτερο μέρος των προβλέψεων έναντι του Bimodal
2. Ο TAGE αναλαμβάνει τις διακλαδώσεις οι οποίες απαιτούν σύνθετους χειρισμούς και μεγάλο μέγεθος ιστορίας για την ορθή πρόβλεψη τους
3. Ο μηχανισμός αντιμετωπίζει προβλήματα Aliasing
4. Η αύξηση του μεγέθους δεν είναι ανάλογη με την χρήση των κελιών του TAGE
5. Ο TAGE με μέγεθος 8KB αντιμετωπίζει πρόβλημα ανεπαρκούς χωρητικότητας σε σχέση με τον 64KB ο οποίος αντιμετωπίζει προβλήματα μη αποδοτικής χρήσης του διαθέσιμου χώρου

Το υπόλοιπο της διατριβής περιλαμβάνει στο κεφάλαιο 3 βασικές έννοιες και μηχανισμούς πρόβλεψης διακλαδώσεων οι οποίοι είναι απαραίτητοι για την κατανόηση του τρόπου λειτουργίας του TAGE και των επιμέρους μηχανισμών του, ο τρόπος συμπεριφοράς του και τα προβλήματα που αντιμετωπίζει.

Κεφάλαιο 2

Τεχνολογίες – Εργαλεία

2.1 Προσομοιωτής

2.2 Traces

2.1 Προσομοιωτής

Για την διεκπεραίωση αυτής της μελέτης χρησιμοποιήσαμε τον προσομοιωτή του διαγωνισμού CBP-2016 [1], τον οποίο οι διαγωνιζόμενοι χρησιμοποίησαν για την ανάπτυξη και αξιολόγηση μηχανισμών πρόβλεψης διακλαδώσεων. Είναι γραμμένος στην γλώσσα προγραμματισμού C++ και έχει διατεθεί στην ιστοσελίδα του διαγωνισμού για ελεύθερη πρόσβαση.

2.2 Traces

Τα Traces, είναι ένα σύνολο εντολών οι οποίες εξάχθηκαν από προγράμματα. Σκοπός τους είναι η γρήγορη προσομοίωση προγραμμάτων συγκεκριμένων σκοπών, όπως για παράδειγμα η περίπτωση που θα εξετάσουμε η οποία χρειάζεται μόνο τις εντολές διακλάδωσης. Τα Traces τα οποία έχουμε χρησιμοποιήσει είναι τα ίδια τα οποία χρησιμοποίησαν στον διαγωνισμό για την αξιολόγηση των διαγωνιζομένων και διατίθενται στην ιστοσελίδα του διαγωνισμού. Θα αναφερόμαστε σε αυτά ως προγράμματα, σε ολόκληρη την διπλωματική εργασία. Συνολικά είχαμε στη διάθεση μας 432 Traces.

Κεφάλαιο 3

Απαιτούμενη Γνώση

3.1 Εισαγωγή Κεφαλαίου

3.2 Διασωλήνωση

3.2.1 Κίνδυνοι Διασωλήνωσης

3.2.1.1 Δομικοί Κίνδυνοι

3.2.1.2 Κίνδυνοι Δεδομένων

3.2.1.3 Κίνδυνοι Ροής

3.2.2 Τρόποι Αντιμετώπισης των Κινδύνων Διασωλήνωσης

3.3 Μηχανισμοί Πρόβλεψης Διακλαδώσεων

3.3.1 Bimodal

3.3.2 Global History Bimodal

3.3.3 Gshare

3.3.4 TAGE

3.3.5 Προβλήματα στους Μηχανισμούς Πρόβλεψης Διακλαδώσεων

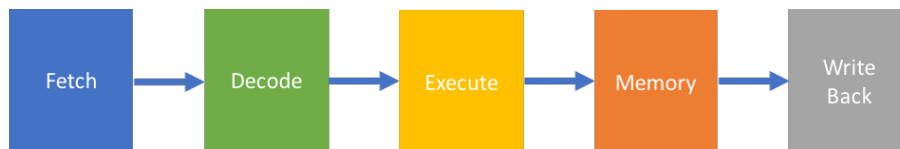
3.1 Εισαγωγή Κεφαλαίου

Σε αυτό το κεφάλαιο, θα αναφερθούμε στην Διασωλήνωση και πως βοηθά την επίδοση. Ακολούθως, θα μελετήσουμε τους κινδύνους οι οποίοι εγείρονται κατά την εφαρμογή της στους επεξεργαστές, καθώς επίσης και οι λύσεις που έχουν δοθεί στα προβλήματα αυτά. Θα μελετήσουμε την αρχιτεκτονική εκτέλεσης εντολών εκτός σειράς και την σημαντικότητα των κινδύνων ροής οι οποίοι αποτελούν το κεντρικό θέμα της παρούσας διπλωματικής εργασίας. Στην συνέχεια θα εξετάσουμε μερικούς από τους πιο ευρέως γνωστούς μηχανισμούς πρόβλεψης διακλαδώσεων εξετάζοντας ταυτόχρονα τα χαρακτηριστικά και τους μηχανισμούς τους οποίους υλοποιούν. Τέλος θα εξετάσουμε τα προβλήματα που αντιμετωπίζουν οι μηχανισμοί πρόβλεψης διακλαδώσεων και πιο συγκεκριμένα θα αναφερθούμε στο Aliasing.

3.2 Διασωλήνωση

Η διασωλήνωση επέτρεψε στους επεξεργαστές την ραγδαία βελτίωση της επίδοσης τους, όντας όμως ταυτόχρονα ένας σχετικά απλός μηχανισμός. Η διασωλήνωση επέτρεψε την παράλληλη εκτέλεση εντολών τα οποία βρίσκονταν σε διαφορετικά στάδια της. Η διασωλήνωση, ουσιαστικά διαχωρίζει το μονοπάτι εκτέλεσης μιας εντολής σε αυτόνομα στάδια τα οποία μπορούν να υποστηρίξουν την εκτέλεση διαφορετικής εντολής σε κάθε ένα από αυτά τα στάδια.

Ένα απλό μοντέλο Διασωλήνωσης αποτελείται από τα εξής στάδια εκτέλεσης μιας εντολής:



Σχήμα 3.1 Απλό Μοντέλο Διασωλήνωσης

Fetch:	Φόρτωση της επόμενης εντολής η οποία πρόκειται να εκτελεστεί
Decode:	Αποκωδικοποίηση της εντολής και φόρτωση των τιμών από τους απαιτούμενους καταχωρητές
Execute:	Εκτέλεση της εντολής
Memory:	Αποθήκευση αποτελέσματος στην μνήμη τυχαίας προσπέλασης αν απαιτείται από την εντολή
Write Back:	Αποθήκευση του αποτελέσματος στον καταχωρητή προορισμού

Η διασωλήνωση επιτρέπει σε κάθε κύκλο μηχανής να ξεκινάει μια καινούρια εντολή και αφού ολοκληρωθεί η πρώτη εντολή, μετά από τόσους κύκλους όσα και τα τμήματα της διασωλήνωσης, ακολούθως τελειώνει μία εντολή σε κάθε κύκλο μηχανής.

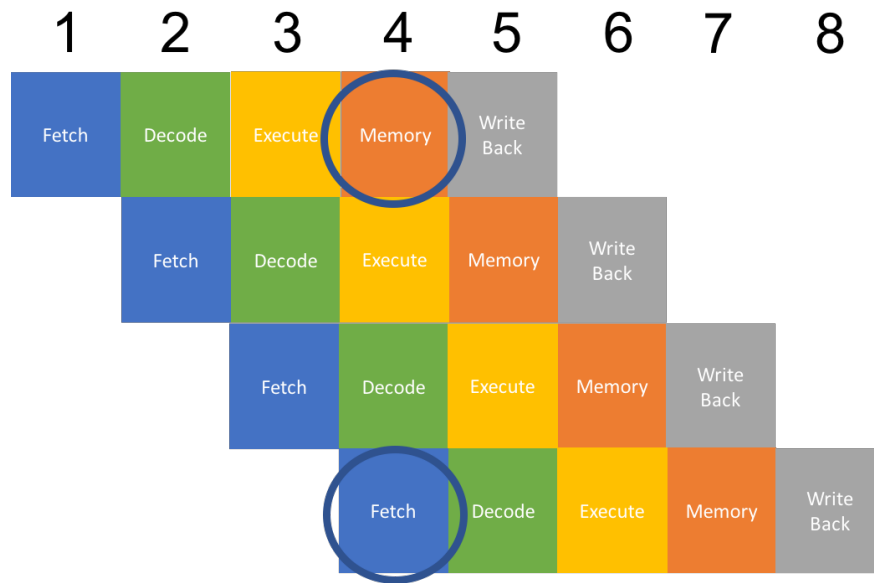
3.2.1 Κίνδυνοι Διασωλήνωσης

Ενώσω η διασωλήνωση προσφέρει αρκετά πλεονεκτήματα και υπόσχεται άμεση βελτίωση της επίδοσης του επεξεργαστή εντούτοις εγείρει τρεις πολύ σημαντικούς κινδύνους. Στο υποκεφάλαιο αυτό θα αναλύσουμε τους κινδύνους που αντιμετωπίζει η διασωλήνωση καθώς και απλές λύσεις αντιμετώπισης τους, οι οποίες κοστίζουν όμως σε επίδοση. Σε επόμενο υποκεφάλαιο θα μελετήσουμε λύσεις οι οποίες λύνουν τους κινδύνους αυτούς προσθέτοντας επιπλέον υλικό και βελτιώνοντας την επίδοση του επεξεργαστή. Αυτοί οι κίνδυνοι είναι οι ακόλουθοι:

- Δομικοί Κίνδυνοι
- Κίνδυνοι Δεδομένων
- Κίνδυνοι Ροής

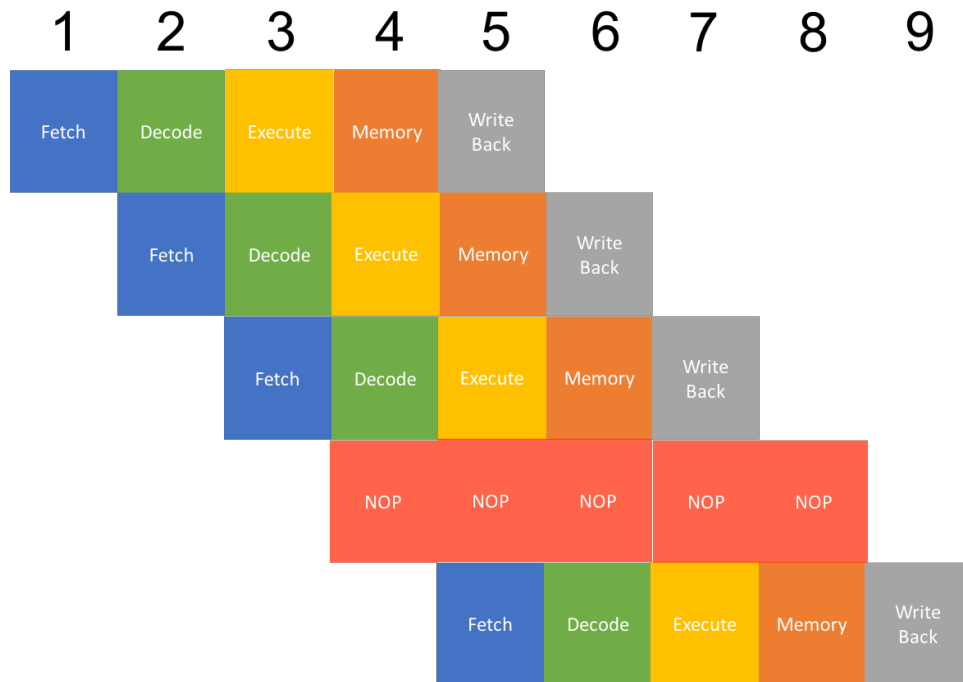
3.2.1.1 Δομικοί Κίνδυνοι

Οι Δομικοί Κίνδυνοι εγείρονται όταν συνδυασμοί εντολών δεν μπορούν να εκτελεστούν ταυτόχρονα διότι χρειάζονται περισσότερους πόρους από αυτούς που διαθέτει η μηχανή. Αυτό μπορεί να συμβεί, για παράδειγμα, όταν υπάρχει μόνο μία κοινή πόρτα πρόσβασης για διάβασμα/γράψιμο σε μια μονάδα μνήμης ή στο αρχείο καταχωρητών. Αυτό έχει ως αποτέλεσμα να μην μπορούν 2 εντολές οι οποίες βρίσκονται σε διαφορετικά τμήματα της διασωλήνωσης να πραγματοποιήσουν ταυτόχρονη πρόσβαση σε αυτήν. Αυτό ακριβώς το σενάριο παρουσιάζουμε στο Σχήμα 3.2.



Σχήμα 3.2 Στον 4^ο Κύκλο Μηχανής παρουσιάζεται Δομικός κίνδυνος λόγω της ύπαρξης μιας πόρτας πρόσβασης στην μνήμη

Μια απλή λύση είναι η εισαγωγή του Bubble ή NOP (No-Operation) όπως αυτή παρουσιάζεται στο Σχήμα 3.3. Αυτή η λύση εισάγει μια μικρή παύση στην εκτέλεση της πιο πρόσφατης από τις δύο εντολές. Με αυτό δίνει προτεραιότητα στην εκτέλεση της εντολής που προηγείται στην Διασωλήνωση και στην συνέχεια η ροή επανέρχεται. Η αντιμετώπιση αυτού του κινδύνου με την συγκεκριμένη λύση απαιτεί κόστος 1 Κύκλο Μηχανής για κάθε φορά που εγείρεται ένας τέτοιος κίνδυνος με αποτέλεσμα την μείωση της επίδοσης του επεξεργαστή. Επίσης η λύση αυτή απαιτεί την αναγνώριση του Κινδύνου και την έγκαιρη αντιμετώπιση του. Ακόμη μία λύση είναι η προσθήκη επιπλέον υλικού έτσι ώστε να μπορούν για παράδειγμα η υποστήριξη ταυτόχρονης πρόσβασης δύο εντολών στην μνήμη



Σχήμα 3.3 Εισαγωγή NOP στον 4^ο Κύκλο Μηχανής όπου παρουσιάζεται
Δομικός κίνδυνος λόγω της ύπαρξης μιας πόρτας πρόσβασης
στην μνήμη

3.2.1.2 Κίνδυνοι Δεδομένων

Οι Κίνδυνοι Δεδομένων εμφανίζονται όταν εντολές, οι οποίες βρίσκονται σε σχετικά μικρή απόσταση εκτέλεσης, έχουν μεταξύ τους εξάρτηση. Οι τύποι εξαρτήσεων μεταξύ εντολών είναι οι ακόλουθοι:

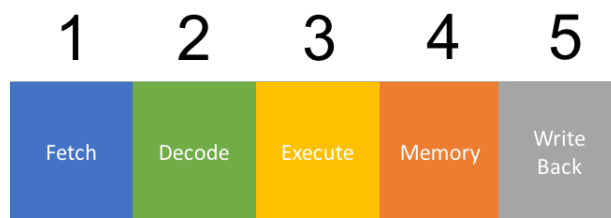
- Read-After-Write (RAW)
- Write-After-Read (WAR)
- Write-After-Write (WAW)
- Read-After-Read (RAR)

Από αυτούς τους τέσσερεις τύπους εξαρτήσεων μεταξύ των εντολών μόνο οι τρεις πρώτοι μπορούν να αποτελέσουν κίνδυνο στην Διασωλήνωση. Εάν δεν αντιμετωπιστούν κατάλληλα, οι κίνδυνοι δεδομένων δημιουργούν προβλήματα στην ορθή ροή δεδομένων και λανθασμένη εκτέλεση προγραμμάτων.

3.2.1.3 Κίνδυνοι Ροής

Οι διακλαδώσεις είναι εντολές οι οποίες επιτρέπουν στα προγράμματα να αλλάζουν την ροή τους.

Οι κίνδυνοι ροής προκύπτουν από το γεγονός ότι η Διασωλήνωση θα πρέπει να εισάγει μία εντολή σε κάθε κύκλο μηχανής. Το αποτέλεσμα όμως της διακλάδωσης, το οποίο είναι η διεύθυνση της επόμενης εντολής που θα εκτελεστεί, είναι διαθέσιμο σε μεταγενέστερο στάδιο, κατά το τέλος του τρίτου σταδίου του Execution, όπως παρουσιάζουμε στο Σχήμα 3.4.



Σχήμα 3.4 Εκτέλεση εντολής με διασωλήνωση

3.2.2 Τρόποι Αντιμετώπισης των Κινδύνων Διασωλήνωσης

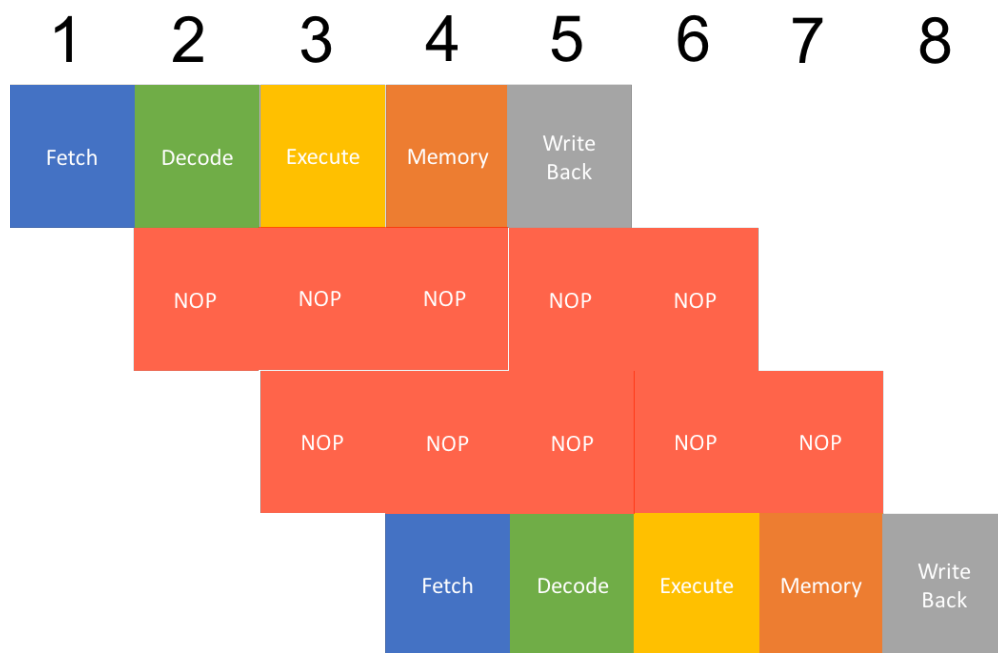
Forwarding

Το Forwarding είναι ένας μηχανισμός ο οποίος επιτρέπει στην Διασωλήνωση να μεταβιβάζει δεδομένα από μία εντολή σε μία άλλη. Αυτός ο μηχανισμός λύνει τα προβλήματα των κινδύνων δεδομένων τύπου RAW, και δεν χρειάζεται ο επεξεργαστής να σπαταλήσει κύκλους μηχανής (Bubbles).

Αντιμετώπιση Κινδύνων Ροής

Όπως έχουμε αναφέρει κατά την ανάπτυξη των κινδύνων ροής, η ύπαρξη τους στηρίζεται στο γεγονός ότι το αποτέλεσμα της εκτέλεσης της εντολής διακλάδωσης δεν είναι διαθέσιμο πριν την έναρξη της επόμενης εντολής. Αυτό που συμπεραίνουμε είναι το

γεγονός ότι είτε ο επεξεργαστής θα πρέπει να εισάγει δύο NOPs για τους δύο κύκλους μηχανής που χρειάζονται μέχρι να είναι διαθέσιμο το αποτέλεσμα της διακλάδωσης, όπως παρουσιάζουμε στο Σχήμα 3.5, είτε να προβλέψει το αποτέλεσμα αυτής και να εισάγει αμέσως την επόμενη εντολή. Η πρόβλεψη μπορεί να γίνει με βάση πολιτικής, κανόνων ή μηχανισμού πρόβλεψης διακλαδώσεων.



Σχήμα 3.5 Διασωλήνωση χωρίς μηχανισμό πρόβλεψης διακλαδώσεων

Ας υποθέσουμε το ακόλουθο πρόγραμμα:

```

1:  bne R1 R2 L1

2:  add R5 R6 R5

3:  add R7 R3 R2

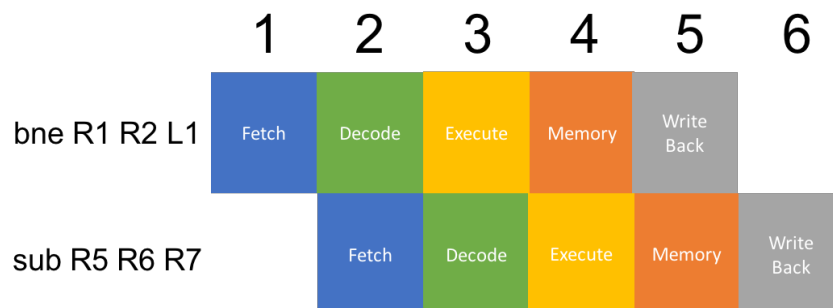
    L1:

4:  sub R5 R6 R7
    
```

Για σκοπούς παραδείγματος ας υποθέσουμε ότι η επόμενη εντολή που θα πρέπει να εκτελεστεί μετά την διακλάδωση είναι η 4^η εντολή.

Η πρόβλεψη μπορεί να έχει μόνο δύο αποτελέσματα, σωστό ή λάθος. Η πρόβλεψη θεωρείται σωστή μόνο όταν η διεύθυνση της εντολής η οποία εισάχθηκε στην Διασωλήνωση είναι ίση με το αποτέλεσμα της διακλάδωσης, ενώ λάθος θεωρείται όταν είναι άνιση.

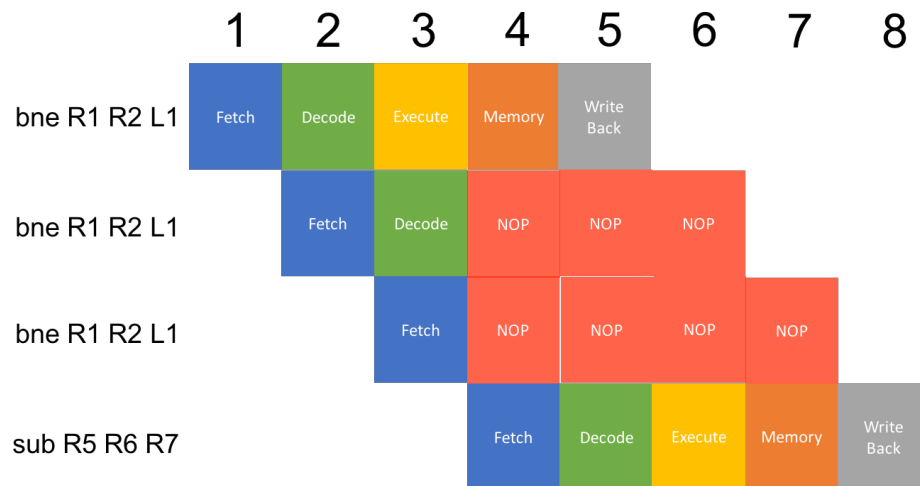
Στο Σχήμα 3.6 παρουσιάζουμε την εκτέλεση των εντολών στην διασωλήνωση κατά την ορθή πρόβλεψη της διακλάδωσης από τον μηχανισμό πρόβλεψης.



Σχήμα 3.6 Διασωλήνωση κατά την ορθή πρόβλεψη του μηχανισμού πρόβλεψης διακλαδώσεων

Σε περίπτωση όπου η πρόβλεψη είναι λάθος εγείρει η ίδια η πρόβλεψη των διακλαδώσεων τους δικούς της κινδύνους, διότι ήδη μέσα στις Διασωλήνωση βρίσκονται εντολές οι οποίες ξεκίνησαν την εκτέλεση τους αλλά δεν θα έπρεπε. Το πιο σημαντικό κριτήριο στην σχεδίαση ενός επεξεργαστή πρέπει να είναι ότι ο ίδιος θα πρέπει να διατηρεί την αρχιτεκτονική του κατάσταση. Δηλαδή, σε περίπτωση λάθους, ο ίδιος θα μπορεί να επαναφέρει την κατάσταση του (αρχείο καταχωρητών, PC κλπ) στο ίδιο σημείο όπου βρισκόταν πριν από το λάθος το οποίο έχει εισέλθει. Συγκεκριμένα για την πρόβλεψη διακλαδώσεων, όσον αφορά το απλό μοντέλο Διασωλήνωσης που έχουμε αναφέρει πιο πάνω, απαιτεί την εισαγωγή NOPs σε όλα τα στάδια της Διασωλήνωσης πριν από την διακλάδωση. Δηλαδή, θα εισαχθούν συνολικά δύο NOPs στα στάδια Fetch και Decode, και η αμέσως επόμενη εντολή που θα εισέλθει στη Διασωλήνωση θα είναι αυτή η διεύθυνση της οποίας είναι ίσο με το αποτέλεσμα της διακλάδωσης,. Με αυτόν τον τρόπο έχουμε καταφέρει να διατηρήσουμε την αρχιτεκτονική κατάσταση του επεξεργαστή αφού δεν επιτρέψαμε την ολοκλήρωση των λανθασμένων εντολών και

έχουμε επανακαθορίσει την ροή του προγράμματος στην ορθή ροή, όπως διακρίνουμε Σχήμα 3.7.



Σχήμα 3.7 Διασωλήνωση κατά την λανθασμένη πρόβλεψη του μηχανισμού πρόβλεψης διακλαδώσεων

3.3 Μηχανισμοί Πρόβλεψης Διακλαδώσεων

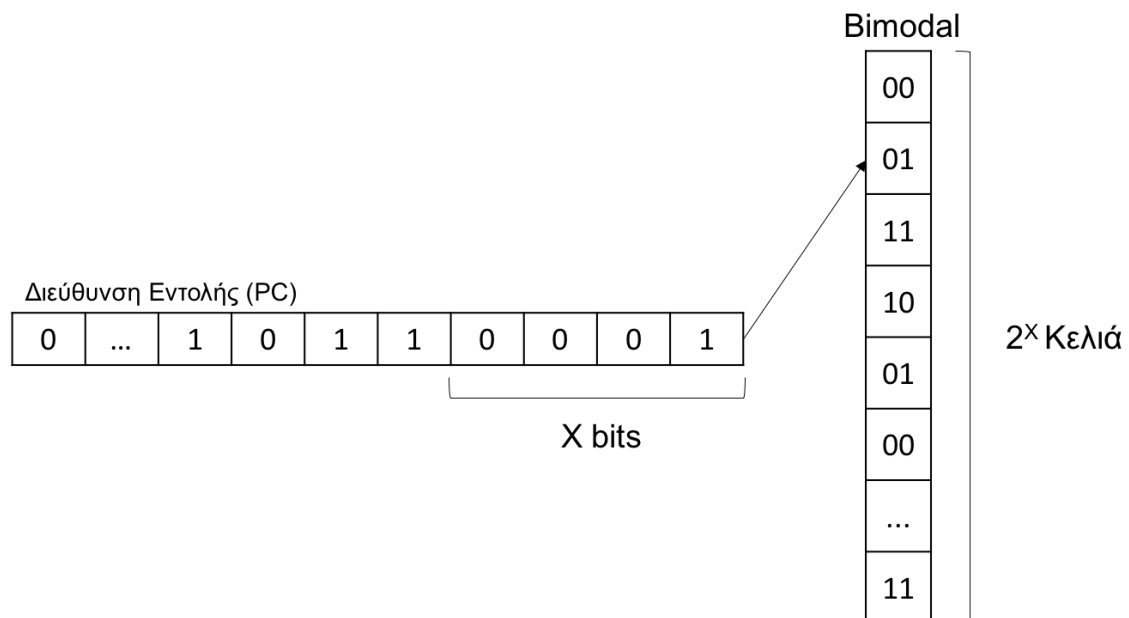
Το κύριο θέμα της παρούσας Διπλωματικής Εργασίας περιστρέφεται γύρω από τους Κινδύνους Ροής τους οποίους αντιμετωπίζει η Διασωλήνωση και πιο συγκεκριμένα τους μηχανισμούς για την πρόβλεψη των αποτελεσμάτων των διακλαδώσεων πριν την ολοκλήρωση της εκτέλεσης τους. Όπως έχουμε αναφέρει πιο πάνω, οι μηχανισμοί πρόβλεψης διακλαδώσεων επιτρέπουν στους επεξεργαστές να αυξήσουν την επίδοση τους, με βασικό πλεονέκτημα την εκτέλεση εντολών πριν από την ολοκλήρωση της εκτέλεσης της διακλάδωσης.

Σε περίπτωση όπου η πρόβλεψη είναι λάθος τότε το κόστος της λάθος πρόβλεψης είναι τουλάχιστο ίσο με τους κύκλους μηχανής από την εισαγωγή την πρώτης εντολής, μετά την διακλάδωση, μέχρι και την ολοκλήρωση της εκτέλεσης της διακλάδωσης όπου το αποτέλεσμα της είναι διαθέσιμο.

Έχουν προταθεί πολλοί μηχανισμοί πρόβλεψης, παρακάτω θα αναφέρουμε τους πιο βασικούς από αυτούς.

3.3.1 Bimodal

Ο Bimodal ^[10] είναι ο πιο απλός μηχανισμός πρόβλεψης, εν συγκρίσει πάντοτε με τους υπόλοιπους μηχανισμούς που θα εξετάσουμε. Ο Bimodal διαθέτει ένα πίνακα μεγέθους 2^X , στο κάθε κελί του οποίου υπάρχει ένα μετρητής 2-bit, στον οποίο θα αναφερόμαστε ως Prediction Counter. Η πολιτική της πρόβλεψης και του Update θα επεξηγηθεί πιο κάτω. Η πρόσβαση σε ένα συγκεκριμένο κελί του πίνακα, Indexing, γίνεται αποκλειστικά με το PC της διακλάδωσης της οποίας θέλουμε μια πρόβλεψη, πχ με τα X πρώτα bits του PC, όπως διακρίνουμε στο Σχήμα 3.8.



Σχήμα 3.8 Μέθοδος Indexing στον Bimodal

Prediction Counter

Ο Prediction Counter ^[10] είναι αυτός ουσιαστικά που δίνει την πρόβλεψη για την διακλάδωση και συνήθως αποτελείται από το bit που δίνει τη πρόβλεψη και ακολούθως τα Hysterisis bits. Μια διακλάδωση μπορεί να έχει μόνο 2 αποτελέσματα Taken και Not-Taken. Όταν ένας Prediction Counter αποτελείται από 2 bits, το Most Significant Bit

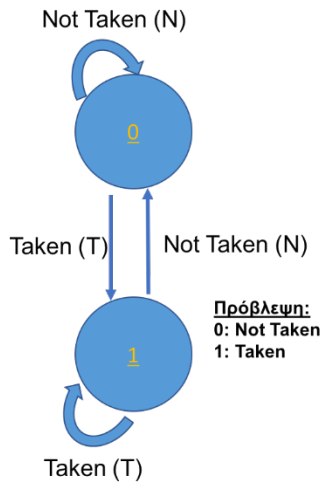
(MSB) δίνει την πρόβλεψη ανάλογα με την τιμή του, όταν είναι 1 είναι Taken και όταν είναι 0 είναι Not-Taken, όπως μπορούμε να διακρίνουμε στο Σχήμα 3.9.

Τα Hysterisis bits επιτρέπουν στους Predictions Counters να έχουν περισσότερη αντοχή στην εναλλαγή των προβλέψεων τους λόγω Aliasing (συγκρούσεων πολλών διακλαδώσεων σε έναν Prediction Counter). Ο Bimodal διαθέτει 1 Hysterisis bit για κάθε έναν Prediction Counter όπως φαίνεται στο Σχήμα 3.10.

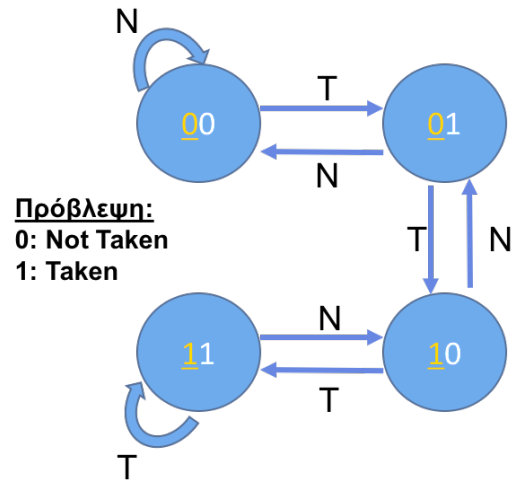
Update

Όταν το αποτέλεσμα της διακλάδωσης είναι γνωστό τότε πραγματοποιείται το Update του Prediction Counter. Και πάλι, χρησιμοποιώντας την διεύθυνση της διακλάδωσης, βρίσκουμε τον Prediction Counter που έδωσε την πρόβλεψη και τον αυξάνουμε κατά 1 αν ήταν Taken ή τον μειώνουμε κατά 1 αν ήταν Not-Taken.

Χωρίς την ύπαρξη Hysterisis bits στον Prediction Counter, ο ίδιος, αλλάζει πολύ εύκολα την κατεύθυνση του σε κάθε λάθος πρόβλεψη σύμφωνα με το Σχήμα 3.9. Αυτό γιατί διαθέτει μόνο δύο καταστάσεις, δηλαδή ένα bit, το οποίο χρησιμοποιείται για την πρόβλεψη της κατεύθυνσης που θα ακολουθήσει η διακλάδωση. Η ύπαρξη Aliasing, το οποίο θα εξετάσουμε στην συνέχεια θα είχε καταστροφικό αποτέλεσμα εάν οι εμπλεκόμενες διακλαδώσεις είχαν αντίθετες κατευθύνσεις και η εκτέλεση τους στην ροή του προγράμματος ήταν εναλλάξ. Με τον Prediction Counter του σχήματος 3.9, όλες οι προβλέψεις θα ήταν λανθασμένες. Αντίθετα, χρησιμοποιώντας αυτόν του σχήματος 3.10 αν υποθέσουμε για σκοπούς παραδείγματος ότι τα Hysterisis bits του είναι ίσο με 1, τότε μόνο η μία από τις δύο διακλαδώσεις θα είχε λανθασμένες προβλέψεις με αποτέλεσμα να έχουμε τις μισές σε σχέση με το πρώτο παράδειγμα.



Σχήμα 3.9 Prediction Counter
χωρίς Hysterisis Bits



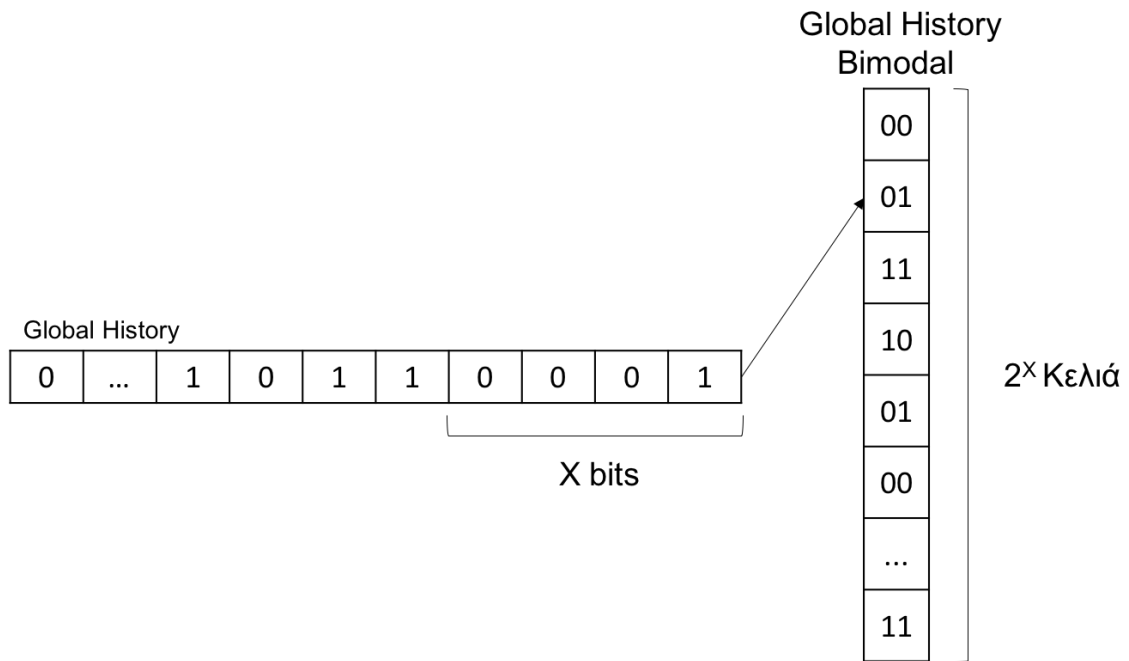
Σχήμα 3.10 Prediction Counter με 1 Hysterisis Bit

3.3.2 Global History Bimodal

Ο Global History Bimodal [10], όπως διακρίνουμε στο Σχήμα 3.11, είναι ο ίδιος με τον Bimodal με την μόνη διαφορά ότι το Indexing γίνεται με βάση το Global History αντί την διεύθυνση της διακλάδωσης με τον ίδιο τρόπο όπως στον Bimodal.

Global History

Το Global History σε αυτόν τον μηχανισμό όπως και στον Gshare τον οποίο θα εξετάσουμε στην συνέχεια, είναι στην πραγματικότητα ένας καταχωρητής ο οποίος σε κάθε διακλάδωση αφού πραγματοποιήσει αριστερή ολίσθηση κατά μία θέση αποθηκεύει ένα bit ανάλογα με την κατεύθυνση που ακολούθησε κάθε διακλάδωση, Taken και Not Taken.



Σχήμα 3.11 Μέθοδος Indexing στον Global History Bimodal

3.3.3 Gshare

Ο Gshare^[10] προτάθηκε από τον McFarling και στην ουσία είναι ένας Global History Bimodal αλλά χρησιμοποιεί επίσης και την διεύθυνση της διακλάδωσης για το Indexing, όπως διακρίνουμε στο Σχήμα 3.12, δημιουργώντας έτσι μία πιο σύνθετη συνάρτηση η οποία μειώνει το Aliasing σε σχέση με τους άλλους δύο. Αυτό διότι, η σύνθεση πολλών μεταβλητών για την παραγωγή του Index, καταφέρνει να διασπάσει με πιο ομοιογενή τρόπο τις διακλαδώσεις στα κελιά του πίνακα συμπεριλαμβάνοντας περισσότερη πληροφορία.



Σχήμα 3.12 Μέθοδος Indexing στον Gshare

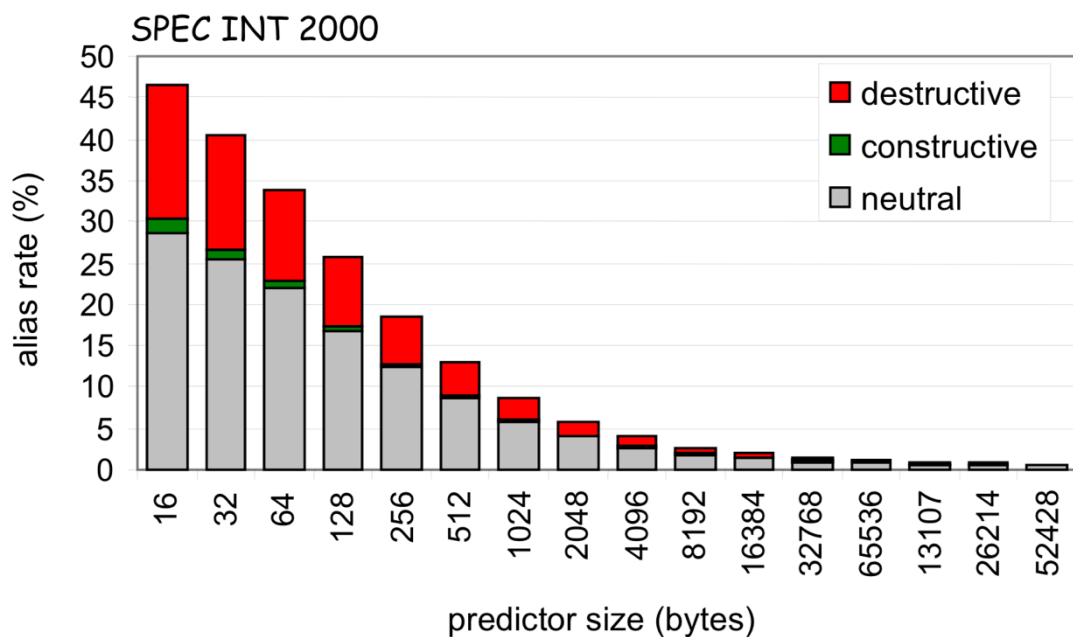
3.3.4 TAGE

Ο TAGE^[1-7] είναι ένας σύγχρονος, state-of-the-art, μηχανισμός πρόβλεψης, ο οποίος σχεδιάστηκε από τον André Seznec. Ο συγκεκριμένος μηχανισμός βρίσκεται στο επίκεντρο της παρούσας Διπλωματικής εργασίας και θα τον εξετάσουμε με λεπτομέρεια σε επόμενο κεφάλαιο.

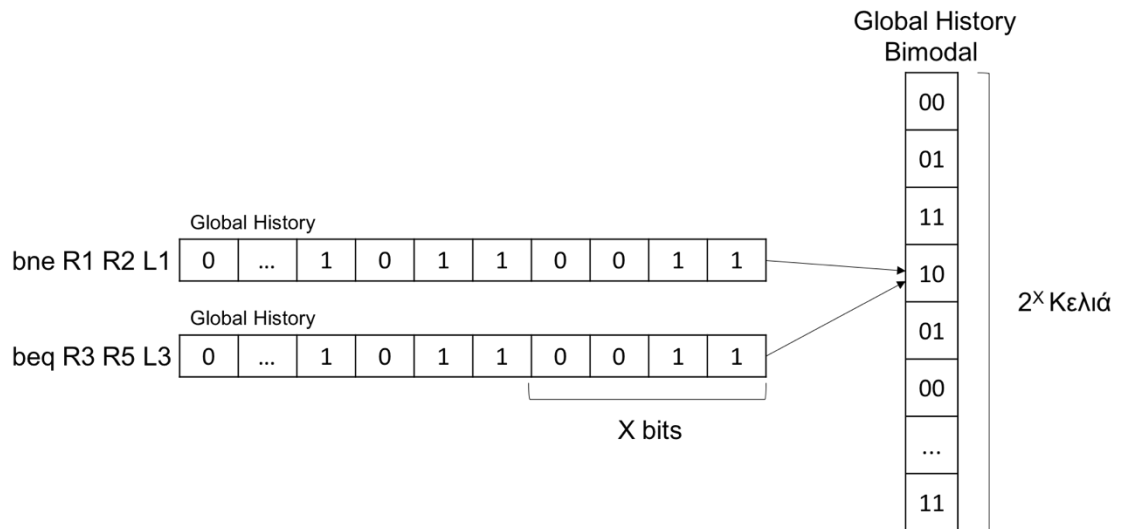
3.3.5 Προβλήματα στους Μηχανισμούς Πρόβλεψης Διακλαδώσεων

Ένα από τα πιο μεγάλα προβλήματα που αντιμετωπίζουν όλοι οι μηχανισμοί πρόβλεψης είναι το Aliasing. Το Aliasing προκύπτει όταν, δύο ή περισσότερες διακλαδώσεις χρησιμοποιούν τους ίδιους μετρητές πρόβλεψης. Δηλαδή, για παράδειγμα στον Bimodal παίρνουν την πρόβλεψη από τον ίδιο Prediction Counter. Αυτό συμβαίνει επειδή χρησιμοποιώντας ένα υποσύνολο της διεύθυνσης διακλάδωσης, λόγω του περιορισμένου μεγέθους του πίνακα, πολλές διακλαδώσεις παράγουν το ίδιο Index. Αυτό μπορεί να φέρει αρνητικά αποτελέσματα αφού, οι δύο διακλαδώσεις θα ενημερώνουν τον ίδιο Prediction Counter με αποτέλεσμα να δίνονται πιθανώς λάθος προβλέψεις. Το Aliasing μπορεί να είναι είτε εποικοδομητικό είτε καταστροφικό με τα δεδομένα σύμφωνα με το Σχήμα 3.13 [11] να δείχνουν ότι είναι περισσότερο καταστροφικό παρά εποικοδομητικό.

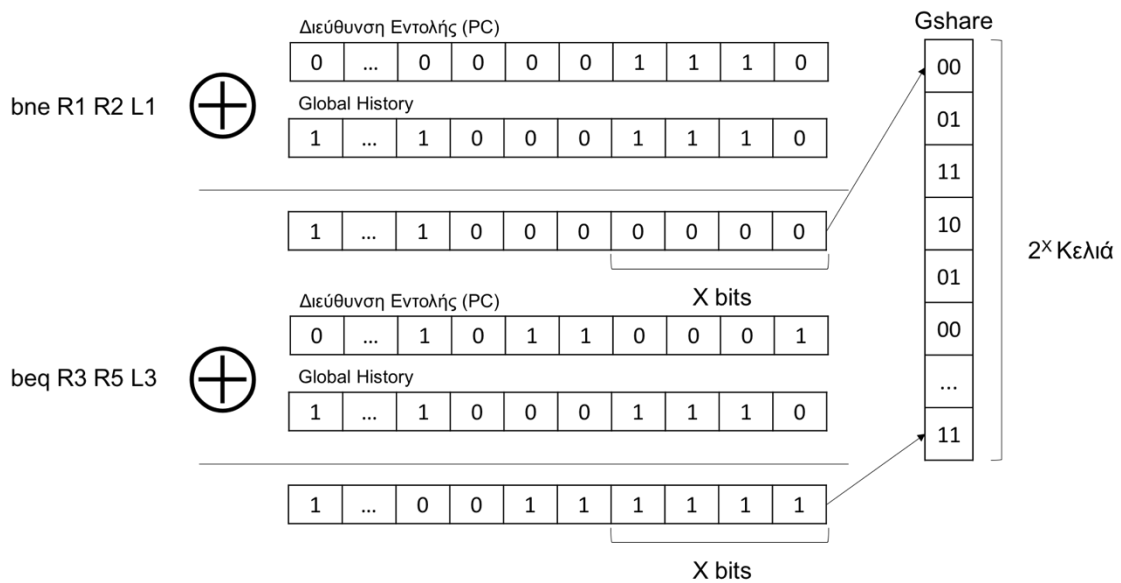
Επίσης, καθώς ο μηχανισμός πρόβλεψης αυξάνει το μέγεθος του μειώνει παράλληλα το Aliasing διότι αναλογικά κάθε κελί αυτού αντιστοιχεί σε μικρότερο αριθμό από διακλαδώσεις. Στο Σχήμα 3.14, δύο διακλαδώσεις παράγουν το ίδιο Global History με αποτέλεσμα να αντιμετωπίζουμε πρόβλημα Aliasing αφού θα γίνει πρόσβαση στο ίδιο κελί και από τις δύο διακλαδώσεις. Η χρήση του μηχανισμού πρόβλεψης διακλαδώσεων Gshare λύνει αυτού του είδους τα προβλήματα, όπως διακρίνουμε στο Σχήμα 3.15, χρησιμοποιώντας ως επιπρόσθετη πληροφορία την διεύθυνση της διακλάδωσης. Βέβαια αυτό, είναι ορθό μόνο όταν τα παραγόμενα bits μετά από την πράξη XOR που εφαρμόζει ο Gshare, δεν είναι και πάλι τα ίδια. Στο συγκεκριμένο παράδειγμα, η χρήση του Gshare δεν θα αντιμετώπιζε προβλήματα Aliasing, αφού το παραγόμενο αποτέλεσμα είναι διαφορετικό για τις δύο αυτές διακλαδώσεις.



Σχήμα 3.13 Σχέση μεταξύ του ρυθμού Aliasing και του μεγέθους του μηχανισμού πρόβλεψης διακλαδώσεων



Σχήμα 3.14 Παράδειγμα Aliasing μεταξύ δύο διακλαδώσεων οι οποίες παράγουν το ίδιο Global History με αποτέλεσμα να οδηγούνται στην πρόσβαση του ίδιου κελιού



Σχήμα 3.14 Παράδειγμα μηχανισμού πρόβλεψης διακλαδώσεων Gshare όπου επιλύει το πρόβλημα Aliasing που αντιμετωπίζουμε στο Σχήμα 3.14

Κεφάλαιο 4

Μηχανισμός Πρόβλεψης Διακλαδώσεων TAGE

4.1 Γενική Εισαγωγή στην Μηχανισμό Πρόβλεψης Διακλαδώσεων TAGE

4.2 Μηχανισμοί και Συστατικά

4.3 Tag

4.4 Prediction Counter

4.5 Useful Counter

4.6 Μέγεθος Ιστοριών και ο Ρόλος τους

4.7 Υπολογισμός Index και Tag

4.7.1 Global History

4.7.2 Folded Histories

4.7.3 Path History

4.7.4 Υπολογισμός Index

4.7.5 Υπολογισμός Tag

4.8 Πρόβλεψη Διακλάδωσης

4.9 Δέσμευση

4.10 Ενημέρωση

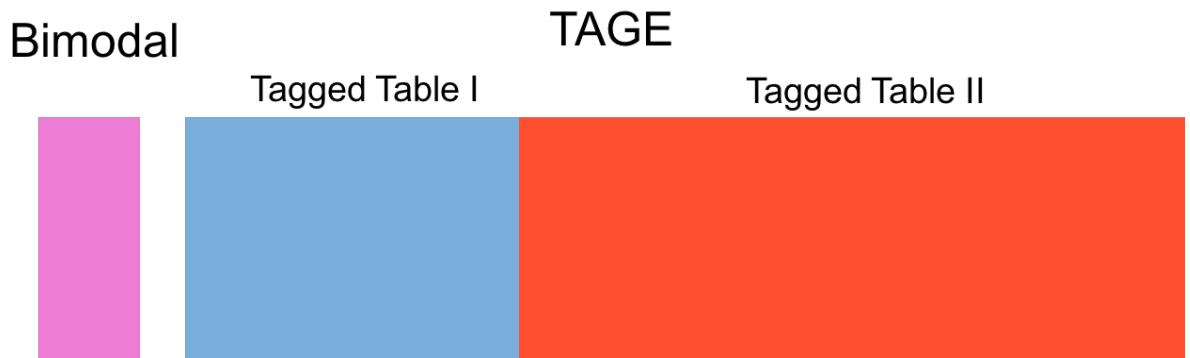
4.11 Διαφορές μεταξύ της έκδοσης 8KB και 64KB

4.1 Γενική Εισαγωγή στην Μηχανισμό Πρόβλεψης Διακλαδώσεων TAGE

Ο TAGE ^[1-7] είναι ένας σύγχρονος μηχανισμός πρόβλεψης. Μία παραλλαγή του οποίου, ο TAGE-SC-L, έχει κερδίσει στο διαγωνισμό CBP 2016. Ο συγκεκριμένος μηχανισμός είναι ένα σύνθετος και πολύπλοκος μηχανισμός ο οποίος προσπαθεί να εκμεταλλευτεί στο έπακρο τις πληροφορίες που είναι διαθέσιμες στην διασωλήνωση σε σχέση με εντολές ροής ελέγχου για να μπορεί να προβλέπει με μεγάλη ακρίβεια τις διακλαδώσεις. Σε αυτό το κεφάλαιο θα εξετάσουμε τους μηχανισμούς που υλοποιεί ο TAGE, τα συστατικά που τον διέπουν και τις πολιτικές τις οποίες χρησιμοποιεί για την Πρόβλεψη (Prediction), Δέσμευση (Allocation), Ενημέρωση (Update) και Indexing.

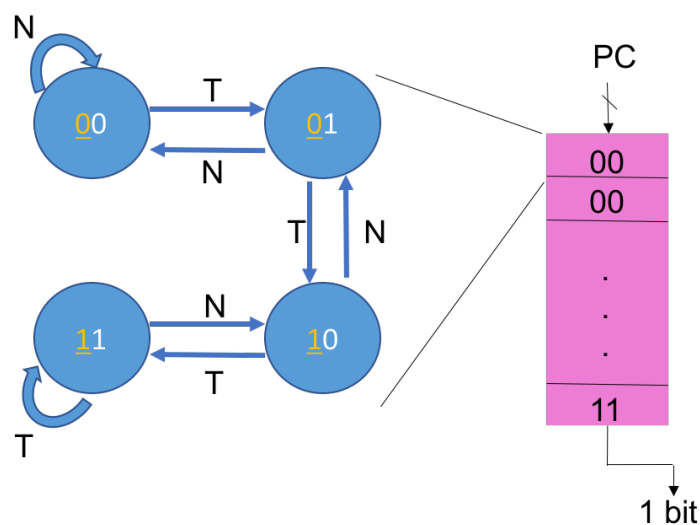
4.2 Μηχανισμοί και Συστατικά

Ο TAGE, στην πραγματικότητα, αποτελείται από δύο μηχανισμούς πρόβλεψης, τον Bimodal και τον TAGE. Στο [1] περιγράφονται δύο εκδόσεις, με διαφορετικό μέγεθος, 64KB και 8KB. Εμείς θα περιγράψουμε στον παρόν σημείο τον TAGE με χωρητικότητα 64KB και στην συνέχεια θα αναφέρουμε τις διαφορές που υπάρχουν με τον 8KB.



Σχήμα 4.1 Συστατικά Μηχανισμού Πρόβλεψης TAGE

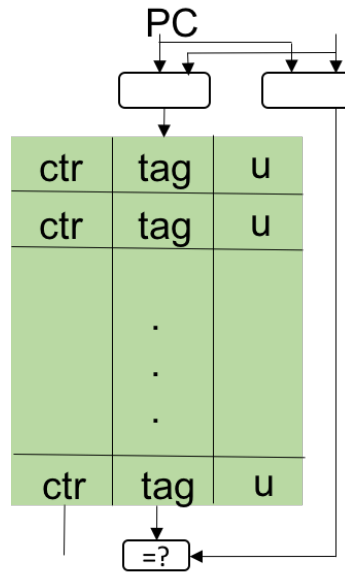
Ως πρώτη και βασική πηγή πρόβλεψης, ο μηχανισμός αυτός, διατηρεί έναν Bimodal ο οποίος αποτελείται από 8192 κελιά. Επίσης κάθε Prediction Counter σε κάθε κελί του διαθέτει και ένα Hysteresis Bit το οποίο μοιράζεται με άλλους 3 γείτονες του με σκοπό την εξοικονόμηση χώρου. Το Index Function του, δηλαδή η συνάρτηση υπολογισμού του κελιού που θα χρησιμοποιήσει σαν πρόβλεψη, έχει ως μοναδική μεταβλητή την διεύθυνση της διακλάδωσης.



Σχήμα 4.2 Δομή Bimodal

Η δεύτερη πηγή πρόβλεψης είναι ο TAGE, ο οποίος αποτελείται ουσιαστικά από δύο Πίνακες (Tagged Tables). Ο πρώτος πίνακας αποτελείται από 10,240 κελιά, ενώ ο δεύτερος πίνακας αποτελείται από 20,480 κελιά. Κάθε κελί αποτελείται από τρία συστατικά:

- Tag (8 bits για τον 1ο Πίνακα και 12 bits για τον 2ο Πίνακα)
- Prediction Counter (3 bits)
- Useful Counter (1 bit)



Σχήμα 4.3 Δομή ενός Φυσικού Πίνακα στον TAGE

Ακολουθώς θα μελετήσουμε τα τρία συστατικά που αποτελούν ένα κελί (Tag, Prediction Counter και Useful Counter), τον σκοπό ύπαρξης τους και τον ρόλο που έχουν να επιτελέσουν στην διαδικασία Πρόβλεψης της κατεύθυνσης των Διακλαδώσεων μέσα από μια σειρά από αποφάσεις.

4.3 Tag

Το Tag μπορούμε να το περιγράψουμε ως μια υπογραφή της κάθε διακλάδωσης, η οποία αποθηκεύεται σε κάθε κελί κατά την διαδικασία της Δέσμευσης. Με αυτόν τον τρόπο, μας δίνεται η δυνατότητα σε μια μελλοντική στιγμή της διαδικασίας πρόβλεψης μιας Διακλάδωσης να επιβεβαιωθεί ότι πρόκειται για την ίδια διακλάδωση η οποία έχει δεσμεύσει το συγκεκριμένο κελί πριν το χρησιμοποιήσει για πρόβλεψη. Με αυτό τον

τρόπο ο TAGE, καταφέρνει να μειώσει δραστικά το Aliasing διότι προσθέτει μία ακόμη διαδικασία εγκυρότητας ότι πρόκειται για την ίδια διακλάδωση η οποία είχε δεσμεύσει το συγκεκριμένο κελί.

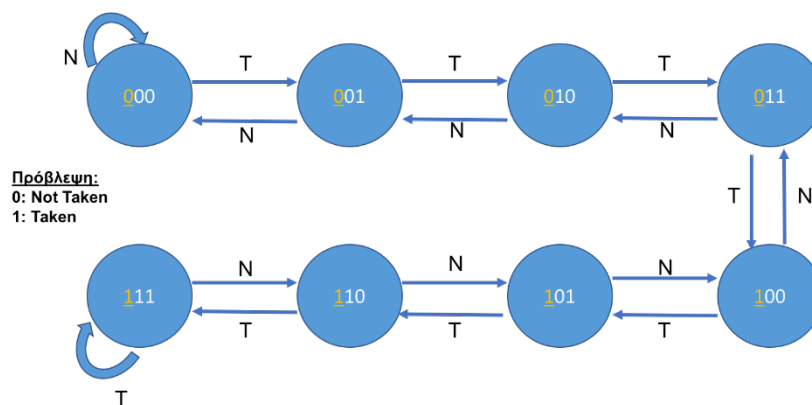
Αυτό όμως δεν εξ υπακούει ότι ο TAGE εξαλείφει το Aliasing, διότι ακόμη και με τη χρήση του Tag μπορεί δύο διαφορετικές διακλαδώσεις να πραγματοποιήσουν πρόσβαση στο ίδιο κελί (ίδια Index) και να διαθέτουν επίσης το ίδιο Tag. (λόγο του ότι το μέγεθος του είναι περιορισμένο)

Το Tag υπολογίζεται με ένα σύνθετο και πολύπλοκο τρόπο τον οποίο θα εξετάσουμε στην συνέχεια του κεφαλαίου αυτού.

4.4 Prediction Counter

Ο Prediction Counter χρησιμοποιεί την ίδια λογική με τον Prediction Counter που διαθέτει ο Bimodal και έχουμε περιγράψει σε προηγούμενο κεφάλαιο. Η μόνη διαφορά είναι το γεγονός ότι κάθε Prediction Counter που διαθέτει ο TAGE αποτελείται από δύο Hysteresis bits αντί για ένα.

Στο σχεδιάγραμμα που ακολουθεί φαίνεται η Πολιτική Ενημέρωσης του Prediction Counter με 2 Hysteresis Bits.



Σχήμα 4.4 Prediction Counter με 2 Hysteresis Bits

4.5 Useful Counter

Ο Useful Counter είναι στην ουσία ένας μετρητής ο οποίος βοηθάει τον μηχανισμό στην ορθότερη Πρόβλεψη και Δέσμευση των διακλαδώσεων στον TAGE. Ο Useful Counter αποτελείται από 1 bit. Σε μεταγενέστερο υπό-κεφάλαιο θα περιγράψουμε πως ο TAGE αξιοποιεί αυτόν τον μετρητή, όταν θα επεξηγούμε τις πολιτικές Πρόβλεψης και Δέσμευσης.

4.6 Μέγεθος Ιστοριών και ο Ρόλος τους

Ο TAGE υποστηρίζει ένα μηχανισμό ο οποίος εκμεταλλεύεται διαφορετικά μεγέθη ιστορίας στην προσπάθεια του να διαχωρίσει τις διακλαδώσεις με βάση αυτές και να τους δώσει το μέγεθος ιστορίας που χρειάζονται για να πετύχει υψηλή ακρίβεια.

Υποστηρίζονται 21 Μεγέθη Ιστοριών, από τα οποία, 7 είναι εις διπλούν, δηλαδή διαθέτουν ίσο μέγεθος ιστορίας, ενώ τα υπόλοιπα 14 έχουν μοναδικό μέγεθος ιστορίας. Τα μεγέθη Ιστορίας 1 – 6 δρομολογούνται στον Φυσικό Πίνακα 1 ενώ οι μεγέθη Ιστορίας 7 – 21 δρομολογούνται στον Φυσικό Πίνακα 2, όπως φαίνεται στο Σχήμα 4.5. Τα μεγέθη ιστορίας που έχουν επιλεγθεί είναι αποτελέσματα γεωμετρικής προόδου. Κάθε αριθμός που παρουσιάζεται στο Σχήμα 4.5 αντιστοιχεί με ένα μέγεθος ιστορίας το οποίο λαμβάνεται υπόψη σε μηχανισμούς τους οποίους θα εξετάσουμε στην συνέχεια αυτού του κεφαλαίου.

Tagged Table 1

1	2	3	4	5	6
---	---	---	---	---	---

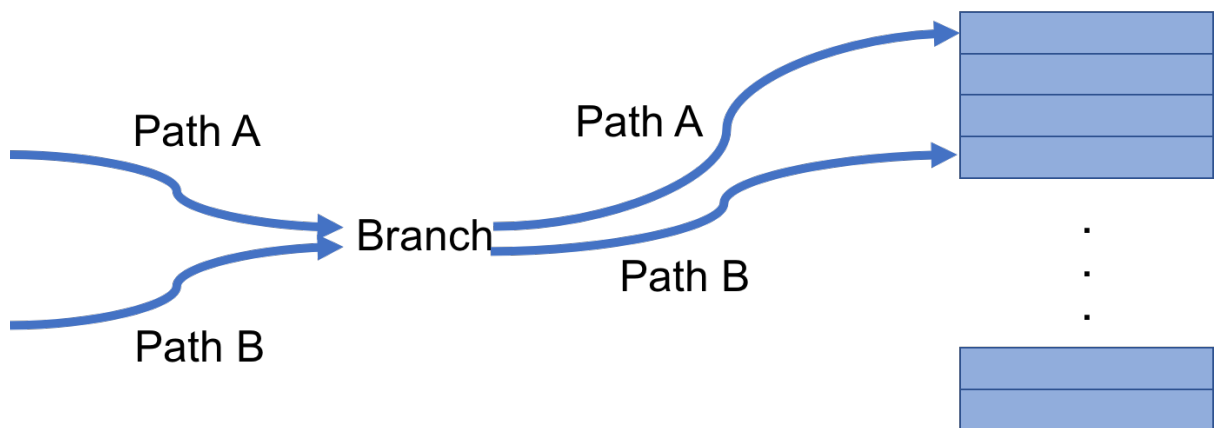
Tagged Table 2

7	8	9	10	11	12	13	14	15	16	17	18	19	20	21
---	---	---	----	----	----	----	----	----	----	----	----	----	----	----

Σχήμα 4.5 Υποστηριζόμενα Μεγέθη Ιστοριών

Αυτό επιτρέπει στην συνάρτηση Index, να αντιστοιχεί μια διακλάδωση με διαφορετικά μεγέθη Ιστορίας σε διαφορετικά κελιά στους Φυσικούς Πίνακες.

Η ροή που έχει διαγραφεί σε ένα πρόγραμμα επηρεάζει άμεσα τις διακλαδώσεις αφού αυτές είναι άμεσα συνδεδεμένες με άλλες διακλαδώσεις. Η δημιουργία διαφορετικών διαδρομών σε ένα μήκος ιστορίας ενδέχεται να δημιουργεί διαφορετικές μεταβλητές που χρησιμοποιούνται κατά την διαδικασία του Indexing. Αυτό έχει ως αποτέλεσμα την δέσμευση πολλαπλών κελιών στους Φυσικούς Πίνακες από διακλαδώσεις οι οποίες έχουν προέλθει από διαφορετικό μονοπάτι και όπως διαφαίνεται στο Σχήμα 4.6.



Σχήμα 4.6 Μια συγκεκριμένη διακλάδωση μπορεί να πάρει την πρόβλεψη της από διαφορετικό κελί λόγω του διαφορετικού μονοπατιού που έχει ακολουθήσει το πρόγραμμα

4.7 Υπολογισμός Index και Tag

Οι διακλαδώσεις χωρίζονται σε 4 συνολικά κατηγορίες ανάλογα με τον τύπο και την λειτουργία τους. Κάθε διακλάδωση υπάγεται σε κάποια από αυτές τις 4 κατηγορίες. Η κατηγορία έχει μεγάλο ρόλο, διότι σε αυτές βασίζονται το Global και το Path History τα οποία χρησιμοποιούνται για τον υπολογισμό των Index και Tag. Επίσης, είναι χρήσιμο και στην Πρόβλεψη και Δέσμευση αφού ο μηχανισμός επικεντρώνεται μόνο στις διακλαδώσεις οι οποίες μπορούν να έχουν δύο κατευθύνσεις. Αυτές οι διακλαδώσεις ορίζονται ως Conditional, διότι η κατεύθυνση τους εξαρτάται από τα δεδομένα εισόδου τους που παράγονται από άλλες εντολές.

Κατηγορία	Τύπος Διακλάδωσης
1	Return Unconditional
	Jump Indirect Unconditional
	Call Indirect Unconditional
2	Jump Direct Unconditional
	Call Direct Unconditional
3	Jump Direct Conditional
	Call Direct Conditional
4	Return Conditional
	Jump Indirect Unconditional
	Call Indirect Unconditional

Σχήμα 4.7 Κατηγορίες διαχωρισμού Τύπου Διακλάδωσης

4.7.1 Global History

Το Global History είναι ένας μεγάλος κυκλικός καταχωρητής ο οποίος αποθηκεύει την πρόσφατη διαδρομή που ακολούθησαν οι διακλαδώσεις του προγράμματος μέχρι μια συγκεκριμένη χρονική στιγμή. Ο μέγεθος του Global History είναι ένας πολύ σημαντικός παράγοντας, ο οποίος επηρεάζει άμεσα την ακρίβεια ενός μηχανισμού πρόβλεψης διακλαδώσεων. Ο TAGE, τον οποίο εξετάζουμε, χρησιμοποιεί το Global History με μέγεθος ~3000 bits τόσο στην διαδικασία του Indexing, όσο και του υπολογισμού του Tag μέσω των Folded Histories τα οποία θα εξετάσουμε στην συνέχεια.

Για τις Κατηγορίες Διακλαδώσεων 2, 3 και 4 εισάγει τα 2 Least Significant Bits (LSB) ενώ για την κατηγορία 1 εισάγει 3 LSB του F στον καταχωρητή Global History βάση της παρακάτω συνάρτησης:

$$F = PC \text{ XOR } (PC \gg 2) \text{ XOR Prediction Bit}$$

Αν η κατηγορία της διακλάδωσης είναι η πρώτη τότε:

$$F = F \text{ XOR } (\text{Target_Address} \gg 2)$$

4.7.2 Folded Histories

Τα Folded Histories, χρησιμοποιούνται έτσι ώστε να συμπίεσουν το Global History σε καταχωρητές μικρότερου μεγέθους. Αυτό γίνεται διότι για τις περισσότερες διακλαδώσεις η ιστορία είναι πολύ μεγάλη, με αποτέλεσμα να πρέπει να συμπιεστεί σε ένα μικρό αριθμό από bits. Κάθε Folded History είναι στην πραγματικότητα ένας κυκλικός καταχωρητής.

Ο TAGE, διαθέτει δύο τύπους από Folded Histories ανάλογα με την διαδικασία που θα εκτελέσει, Index και Tag. Για κάθε τύπο, υπάρχουν τόσα Folded Histories όσα και τα Μεγέθη Ιστοριών που υπάρχουν στον TAGE. Κάθε Folded History, οποιουδήποτε τύπου εξαρτάται από τις εξής παραμέτρους:

Πραγματικό Μέγεθος Ιστορίας (ΠΜΙ): Το Πραγματικό Μέγεθος Ιστορίας αποτελεί το μέγεθος ιστορίας που αντιπροσωπεύει η συμπιεσμένη τιμή που βρίσκεται στο παρόν Folded History.

Συμπιεσμένο Μέγεθος (ΣΜ): Το Συμπιεσμένο Μέγεθος είναι το μέγεθος του καταχωρητή του Folded History.

Outpoint (OP): Το Outpoint είναι ίσο με την μαθηματική συνάρτηση: $\text{Outpoint} = \text{Πραγματικό Μέγεθος} \bmod \text{Συμπιεσμένο Μέγεθος}$.

Τα Folded Histories για το Index, διαθέτουν Συμπιεσμένο μέγεθος ίσο με 10.

Η ύπαρξη ενός Folded History για κάθε Μέγεθος Ιστορίας για τον υπολογισμό του Tag, συναντά προβλήματα καθώς είναι ευαίσθητος σε περιοδικά μοτίβα που επαναλαμβάνονται^[10]. Η λύση που εισάχθηκε για την αντιμετώπιση αυτού του προβλήματος είναι η ύπαρξη ενός δεύτερου Folded History για κάθε Μέγεθος Ιστορίας. Για κάθε Μέγεθος Ιστορίας, τα δύο Folded Histories έχουν το ίδιο ΠΜΙ αλλά όσον αφορά το ΣΜ, ο πρώτος είναι ίσος με το μέγεθος του Tag στο συγκεκριμένο Μέγεθος Ιστορίας ενώ ο Δεύτερος είναι ίσος με το μέγεθος του πρώτου μείον ένα.

4.7.3 Path History

Το Path History είναι επίσης, ένας κυκλικός καταχωρητής μεγέθους 27 bits, ο οποίος, αναλαμβάνει να αποθηκεύσει τις διαδρομές των διακλαδώσεων με βάση τις διευθύνσεις τους. Η διαφορά μεταξύ Global και Path History, βασίζεται στο ότι το Global History λαμβάνει υπόψη του το αποτέλεσμα της διακλάδωσης, Taken – Not Taken, ενώ το Path History το PC.

Για τις Κατηγορίες Διακλαδώσεων 2, 3 και 4:

$$P = PC \text{ XOR } (PC \gg 2) \text{ XOR } (PC \gg 4)$$

Αν η κατηγορία της διακλάδωσης είναι η πρώτη τότε:

$$P = P \text{ XOR } (\text{Target_Address} \gg 2) \text{ XOR } (\text{Target_Address} \gg 4)$$

Η ενημέρωση του καταχωρητή γίνεται με την ακόλουθη διαδικασία σε κάθε επανάληψη:

- Ο καταχωρητής Path History πραγματοποιεί αριστερή ολίσθηση κατά μία θέση
- Το αποτέλεσμα της συνάρτησης P γίνεται XORed με τον καταχωρητή του Path History
- Το αποτέλεσμα της συνάρτησης P πραγματοποιεί δεξιά ολίσθηση κατά μία θέση

Για τις κατηγορίες διακλαδώσεων 2, 3 και 4 η επανάληψη της πιο πάνω διαδικασίας εφαρμόζεται 2 φορές ενώ για την κατηγορία διακλαδώσεων 1 η επανάληψη της διαδικασίας εφαρμόζεται 3 φορές.

4.7.4 Υπολογισμός Index

Το Index αποτελεί ουσιαστικά την συνάρτηση η οποία αντιστοιχεί την θέση που βρίσκεται η διακλάδωση στον Φυσικό Πίνακα για το συγκεκριμένο Μέγεθος Ιστορίας. Το Index υπολογίζεται για κάθε Μέγεθος Ιστορίας με βάση τις ακόλουθες μεταβλητές:

- Διεύθυνση Διακλάδωσης

- Μέγεθος Ιστορίας
- Ο Αύξων Αριθμός του συγκεκριμένου Μεγέθους Ιστορίας
- Folded Histories του Index
- Path History

4.7.5 Υπολογισμός Tag

Το Tag αποτελεί την υπογραφή της διακλάδωσης η οποία αποθηκεύεται στο κελί του Φυσικού Πίνακα μαζί με την πρόβλεψη. Αυτό γίνεται έτσι ώστε να επιβεβαιώνεται ότι πρόκειται για την ίδια διακλάδωση προτού αξιοποιηθεί η πρόβλεψη από το συγκεκριμένο κελί. Το Tag υπολογίζεται με βάση τις ακόλουθες μεταβλητές:

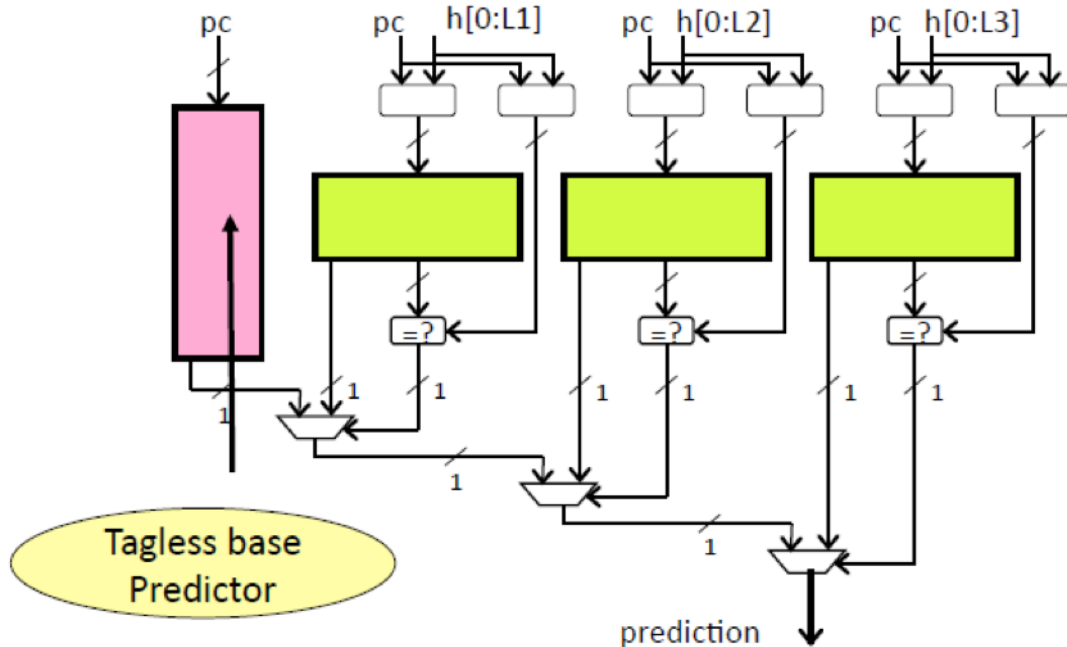
- Διεύθυνση Διακλάδωσης
- Folded Histories του Tag

4.8 Πρόβλεψη Διακλάδωσης

Η πρόβλεψη της διακλάδωσης μέσα από αυτόν τον μηχανισμό αποτελεί έναν σύνθετο και συνάμα πολύπλοκο αλγόριθμο ο οποίος χρησιμοποιεί όλους τους μετρητές και τα συστατικά τα οποία διαθέτει.

Η διαδικασία της πρόβλεψης ξεκινάει από το τελευταίο Μέγεθος Ιστορίας μέχρι και τον Bimodal. Συνολικά υπολογίζονται 22 Indices, 21 Indices για τα Μεγέθη Ιστοριών και 1 Index για τον Bimodal και 21 Tags. Για κάθε διαδικασία πρόβλεψης αντιστοιχούν 2 προβλέψεις τις οποίες ονομάζουμε ως Πρωτεύων και εναλλακτική. Για κάθε Index όσον αφορά τον TAGE, αντιπαραβάλλεται το αποθηκευμένο Tag με αυτό που έχει υπολογιστεί. Σε περίπτωση ισότητας, το κελί θεωρείται ορθό για την συγκεκριμένη διακλάδωση και αποθηκεύονται οι πληροφορίες ως Πρωτεύων Πρόβλεψη. Η διαδικασία συνεχίζει από το σημείο αυτό για την εξακρίβωση της Εναλλακτικής Πρόβλεψης με την ίδια διαδικασία. Για τον Bimodal, ο οποίος δεν διαθέτει Tag, η Πρόβλεψη αποκτάτε μόνο με το Index χωρίς οποιαδήποτε άλλη λογική επιβεβαίωσης του κελιού. Σε περίπτωση όπου δεν υπάρχει Πρωτεύων πρόβλεψη στον TAGE, τότε νοείται ότι δεν υπάρχει ούτε

Εναλλακτική Πρόβλεψη αφού, η Πρωτεύων και η Εναλλακτική είναι οι ίδιες και έχουν επέλθει από τον Bimodal.



Σχήμα 4.8 Διάταξη ολόκληρου του Μηχανισμού Πρόβλεψης TAGE [1]

Μετά την διαδικασία αυτή, ο μηχανισμός αυτός, επιλέγει ποια από τις 2 προβλέψεις θα χρησιμοποιήσει μέσω μιας πολιτικής και κάποιων επιπλέον μετρητών τους οποίους διατηρεί. Ο μηχανισμός, διαθέτει 8 μετρητές οι οποίοι καταγράφουν ο κάθε ένας για ένα σύνολο από Ιστορίες, ποια πρόβλεψη από τις 2 ήταν πιο ακριβής, με κύριο σκοπό την εξακρίβωση αν η Πρωτεύουσα Πρόβλεψη έχει Δεσμευτεί στον πρόσφατο παρελθόν με αποτέλεσμα η Εναλλακτική της Πρόβλεψη να είναι πιο ακριβής. Ο τρόπος επιλογής του μετρητή που θα λάβει υπόψη του, εξαρτάται από τον αύξων αριθμό του Μεγέθους Ιστορίας όπου προέρχεται η Πρωτεύουσα Πρόβλεψη και το αν τα Hysteresis bits του Prediction Counter της Εναλλακτικής Πρόβλεψης είναι μεγαλύτερα ή ίσα του 1. Αν ο μετρητής στον οποίο αντιστοιχεί η Πρωτεύουσα Πρόβλεψη είναι αρνητικός τότε ποιο ακριβής η Εναλλακτική ενώ όταν είναι θετικός θεωρείται πιο ακριβής η Πρωτεύουσα Πρόβλεψη.

Στην συνέχεια εφαρμόζεται η ακόλουθη πολιτική Πρόβλεψης έτσι ώστε να εξαχθεί η Τελική Πρόβλεψη. Η Τελική Πρόβλεψη είναι η Πρωτεύων είτε όταν, ο μετρητής είναι θετικός είτε όταν ο Prediction Counter της Πρωτεύουσας Πρόβλεψης δεν είναι αδύναμος, δηλαδή τα Hysterisis bits του είναι ίσο με μηδέν. Διαφορετικά η Τελική Πρόβλεψη είναι η Εναλλακτική.

4.9 Δέσμευση

Η Δέσμευση είναι η διαδικασία κατά την οποία μια διακλάδωση, προσπαθεί να δεσμεύσει χώρο στον TAGE. Η Δέσμευση συμβαίνει όταν η πρόβλεψη μιας διακλάδωσης ήταν λανθασμένη με αποτέλεσμα, είτε της ανάγκης για δέσμευση ενός κελιού στον TAGE ερχόμενη από τον Bimodal είτε της ανάγκης για δέσμευση ενός κελιού χρησιμοποιώντας μεγαλύτερη ιστορία.

Η διαδικασία αυτή παράγει για κάθε επόμενο μέγεθος ιστορίας το Index που αφορά την διακλάδωση σε αυτό το κελί και ακολούθως εξετάζει τους ακόλουθους κανόνες οι οποίοι διέπουν την πολιτική δέσμευσης του μηχανισμού αυτού. Το εξεταζόμενο κελί θα πρέπει, ο Useful Counter του να είναι ίσος με μηδέν και τα Hysterisis bits του Prediction Counter του να είναι μικρότερα ή ίσα του 1. Αυτή η πολιτική επιβάλλει ότι καμιά διακλάδωση δεν θα είναι σε θέση να αντικαταστήσει μια άλλη διακλάδωση η οποία δίνει ορθές προβλέψεις.

Αφού βρεθεί ένα κατάλληλο κελί, τότε, αποθηκεύεται το Tag της διακλάδωσης, το Prediction bit του Prediction Counter γίνεται ίσο με την τελευταία του κατεύθυνση και τα Hysterisis bits και το Useful Counter ίσα με μηδέν.

4.10 Ενημέρωση

Η Ενημέρωση είναι η διαδικασία όπου αφού έχει παρθεί η Πρόβλεψη, η διακλάδωση έχει ολοκληρώσει την εκτέλεση της και απομένει η εκτίμηση αν η πρόβλεψη ήταν ορθή ή όχι. Γενικά, η διαδικασία της ενημέρωσης του μηχανισμού αυτού είναι η πιο σύνθετη και πολύπλοκη διαδικασία η οποία βασίζεται σε μια πολιτική με αρκετούς κανόνες και διαδικασίες. Η αποτίμηση της πρόβλεψης μπορεί αν πάρει μόνο δύο τιμές, ορθή ή λάθος. Όμως, η διαδικασία λαμβάνει υπόψη της και τις παραμέτρους αν η πρόβλεψη είχε παρθεί

από τον Bimodal ή τον TAGE, αν ήταν η Πρωτεύουσα ή η Εναλλακτική Πρόβλεψη και άλλα τα οποία θα εξετάσουμε στην συνέχεια.

Αν η πρόβλεψη ήταν ορθή, τότε, ενημερώνεται ο Prediction Counter της Πρωτεύουσας Πρόβλεψης όπως έχουμε περιγράψει σε προηγούμενο υπό-κεφάλαιο. Επίσης, αν ο Prediction Counter της Πρωτεύουσας Πρόβλεψης είναι αδύναμος, δηλαδή τα Hysteresis bits του είναι μικρότερα ή ίσα του 1 τότε ενημερώνεται και ο Prediction Counter της Εναλλακτικής Πρόβλεψης.

Αν η πρόβλεψη ήταν λανθασμένη, ενημερώνεται ο Prediction Counter της Πρωτεύουσας Πρόβλεψης και στην συνέχεια εάν η Πρωτεύουσα Πρόβλεψη δεν έχει επέλθει από την ιστορία με το μεγαλύτερο μέγεθος τότε ξεκινάει η προσπάθεια για την δέσμευση μέχρι 2 νέων κελιών, όπως έχουμε περιγράψει πιο πάνω, σε 2 διαφορετικά μεγέθη ιστορίας τα οποία είναι μεγαλύτερα από το πρωτεύων.

Εκτός από τον Prediction Counter, ενημερώνεται ανάλογα μέσω μιας άλλης διαφορετικής πολιτικής ο Useful Counter της Πρωτεύουσας Πρόβλεψης. Ο Useful Counter της Πρωτεύουσας Πρόβλεψης μειώνεται κατά 1 όταν ο Prediction Counter της μετά την πιο πάνω ενημέρωση γίνεται αδύναμος, δηλαδή τα Hysteresis bits του είναι ίσο με μηδέν. Επίσης, μειώνεται κατά 1 όταν ισχύουν τα ακόλουθα:

- Τόσο η Εναλλακτική όσο και η Πρωτεύουσα Πρόβλεψη έχουν επέλθει από τον TAGE
- Και οι δύο προβλέψεις είναι ορθές
- Ο Prediction Counter της Εναλλακτικής Πρόβλεψης είναι δυνατός, δηλαδή τα Hysteresis bits του είναι ίσο με 3
- Ο Useful Counter της Πρωτεύουσας Πρόβλεψης είναι δυνατός
- Με την πιο πάνω προσθήκη, ο μηχανισμός προσπαθεί, να απαλείψει τα κελιά τα οποία οι Εναλλακτικές τους Προβλέψεις δίνουν ορθές προβλέψεις.

Στην αντίθετη περίπτωση, ο Useful Counter της Πρωτεύουσας Πρόβλεψης, αυξάνεται κατά 1 όταν η Πρωτεύουσα πρόβλεψη ήταν ορθή ενώ η Εναλλακτική Πρόβλεψη ήταν λανθασμένη.

Ο TAGE διαθέτει επιπλέον ένα μηχανισμό ο οποίος είναι υπεύθυνος να ελέγχει τον αριθμό των αποτυχημένων προσπαθειών δέσμευσης κελιών λόγω του Useful bit, και όταν αυτός ο αριθμός υπερβεί ένα συγκεκριμένο αριθμό τότε, όλα τα Useful Counters των κελιών του Φυσικού Πίνακα 1 μειώνονται κατά 1. Αυτό επιτρέπει σε νέες διακλαδώσεις να πάρουν τα κελιά διακλαδώσεων οι οποίες δεν εκμεταλλεύονται πλέον το συγκεκριμένο κελί αφού έχουν προχωρήσει σε μεγαλύτερα μεγέθη ιστορίας.

4.11 Διαφορές μεταξύ της έκδοσης 8KB και 64KB

Ο 8KB διαθέτει το μισό μέγεθος Bimodal απ' ότι ο 64KB και 2 φυσικούς Πίνακες επίσης, μεγέθους 896 και 1408 κελιών αντίστοιχα. Ο Useful Counter κάθε κελιού είναι μεγέθους 2 bit. Επίσης, διαθέτει συνολικά 21 διαφορετικά μεγέθη ιστορίας.

Κεφάλαιο 5

TAGE Characterization

5.1 TAGE-SC-L

5.1.1 Statistical Corrector

5.1.2 Loop Predictor

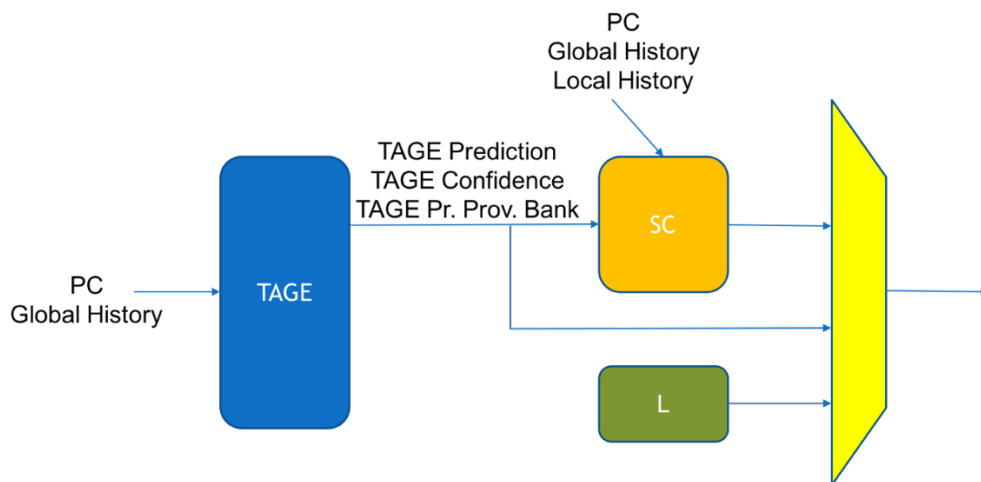
5.1.3 Σύγκριση μεταξύ TAGE-SC-L και TAGE

5.2 Σύγκριση μεταξύ TAGE 8KB και 64KB

5.3 Ανάλυση Heat Maps στον TAGE 64KB

5.1 TAGE-SC-L

Ο TAGE-SC-L προτάθηκε από τον Seznec [1-7] επίσης και έλαβε μέρος στο διαγωνισμό CBP-2016 [1-7] με δύο διαφορετικές εκδόσεις, 8KB και 64KB. Σημαντικό να αναφέρουμε ότι η έκδοση με μέγεθος 8KB κέρδισε τον διαγωνισμό αυτό. Ο μηχανισμός αυτός δεν διαφέρει καθόλου από αυτόν τον οποίο έχουμε εξετάσει λεπτομερώς στο προηγούμενο κεφάλαιο, με την διαφορά ότι προστίθενται δύο επιπλέον μηχανισμοί, Statistical Corrector και Loop Predictor. Η διάταξη αυτού του μηχανισμού πρόβλεψης διακλαδώσεων παρουσιάζεται στο Σχήμα 5.1. Να αναφέρουμε ότι ο Μηχανισμός αυτός διαθέτει Bimodal, ο οποίος συμπεριλαμβάνεται μέσα στον TAGE.



Σχήμα 5.1 Συστατικά Μηχανισμού Πρόβλεψης TAGE-SC-L

5.1.1 Statistical Corrector

Η προσθήκη αυτού του μηχανισμού στο γενικό σύνολο του TAGE οφείλεται στο γεγονός ότι ο ίδιος ο TAGE δεν μπορεί να προβλέψει διακλαδώσεις οι οποίες παρουσιάζουν στατιστική κλίση. Ο σκοπός του μηχανισμού αυτού είναι η αναγνώριση διακλαδώσεων που παρουσιάζουν κλίση και η αναστροφή του αποτελέσματος που δόθηκε από τον TAGE. Ως είσοδο, του δίνονται τα ακόλουθα:

- Πρόβλεψη διακλάδωσης από τον TAGE
- Η σιγουριά της πρόβλεψης από τον TAGE, μέσω του Useful Counter
- Ο αύξων αριθμός της ιστορίας από την οποία πάρθηκε η πρόβλεψη του TAGE
- Η διεύθυνση εντολής της διακλάδωσης
- Global History
- Local History

Ο Statistical Corrector είναι ένας μηχανισμός πρόβλεψης τύπου Multi-GEHL SC [1, 7] ο οποίος στηρίζεται στις έννοιες της μηχανικής μάθησης και των τεχνητών νευρωνικών δικτύων.

Το Local History, το οποίο είναι το μοναδικό συστατικό το οποίο εισάγεται στο παρόν σημείο είναι πολλοί καταχωρητές οι οποίοι αποθηκεύουν, οι οποίοι ο κάθε ένας αποθηκεύει την ιστορία μόνο μιας συγκεκριμένης διακλάδωσης με τον ίδιο τρόπο όπως ο μηχανισμός Global History και Bimodal. Αυτό το συστατικό είναι πολύ ακριβό τόσο ως προς το επιπλέον υλικό όσο και ως προς την πολυπλοκότητα του ίδιου του μηχανισμού.

5.1.2 Loop Predictor

Ο Loop Predictor είναι ο δεύτερος μηχανισμός ο οποίος προστέθηκε, ο οποίος ο σκοπός του είναι η αναγνώριση βασικών επαναληπτικών βρόχων με σταθερό αριθμό επαναλήψεων.

5.1.3 Σύγκριση μεταξύ TAGE-SC-L και TAGE

Πρώτα θα αναλύσουμε την επίδοση του TAGE με την ύπαρξη και χωρίς την ύπαρξη των Statistical Predictor και Loop Predictor ώστε να εκτιμήσουμε το πραγματικό όφελος των δύο αυτών επιπρόσθετων μηχανισμών.

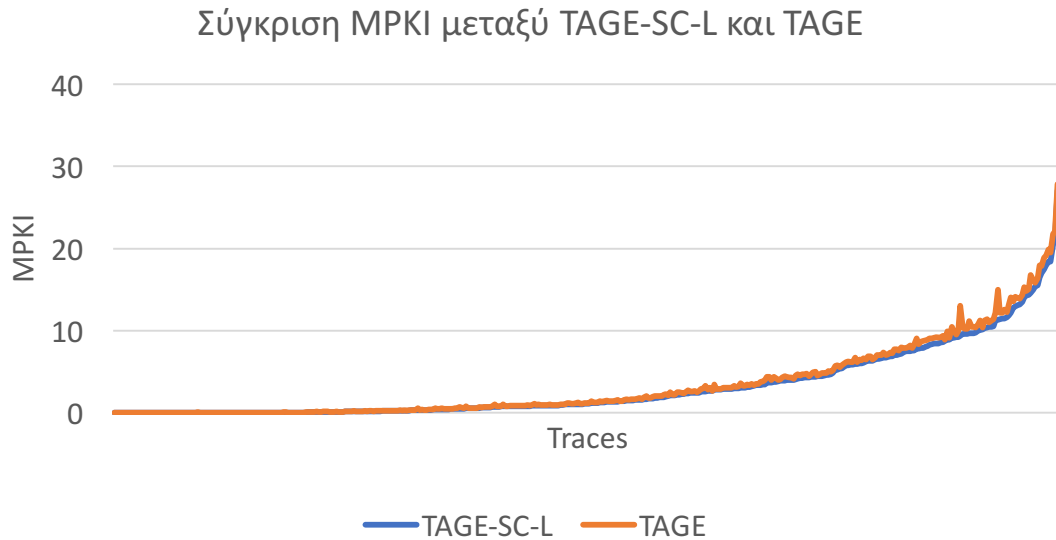
Αφού έχουμε προσομοιώσει τα προγράμματα που έχουμε στην διάθεση μας για τους μηχανισμούς πρόβλεψης διακλαδώσεων TAGE-SC-L και TAGE με μέγεθος 64KB και έχουμε εξάγει τα μετρικά και τα αποτελέσματα μας.

Πρώτα όμως, να αναφέρουμε ότι η ακρίβεια των δύο μηχανισμών είναι:

- TAGE-SC-L : 3.33 MPKI
- TAGE : 3.61 MPKI

Η διαφορά στην ακρίβεια του TAGE, δηλαδή χωρίς τους μηχανισμούς Statistical Corrector και Loop Predictor, βρίσκεται στο 0.28 MPKI.

Στο Σχήμα 5.2 παρουσιάζουμε τα Mispredictions Per Kilo Instructions (MPKI) του TAGE-SC-L και TAGE για όλα τα προγράμματα που έχουμε προσομοιώσει. Μπορούμε να παρατηρήσουμε ότι σχεδόν σε όλα τα προγράμματα ο TAGE παρουσιάζει μείωση στην ακρίβεια των προβλέψεων αλλά το κόστος υλικού και πολυπλοκότητας κατασκευής σε σχέση με την διαφορά στην ακρίβεια που παρουσιάζουν οι δύο είναι μικρή.



Σχήμα 5.2 TAGE-SC-L MPKI εν συγκρίσει με το Cumulative Percentage της διαφοράς του MPKI μεταξύ των TAGE και TAGE-SC-L

5.2 Σύγκριση μεταξύ TAGE 8KB και 64KB

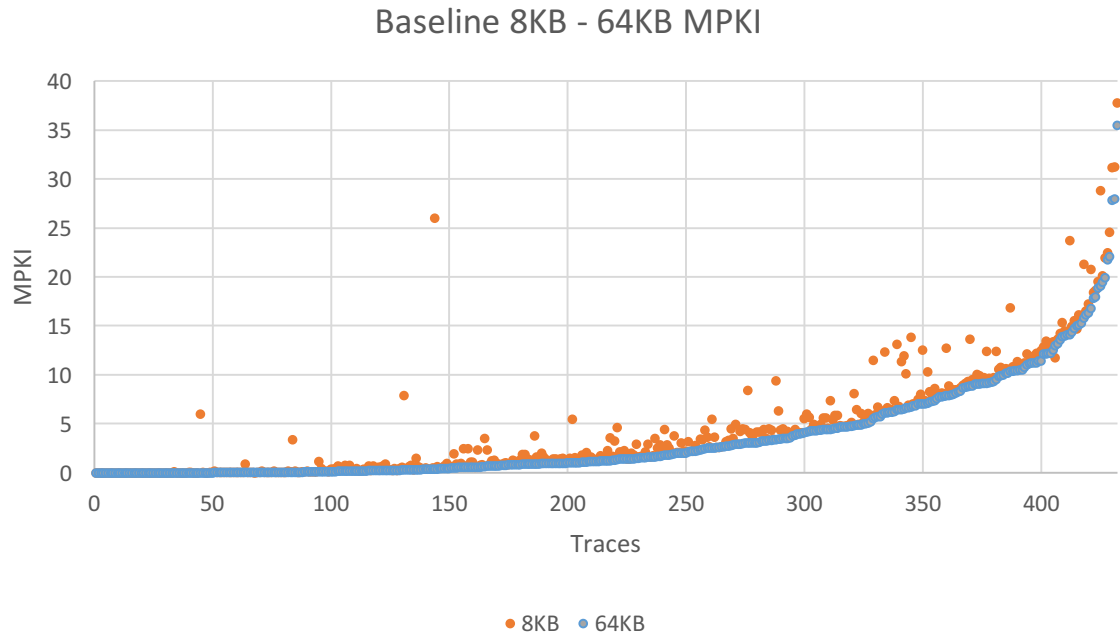
Η ανάλυση μεταξύ των δύο μηχανισμών πρόβλεψης θα μας επιδείξει μια αρχική κατάσταση στο πως ο μηχανισμός συμπεριφέρεται ανάλογα με τα συστατικά που διαθέτει. Μπορούμε να αναλογιστούμε ότι εφόσον ο 64 KB είναι 8 φορές μεγαλύτερος απ' ότι ο 8KB, περιμένουμε σίγουρα βελτίωση όσο αφορά το MPKI.

Αφού, έχουμε προσομοιώσει τα αποτελέσματα των προγραμμάτων και έχουμε εξάγει τα μετρικά και τα αποτελέσματα από αυτά, θα αναλύσουμε καταρχάς τις διαφορές Overall, και στην συνέχεια θα εμβαθύνουμε έτσι ώστε να εξετάσουμε τις συμπεριφορές των προγραμμάτων ατομικά. Αυτό, μας επιτρέπει στο να έχουμε μια πιο σφαιρική εικόνα της συμπεριφοράς των 2 μηχανισμών οι οποίοι βασίζονται στα ίδια συστατικά αλλά διαφέρουν στο μέγεθος. Το μέγεθος θα είναι ένα βασικό κριτήριο, αφού θα εξετάσουμε αν πραγματικά ο 64KB χρησιμοποιεί αποδοτικά τον επιπλέον χώρο που διαθέτει από τον 8KB.

Accuracy – Distribution - MPKI								
	Properties		Overall Accuracy	Average				MPKI
				Accuracy		Distribution		
	Size (KB)	Useful Counter Width		Bimodal	TAGE	Bimodal	TAGE	
Baseline	64	1	97,36%	99,71%	96,39%	32,13%	67,87%	3,61
	8	2	96,84%	99,30%	95,31%	39,39%	60,61%	4,43

Σχήμα 5.4 Παρουσιάζει τους Μέσους Όρους της ακρίβειας, της κατανομής και του MPKI για τα δύο μεγέθη 8KB και 64KB

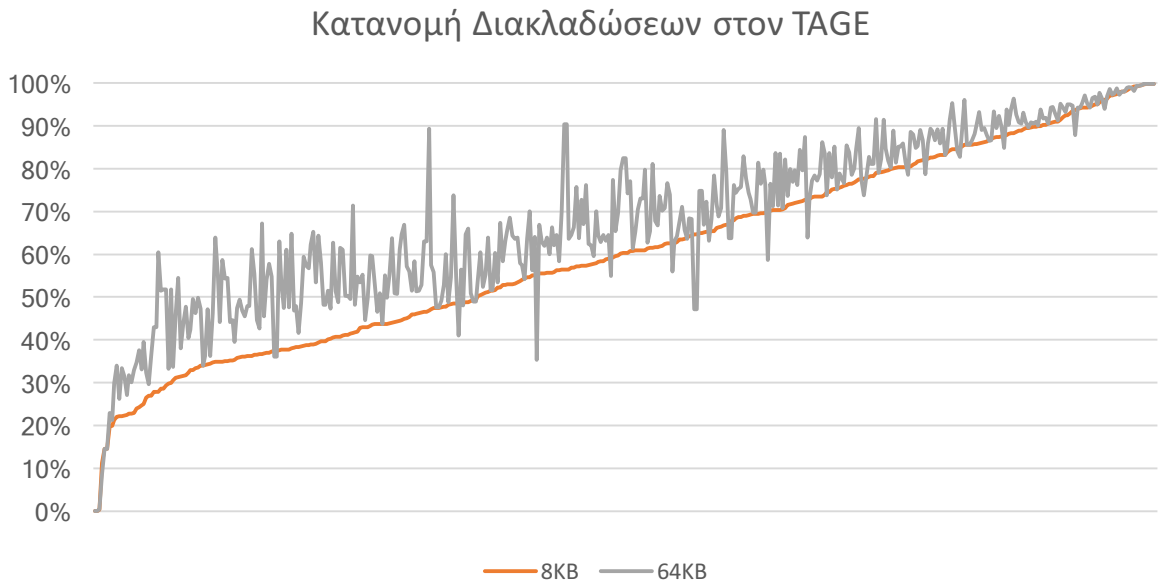
Όπως παρουσιάζουμε στο Σχήμα 5.4, μπορούμε να διαπιστώσουμε ότι το μεγαλύτερο ποσοστό προβλέψεων έρχεται από τον TAGE με 67,87% και 60,61% αντίστοιχα. Επίσης, μπορούμε να παρατηρήσουμε ότι η ακρίβεια του Bimodal είναι σημαντικά υψηλή και αγγίζει τα ποσοστά των 99,71% και 99,30% αντίστοιχα. Αυτό μας υποδεικνύει γενικά ότι ο Bimodal έχει υψηλή ακρίβεια όταν ο TAGE αναλαμβάνει τις δύσκολες διακλαδώσεις οι οποίες απαιτούν σύνθετους χειρισμούς και πολλαπλά κελιά στους Φυσικούς Πίνακες για την ορθή πρόβλεψη τους. Η ακρίβεια του TAGE, η οποία φτάνει στα 96,39% και 95,31% αντίστοιχα μπορούμε να πούμε ότι είναι αρκετά υψηλή αν αναλογιστούμε πρώτα το γεγονός ότι ο TAGE αναλαμβάνει όχι μόνο όλες τις δύσκολες διακλαδώσεις αλλά και όλες τις διακλαδώσεις οι οποίες έτυχαν λανθασμένης πρόβλεψης από τον Bimodal όπου και στην συνέχεια έχει αναλάβει ο TAGE την πρόβλεψη τους. Επίσης, η κατανομή των διακλαδώσεων είναι μεγαλύτερη, άρα ο TAGE όχι μόνο αναλαμβάνει περισσότερες διακλαδώσεις και συνάμα προβλέψεις αλλά οι περισσότερες από αυτές είναι πιο δύσκολες.



Σχήμα 5.5 Παρουσιάζει τα αποτελέσματα του MPKI για όλα τα Traces για τα δύο μεγέθη 8KB και 64KB

Η πιο πάνω γραφική παράσταση, που παραθέτουμε στο Σχήμα 5.5, παρουσιάζει για όλα τα προγράμματα που εξετάσαμε, το MPKI στον άξονα των y για τους μηχανισμούς πρόβλεψης 8KB και 64KB. Αυτό που μπορούμε να εξάγουμε από το γράφημα αυτό είναι το γεγονός ότι επιβεβαιώνουμε τις προσδοκίες μας, αφού MPKI στον 64KB είναι σημαντικά μικρότερο απ' ό τι στον 8KB. Σημαντικό να αναφέρουμε ότι ένα συγκεκριμένο πρόγραμμα παρουσιάζει βελτίωση της τάξης των 68x ενώ η μέση αύξηση του MPKI στον 8KB κυμαίνεται το 2-5x.

Το ερώτημα που εγείρεται είναι αν τα προγράμματα, στον Μηχανισμός Πρόβλεψης με χωρητικότητα 64KB, εκμεταλλεύονται επαρκώς τον επιπλέον χώρο που διαθέτει έναντι του 8KB. Επίσης ακόμη ένα ερώτημα είναι αν εκτός από τον επιπρόσθετο χώρο που διαθέτει να έχει βοηθήσει το γεγονός ότι με την αύξηση αυτή, οι διακλαδώσεις έχουν διασπαρθεί πιο καλά και το Aliasing έχει μειωθεί. Η μείωση του Aliasing, όπως έχουμε ήδη αναφέρει σε προηγούμενο κεφάλαιο, είναι ένα σημαντικό πρόβλημα που οι μηχανισμοί πρόβλεψης προσπαθούν να ελαχιστοποιήσουν.



Σχήμα 5.6 Παρουσιάζει την κατανομή των προβλέψεων στον TAGE
για τα δύο μεγέθη 8KB και 64KB

Στην πιο πάνω γραφική παράσταση, που παραθέτουμε στο Σχήμα 5.6, παρουσιάζουμε την κατανομή των προβλέψεων στον TAGE για τα δύο διαφορετικά μεγέθη, 8KB και 64KB, του μηχανισμού πρόβλεψης. Όπως μπορούμε να διακρίνουμε, τα μισά προγράμματα παραδίδουν την ευθύνη πρόβλεψης της διακλάδωσης έως και στο 60% στον TAGE τόσο για τον 64KB όσο και για τον 8KB. Κάποια προγράμματα αγγίζουν και ποσοστό μεγαλύτερο του 90% που μας υποδεικνύει την απρόβλεπτη τους συμπεριφορά. Αυτό μπορούμε να το συμπεράνουμε καθώς, αφού έχουμε μελετήσει τις διαδικασίες και τους μηχανισμούς που διαθέτει ο συγκεκριμένος μηχανισμός πρόβλεψης, ο TAGE αναλαμβάνει τις διακλαδώσεις οι οποίες χρειάζονται ιδιαίτερο χειρισμό για την ορθή πρόβλεψη τους κάτι που αποτυγχάνει να εφαρμόσει ο Bimodal.

Επίσης, μπορούμε να διακρίνουμε ότι ενόσω ο 64KB διαθέτει 8 φορές μεγαλύτερο μέγεθος και ο Bimodal 2 φορές μεγαλύτερο μέγεθος, εντούτοις, μπορούμε να διακρίνουμε ότι ο TAGE στον 64KB αναλαμβάνει μεγαλύτερο μέρος προβλέψεων απ' ότι στον 8KB. Το γεγονός αυτό μας φανερώνει ότι τα προγράμματα στον 8KB δυσκολεύονται να δεσμεύσουν χώρο στον TAGE λόγω της μεγάλης ζήτησης που επικρατεί από τις διακλαδώσεις. Ο 64KB όμως, ο οποίος διαθέτει 8 φορές μεγαλύτερο

χώρο, δημιουργεί την ελευθερία στα προγράμματα της δέσμευσης χώρου σε αυτόν από τις διακλαδώσεις.

Σε κάποια προγράμματα μπορούμε να διακρίνουμε ότι το πρόσημο της διαφοράς μεταξύ του 64KB και του 8KB έχει αρνητικό πρόσημο κάτι που μας υποδεικνύει ότι ο διπλασιασμός του μεγέθους του Bimodal, αύξησε την ακρίβεια του λόγω της μείωσης του Aliasing μεταξύ των διακλαδώσεων.

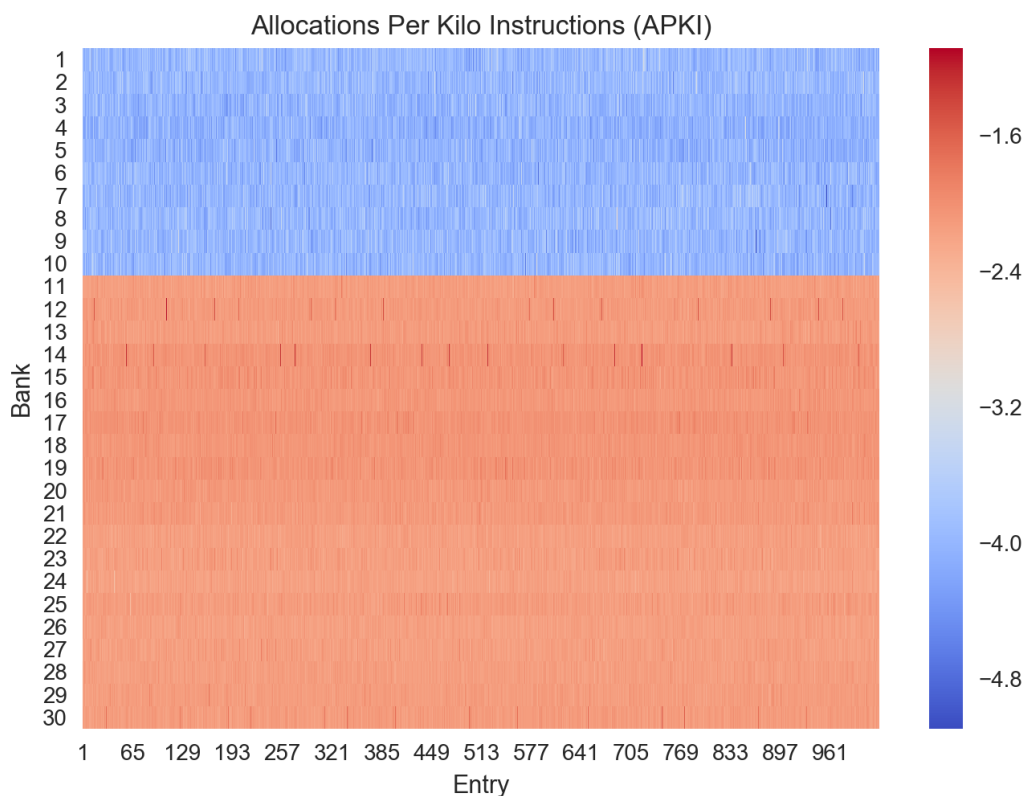
5.3 Ανάλυση Heat Maps στον TAGE

Σε αυτήν την συγκεκριμένη ανάλυση θα εστιάσουμε την προσοχή μας στον TAGE με χωρητικότητα 64KB και θα μελετήσουμε τη συνολική συμπεριφορά του με βάση Heat Maps που έχουμε παράγει για όλα τα προγράμματα που έχουμε προσομοιώσει. Τα Heat Maps θα μας βοηθήσουν να απεικονίσουμε και να συλλάβουμε οπτικά την πραγματική συμπεριφορά και τον αντίκτυπο των προγραμμάτων στους Φυσικούς Πίνακες του TAGE. Θα εξετάσουμε τα Heat maps των Allocations Per Kilo Instructions (APKI), Correct Predictions Per Kilo Instructions (CPKI) και MPKI. Το APKI αντιπροσωπεύει τον αριθμό των δεσμεύσεων, το CPKI τον αριθμό των ορθών προβλέψεων και το MPKI τον αριθμό των λανθασμένων προβλέψεων σε συγκεκριμένο κελί ανά 1000 εντολές μηχανής.

Αρχικά αφού προσομοιώσαμε όλα τα προγράμματα, περισυλλέξαμε τον αριθμό των Δεσμεύσεων, ορθών και λανθασμένα προβλέψεων για κάθε κελί των Φυσικών Πινάκων του TAGE για αυτά. Στην συνέχεια, εφαρμόσαμε σε αυτά μια μορφή κανονικοποίησης (normalization) χρησιμοποιώντας τον αριθμό των εντολών κάθε προγράμματος. Κάθε κελί του Heat Map αντιπροσωπεύει ένα κελί των Φυσικών Πινάκων. Τέλος, αφού υπολογίσαμε τους μέσους όρους κάθε κελιού των Φυσικών Πινάκων των τριών αυτών μετρικών για όλα τα προγράμματα που προσομοιώσαμε, κατασκευάσαμε τα οπτικά Heat Maps που θα εξετάσουμε.

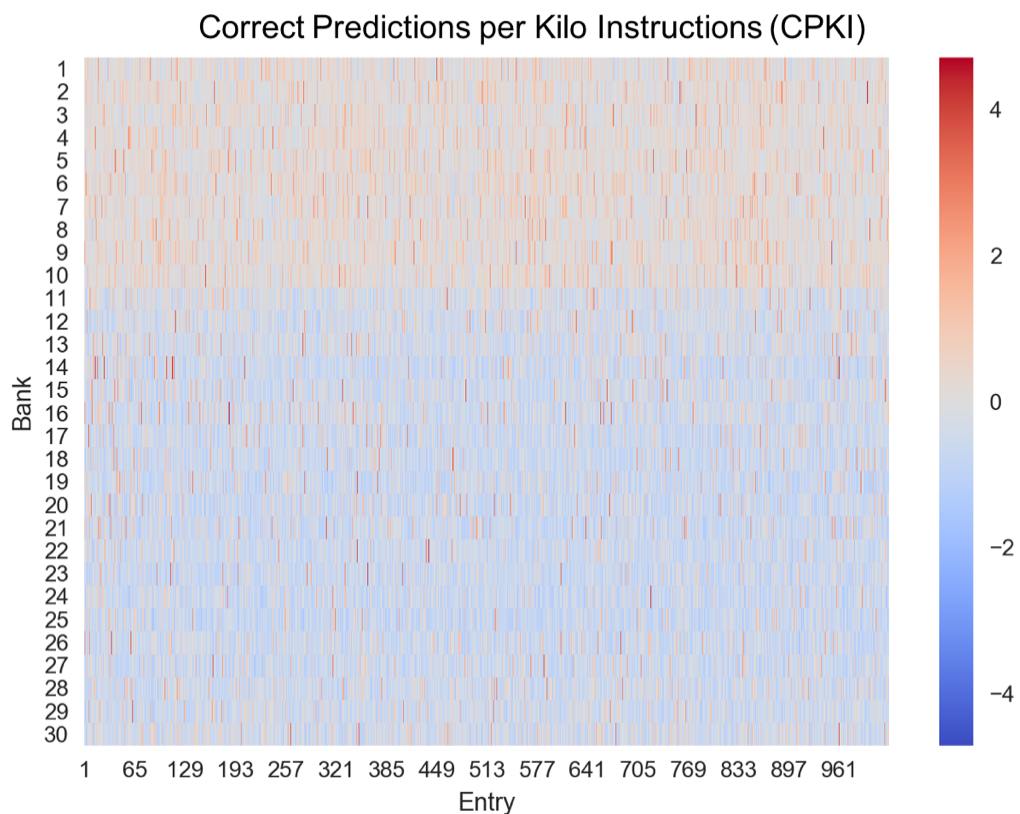
Εξετάζοντας το Heat Map για το APKI, Σχήμα 5.7, συμπεραίνουμε ότι υπάρχει μεγάλη διαφορά μεταξύ των δύο Φυσικών Πινάκων του TAGE ως προς τις δεσμεύσεις διακλαδώσεων. Αυτό μπορεί να συμβαίνει για δύο λόγους

1. Οι περισσότερες διακλαδώσεις χρειάζονται χρήση μεγαλύτερης ιστορίας από αυτές που δρομολογούνται στον πρώτο Φυσικό Πίνακα
2. Λόγω της πολιτικής δέσμευσης που ακολουθεί ο TAGE, όπου κατά την λάθος πρόβλεψη μιας διακλάδωσης δεσμεύονται έως και 2 νέα κελιά στους Φυσικούς Πίνακες. Έτσι, δίνει προτεραιότητα στον δεύτερο Φυσικό Πίνακα αφού αν η πρόβλεψη έχει παρθεί από τον πρώτο, τα Μεγέθη Ιστορίας που δρομολογούνται σε αυτόν είναι σημαντικά λιγότερα και δεσμεύει πιο εύκολα κελιά στον δεύτερο.



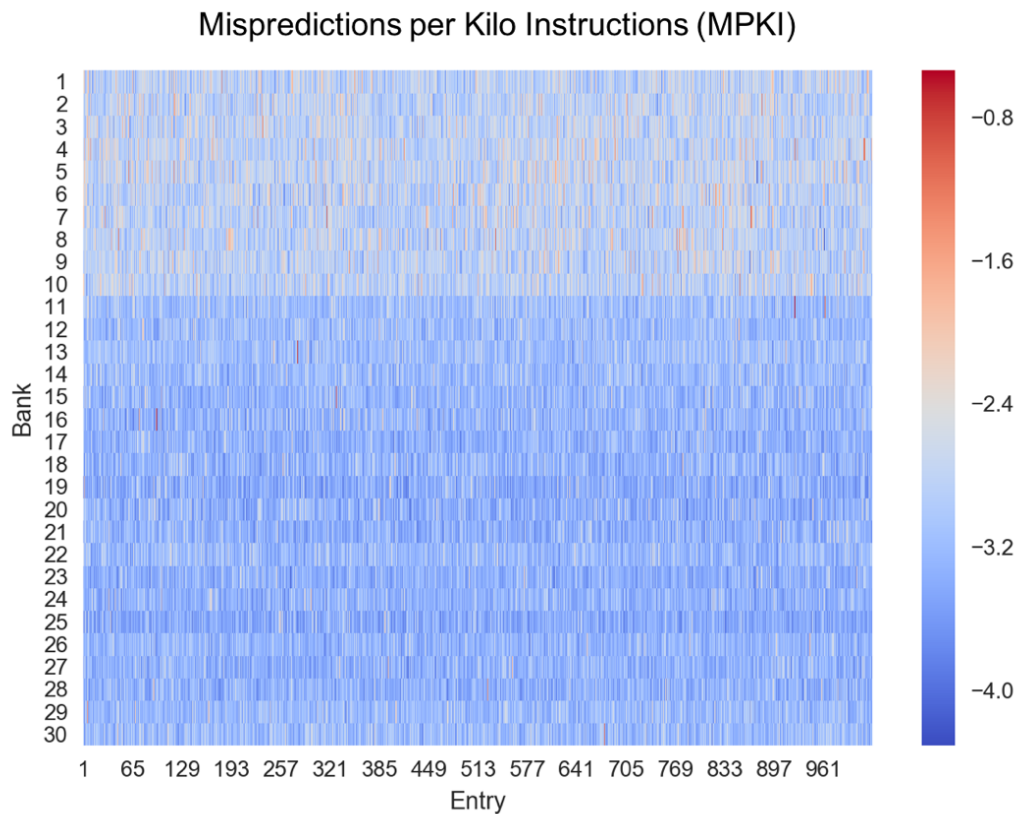
Σχήμα 5.7 Παρουσιάζει το APKI για τους 2 Φυσικούς Πίνακες του TAGE σε λογαριθμική κλίμακα

Εξετάζοντας το Heat Map για το CPKI, Σχήμα 5.8, παρατηρούμε ότι ενόσω το APKI στον πρώτο Φυσικό Πίνακα ήταν χαμηλότερο του δεύτερου, εντούτοις το CPKI που πετυχαίνει ο πρώτος Φυσικός Πίνακας είναι μεγαλύτερο από του δεύτερου. Μπορούμε επίσης να διακρίνουμε ότι στον δεύτερο Φυσικό Πίνακα μεγάλο μέρος των κελιών του συνεισφέρει ελάχιστα στις ορθές προβλέψεις. Το ίδιο μπορούμε να διακρίνουμε και στον πρώτο όμως σε τα κελιά αυτά δίνουν περισσότερες ορθές προβλέψεις από αυτά του δεύτερου. Τέλος, διακρίνουμε αρκετά κελιά στον δεύτερο και ελάχιστα στον πρώτο Φυσικό Πίνακα τα οποία δίνουν με διαφορά τις περισσότερες ορθές προβλέψεις από όλα τα κελιά.



Σχήμα 5.8 Παρουσιάζει το CPKI για τους 2 Φυσικούς Πίνακες του TAGE σε λογαριθμική κλίμακα

Τέλος, εξετάζοντας το Heat Map του MPKI, Σχήμα 5.9, παρατηρούμε ότι το μεγαλύτερο μέρος των λανθασμένων προβλέψεων του TAGE προέρχεται από τον πρώτο Φυσικό Πίνακα. Αυτό είναι κάτι το οποίο αναμέναμε καθώς τα κελιά που δίνουν τις περισσότερες ορθές προβλέψεις είναι αναμενόμενο να δίνουν και τις περισσότερες λανθασμένες. Επίσης, μπορούμε να διακρίνουμε ξεκάθαρα τη διαφορά του αριθμού των λανθασμένων προβλέψεων που προέρχονται από τον δεύτερο αφού είναι αρκετά μικρότερος. Ακόμη, μπορούμε να διακρίνουμε αρκετά κελιά τόσο στον δεύτερο όσο και στον πρώτο Φυσικό Πίνακα τα οποία δίνουν τις περισσότερες λανθασμένες προβλέψεις από όλα τα κελιά.



Σχήμα 5.9 Παρουσιάζει το MPKI για τους 2 Φυσικούς Πίνακες του TAGE σε λογαριθμική κλίμακα

Κεφάλαιο 6

Έννοια των Κενών Κελιών

6.1 Εισαγωγή Κεφαλαίου

6.2 Ανάλυση αποτελεσμάτων της προσομοίωσης TAGE με κενά κελιά

6.1 Εισαγωγή Κεφαλαίου

Όπως έχουμε διατυπώσει σε προηγούμενο κεφάλαιο, εγείρεται το ερώτημα αν τα πλείστα προγράμματα εκμεταλλεύονται την χωρητικότητα του 64KB TAGE και αν η αύξηση του μεγέθους έχει βοηθήσει στην καλύτερη διασπορά των διακλαδώσεων σε αυτόν με αποτέλεσμα την μείωση του Aliasing.

Η πειραματική διαδικασία που έχουμε ακολουθήσει είναι η εισαγωγή κενών κελιών – τρυπών στον TAGE τα οποία υπολογίζονται κατά την διαδικασία του Indexing αλλά δεν υπάρχουν στις υπόλοιπες διαδικασίες. Αυτό θα επέτρεπε την δημιουργία μιας νοητής μεγέθυνσης του TAGE, διατηρώντας τον όμως στο ίδιο μέγεθος από το αρχικό του. Με αυτή τη συγκεκριμένη πειραματική διαδικασία θα μπορούσαμε να εξετάσουμε κατά πόσο η εκτίμηση αυτή ισχύει. Θα μπορούσαμε να αναμένουμε βελτίωση, μείωση της ακρίβειας ή ισότητα. Αυτό, διότι με την αύξηση των συνολικών θέσεων στον TAGE επηρεάζουμε ταυτόχρονα το Index και Tag με αποτέλεσμα να μην μπορούμε να προβλέψουμε το αποτέλεσμα του συγκεκριμένου πειράματος στην εκτέλεση συγκεκριμένου προγράμματος. Μπορούμε να διακρίνουμε στο Σχήμα 6.1 πως μετατρέπεται ο Φυσικός Πίνακας του TAGE με την χρήση της έννοιας των κενών κελιών. Αξίζει να σημειώσουμε την παρατήρηση ότι διπλασιάζουμε μεν τα κελιά που διαθέτει ο Φυσικός Πίνακας αλλά λόγω του γεγονότος ότι τα μισά από αυτά είναι κενά, δηλαδή δεν μπορείς να εκτελέσεις πρόσβαση, είναι ίσο με τα κελιά που υπήρχαν πριν από την εισαγωγή των κενών κελιών. Ακόμη, αλλάζει ο τρόπος Indexing και Tagging, αφού το μέγεθος του πίνακα αλλάζει δημιουργώντας μια αλυσιδωτή αλλαγή στις μεταβλητές οι οποίες αποτελούν τον τρόπο υπολογισμού τους.

Η εισαγωγή των κενών κελιών στους δύο Φυσικούς Πίνακες του TAGE είναι τυχαία με την προσθήκη όμως κάποιων επιπρόσθετων κανόνων. Η προσθήκη των κανόνων εισάχθηκε έτσι ώστε να μειώσουμε το bias που προκύπτει από την τυχαία αυτή κατανομή.



Σχήμα 6.1 Παρουσιάζει την κατανομή των προβλέψεων στον TAGE
για τα δύο μεγέθη 8KB και 64KB

6.2 Ανάλυση αποτελεσμάτων της προσομοίωσης TAGE με κενά κελιά

Αποφασίσαμε στην συγκεκριμένη ανάλυση να χρησιμοποιήσουμε τον μηχανισμό TAGE με μέγεθος 8KB. Διπλασιάσαμε το μέγεθος των δύο Φυσικών Πινάκων του TAGE, κρατώντας τον Bimodal στο ίδιο μέγεθος. Ταυτόχρονα επεκτείναμε τον προσομοιωτή, προσθέτοντας έναν αλγόριθμο τυχαίας κατανομής κενών κελιών ελαχιστοποιώντας το bias που προκύπτει με ένα σύνολο κανόνων. Αυτό μας έδωσε την δυνατότητα να δημιουργήσουμε τον μηχανισμό πρόβλεψης διακλαδώσεων TAGE με κενά κελιά, έτσι ώστε να μπορούμε να προσομοιώσουμε τα προγράμματα τα οποία έχουμε στην διάθεση μας και να αναλύσουμε και συγκρίνουμε τα αποτελέσματα μας.

Ακολούθως θα εξετάσουμε τα αποτελέσματα που έχουμε εξάγει από την προσομοίωση των προγραμμάτων χρησιμοποιώντας τον μηχανισμό πρόβλεψης TAGE με κενά κελιά και συνολικό μέγεθος 8KB. Αρχικά θα μελετήσουμε τα συνολικά αποτελέσματα τα οποία παρουσιάζουμε στο Σχήμα 6.2. Όπως μπορούμε να παρατηρήσουμε, η συνολική ακρίβεια του μηχανισμού μειώνεται καθώς οι δύο μηχανισμοί του, Bimodal και TAGE, παρουσιάζουν με τη σειρά τους μικρή μείωση στην ακρίβεια τους. Επίσης, όσο αφορά την κατανομή των προβλέψεων ανάμεσα στους δύο μηχανισμούς, μπορούμε να διακρίνουμε μικρή μείωση στον αριθμό των προβλέψεων διακλαδώσεων που εξέρχονται

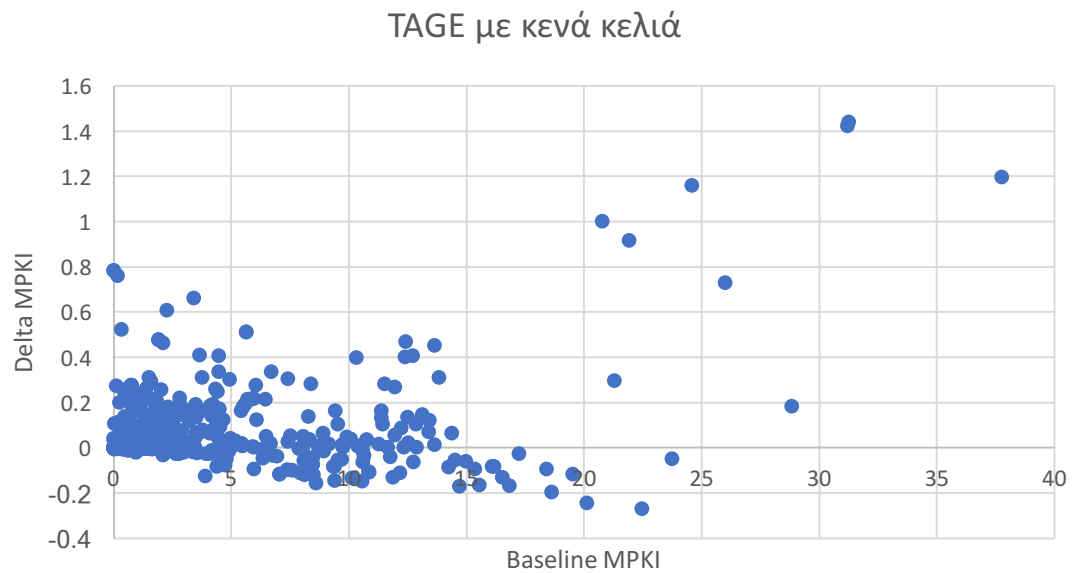
από τον TAGE. Τέλος, στον παρόντα Πίνακα μπορούμε να παρατηρήσουμε την μικρή αύξηση του MPKI, το οποίο συνιστά μείωση της ακρίβειας του μηχανισμού πρόβλεψης TAGE με κενά κελιά έναντι του αρχικού TAGE.

Accuracy – Distribution - MPKI								
	Properties		Overall Accuracy	Average				MPKI
				Accuracy		Distribution		
	Size (KB)	Useful Counter Width		Bimodal	TAGE	Bimodal	TAGE	
Baseline	64	1	97,36%	99,71%	96,39%	32,13%	67,87%	3,61
	8	2	96,84%	99,30%	95,31%	39,39%	60,61%	4,43
With Holes	8	2	96,78%	99%	95,22%	40,94%	59,06%	4,52

Σχήμα 6.2 Παρουσιάζει τους Μέσους Όρους της ακρίβειας, της κατανομής και του MPKI για τα δύο μεγέθη 8KB και 64KB του TAGE και τον TAGE με τρύπες που έχουμε δημιουργήσει

Η γραφική παράσταση, που παραθέτουμε στο Σχήμα 6.3, παρουσιάζει για όλα τα προγράμματα που εξετάσαμε, την διαφορά του MPKI μεταξύ του TAGE με χωρητικότητα 8KB και του TAGE με κενά κελιά στον άξονα των y και το MPKI του TAGE με χωρητικότητα 8KB στον χ άξονα. Όπως μπορούμε να παρατηρήσουμε μειώνεται η ακρίβεια του μηχανισμού στα περισσότερα προγράμματα, ποσοστό 70,83%, με μέση αύξηση 0,1391 MPKI. Ταυτόχρονα όμως, βελτιώνεται η ακρίβεια του μηχανισμού σε αρκετά προγράμματα με ποσοστό 22,92% και μέση βελτίωση 0,0570 MPKI. Τέλος, η ακρίβεια σε ένα μικρό ποσοστό, 6,25% των προγραμμάτων παραμένει η ίδια. Αυτό μας οδηγεί στο συμπέρασμα ότι ενόσω τα συνολικά αποτελέσματα μας υπόδειξαν ότι έχει μειωθεί η ακρίβεια του μηχανισμού, εντούτοις, κάποια από τα προγράμματα βελτιώνουν την ακρίβεια του. Η μείωση της συνολικής ακρίβειας του

μηχανισμού βασίζεται στο γεγονός ότι η μέση αύξηση του MPKI είναι μεγαλύτερη από την μέση μείωση του.



Σχήμα 6.3 Παρουσιάζει την διαφορά του MPKI μεταξύ του TAGE με χωρητικότητα 8KB και του TAGE με κενά κελιά σε σχέση με το MPKI του TAGE με χωρητικότητα 8KB

Κεφάλαιο 7

Συμπεράσματα και Μελλοντική Εργασία

7.1 Συμπεράσματα

7.2 Μελλοντική Εργασία

7.1 Συμπεράσματα

Μέσα από την ολοκλήρωση αυτής της Διπλωματικής Εργασίας, μπορούμε να κατανοήσουμε πλήρως τον τρόπο λειτουργίας κάθε μηχανισμού που διαθέτει ο μηχανισμός πρόβλεψης διακλαδώσεων TAGE και μπορούμε να αναγνωρίσουμε κάθε ένα από αυτά.

Επίσης, έχουμε αναλύσει την συμπεριφορά των μηχανισμών που έχουμε σχολιάσει εκτενών σε προηγούμενα κεφάλαια. Τα συμπεράσματα που μπορούμε να κρατήσουμε είναι τα ακόλουθα Αρχικά, έχουμε μελετήσει την σπουδαιότητα της ύπαρξης των Bimodal και TAGE και τους ρόλους που έχουν να επιτελέσουν με σκοπό την ορθή πρόβλεψη των διακλαδώσεων. Ακόμη, έχουμε θέσει το γεγονός ότι η αύξηση του μεγέθους του μηχανισμού σε σχέση με την χρησιμοποίηση των επιπλέον κελιών δεν είναι άμεσα συνυφασμένη κάτι που μας υποδεικνύει ότι ο TAGE αντιμετωπίζει προβλήματα Aliasing. Επίσης, η αύξηση του μεγέθους του μηχανισμού επηρεάζει θετικά την ακρίβεια των προβλέψεων.

Τέλος, έχουμε μελετήσει μια αρχική προσπάθεια, την εισαγωγή κενών κελιών στον TAGE. Ο σκοπός της προσπάθειας αυτής είναι η αντιμετώπιση του Aliasing που αντιμετωπίζει ο μηχανισμός με την εισαγωγή κενών κελιών έτσι ώστε να μεγαλώσει ο χώρος που διαθέτει ο TAGE χωρίς όμως την αύξηση του πραγματικού του μεγέθους. Έτσι διατηρούμε τον μηχανισμό στο αρχικό του μέγεθος. Ο τελικός στόχος όμως είναι η σημαντική μείωση του Aliasing με άμεση ανταπόκριση την βελτίωση της ακρίβειας του.

7.2 Μελλοντική Εργασία

Η παρούσα Διπλωματική Εργασία αποτελεί την απαρχή μιας πιο μεγάλης έρευνας η οποία περιστρέφεται γύρω από τον Μηχανισμό Πρόβλεψης Διακλαδώσεων, TAGE. Έχουμε εφοδιαστεί με όλες τις απαραίτητες γνώσεις του τρόπου λειτουργίας κάθε μηχανισμού που διαθέτει για την κατανόηση της γενικής λειτουργίας του. Ακόμη παρέχουμε την αρχική ανάλυση του TAGE και μιας αρχικής ιδέας αντιμετώπισης του Aliasing. Σαν συνέχεια της εργασίας αυτής, θα ήθελα στο αυτό το σημείο, να προτείνω την πιο λεπτομερή ανάλυση του TAGE με κενά κελιά και πως αυτός συμπεριφέρεται. Επίσης, ίσως η περαιτέρω ανάλυση του αρχικού TAGE να μπορούσε να μας προϋδεάσει με κάποια άλλη διαφορετική λύση η οποία θα μπορούσε να αναπτυχθεί.

Η συνέχεια αυτής της εργασίας θα πρέπει να εστιάσει στην αύξηση της ακρίβειας του TAGE, μέσα από περεταίρω ανάλυση, δοκιμές διαφορετικής κατανομής των κενών κελιών και διαφορετικούς τρόπους προσέγγισης για την λύση του προβλήματος. Ακόμη, όπως έχουμε αναφέρει οι διακλαδώσεις στον TAGE με μέγεθος 8KB δυσκολεύονται να δεσμεύσουν χώρο στους Φυσικούς Πίνακες, λόγω ανεπάρκειας χώρου. Ταυτόχρονα έχουμε όμως υψηλή ακρίβεια στον Bimodal κάτι που μπορεί να μας υποδεικνύει ότι μπορούμε να ανεχτούμε κάποιες λάθος προβλέψεις πριν δεσμεύσουμε χώρο στους Φυσικούς Πίνακες. Αυτό μπορεί να αυξήσει την ακρίβεια λόγω του ότι οι Φυσικοί Πίνακες θα φιλοξενούν διακλαδώσεις οι οποίες πραγματικά χρειάζονται να χρησιμοποιήσουν κάποια ιστορία για την ορθή πρόβλεψη τους. Επίσης, θα ήταν ενδιαφέρον να μελετήσουμε κατά πόσο η αλλαγή του Bimodal σε Gshare και η χρήση ενός μικρού μεγέθους ιστορίας μπορεί να μειώσει σημαντικά το Aliasing που αντιμετωπίζει και να βελτιώσει ταυτόχρονα την ακρίβεια του. Η αλυσιδωτή αυτή αντίδραση θα μπορεί να απελευθερώσει χώρο στους Φυσικούς Πίνακες για να φιλοξενήσει άλλες διακλαδώσεις οι οποίες πραγματικά χρειάζονται μεγάλο μέγεθος ιστορίας. Η μείωση του Aliasing, επιφέρει βελτίωση της ακρίβειας όπως έχουμε εξετάσει. Η προσθήκη ενός μηχανισμού με ένα σύνολο από μετρητές παρόμοιο με αυτόν που διαθέτει η διαδικασία της Πρόβλεψης, θα μπορούσε δυναμικά να αναλύει τον αριθμό των προσβάσεων στα κελιά από διαφορετικές διακλαδώσεις, μέσω της ανισότητας μεταξύ των Tags, και να αυξάνει ή να μειώνει δυναμικά τον αριθμό των bits των Useful Counters του συγκεκριμένου μεγέθους ιστορίας. Αυτή η δυναμική ρύθμιση μπορεί να εφαρμόζεται ανάμεσα στο Useful Counter και στον Prediction Counter.

Βιβλιογραφία

- [1] André Seznec, TAGE-SC-L Branch Predictors Again, In Proceedings of the 5th Championship on Branch Prediction, <http://www.jilp.org/cbp2016/>, 2016.
- [2] André Seznec and Pierre Michaud, A case for (partially)-tagged geometric history length predictors, Journal of Instruction Level Parallelism (<http://www.jilp.org/vol8>), April 2006.
- [3] André Seznec, The L-TAGE branch predictor, Journal of Instruction Level Parallelism (<http://www.jilp.org/vol9>), May 2007.
- [4] André Seznec, Tage-sc-l branch predictors. In Proceedings of the 4th Championship on Branch Prediction, <http://www.jilp.org/cbp2014/>, 2014.
- [5] André Seznec. A 64 kbytes ISL-TAGE branch predictor. In *Proceedings of the 3rd Championship Branch Prediction*, June 2011.
- [6] André Seznec. A new case for the tage branch predictor. In *Proceedings of the MICRO 44*, 2011.
- [7] A. Seznec, Analysis of the O-GEHL branch predictor, In Proceedings of the 32nd Annual International Symposium on Computer Architecture, June 2005.
- [8] P. Michaud. A ppm-like, tag-based predictor. In *The 1st JILP Championship Branch Prediction Competition (CBP-1)*, 2004
- [9] R. M. Tomasulo, An Efficient Algorithm for Exploiting Multiple Arithmetic Units, <https://www.cs.virginia.edu/~evans/greatworks/tomasulo.pdf>
- [10] Scott McFarling, Combining Branch Predictors, WRL Tech. Report TN-36 (<http://www.hpl.hp.com/techreports/Compaq-DEC/WRL-TN-36.pdf>), June 1993
- [11] Veerle Desmet, Hans Vandierendonck, Koen De Bosschere: Clustered indexing for branch predictors. Microprocessors and Microsystems 31(3): 168-177 (2007)