

Αλγόριθμοι Ανίχνευσης Ελαττωμάτων Θερμικής

Αλληλεπίδρασης σε Μνήμες PCM

Η Μεταπτυχιακή Διπλωματική Εργασία

υποβάλλεται στην ορισθείσα

από τη Συνέλευση

του Τμήματος Μηχανικών Η/Υ και Πληροφορικής

Εξεταστική Επιτροπή

από τον

Σπυρίδων Σπυρίδωνος

ως μέρος των υποχρεώσεων για την απόκτηση του

ΔΙΠΛΩΜΑΤΟΣ ΜΕΤΑΠΤΥΧΙΑΚΩΝ ΣΠΟΥΔΩΝ

ΣΤΗ ΜΗΧΑΝΙΚΗ ΔΕΔΟΜΕΝΩΝ ΚΑΙ ΥΠΟΛΟΓΙΣΤΙΚΩΝ

ΣΥΣΤΗΜΑΤΩΝ

ΜΕ ΕΙΔΙΚΕΥΣΗ

ΣΤΑ ΠΡΟΗΓΜΕΝΑ ΥΠΟΛΟΓΙΣΤΙΚΑ ΣΥΣΤΗΜΑΤΑ

Πανεπιστήμιο Ιωαννίνων

Πολυτεχνική Σχολή

Ιωάννινα 2022

Εξεταστική επιτροπή:

- *Γεώργιος Τσιατούχας*, Καθηγητής Τμήμα Μηχανικών Η/Υ και Πληροφορικής, Πανεπιστήμιο Ιωαννίνων (Επιβλέπων)
- *Αριστείδης Ευθυμίου*, Επίκ. Καθηγητής Τμήμα Μηχανικών Η/Υ και Πληροφορικής, Πανεπιστήμιο Ιωαννίνων
- *Βασίλειος Τενέντες*, Επίκ. Καθηγητής Τμήμα Μηχανικών Η/Υ και Πληροφορικής, Πανεπιστήμιο Ιωαννίνων

ΠΕΡΙΕΧΟΜΕΝΑ

<i>Καταλογος Σχηματων</i>	iii
<i>Καταλογος Πινακων</i>	v
<i>Καταλογος Αλγοριθμων</i>	vi
<i>Περιληψη</i>	vii
<i>Extended Abstract</i>	ixx
ΚΕΦΑΛΑΙΟ 1 Εισαγωγή	1
1.1 Στόχοι	1
1.2 Δομή της Διατριβής	2
ΚΕΦΑΛΑΙΟ 2 Παρουσιαση των Μνημων PCM	3
2.1 Εφαρμογές και Χρήση των PCM	4
2.2 Δομή Κελιού και Λειτουργία των Μνημών PCM	5
2.2.1 Δομή Κελιού.....	5
2.2.2 Λειτουργία των Μνημών PCM.....	7
ΚΕΦΑΛΑΙΟ 3 Μοντελα Σφαλαματων και Δοκιμη των Μνημων PCM	14
3.1 Αξιοπιστία των Κελιών στις Μνήμες PCM	14
3.2 Μοντέλα Σφαλαμάτων σε Μνήμες PCM	18
3.2.1 Γενική Περιγραφή των Σφαλαμάτων σε Μνήμες PCM.....	18
3.2.2 Θερμική Αλληλεπίδραση (Thermal Crosstalk)	21
3.3 Αλγόριθμοι Δοκιμής σε Μνήμες PCM	25
3.3.1 Μελέτη Βιβλιογραφίας για τις Βέλτιστες Πρακτικές Δοκιμής στις Μνήμες PCM.....	25
3.3.2 Μοντέλο Σφαλαμάτων Ευαισθησίας στο Μοτίβο της Γειτονιάς (Neighborhood Pattern Sensitive Faults).....	32

ΚΕΦΑΛΑΙΟ 4	Αλγόριθμοι για τη Δοκιμή της Θερμικής Αλληλεπίδρασης στις	
	Μνημες PCM	42
4.1	Τυπική Δοκιμή Μνημών PCM	43
4.2	Ενισχυμένη Δοκιμή Μνημών PCM.....	44
4.3	Προηγμένη Δοκιμή Μνημών PCM	51
ΚΕΦΑΛΑΙΟ 5	Συμπερασματα	63
	<i>Βιβλιογραφια</i>	<i>65</i>
	<i>Ευρετήριο</i>	<i>71</i>

ΚΑΤΑΛΟΓΟΣ ΣΧΗΜΑΤΩΝ

Σχήμα 2.1: Σχηματική Διατομή ενός Κελιού PCM.....	6
Σχήμα 2.2: Συστοιχία Μνήμης PCM.....	7
Σχήμα 2.3: Συστοιχία Μνήμης PCM σε 3-Δ Μορφή.....	7
Σχήμα 2.4: Συνθήκες Μετάβασης.....	9
Σχήμα 2.5: Παλμοί Προγραμματισμού για τα Κελιά PCM.....	10
Σχήμα 2.6: Χαρακτηριστική I-V ενός κελιού PCM.....	11
Σχήμα 2.7: R-I Χαρακτηριστική.....	12
Σχήμα 3.1: Αντίσταση Κελιού στις Καταστάσεις SET και RESET.....	15
Σχήμα 3.2: Αντοχή Κελιού ως Συνάρτηση του Πλάτους του Παλμού RESET.....	16
Σχήμα 3.3: Γραφική Παράσταση Arrhenius.....	17
Σχήμα 3.4: Σχηματική Περιγραφή της Διαταραχής Προγράμματος στις PCM.....	17
Σχήμα 3.5: Η Αντίσταση των Κελιών στην Κατάσταση RESET.....	18
Σχήμα 3.6: Επίδραση Διαταραχής Εγγραφής σε Κελιά PCM.....	23
Σχήμα 3.7: WD κατά Μήκος μιας Γραμμής Bit.....	24
Σχήμα 3.8: Αλγόριθμος Πορείας Ανίχνευσης Λαθών PCM.....	28
Σχήμα 3.9: Snake Addressing Mode.....	30
Σχήμα 3.10: Ορολογία Μοντέλου Σφαλμάτων Ευαισθησίας στο Μοτίβο της Γειτονιάς.....	33
Σχήμα 3.11: Όλα τα Μοτίβα Ενεργής Γειτονιάς.....	36
Σχήμα 3.12: Γράφημα Euler για 3-bit Μοτίβα.....	38
Σχήμα 3.13: Γειτονιές Πλακιδίων Τύπου-1.....	39
Σχήμα 3.14: Γειτονιές Πλακιδίων Τύπου-1 για τα Κελιά	

Βάσης 0, 1, 2, 3 και 4.....	40
Σχήμα 3.15: Επιγραφές Κελιών με τη Μέθοδο των Δύο Ομάδων.....	41
Σχήμα 4.1: Σχήμα Διευθυνσιοδότησης για τη Μετάβαση Τριών Κελιών.....	45
Σχήμα 4.2: Σχήμα Διευθυνσιοδότησης για τη Μετάβαση Τεσσάρων Κελιών.....	52
Σχήμα 4.3: Η Πολυπλοκότητα ως Συνάρτηση των Αριθμών των Κελιών που Γίνονται RESET.....	62

ΚΑΤΑΛΟΓΟΣ ΠΙΝΑΚΩΝ

Πίνακας 3.1: Μοντέλα Σφαλμάτων και Συνθήκες Δοκιμής των Μνημών PCM 20

ΚΑΤΑΛΟΓΟΣ ΑΛΓΟΡΙΘΜΩΝ

Αλγόριθμος 4.1 3-cell transition PCM Testing (Group-1).....	46
Αλγόριθμος 4.2 3-cell transition PCM Testing (Group-2).	48
Αλγόριθμος 4.3 right_arrow(i).....	52
Αλγόριθμος 4.4 right_arrow(i).....	53
Αλγόριθμος 4.5 4-cell transition PCM Testing (Group-1).....	55
Αλγόριθμος 4.6 4-cell transition PCM Testing (Group-2).	58

ΠΕΡΙΛΗΨΗ

Σπυρίδων Σπυρίδωνος, Δ.Μ.Σ. στη Μηχανική Δεδομένων και Υπολογιστικών Συστημάτων, Τμήμα Μηχανικών Η/Υ και Πληροφορικής, Πολυτεχνική Σχολή, Πανεπιστήμιο Ιωαννίνων, Μάρτιος 2022

Αλγόριθμοι Ανίχνευσης Ελαττωμάτων Θερμικής Αλληλεπίδρασης σε Μνήμες PCM
Επιβλέπων: Γεώργιος Τσιατούχας Καθηγητής

Η Μνήμη Αλλαγής Φάσης (Phase Change Memory) είναι ένας τύπος μη πτητικής μνήμης που είναι πολύ πιθανό να αντικαταστήσει την σε ευρεία χρήση μνήμη flash. Η τρέχουσα έρευνα σχετικά με τη μνήμη αλλαγής φάσης στοχεύει στην ενσωμάτωσή της στην κατασκευαστική διαδικασία της τεχνολογίας CMOS που χρησιμοποιείται σήμερα καθώς και στην αύξηση της αξιοπιστίας αυτών των μνημών. Ένα στοιχείο μνήμης αλλαγής φάσης, μπορεί να υπάρχει σε δύο σταθερές καταστάσεις, είτε με υψηλή, είτε με χαμηλή τιμή αντίστασης. Η κατάσταση υψηλής αντίστασης είναι γνωστή ως άμορφη φάση (ή κατάσταση RESET) ενώ η κατάσταση χαμηλής αντίστασης είναι γνωστή ως κρυσταλλική φάση (ή κατάσταση SET). Κατά τη διάρκεια της μετάβασης σε κατάσταση RESET εφαρμόζεται ένας ισχυρός παλμός με αποτέλεσμα να παράγεται σημαντική ποσότητα θερμότητας. Διάφορες μελέτες εντόπισαν πως η διάχυση αυτής της θερμότητας είναι ικανή να προκαλέσει αλλοίωση στα αποθηκευμένα δεδομένα των γειτονικών κελιών μέσω του φαινομένου της θερμικής αλληλεπίδρασης (thermal crosstalk). Σε αυτήν την εργασία παρουσιάζουμε αλγορίθμους βασισμένους στο μοντέλο σφαλμάτων ευαισθησίας στο μοτίβο γειτονιάς (Neighborhood Pattern Sensitive Fault – NPSF), με σκοπό τη δοκιμή (testing) για την ανίχνευση των κελιών του επηρεάζονται από

το φαινόμενο της θερμικής αλληλεπίδρασης και την παροχή υψηλής αξιοπιστίας μνημών βασισμένες στις μνήμες αλλαγής φάσης.

EXTENDED ABSTRACT

Spyridon Spyridonos, M.Sc. in Data and Computer Systems Engineering, Department of Computer Science and Engineering, School of Engineering, University of Ioannina, Greece, March 2022

Algorithms for Thermal Crosstalk Fault Detection in PCM

Advisor: Yiorgos Tsiatouhas Professor

Phase Change Memory is a type of non-volatile memory that is very likely to replace the widely used flash memory. Current research on phase shift memory targets its integration into the manufacturing process of the CMOS technology that is currently in use as well as its reliability. The chalcogenide material that is used in a PCM cell can exist in two steady states with either a high or a low resistance value respectively. The high resistance state is known as the amorphous phase (or RESET state) while the low resistance state is known as the crystalline phase (or SET state). During the transition to the RESET phase, a strong pulse is applied resulting in the generation a significant amount of heat. Various studies have found that the diffusion of this heat is capable of causing damage to the stored data of neighboring cells through the phenomenon of thermal interaction (thermal crosstalk). In this work we present new test algorithms that are based on the Neighborhood Pattern Sensitive Fault (NPSF) model, for the detection of cells affected by the phenomenon of thermal interaction and consequently to provide phase change memories of high reliability.

ΚΕΦΑΛΑΙΟ 1

ΕΙΣΑΓΩΓΗ

1.1 Στόχοι

1.2 Δομή της Διατριβής

1.1 Στόχοι

Η παρούσα διατριβή επικεντρώνεται στο φαινόμενο της θερμικής αλληλεπίδρασης μεταξύ των γειτονικών κελιών στις μνήμες αλλαγής φάσης (Phase Change Memories - PCM), το οποίο είναι ένα από τα κύρια θέματα αξιοπιστίας στο συγκεκριμένο τύπο μνημών καθώς οι τεχνολογίες των ημιαγωγών κλιμακώνονται προς τα κάτω. Υπό τη θερμική αλληλεπίδραση, η αλλαγή της κατάστασης σε ένα κελί ή σε περισσότερα κελιά μιας γειτονιάς δύναται να επιφέρει μεταβολή στα δεδομένα ενός άλλου κελιού στη συγκεκριμένη γειτονιά. Για την αντιμετώπιση του φαινομένου, παρουσιάζονται νέοι αλγόριθμοι για τη δοκιμή (testing) και την ανίχνευση των κελιών που επηρεάζονται από τη θερμική αλληλεπίδραση στις μνήμες PCM.

1.2 Δομή της Διατριβής

Η διατριβή περιέχει πέντε κεφάλαια. Στο κεφάλαιο 2, γίνεται αναφορά στη δομή, τον τρόπο λειτουργίας και τις εφαρμογές των μνημών PCM. Στο κεφάλαιο 3, παρουσιάζονται τα μοντέλα σφαλμάτων στις μνήμες PCM, τίγονται θέματα αξιοπιστίας κατά τη λειτουργία της μνήμης και γίνεται εκτενής αναφορά στο φαινόμενο της θερμικής αλληλεπίδρασης. Ακολούθως, παρουσιάζονται, αλγόριθμοι στη διεθνή βιβλιογραφία οι οποίοι έχουν προταθεί για τη δοκιμή των μνημών PCM. Επίσης, πραγματοποιείται ανάλυση του γενικού μοντέλου σφαλμάτων ευαισθησίας στο μοτίβο γειτονιάς (Neighborhood Pattern Sensitive Fault model – NPSF) και περιγράφονται αλγόριθμοι για την ανίχνευση των σχετικών σφαλμάτων. Ακολούθως, στο κεφάλαιο 4, προτείνονται νέοι αλγόριθμοι για τη δοκιμή και ανίχνευση του φαινομένου της θερμικής αλληλεπίδρασης στις μνήμες PCM, κάτω από διάφορες εναλλακτικές συνθήκες ενεργοποίησης των κελιών της μνήμης. Τέλος, στο κεφάλαιο 5 καταγράφονται τα συμπεράσματα της διατριβής.

ΚΕΦΑΛΑΙΟ 2

ΠΑΡΟΥΣΙΑΣΗ ΤΩΝ ΜΝΗΜΩΝ PCM

2.1 Εφαρμογές και Χρήση των Μνημών PCM

2.2 Δομή Κελιού και Λειτουργία των Μνημών PCM

Καθώς ο ρυθμός κλιμάκωσης των μνημών Flash φθίνει, διερευνώνται εναλλακτικές ιδέες όσον αφορά τις μη-πτητικές μνήμες, προκειμένου να επιτευχθούν μεγαλύτερες δυνατότητες κλιμάκωσης και υψηλότερη αξιοπιστία. Ανάμεσα σε αυτές είναι οι Μνήμες Αλλαγής Φάσης (Phase Change Memories – PCM), όπου άρχισαν να προσελκύουν μεγάλο ενδιαφέρον. Αυτές οι μνήμες προτάθηκαν αρχικά από τον S. Ovshinsky, ο οποίος στα τέλη της δεκαετίας του 1960, παρατήρησε την αναστρέψιμη εναλλαγή της αγωγιμής κατάστασης σε υλικά χαλκογονιδίου [1].

Τα χαλκογονίδια είναι ημιαγώγιμα γυαλιά φτιαγμένα από στοιχεία της ομάδας 16 του περιοδικού πίνακα, όπως το θείο, το σελήνιο και το τελλούριο. Το As-Se (αρσενικό-σεληνίδιο) διερευνήθηκε προκειμένου να χρησιμοποιηθεί στην ξηρογραφία, ενώ η ικανότητα των γυαλιών Ge-Te (γερμάνιου-τελλούριου) να μετασχηματίζονται γρήγορα μεταξύ κρυσταλλικής-άμορφης φάσης χρησιμοποιήθηκε στα επανεγγράψιμα οπτικά μέσα (CD, DVD). Στην τελευταία εφαρμογή, η επιλεκτική κρυστάλλωση/αμορφοποίηση προκαλείται από μία δέσμη λέιζερ. Οι δυαδικές πληροφορίες διαβάζονται εκμεταλλευόμενοι την αλλαγή στην οπτική ανακλαστικότητα μεταξύ της άμορφης και της κρυσταλλικής κατάστασης. Τα υλικά

χαλκογονιδίου παρουσιάζουν επίσης αρνητική αντίσταση και δισταθερή συμπεριφορά. Μπορούν να αλλάξουν από κατάσταση υψηλής σε χαμηλή αντίσταση και επομένως μπορούν να χρησιμοποιηθούν ως μνήμη στερεάς κατάστασης [2].

Η αρχή της RAM μνήμης χαλκογονιδίου προτάθηκε πρώτα από τον Dewald το 1962 [2]. Ωστόσο, στις αρχές της δεκαετίας του 2000 οι βιομηχανίες ημιαγωγών εξέτασαν για πρώτη φορά την εκμετάλλευση της ίδιας ιδέας για μνήμες στερεάς κατάστασης (solid-state) μεγάλου μεγέθους [2-7]. Σήμερα οι PCM θεωρούνται υποψήφια για να γίνουν η κυρίαρχη μη-πτητική τεχνολογία λόγω της μεγάλης αντοχής [8-10], του γρήγορου προγραμματισμού και των χρόνων προσπέλασης και της ευρείας επεκτασιμότητας [2], [11].

2.1 Εφαρμογές και Χρήση των Μνημών PCM

Ένα ελκυστικό χαρακτηριστικό των PCM είναι ότι τα αποθηκευμένα δεδομένα διατηρούνται για πολύ μεγάλο χρονικό διάστημα (περίπου 10 χρόνια σε θερμοκρασία δωματίου), παρ'ότι γράφονται σε μόνο μερικά νανοδευτερόλεπτα. Αυτή η ιδιότητα επιτρέπει στις PCM να χρησιμοποιηθούν ως μη-πτητικό μέσο αποθήκευσης όπως οι μνήμες Flash και οι σκληροί δίσκοι, ενώ λειτουργούν σχεδόν τόσο γρήγορα όσο οι υψηλής απόδοσης πτητικές μνήμες όπως η DRAM [12].

Μία ακόμη ενδιαφέρουσα εφαρμογή για τις PCM είναι ο non-von Neumann υπολογισμός. Σε αυτό το υπολογιστικό μοντέλο, οι συσκευές μνήμης δεν χρησιμοποιούνται μόνο για την αποθήκευση δεδομένων αλλά επίσης και για την εκτέλεση ορισμένων υπολογιστικών εργασιών. Έχοντας μια συσκευή μνήμης που μπορεί να υπολογίζει, εξαλείφεται η ανάγκη μεταφοράς δεδομένων από και προς των μονάδων υπολογισμού (CPU) και των μονάδων μνήμης (DRAM) που είναι φυσικά διαχωρισμένες σε συμβατικούς υπολογιστές. Αυτός ο φυσικός διαχωρισμός και οι συσχετιζόμενες μεταφορές δεδομένων είναι αναμφισβήτητα ένα από τα κύρια κωλύματα των παραδοσιακών υπολογιστών von Neumann, καθώς η πρόσβαση στη μνήμη συνήθως καταναλώνει 100 έως 1000 φορές περισσότερη ενέργεια από μια λειτουργία της CPU [12], [13].

Ως τεχνολογία μνήμης, οι δυνατότητες των PCM έχουν παρουσιαστεί σε αρκετές εργασίες τα τελευταία 10 χρόνια και μερικές από τις κύριες εναπομείναντες

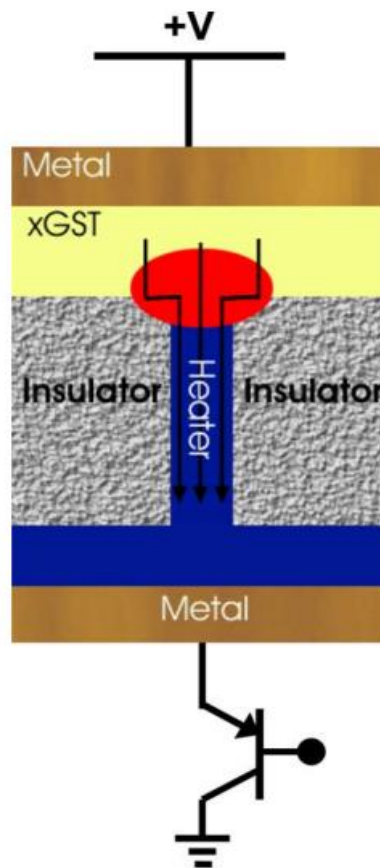
προκλήσεις είναι αυτές που σχετίζονται αναμφισβήτητα με το κόστος, την κατασκευή σε επίπεδο προϊόντος (product-level fabrication) και την ενσωμάτωση υψηλού επιπέδου (high-level integration) σε ένα υπολογιστικό σύστημα [12], [14]. Η επιτυχημένη κυκλοφορία του Intel Optane το 2018, μιας μη πτητικής μνήμης που βασίζεται στις PCM, που μπορεί να χρησιμοποιηθεί για τη βελτίωση του υπάρχοντος συστήματος αποθήκευσης μνήμης, καταδεικνύει τη βιωσιμότητα των PCM για χρήση ως ψηφιακή μνήμη σε ένα τυπικό σύστημα υπολογιστών. Εξαιτίας αυτού, η λεπτομερής κατανόηση των υποκείμενων φυσικών μηχανισμών και της δυναμικής κατάστασης του PCM είναι σημαντική για να ανακαλύψουμε πώς μπορεί να βελτιστοποιηθεί περαιτέρω η τεχνολογία. Μια τέτοια κατανόηση θα ήταν επίσης χρήσιμη για να μάθουμε πώς οι ιδιότητες PCM μπορούν να χρησιμοποιηθούν καλύτερα για αναδυόμενες εφαρμογές υπολογιστών του μοντέλου non-von Neumann. Παρά το γεγονός ότι το φαινόμενο της μνήμης σε υλικά αλλαγής φάσης (phase-change materials) ανακαλύφθηκε πριν από περισσότερα από 50 χρόνια, υπάρχουν πολλά ανοιχτά ερωτήματα σχετικά με την ηλεκτρική μεταφορά, τον μηχανισμό κρυστάλλωσης, τις επιδράσεις της χαλάρωσης και την εγγενή στοχαστικότητα στις PCM, τα οποία είναι όλα σημαντικά στην αρχή λειτουργίας του [12].

2.2 Δομή Κελιού και Λειτουργία των Μνημών PCM

2.2.1 Δομή Κελιού

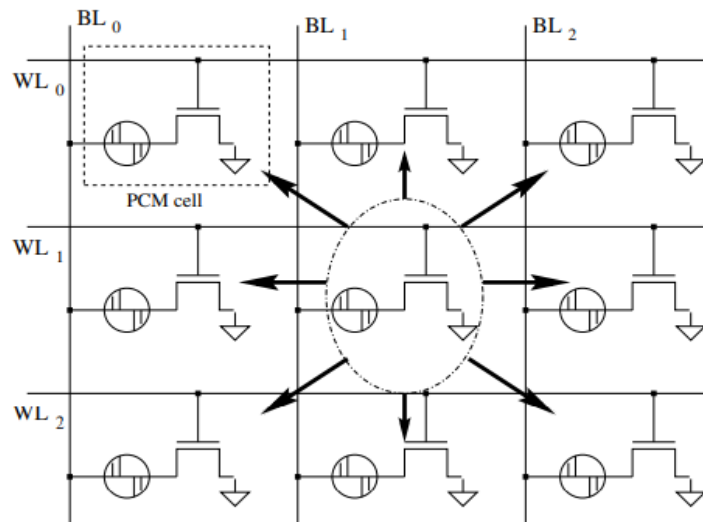
Το κελί μνήμης PCM (Σχήμα 2.1 [2]) αποτελείται από ένα τρανζίστορ και μία αντίσταση. Η αντίσταση, που είναι ένα στρώμα χαλκογονιδίου ($\text{Ge}_2\text{Sb}_2\text{Te}_5$ ή GST), βρίσκεται ανάμεσα από μία μεταλλική επαφή στο πάνω μέρος και ένα ηλεκτρόδιο με αντίσταση στο κάτω μέρος. Ανάλογα με το αν το υλικό χαλκογονιδίου είναι σε κρυσταλλική μορφή (κατάσταση SET) ή σε άμορφη μορφή (κατάσταση RESET), η αντίσταση του κελιού αλλάζει κατά τάξεις μεγέθους [2]. Οι καταστάσεις SET και RESET αναφέρονται σε καταστάσεις χαμηλής και υψηλής αντίστασης αντίστοιχα. Όπως κατασκευάζεται, το υλικό αλλαγής κατάστασης είναι στην κρυσταλλική, χαμηλής αντίστασης κατάσταση διότι η θερμοκρασία επεξεργασίας των μεταλλικών στρωμάτων διασύνδεσης (BEOL) είναι επαρκής για να κρυσταλλώσει το υλικό

αλλαγής φάσης [15]. Η αντίσταση λειτουργεί σαν επιλογέας, οδηγώντας το ρεύμα στις φάσεις ανάγνωσης και εγγραφής. Έχουν παρουσιαστεί λύσεις που χρησιμοποιούν διόδους, διπολικά τρανζίστορ και MOSFETs ως επιλογείς [4-7].

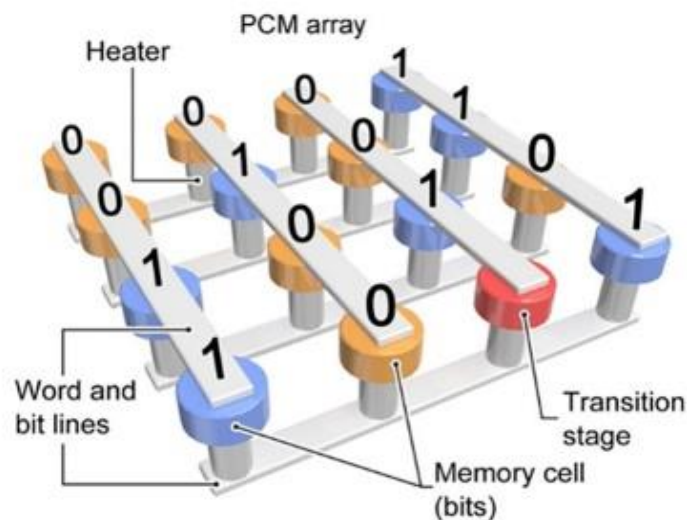


Σχήμα 2.1: Σχηματική Διατομή ενός Κελιού PCM.
Η Ενεργή Περιοχή Γειτνιάζει με τη Διεπαφή GST-heater.

Επίσης, στο Σχήμα 2.2 [20] παρουσιάζεται μία συστοιχία μνήμης PCM (ένα παράδειγμα συστοιχίας 3×3). Στο Σχήμα 2.3 [25] παρουσιάζεται παρόμοια μία συστοιχία μνήμης PCM σε 3-Δ μορφή. Ο πυρήνας ενός κελιού PCM φαίνεται σε ένα διακεκομμένο πλαίσιο στο Σχήμα 2.2 και έχει μοντελοποιηθεί χρησιμοποιώντας ένα συμπαγές μοντέλο MOSFET με μεταβλητή αντίσταση συνδεδεμένη στον ακροδέκτη της υποδοχής (drain terminal). Η υποδοχή του κελιού συνδέεται στη γραμμή bit (bitline-BL), ο ακροδέκτης πύλης (gate terminal) συνδέεται με τη γραμμή λέξης (wordline-WL) και ο ακροδέκτης πηγής (source terminal) είναι γειωμένος [20].



Σχήμα 2.2: Συστοιχία Μνήμης PCM.



Σχήμα 2.3: Συστοιχία Μνήμης PCM σε 3-Δ Μορφή.

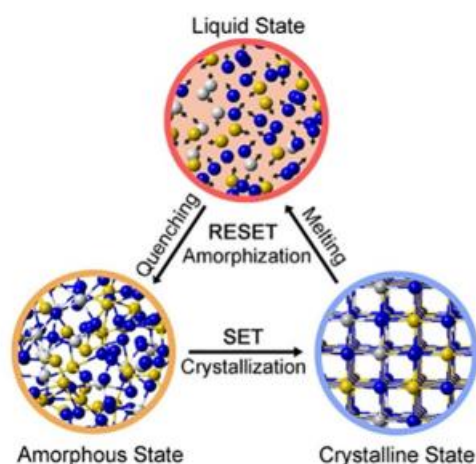
2.2.2 Λειτουργία των Μνημών PCM

Ο προγραμματισμός της μνήμης επιτυγχάνεται με την οδήγηση του κελιού με ένα παλμό ρεύματος για χρόνο 50–100 ns, προκαλώντας έτσι τη αλλαγή φάσης του στρώματος χαλκογονιδίου κοντά στην διεπαφή με το κάτω ηλεκτρόδιο. Η δομή έκκλησης θερμότητας (heater) κρατά το θερμαινόμενο σημείο μακριά από την επαφή του κρύου μετάλλου, ενώ το σχήμα του έχει σχεδιαστεί για να περιορίζει τη ροή του ρεύματος και την παραγωγή θερμότητας. Η άμορφη κατάσταση (RESET) επιτυγχάνεται οδηγώντας τη συσκευή με παλμό ρεύματος που διαρκεί λιγότερο

από 100 ns. Συγκεκριμένα, εφαρμόζεται υψηλή τάση στη BL του κελιού ενδιαφέροντος ενεργοποιώντας το τρανζίστορ διέλευσης (pass transistor) εφαρμόζοντας την κατάλληλη τάση στην πύλη. Για τα υπόλοιπα κελιά, οι BL και οι WL τους είναι γειωμένες [24]. Η θερμοκρασία του GST, κοντά στη δομή έκκλισης θερμότητας, ανεβαίνει πάνω από τη θερμοκρασία τήξης ($T_{\text{melt}} = 620 \text{ }^\circ\text{C}$) και έπειτα η συσκευή ψύχεται γρήγορα στο τέλος του τρέχοντος παλμού. Δεδομένου ότι το λιωμένο υλικό δεν έχει χρόνο να αναδιατάξει τους δεσμούς, αφήνεται στην άμορφη φάση. Η αντίσταση ενός κελιού με μια άμορφη περιοχή που καλύπτει το επάνω άκρο της δομής έκκλισης θερμότητας μπορεί να είναι της τάξης των MΩ [2].

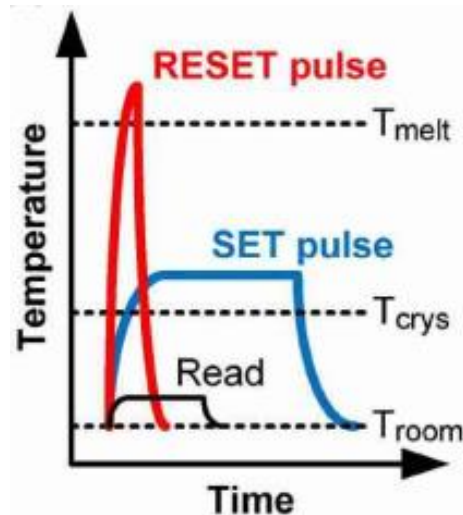
Για την αντίθετη μετάβαση σε κατάσταση SET, το κελί οδηγείται από έναν παρόμοιο παλμό αλλά με χαμηλότερη τιμή. Η λειτουργία πραγματοποιείται εφαρμόζοντας υψηλή τάση στη BL του κελιού ενδιαφέροντος ενώ ενεργοποιείται το τρανζίστορ διέλευσης (πολώνοντας τη WL). Όσον αφορά τα υπόλοιπα κελιά, οι BL και οι WL τους είναι γειωμένες προκειμένου να μην επιτραπεί ο προγραμματισμός τους [24]. Ο παλμός θερμαίνει το GST κοντά στο κάτω ηλεκτρόδιο περίπου στους 550°C (T_{cryst}). Αυτή η θερμοκρασία είναι μικρότερη από τη θερμοκρασία τήξης αλλά αρκετά υψηλή ώστε να πραγματοποιήσει την μετάβαση από την άμορφη στην κρυσταλλική κατάσταση σε περίπου 100ns. Μόλις ο παλμός ολοκληρωθεί, το GST έχει ανακτήσει την κρυσταλλική, δηλαδή την κατάσταση χαμηλής αντίστασης. Η αντίστοιχη αντίσταση του κελιού είναι της τάξης των kΩ. Αφού και οι δύο προγραμματιστικές μεταβάσεις χρησιμοποιούν παλμούς 50-100ns, αυτή η νέα μνήμη μπορεί να εγγραφεί γρήγορο προγραμματισμό, που είναι σημαντικό πλεονέκτημα απόδοσης σε σχέση με τις κυρίαρχες Flash συσκευές [2]. Στο Σχήμα 2.4 [25] παρουσιάζεται ένα διάγραμμα μεταβάσεων μεταξύ της άμορφης και κρυσταλλικής φάσης.

Η ανάγνωση (reading) επιτυγχάνεται πολώνοντας το κελί και μετρώντας το ρεύμα που το διατρέχει. Μερικές εκατοντάδες mV στην κατάσταση SET παράγουν 50-100μΑ. Αυτό το ρεύμα είναι ικανό να φορτίσει τη χωρητικότητα της BL μιας συστοιχίας μνήμης (memory array), καθιστώντας δυνατή την ανάγνωση ενός λογικού “1” σε 50ns. Η ίδια πόλωση του κελιού στην κατάσταση RESET δεν είναι ικανή να δημιουργήσει αρκετό ρεύμα για να πυροδοτήσει τον ενισχυτή ανίχνευσης (sensing amplifier), με αποτέλεσμα την ανάγνωση ενός λογικού “0” [2].



Σχήμα 2.4: Συνθήκες Μετάβασης μεταξύ Κρυσταλλικής και Άμορφης Φάσης σε ένα Κελί PCM.

Κατά τη λειτουργία ανάγνωσης, εφαρμόζεται κατάλληλη τάση στην BL του κελιού ενδιαφέροντος ενώ ενεργοποιείται το τρανζίστορ διέλευσης πολώνοντας τον ακροδέκτη της πύλης με την κατάλληλη τάση. Η χαμηλή τάση στην BL απαιτείται προκειμένου να δημιουργηθεί το κατάλληλο ρεύμα ανάγνωσης και να αποφευχθεί η ακούσια τροποποίηση των αποθηκευμένων δεδομένων στο συγκεκριμένο κελί. Για παράδειγμα, για να διαβαστεί ένα κελί, η BL πολώνεται με τάση ανάγνωσης (0.4V) και το κελί επιλέγεται εφαρμόζοντας την απαιτούμενη τάση (1.2V) στην WL του [24]. Στο Σχήμα 2.5 [15] που παρουσιάζεται παρακάτω, φαίνονται οι παλμοί SET, REST, καθώς και Read.

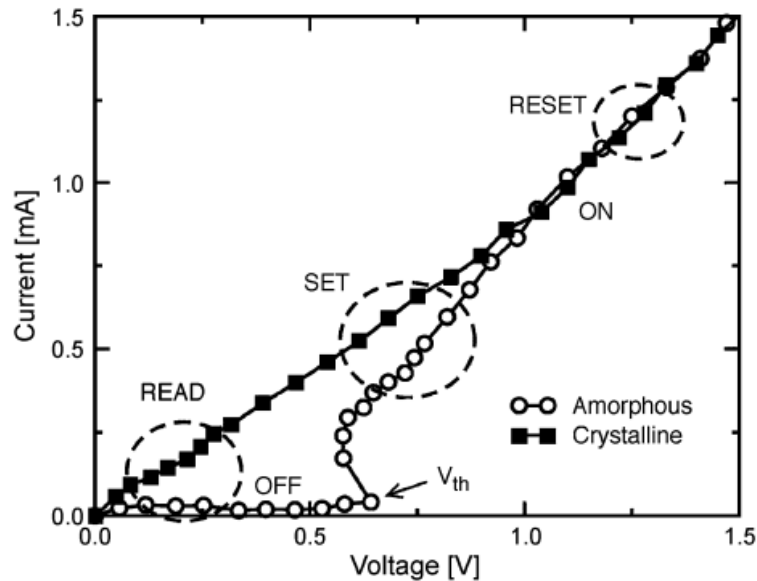


Σχήμα 2.5: Παλμοί Προγραμματισμού για τα Κελιά PCM.

Προκειμένου να γίνει η μετάβαση σε ένα κελί, από την άμορφη στην κρυσταλλική μορφή (RESET- SET μετάβαση), η συσκευή πρέπει να οδηγηθεί από περίπου 0.5 mA. Αφού στην άμορφη κατάσταση το κελί έχει περίπου 1 MΩ αντίσταση, θα απαιτούνταν τάση μεγαλύτερη από τα 500 V προκειμένου να πραγματοποιηθεί η συγκεκριμένη μετάβαση. Καμία πρακτική ηλεκτρική μνήμη δεν θα μπορούσε να λειτουργήσει με μία τέτοια απαίτηση τάσης. Η συμπεριφορά της τάσης (Σχήμα 2.6 [15]) ενός κελιού στην άμορφη (RESET) κατάσταση καθιστά δυνατή την υπέρβαση αυτού του περιορισμού. Το κελί στην κατάσταση υψηλής αντίστασης έχει χαμηλή αγωγιμότητα μόνο σε χαμηλή πόλωση. Καθώς η τάση αυξάνεται πάνω από την τάση κατωφλίου (V_{th}), η τάση επανέρχεται σε προηγούμενη τιμή (snap-back) και το κελί το οποίο βρίσκεται στην άμορφη κατάσταση γίνεται εξαιρετικά αγώγιμο. Μια τέτοια αλλαγή στο κατώφλι καθιστά δυνατή την ύπαρξη μεγάλων ρευμάτων μέσα στο κελί στην κατάσταση RESET ακόμη και αν η τάση είναι πάνω από το 1V [2].

Η φυσική αυτού του περιέργου φαινομένου είναι ένα θέμα που συζητείται εδώ και πολλά χρόνια. Από τότε που ο Ovshinsky ανέφερε την αλλαγή κατωφλίου [1] έχουν προταθεί διαφορετικά μοντέλα. Πολλοί ερευνητές υποστήριξαν την ιδέα ότι αυτή η αλλαγή είναι ουσιαστικά ένα θερμικό φαινόμενο και ότι το ρεύμα σε ένα άμορφο στρώμα αλλάζει λόγω της δημιουργίας ενός θερμού νήματος [17,18]. Αργότερα, ο Adler έδειξε ότι το φαινόμενο δεν είναι θερμικό [19], τουλάχιστον σε

λεπτές μεμβράνες χαλκογονιδίου, σε συμφωνία με την αρχική εικόνα του Ovshinsky.



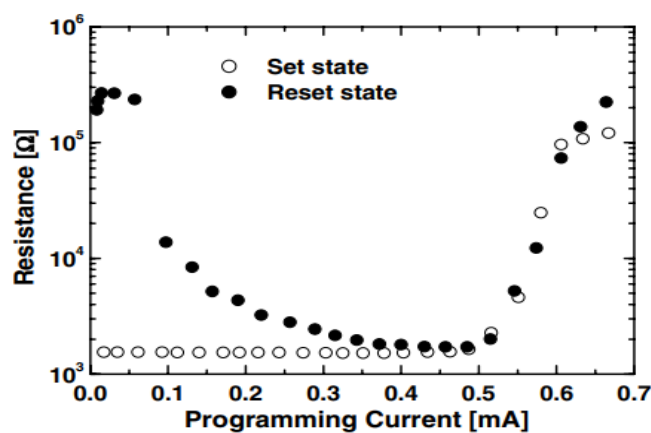
Σχήμα 2.6: Χαρακτηριστική I-V ενός Κελιού PCM για τις Καταστάσεις SET και RESET.

Η ηλεκτρονική μεταγωγή του GST και η φυσική που σχετίζεται με αυτή, έχουν μελετηθεί ποσοτικά χρησιμοποιώντας έναν προσομοιωτή σε επίπεδο υλικού και υπάρχει σύγκλιση μεταξύ των υπολογισμένων και των πειραματικών καμπυλών I-V [21]. Το αποτέλεσμα μπορεί να εξηγηθεί μελετώντας το Σχήμα 2.6. Σε καθεστώς χαμηλής πόλωσης (low bias), το άμορφο επίπεδο GST έχει υψηλή αντίσταση. Ωστόσο, καθώς η τάση αυξάνει, το ρεύμα αρχίζει να αυξάνεται εκθετικά λόγω του ιονισμού κρούσης και εξισορροπείται από τον ανασυνδυασμό μέσω των εντοπισμένων καταστάσεων (localized states). Η μικροσκοπική δομή των άμορφων γυαλιών χαλκογονιδίου χαρακτηρίζεται από διπλά άτομα τελλουρίου που συνδέονται με δεσμούς. Κατά μήκος αυτών των δεσμών υπάρχουν δομικά ελαττώματα, τα λεγόμενα ζεύγη εναλλαγής σθένους, τα οποία είναι υπεύθυνα για την υψηλή πυκνότητα (10^{17} – 10^{20} cm^{-3}) των εντοπισμένων καταστάσεων [22,23]. Καθώς όλα τα ελαττώματα είναι πλήρως κορεσμένα από τους ανασυνδυασμένους φορείς, η εκθετικά αυξανόμενη παραγωγή ρεύματος δεν μπορεί πλέον να εξισορροπηθεί με τον ανασυνδυασμό και ο μόνος τρόπος για να φτάσει το κυκλωματικό στοιχείο σε μια νέα σταθερή κατάσταση είναι να μειώσει γρήγορα

την τάση για να μηδενίσει το ρυθμό παραγωγής [21]. Γίνεται μια επαναφορά της τάσης σε προηγούμενη τιμή και μετά την μεταγωγή το GST είναι ακόμα άμορφο αλλά εξαιρετικά αγώγιμο [2].

Αντίθετα, η I-V καμπύλη του κρυσταλλικού GST (κατάσταση SET) δεν πραγματοποιεί αλλαγή κατωφλίου και συγκλίνει με τη I-V καμπύλη της κατάστασης RESET μετά την αλλαγή. Η κλίση αυτού του τμήματος της καμπύλης καθορίζεται κυρίως από την αντίσταση της δομής έκκλησης θερμότητας, ενώ η πτώση τάσης στο στρώμα GST πλησιάζει μια σταθερή τιμή, την τάση συγκράτησης (holding voltage), V_H , η οποία είναι της τάξης του υλικού της ζώνης απογύμνωσης (0,5 eV) [2].

Το Σχήμα 2.7 που παρουσιάζεται παρακάτω δείχνει τη χαρακτηριστική προγραμματισμού ενός PCM κελιού, που είναι η εξάρτηση της αντίστασης R του κελιού ως συνάρτηση του ρεύματος προγραμματισμού. Οι μαύρες κουκίδες αναφέρονται στην αντίσταση που επιτυγχάνεται οδηγώντας ένα κελί στην κατάσταση RESET. Στη διαδικασία μέτρησης εφαρμόζεται ένας παλμός προγραμματισμού 100 ns και η αντίσταση του κελιού μετά τον προγραμματισμό διαβάζεται στην τάση των 0.2 V. Πριν από την επόμενη μέτρηση, το κελί επαναφέρεται στην αρχική RESET κατάσταση χρησιμοποιώντας έναν κατάλληλο παλμό ρεύματος. Στη συνέχεια, ο κύκλος μέτρησης ξεκινά ξανά οδηγώντας το κελί με έναν νέο παλμό ρεύματος προγραμματισμού 100 ns με διαφορετικό πλάτος [2].



Σχήμα 2.7: Η Αντίσταση του Κελιού ως Συνάρτηση του Ρεύματος Προγραμματισμού (R-I Χαρακτηριστική).

Τρεις διαφορετικές περιοχές μπορούν να παρατηρηθούν: (i) για παλμούς προγραμματισμού κάτω των 50μΑ, η αγωγιμότητα της ON-κατάστασης δεν είναι ενεργοποιημένη και το πολύ μικρό ρεύμα δεν προκαλεί αλλαγή φάσης, (ii) στο εύρος 50-500μΑ, η αντίσταση μειώνεται μετά την κρυστάλλωση του άμορφου GST, (iii) πάνω από 500μΑ, ο παλμός προγραμματισμού λιώνει ένα μέρος του GST κοντά στην διεπαφή με το κάτω ηλεκτρόδιο, αφήνοντάς το σε άμορφη κατάσταση. Η αντίσταση bit που μετριέται μετά τον προγραμματισμό αυξάνεται. Οι λευκές κουκίδες στο Σχήμα 2.7 δείχνουν επίσης τα R-I χαρακτηριστικά για το ίδιο κελί, αλλά ξεκινώντας από την SET κατάσταση. Η τιμή της αντίστασης αλλάζει μόνο όταν το ρεύμα ξεπερνά τα 500μΑ, όπου η χαρακτηριστική επικαλύπτεται με την R-I της RESET κατάστασης. Συμπερασματικά, το κελί PCM μπορεί να μεταβληθεί χρησιμοποιώντας παλμούς των 500μΑ και 700μΑ, αντίστοιχα. Οι παλμοί είναι ανεξάρτητοι της αρχικής κατάστασης του κελιού (αντίσταση). Το κελί μπορεί ως εκ τούτου να ξαναγραφτεί χωρίς να χρειάζεται ενδιάμεση διαγραφή. Η διαφορά της τάξης μεγέθους στις SET και RESET καταστάσεις καθιστά την PCM μνήμη ιδανική για μια λειτουργία πολλαπλών bit (multi-bit operation): σε αυτό το σχήμα η αντίσταση του κελιού μπορεί να ρυθμιστεί μεταξύ των δύο ακραίων τιμών, τοποθετώντας έτσι περισσότερα από δύο επίπεδα ανά κελί. Αυτή η προσέγγιση μπορεί να γίνει μια βιώσιμη επιλογή για περαιτέρω μείωση του κόστους ανά bit των συσκευών PCM [2].

ΚΕΦΑΛΑΙΟ 3

ΜΟΝΤΕΛΑ ΣΦΑΛΜΑΤΩΝ ΚΑΙ ΔΟΚΙΜΗ ΤΩΝ ΜΝΗΜΩΝ PCM

3.1 Αξιοπιστία των Κελιών στις Μνήμες PCM

3.2 Μοντέλα Σφαλμάτων σε Μνήμες PCM

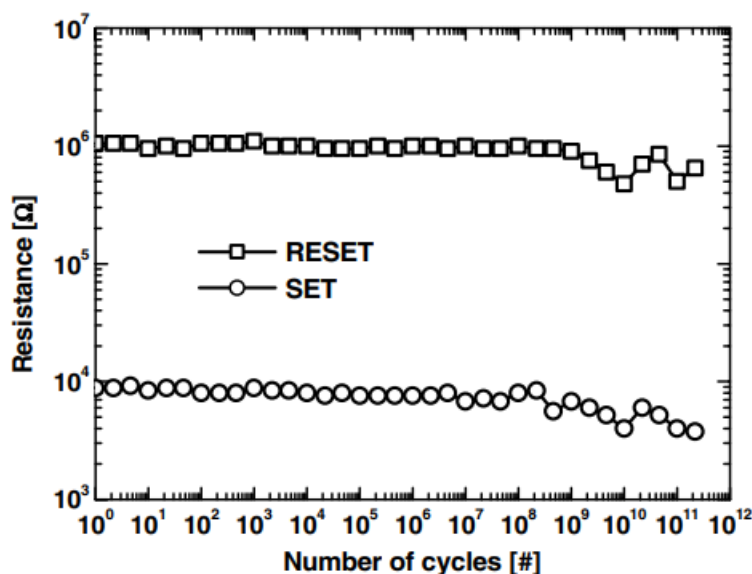
3.3 Αλγόριθμοι Δοκιμής σε Μνήμες PCM

Σε αυτό το κεφάλαιο θα αναφερθούμε στην αξιοπιστία των κελιών μιας μνήμης PCM. Θα αναλύσουμε τα μοντέλα σφαλμάτων που εφαρμόζονται στις μνήμες PCM αλλά θα επικεντρωθούμε περισσότερο στο φαινόμενο της θερμικής αλληλεπίδρασης (thermal crosstalk) μεταξύ γειτονικών κελιών. Επίσης, θα γίνει μία αναφορά στους αλγορίθμους που έχουν προταθεί για τη δοκιμή των μνημών PCM, καθώς και ποια μοντέλα σφαλμάτων καλύπτει ο κάθε αλγόριθμος.

3.1 Αξιοπιστία των Κελιών στις Μνήμες PCM

Ο κρυσταλλικός-άμορφος μετασχηματισμός στο GST είναι γρήγορος και ευσταθής. Το Σχήμα 3.1 [2] δείχνει την αντίσταση των SET και RESET καταστάσεων ως συνάρτηση των κύκλων προγραμματισμού. Κάθε κύκλος προγραμματισμού περιλάμβανε έναν RESET παλμό 40ns και έναν SET παλμό 100ns. Ένα παράθυρο αντίστασης ενός παράγοντα $10^2 \Omega$ διατηρείται για πάνω από 10^{11} κύκλους,

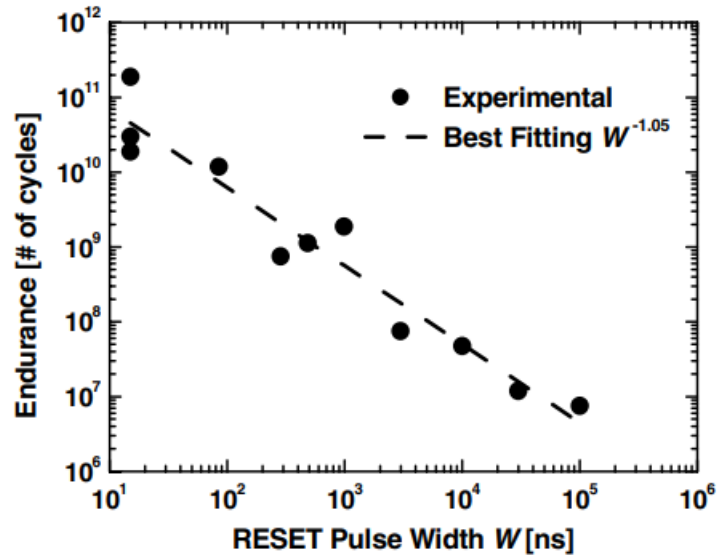
επιβεβαιώνοντας την εξαιρετική σταθερότητα του προγραμματισμού. Ο λόγος για μία τόσο αξιοσημείωτη αντοχή σχετίζεται αυστηρά με τη μικροσκοπική δομή του κράματος GST. Στην άμορφη κατάσταση, οι αλληλεπιδράσεις μεταξύ διαφορετικών δομικών στοιχείων του $\text{Ge}_2\text{Sb}_2\text{Te}_5$ εξασθενούν. Η δομή επομένως αφήνεται να χαλαρώσει, αλλά ούτε οι ισχυροί ομοιοπολικοί δεσμοί σπάνε, ούτε τα άτομα αλλάζουν δραστικά τη θέση τους στο πλέγμα. Το Te-υποπλέγμα διατηρείται εν μέρει καθώς και η διατήρηση της τοπικής δομής γύρω από τα άτομα Sb. Η αρχική κρυσταλλική δομή μπορεί επομένως να ανακτηθεί γρήγορα και αξιόπιστα [2].



Σχήμα 3.1: Αντίσταση Κελιού στις Καταστάσεις SET και RESET ως Συνάρτηση του Αριθμού των Κύκλων SET/RESET.

Μετά από 10^{11} - 10^{12} κύκλους, το κελί PCM αποτυγχάνει είτε με stuck-set (αδυναμία να κάνει SET το κελί) είτε με stuck-reset (αδυναμία να κάνει RESET το κελί). Και οι δύο τρόποι αστοχίας εξαρτώνται ως επί το πλείστον από την ποιότητα του ηλεκτροδίου στο κάτω μέρος της διεπαφής του κελιού, από την προσκόλλησή του με το στρώμα χαλκογονιδίου και από διεργασίες ενδοδιάχυσης που μπορεί να συμβούν μεταξύ του κράματος χαλκογονιδίου και των παρακείμενων υλικών. Η αστοχία αναμένεται να εξαρτηθεί από την ενέργεια που απελευθερώνεται κατά την μετατροπή ανά μονάδα ενεργού όγκου. Το Σχήμα 3.2 [2] επιβεβαιώνει ότι η αντοχή σε σταθερό ρεύμα προγραμματισμού κλιμακώνεται αντίστροφα προς το πλάτος του παλμού RESET. Όσο μεγαλύτερος είναι ο παλμός, τόσο μεγαλύτερη

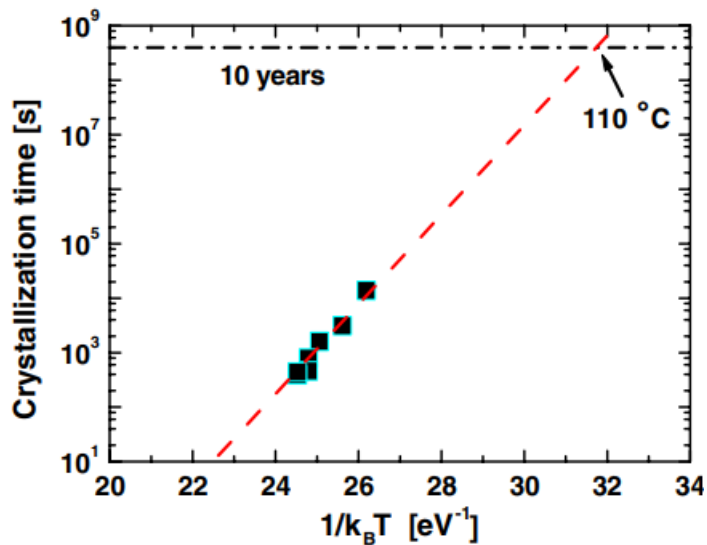
είναι η ενέργεια που απελευθερώνεται ανά παλμό, τόσο ταχύτερη είναι η υποβάθμιση της διεπαφής, ενώ η συνολική ενέργεια που απελευθερώνεται μέχρι την αστοχία του bit παραμένει σχεδόν σταθερή [2].



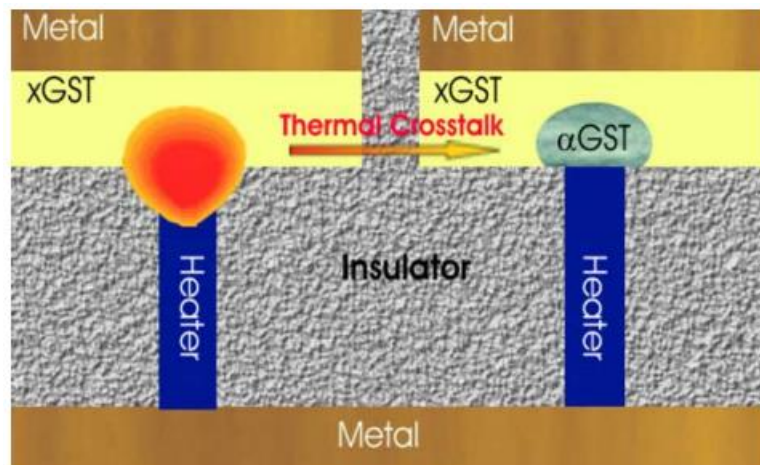
Σχήμα 3.2: Αντοχή Κελιού ως Συνάρτηση του Πλάτους του Παλμού RESET.

Ένα άλλο κρίσιμο ζήτημα για ένα κελί μη πτητικής μνήμης είναι η διατήρηση δεδομένων (data retention), δηλαδή η δυνατότητα διατήρησης των αποθηκευμένων πληροφοριών κατά τη διάρκεια ζωής της μνήμης. Σε αυτό το σημείο η κατάσταση RESET της PCM είναι ιδιαίτερα ευαίσθητη. Το άμορφο GST μπορεί να εξελιχθεί αυθόρμητα προς την πιο σταθερή κρυσταλλική φάση. Το Σχήμα 3.3 [2] δείχνει την γραφική παράσταση Arrhenius για το χρόνο αστοχίας, που ορίζεται ως ο χρόνος που απαιτείται για την αντίσταση να μειωθεί κατά το ήμισυ της τιμής RESET της. Η πειραματική ενέργεια ενεργοποίησης των 2.6 eV αντιστοιχεί σε 10 χρόνια ζωής στη μέγιστη θερμοκρασία των 110 °C [2].

Μία περαιτέρω πιθανή ανησυχία για την αξιοπιστία είναι η διαταραχή του προγραμματισμού (program disturb). Όταν προγραμματίζεται ένα κελί συστοιχίας, τα γειτονικά κελιά μπορεί να υποστούν παρασιτική θέρμανση, η οποία μπορεί να οδηγήσει σε απώλεια δεδομένων (Σχήμα 3.4 [2]). Το Σχήμα 3.5 [2] δείχνει τα αποτελέσματα που προέκυψαν από μία multi-Mb συστοιχία κατασκευασμένη στην τεχνολογία των 0.18μm.



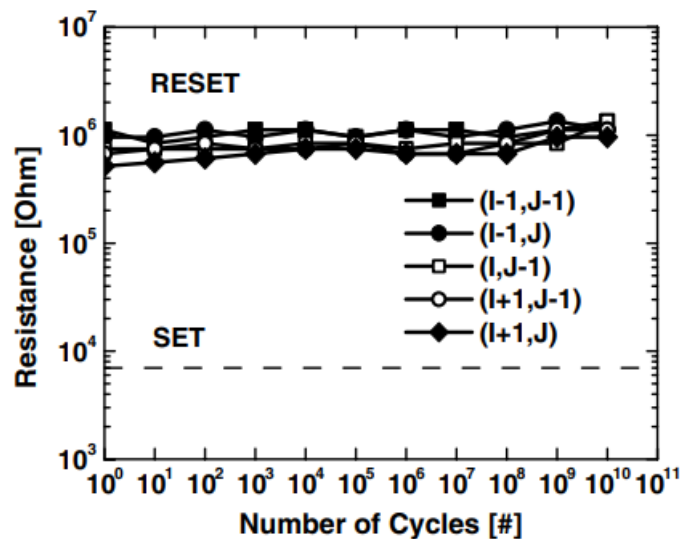
Σχήμα 3.3: Πειραματική Γραφική Παράσταση Arrhenius για το Χρόνο Κρυστάλλωσης της Κατάστασης RESET. Η Διατήρηση Δέκα Χρόνων Απαιτεί Λειτουργία Κάτω από τους 110 °C.



Σχήμα 3.4: Σχηματική Περιγραφή της Διαταραχής Προγράμματος σε Μία Συστοιχία PCM .Η Θερμοκρασία Αυξάνεται Εξαιτίας του Παλμού Προγραμματισμού και Μπορεί να Προκαλέσει ένα Γειτονικό Άμορφο Κελί να Κρυσταλλωθεί.

Η αντίσταση κάποιων κελιών στην άμορφη κατάσταση καταγράφονται ενώ ένα γειτονικό κελί υφίσταται επαναλαμβανόμενο προγραμματισμό (800 μ A, 50 ns παλμοί). Η κατάσταση RESET παραμένει ευσταθής και δεν παρατηρήθηκε απώλεια δεδομένων μέχρι τους 10¹⁰ κύκλους προγραμματισμού. Για να αντιμετωπιστεί ο αντίκτυπος της πιθανής αλληλεπίδρασης σε περαιτέρω

κλιμάκωση της τεχνολογίας, έχουν πραγματοποιηθεί αριθμητικές προσομοιώσεις [26]. Τα αποτελέσματα δείχνουν ότι η διαταραχή του προγραμματισμού δεν θα πρέπει να αποτελεί πρόβλημα τουλάχιστον στον κόμβο των 45 nm. Για κελιά μνήμης σε μικρότερες τεχνολογίες, η αλληλεπίδραση μεταξύ γειτονικών bits μπορεί να επιβάλλει μια απόσταση κελιών ελαφρώς μεγαλύτερη από το επιτρεπτό [2]. Η θερμική αλληλεπίδραση (thermal crosstalk) είναι το κύριο πρόβλημα αξιοπιστίας που μελετάμε σε αυτή την διατριβή και μία εκτενέστερη αναφορά θα γίνει παρακάτω σε αυτό το κεφάλαιο.



Σχήμα 3.5: Η Αντίσταση των Κελιών στην Κατάσταση RESET Μετρήθηκε ενώ ένα Γειτονικό Κελί Γράφεται Επαναλαμβανόμενα.

3.2 Μοντέλα Σφαλμάτων σε Μνήμες PCM

3.2.1 Γενική Περιγραφή των Σφαλμάτων σε Μνήμες PCM

Οι μνήμες PCM είναι επιρρεπείς σε σφάλματα λόγω του ειδικού υλικού που τις αποτελούν, της δομής τους και του μηχανισμού λειτουργίας τους. Η παρουσία μολυσματικών ουσιών ή άλλων ακαθαρσιών στην ενεργό περιοχή του υλικού αλλαγής φάσης μπορεί να οδηγήσει σε παρασιτικά παράλληλα αγώγιμα μονοπάτια στο κελί PCM. Αυτά τα ωμικά ή χωρητικά παρασιτικά φαινόμενα οδηγούν σε ατελές σφάλμα προγραμματισμού 0 (incomplete program fault 0-IPF0), σφάλμα

ασθενούς μετάβασης 0 (weak transition fault 0-WTF0), καταστροφικό σφάλμα εγγραφής 1 (write 1 destructive fault-WDF1), καταστροφικό σφάλμα εγγραφής 0 (write 0 destructive fault-WDF0) ή ασθενές σφάλμα εγγραφής 1 (weak write 1 destructive fault-WWDF1), ανάλογα με την τιμή της παρασιτικής αντίστασης και χωρητικότητας [24,27,28]. Ένα σφάλμα διαταραχής ανάκτησης ανάγνωσης (read recovery disturb fault-RRD), το οποίο προκαλείται από την κατάσταση μη ισορροπίας στο πρόσφατα προγραμματισμένο κελί, μπορεί να προκύψει όταν ένα κελί διαβάζεται αμέσως μετά τον προγραμματισμό του στην κατάσταση RESET [29]. Το κανονικό ρεύμα ανάγνωσης είναι μικρό και δεν μπορεί να δημιουργήσει θερμοκρασία πάνω από την κρυσταλλική θερμοκρασία του υλικού αλλαγής φάσης. Ωστόσο, το ρεύμα που διαρρέει την περιοχή υλικού αλλαγής φάσης μπορεί στην πραγματικότητα να αυξηθεί εάν υπάρχουν ελαττώματα, γεγονός που θα οδηγήσει σε σφάλμα ανάγνωσης (read fault-RD) που καταστρέφει τις πληροφορίες που είναι αποθηκευμένες στο κελί PCM [30,31]. Εφόσον η κατάσταση SET απαιτεί χαμηλότερη θερμοκρασία από την κατάσταση RESET, μόνο η κρυσταλλική κατάσταση λαμβάνεται υπόψη όταν συζητείται ένα σφάλμα RD. Ενδέχεται να προκύψει σφάλμα ελλειπτικού εγγραφής (fault write fault-FWR), το οποίο αλλάζει το αναφερόμενο κελί από την κατάσταση SET στην κατάσταση RESET όταν εκτελείται η λειτουργία ανάγνωσης εάν το κύκλωμα ρεύματος εγγραφής του κελιού PCM ενεργοποιηθεί εσφαλμένα, ειδικά σε περιβάλλον ακτινοβολίας [32]. Ένα σφάλμα μόνιμης τιμής SET (stuck-at SET-SS) προκαλείται από τη διάχυση μεταξύ του υλικού αλλαγής φάσης της PCM και του παρακείμενου υλικού του ή από σφάλματα κυκλώματος, όπως ένα βραχυκύκλωμα μεταξύ της BL και της γείωσης (GND) [20,31]. Παρουσιάζεται σφάλμα μόνιμης τιμής RESET (stuck-at RESET-SR) εάν η περιοχή επαφής μεταξύ του επάνω ηλεκτροδίου και του υλικού αλλαγής φάσης είναι φθαρμένη [2,15] ή εάν υπάρχουν λάθη κυκλώματος (circuit errors) στη διάταξη PCM, όπως γεφύρωση μεταξύ της WL και γείωσης [20]. Εφόσον το κελί PCM αλλάζει την κατάστασή του σύμφωνα με τη θερμότητα Joule που παράγεται από τον παλμό ρεύματος εισόδου, η θερμική αλληλεπίδραση μεταξύ των γειτονικών κελιών μπορεί να οδηγήσει σε σφάλμα διαταραχής εγγύτητας (proximity disturb fault-PDF). Μια περιεκτική περίληψη των μοντέλων σφαλμάτων PCM μπορεί να βρεθεί στις εργασίες [20,24].

Τα μοντέλα σφαλμάτων PCM και οι συσχετιζόμενες συνθήκες δοκιμής συνοψίζονται στον Πίνακα 3.1 [33]. Στον Πίνακα 3.1, τα σφάλματα μοντελοποιούνται με την παραδοσιακή σημειογραφία (S/F/R) [34], όπου το $SE\{w0, w1, r0, r1, 1, 0\}$ είναι η συνθήκη ευαισθητοποίησης σφάλματος, το $FE\{0, 0_m, 1_m, 1\}$ είναι η τιμή του ελαττωματικού κελιού, το $RE\{0, 1, -\}$ είναι η έξοδος της λειτουργίας ανάγνωσης, το 0_m και το 1_m είναι οι οριακές τιμές του κελιού PCM που προκύπτουν από ειδικά σφάλματα, και το '-' χρησιμοποιείται για να δηλώσει αν εκτελείται μια λειτουργία εγγραφής. Για το μοντέλο σφάλματος και την συνθήκη ελέγχου του PDF, οι λειτουργίες πριν και μετά το ';' εκτελούνται στα κελιά θύτες (aggressor cells) και στο κελί θύμα, αντίστοιχα [33].

Πίνακας 3.1: Μοντέλα Σφαλμάτων και Συνθήκες Δοκιμής των Μνημών PCM

Όνομα αστοχίας	Μοντέλο	
	σφάλματος	Συνθήκη δοκιμής
Stuck at SET fault (SS)	$\langle \forall/1 \rangle$	w0/r0
Stuck at RESET fault (SR)	$\langle \forall/0 \rangle$	w1/r1
Incomplete program fault 0 (IPF0)	$\langle x, w0/1/- \rangle$	w0/r0, r0
Weak transition fault 0 (WTF0)	$\langle x, w0/1_m/- \rangle$	w0/r0
Write 1 destructive fault (WDF1)	$\langle x, w1/0/- \rangle$	w1/r1
Write 1 destructive fault (WDF0)	$\langle x, w0/1/- \rangle$	w0/r0
Weak write 1 destructive fault (WWDF1)	$\langle x, w1/1_m/- \rangle$	w1/r1_m
Proximity disturb fault (PDF)	$\langle x, w0; 0/1_m/- \rangle$	x, w0_neighbors; r0
Read recovery disturb fault (RRD)	$\langle 1, w0, r0/0/1_m \rangle$	x, w0, r0(following w0)
Read fault (RD)	$\langle 0, r0/1_m/0 \rangle$	0, r0, r0
False write fault (FWR)	$\langle 1, r1/0/1 \rangle$	1, r1, r1

3.2.2 Θερμική Αλληλεπίδραση (Thermal Crosstalk)

Λαμβάνοντας υπόψη τη τη μεγάλη αύξηση της θερμοκρασίας που απαιτείται για τον προγραμματισμό επαναφοράς (reset programming) της PCM, τα γειτονικά κελιά διαταράσσονται θερμικά κατά τον προγραμματισμό RESET του επιλεγμένου κελιού. Η θερμική αλληλεπίδραση (thermal crosstalk, proximity disturb) μπορεί να οδηγήσει σε μερική κρυστάλλωση του κελιού που είναι σε κατάσταση RESET, προκαλώντας αστοχία συγκράτησης (retention failure). Ταυτόχρονα, η συμπεριφορά μετατόπισης επηρεάζεται επίσης από τη θερμική αλληλεπίδραση επειδή η μετατόπιση εξαρτάται από τη θερμοκρασία [35]. Η συμπεριφορά μετατόπισης που επιταχύνεται από τη θερμική αλληλεπίδραση οδηγεί σε ευρύτερες κατανομές αντίστασης όταν τα κελιά διαταράσσονται θερμικά στο αρχικό στάδιο μετά τον RESET προγραμματισμό [36]. Ως εκ τούτου, η θερμική αλληλεπίδραση είναι το κύριο πρόβλημα για την κλιμάκωση της PCM επειδή η μικρότερη απόσταση μεταξύ των κελιών μπορεί δυνητικά να οδηγήσει σε μεγέθυνση της θερμικής αλληλεπίδρασης.

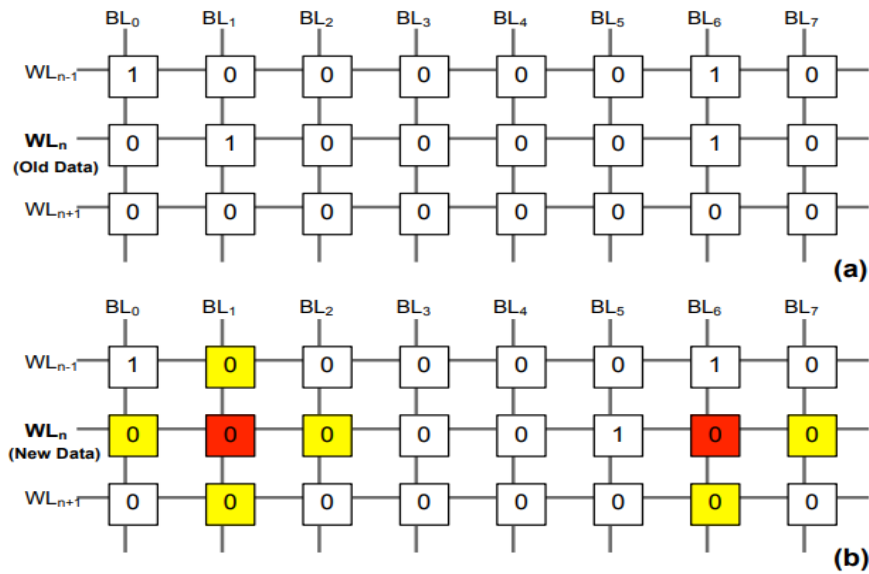
Το μέγεθος και η διάρκεια της θερμικής αλληλεπίδρασης καθώς και οι ιδιότητες κλιμάκωσής τους έχουν μοντελοποιηθεί και προσομοιωθεί [11], [37], [38]. Τα αποτελέσματά τους δείχνουν ότι η ισότροπη κλιμάκωση, όπου οι διαστάσεις των κελιών κλιμακώνονται κατά τον ίδιο παράγοντα κλιμάκωσης, προκαλεί θερμική αλληλεπίδραση σε αναλογική κλίμακα σύμφωνα με τον ίδιο παράγοντα κλιμάκωσης. Επομένως, η ισότροπη κλιμάκωση δεν κάνει τη θερμική αλληλεπίδραση ούτε καλύτερη ούτε χειρότερη από τον τρέχοντα κόμβο. Ωστόσο, όταν οι αποστάσεις των κελιών κλιμακώνονται σε διαδοχικές γενιές τεχνολογίας για τη βελτίωση της πυκνότητας και της απόδοσης, η θερμική αλληλεπίδραση μπορεί να επιδεινωθεί. Επομένως, οποιαδήποτε κίνηση κλιμάκωσης θα πρέπει να περιλαμβάνει την προσεκτική αξιολόγηση των θερμικών διαταραχών [15].

Η κλιμάκωση πέρα από τα 22nm αυξάνει σημαντικά την εγγύτητα μεταξύ των κελιών, καθώς και μειώνει το μέγεθος των κελιών, η θερμότητα που δημιουργείται από τον RESET προγραμματισμό, με ολοένα και μεγαλύτερη πιθανότητα, μπορεί πραγματικά να επηρεάσει την κατάσταση των γειτονικών κελιών [40]. Όταν γίνεται RESET ένα κελί, τα γειτονικά κελιά στην άμορφη κατάσταση μπορούν να

μετακινηθούν στην κρυσταλλική κατάσταση. Αυτό το φαινόμενο αναφέρεται ως επίδραση διαταραχής εγγραφής (write disturbance crosstalk) [39].

Το Σχήμα 3.6(a) [40] δείχνει την αρχιτεκτονική των κελιών PCM, τα οποία αναπτύσσονται στη διασταύρωση των WL και BL. Μετά την εγγραφή στη μνήμη, ορισμένα φυσικά κελιά στην ενεργή WL πρέπει να αλλάξουν την κατάστασή τους (SET και RESET), ενώ άλλα παραμένουν ανέγγιχτα (δηλαδή, σε αδράνεια). Δεδομένου ότι η διαδικασία RESET παράγει σημαντική θερμότητα, μπορεί να διαταράξει τα κελιά που βρίσκονται σε αδράνεια. Αυτά τα αδρανή κελιά που αλλάζουν ακούσια κατά τη διαδικασία εγγραφής είναι κελιά θύματα. Σε αυτήν την περίπτωση, η αντίσταση ενός κελιού θύματος μειώνεται και η λογική του τιμή αλλάζει από '0'→'1'. Η σύγκριση του σχήματος 3.6(a) και (b) δείχνει ότι όταν γίνεται RESET ένα κελί '1'→'0', μπορεί να διαταράξει τα κελιά εντός της ενεργής γραμμής λέξεων WL_n και επίσης σε γειτονικές γραμμές λέξεων $WL_{n\pm 1}$. Τα πιθανά κελιά θύματα που είναι ευάλωτα στη θερμική αλληλεπίδραση αναπαρίστανται με κίτρινο χρώμα στο σχήμα 3.6(b). Για παράδειγμα, θεωρήστε το κελί, $c_{n,1}$, στη διασταύρωση των WL_n και BL_1 . Η τιμή του $c_{n,1}$ ενημερώνεται από '1'→'0' κατά τη διάρκεια μιας νέας εγγραφής. Δεδομένου ότι όλα τα γειτονικά κελιά του $c_{n,1}$ βρίσκονται σε άμορφη κατάσταση, η θερμότητα που προκύπτει από τη διαδικασία RESET μπορεί να ενημερώσει ακούσια τα γειτονικά κελιά με πιθανότητα περίπου 10% (9,9% και 11% σε WL_n και $WL_{n\pm 1}$, αντίστοιχα [39]). Προκειμένου να εκλυθεί η απαραίτητη θερμότητα είναι κρίσιμο να γίνουν RESET διαδοχικά τουλάχιστον δύο γειτονικά κελιά, ενώ στην περίπτωση που γίνουν διαδοχικά και τα τέσσερα γειτονικά κελιά RESET εκλύεται η μέγιστη ποσότητα θερμότητας η οποία δύναται να επηρεάσει το κελί θύτη αλλοιώνοντας τα αποθηκευμένα δεδομένα σε αυτό.

Το Σχήμα 3.7(a) [39] δείχνει δύο κελιά κατά μήκος μιας BL. Ένα αίτημα εγγραφής μπορεί να επαναφέρει μόνο το αριστερό κελί και να αφήσει το δεξί κελί αμετάβλητο. Το τελευταίο αναφέρεται ως αδρανές κελί (idle cell). Λόγω της κρυφής ροής θερμότητας κατά μήκος της BL, εάν το δεξιό κελί βρίσκεται σε κατάσταση RESET, ένα μέρος του άμορφου όγκου της μπορεί να θρυμματιστεί σε μικρά κρυσταλλικά κομμάτια, γεγονός που μειώνει σημαντικά την αντίστασή της, χάνοντας την αποθηκευμένη πληροφορία bit '0' [39,42,43].

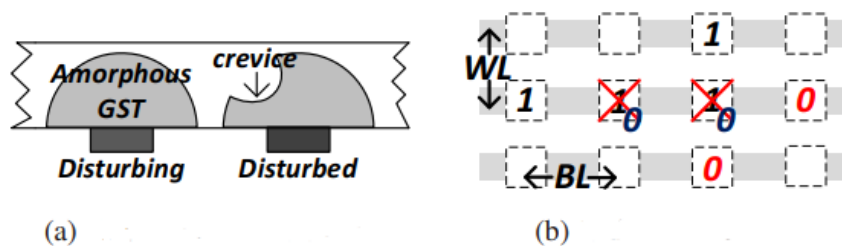


Σχήμα 3.6: Επίδραση Διαταραχής Εγγραφής σε Κελιά PCM. Η Ενεργή Γραμμή Λέξης WL_n με (a) Παλιές Τιμές Δεδομένων και με (b) νέες Τιμές Δεδομένων. Τα Κόκκινα και Κιτρινα Κελιά Αναπαριστούν τα Κελιά Θύτες και Θύματα, Αντίστοιχα.

Η διαταραχή εγγραφής (write disturbance-WD) στις PCM εξαρτάται από τα μοτίβα τιμών που γράφονται. Συγκεκριμένα, το WD μπορεί να συμβεί ευκολότερα μεταξύ ενός κελιού- X που βρίσκεται σε κατάσταση RESET, ενώ το γειτονικό του κελί- Y είναι αδρανές και σε κατάσταση RESET. Αυτό οφείλεται στο ότι: Εφόσον το ρεύμα SET είναι μικρότερο από το 50% του ρεύματος RESET, η αύξηση της θερμοκρασίας κατά τη διάρκεια του SET είναι περίπου τέσσερις φορές χαμηλότερη από εκείνη κατά τη διάρκεια του RESET [41]. Ωστόσο, καθώς η τεχνολογία κλιμακώνεται προς τα κάτω και η απόσταση των κελιών μειώνεται, η θερμική αλληλεπίδραση μπορεί να επηρεάσει και κελιά που είναι αδρανή σε κατάσταση SET, ιδιαίτερα αν διαδοχικά μεταβούν σε κατάσταση RESET τρία ή τέσσερα γειτονικά κελιά, ώστε να μεγιστοποιηθεί η θερμική αλληλεπίδραση. Το Σχήμα 37(b) απεικονίζει τα ευάλωτα κελιά PCM με κόκκινο χρώμα. Για ένα κελί που είναι σε RESET κατάσταση, το γειτονικό του κελί είναι ευάλωτο εάν αποθηκεύει το bit '0' ή το bit '1' και είναι αδρανές. Τα ευάλωτα κελιά μπορεί να εμφανίζονται μεταξύ γραμμών bit κατά μήκος μιας BL ή μεταξύ WL κατά μήκος μιας BL. Τα ευάλωτα κελιά δεν παράγουν πάντα σφάλματα διαταραχής (disturbance errors). Ωστόσο,

στη χειρότερη περίπτωση, μία μετάβαση σε RESET κατάσταση μπορεί να επηρεάσει τέσσερα γειτονικά κελιά [39].

Σύμφωνα με όσα αναφέραμε παραπάνω, ένα κελί που μεταβαίνει από την κατάσταση SET στην κατάσταση RESET έχει περίπου 10% πιθανότητα να διαταράξει μέσω της θερμικής αλληλεπίδρασης τα γειτονικά κελιά. Καθώς η τεχνολογία κλιμακώνεται προς τα κάτω και η απόσταση, όπως επίσης και το μέγεθος των κελιών, μικραίνει, αυτή η πιθανότητα αναμένεται να αυξηθεί. Έτσι στην περίπτωση που περισσότερα από ένα γειτονικά κελιά (δύο, τρία ή τέσσερα αναλόγως) μεταβούν σε κατάσταση RESET η πιθανότητα να επηρεάσουν το κελί θύμα αυξάνεται ανάλογα. Η επίδραση της θερμικής αλληλεπίδρασης στο κελί θύμα μεγιστοποιείται στην περίπτωση μετάβασης τεσσάρων γειτονικών κελιών σε κατάσταση RESET διαδοχικά, και με πιθανότητα περίπου 40% θα αλλοιώσει τα δεδομένα που έχει αποθηκευμένα το κελί θύμα. Προκειμένου, σε κάθε περίπτωση, η επίδραση της θερμικής αλληλεπίδρασης να είναι μέγιστη, είναι κρίσιμο η μετάβαση των γειτονικών κελιών από την κατάσταση SET στην κατάσταση RESET να είναι διαδοχική στο χρόνο, προκειμένου το κελί θύμα να επηρεάζεται συνεχόμενα από την θερμότητα που εκλύεται.



Σχήμα 3.7: (a) WD κατά Μήκος μιας Γραμμής Bit και
(b) Μοτίβο Ευάλωτων Δεδομένων σε PCM.

3.3 Αλγόριθμοι Δοκιμής σε Μνήμες PCM

Σε αυτήν την ενότητα θα παρουσιάσουμε κάποιους αλγορίθμους που έχουν προταθεί για τη δοκιμή (testing) των PCM στην διεθνή βιβλιογραφία. Η υποενότητα 3.3.1 αναφέρεται σε αυτούς τους αλγορίθμους. Η δεύτερη υποενότητα 3.3.2 αναφέρεται στο Μοντέλο Σφαλμάτων Ευαισθησίας στο Μοτίβο της Γειτονιάς (Neighborhood Pattern Sensitive Faults - NPSF) και στους βασικούς αλγορίθμους για την ανίχνευση του συγκεκριμένου τύπου σφαλμάτων.

3.3.1 Μελέτη Βιβλιογραφίας για τις Βέλτιστες Πρακτικές Δοκιμής στις Μνήμες PCM

Προκειμένου να καλυφθούν τα σφάλματα των PCM που παρουσιάστηκαν στην ενότητα 3.2.1, νέοι αλγόριθμοι για δοκιμή έχουν σχεδιαστεί και προταθεί. Το 2006 προτάθηκε ένας αλγόριθμος πορείας (March), με πολυπλοκότητα $11 \cdot M \times N$, όπου τα M και N είναι ο αριθμός των γραμμών και των στηλών, αντίστοιχα, για τον έλεγχο των και σφαλμάτων μόνιμης τιμής (stuck-at) και διαταραχής (disturb) [20]. Ο αλγόριθμος πορείας παρουσιάζεται παρακάτω:

$$\{\uparrow(W1); \uparrow(R1, W0, R0); \downarrow(R0, W1, R1); \downarrow(R1, W0); \uparrow(R0, W1)\}$$

Τα σύμβολα “ \uparrow ” και “ \downarrow ” αντιπροσωπεύουν το εάν η ακολουθία διευθύνσεων προς τη μνήμη προχωρά κατά αύξουσα ή φθίνουσα πορεία αντίστοιχα. Τα R_x και W_x αντιπροσωπεύουν τη λειτουργία ανάγνωσης ή εγγραφής για την αναμενόμενη τιμή “x” (0 ή 1) στο κελί που απευθύνεται η διεύθυνση. Όλες οι λειτουργίες που βρίσκονται μεταξύ αγκύλων γίνονται στο ίδιο κελί πριν ο αλγόριθμος προχωρήσει στο επόμενο κελί. Για παράδειγμα, το στοιχείο πορείας $\uparrow(R1, W0, R0)$ εκτελεί μια λειτουργία ανάγνωσης “1”, ακολουθούμενη από εγγραφή “0”, ακολουθούμενη από μια λειτουργία ανάγνωσης “0” στο κελί διεύθυνσης (ας πούμε κελί i) πριν μεταβεί ο αλγόριθμος στο επόμενο κελί (κελί $i+1$) για την ίδια διεργασία [20].

Ο αλγόριθμος αποτελείται από πέντε κύρια βήματα (πέντε ακολουθίες πορείας). Η πρώτη ακολουθία πορείας, M_0 , αποτελείται από μία μόνο λειτουργία,

συγκεκριμένα τη λειτουργία $W1$ που εφαρμόζεται με τυχαία σειρά διευθύνσεων (το \mathbb{F} σημαίνει μια τυχαία ακολουθία διευθύνσεων). Η ακολουθία $M_1, \uparrow(R1, W0, R0)$, αποτελείται από τρεις πράξεις: διάβασε "1", γράψε "0", διάβασε "0" που εφαρμόζεται σε αύξουσα σειρά διευθύνσεων. Τα υπόλοιπα στοιχεία πορείας μπορούν να εξηγηθούν και να επισημανθούν με παρόμοιο τρόπο [20].

Ακολουθεί η απόδειξη του τρόπου με τον οποίο ο αλγόριθμος πορείας διεγείρει/εντοπίζει τα διαφορετικά σφάλματα, υποθέτοντας έναν πίνακα κελιών ($m \times m$):

1. Η λειτουργία $W1$ στο M_0 αρχικοποιεί τον πίνακα και διεγείρει όλα τα SR σφάλματα.
2. Η λειτουργία $R1$ στο M_1 ανιχνεύει σφάλμα SR για όλα τα κελιά. Το $W0$, η δεύτερη λειτουργία στο M_1 , ενεργοποιεί το SS και το PDF σε όλα τα κελιά. Η τρίτη λειτουργία, $R0$, ανιχνεύει τα SS και PDF σε όλα τα κελιά εκτός από το κελί $C_{(0,0)}$. Επιπλέον, η λειτουργία $R0$ στο M_1 διεγείρει και ανιχνεύει τα RD σφάλματα και διεγείρει τα σφάλματα FWR.
3. Στο M_2 , το $R0$ ανιχνεύει το PDF για το πρώτο κελί και το FWR για όλα τα κελιά. Όλα τα σφάλματα PDF εκτός από το κελί $C_{(m-1,m-1)}$ διεγείρονται και ανιχνεύονται από τις λειτουργίες $W1$ και $R1$ του M_2 αντίστοιχα. Το PDF στο $C_{(m-1,m-1)}$ διεγείρεται μόνο από τη λειτουργία $W1$. Επιπλέον, το $R1$ στο M_2 διεγείρει τα σφάλματα RD και FWR αλλά ανιχνεύει μόνο το σφάλμα FWR.
4. Η λειτουργία $R1$ στο M_3 ανιχνεύει το σφάλμα FWR σε όλα τα κελιά καθώς και το PDF στο κελί $C_{(0,0)}$ που διεγείρεται από το $W0$ του M_3 .
5. Στο M_4 , η λειτουργία $W1$ θα διεγείρει το σφάλμα PDF στο κελί $C_{(m-1,m-1)}$ και η λειτουργία $R0$ θα το ανιχνεύσει.

Ακολούθως, το 2011 προτάθηκε μία βελτιωμένη έκδοση του αλγορίθμου προκειμένου να μειωθεί η πολυπλοκότητα σε $8 \cdot M \times N$ [24]. Εξ αιτίας της λειτουργίας γραμμικής διευθυνσιοδότησης (linear addressing mode), οι αλγόριθμοι αυτοί περιορίζουν την ικανότητά τους να διεγείρουν τα σφάλματα PDF, που οφείλονται στη θερμική αλληλεπίδραση, επειδή δεν μπορούν να χρησιμοποιήσουν και τα τέσσερα γειτονικά κελιά για να διαταράξουν το κελί θύματος.

Οι Zhang et al. [44], το 2012, εφάρμοσαν τον τροποποιημένο αλγόριθμο πορείας για να ελέγξουν σφάλματα παρεμβολής προγραμματισμού (δηλ. PDF), σφάλματα RD και σφάλματα εγγραφής '1/0' και παρατήρησαν ότι το σφάλμα εγγραφής '1' είναι ο κυρίαρχος τύπος σφάλματος που αυξάνεται γραμμικά με τον αριθμό των κύκλων προγραμματισμού.

Ο αλγόριθμος παρουσιάζεται στο Σχήμα 3.8 [44]. Κάθε στοιχείο εφαρμόζει ένα σύνολο λειτουργιών ανάγνωσης/εγγραφής σε μια δεδομένη διεύθυνση μνήμης. Ο αλγόριθμος πορείας τρέχει σε διαδοχικές διευθύνσεις κατά μήκος της κατεύθυνσης του συμβόλου \blacktriangledown . Αυτός ο αλγόριθμος πορείας ανιχνεύει τέσσερις τύπους σφαλμάτων ως εξής [44]:

- WDF1: Ανιχνεύεται στο M_3 . Δύο λειτουργίες ανάγνωσης λαμβάνουν δύο τιμές που αναμένεται να είναι 1. Ωστόσο, εάν εμφανιστεί ένα WDF1 σε οποιοδήποτε bit, οι επιστρεφόμενες τιμές από τις λειτουργίες ανάγνωσης θα πρέπει να είναι 0. Μπορεί να γίνει μια λειτουργία Bit-OR μεταξύ δύο τιμών για να εντοπίσουμε το WDF1. Το Bit-OR αποκλείει το σφάλμα RD.
- WDF0: Όμοια με το WDF1, το WDF0 μπορεί να ανιχνευθεί στο M_0 . Δύο τιμές από την ανάγνωση αναμένεται να είναι 0. Γίνεται μια λειτουργία Bit-AND μεταξύ τους για να ανιχνευθεί το WDF0. Το Bit-AND αποκλείει το σφάλμα RD.
- PDF: Όλα τα PDFs ανιχνεύονται από το M_0 και το M_3 . Εάν παρουσιαστεί ένα PDF, τα δεδομένα που διαβάζονται από το M_2 θα πρέπει να διαφέρουν και από τα δύο δεδομένα που διαβάστηκαν από το M_0 . Το M_5 λειτουργεί με τον ίδιο τρόπο. Ο λόγος για να προστεθεί το M_1 και το M_4 είναι για να διακριθούν τα σφάλματα RD από τα σφάλματα PDF. Ένα σφάλμα PDF παρουσιάζεται μόνο εάν τα M_1 και M_2 διαβάζουν τα ίδια δεδομένα από μια συγκεκριμένη τοποθεσία. Τα M_4 και M_5 λειτουργούν με τον ίδιο τρόπο όπως τα M_1 και M_2 .
- RD: Στα M_0 και M_3 , εμφανίζεται ένα RD εάν δύο αναγνώσεις από την ίδια διεύθυνση είναι διαφορετικές μεταξύ τους.

addr	March 0	March 1	March 2
0	W0,R0,R0	R0	R0
1	W0,R0,R0	R0	R0
⋮			
31	↙ W0,R0,R0	↙ R0	↙ R0

addr	March 3	March 4	March 5
0	W1,R1,R1	R1	R1
1	W1,R1,R1	R1	R1
⋮			
31	↙ W1,R1,R1	↙ R1	↙ R1

Σχήμα 3.8: Αλγόριθμος Πορείας Ανίχνευσης Λαθών PCM.

Το 2013, οι Kannan et al. [45] εφάρμοσαν μια μέθοδο κρυφού μονοπατιού (sneak path) για να ελέγξουν τις μη πτητικές μνήμες με έναν αμφίδρομο επιλογέα (Memristor ή PCM) σε κάθε κελί. Ο αλγόριθμός τους έχει πολυπλοκότητα μόνο $3.8 \cdot M \times N$ όταν εφαρμόζεται στις PCM, επειδή για τη δοκιμή χρησιμοποιείται η Περιοχής Ανίχνευσης (Region of Detection-RoD), η οποία είναι μια ομάδα κελιών μνήμης που μπορούν να δοκιμασθούν ταυτόχρονα και όχι μεμονωμένα.

Ο αλγόριθμος πορείας που προτείνουν καλύπτει τα σφάλματα PDF, RRD, RD, FWR, SS και SR και δίνεται παρακάτω:

$$\text{March - PCM} : \{ \uparrow (w_0) ; \uparrow (r_0, w_1, r_1); \downarrow (r_1, w_0, r_0); \downarrow (r_0) \}$$



Ο χρόνος ελέγχου για αυτόν τον αλγόριθμο πορείας είναι 5 αναγνώσεις και 3 εγγραφές με αποτέλεσμα $8 \cdot M \times N$ προσπελάσεις μνήμης. Επιπλέον, τα σφάλματα που συμβαίνουν επίσης στις μνήμες PCM προκαλούνται λόγω της ποσότητας ρεύματος που ρέει μέσα από ένα κελί μνήμης. Για παράδειγμα, κατά τη λειτουργία ανάγνωσης, η ποσότητα του ρεύματος που διαρρέει το στοιχείο είναι της τάξης του $1 \mu\text{A}$ [45]. Παρουσία ελαττωμάτων, η ποσότητα του ρεύματος που ρέει στο κελί αυξάνεται οδηγώντας σε αύξηση της ποσότητας θερμότητας που παράγεται από το υλικό PCM. Αυτή η τοπική θερμότητα έχει ως αποτέλεσμα την αλλαγή της κατάστασης των γειτονικών στοιχείων μνήμης PCM. Αυτός ο τύπος σφάλματος ευαισθητοποιείται με τη χρήση δύο λειτουργιών r_0 διαδοχικά για τη δημιουργία

αρκετού ρεύματος για την εμφάνιση του φαινομένου της επίδρασης της θερμότητας. Λόγω του αυξημένου ρεύματος κατά τη δοκιμή κρυφών μονοπατιών, είναι πολύ πιο απλό να ευαισθητοποιηθούν αυτοί οι τύποι σφαλμάτων που είναι μοναδικοί για τις μνήμες PCM. Ως εκ τούτου, η ακολουθία ελέγχου για τη δοκιμή κρυφών μονοπατιών είναι:

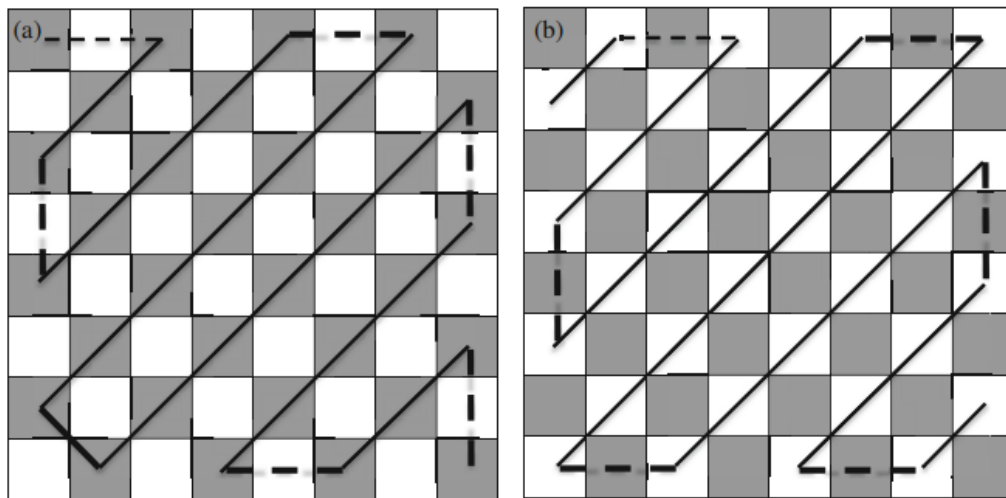
$$\{\uparrow (w0) ; \uparrow (r0) ; \uparrow (w1) ; \uparrow (r1); \downarrow (r1); \downarrow (w0); \downarrow (r0)\}.$$

Το κόστος είναι 3 εγγραφές και 4 αναγνώσεις, μειώνοντας τον χρόνο δοκιμής σε $3 \cdot M \times N + 4 \cdot M \times N / 5$ (μείωση 52,5% στο χρόνο δοκιμής) [45].

Τέλος, το 2016, οι C. Xiaole et al., πρότειναν ένα νέο σχήμα διευθυνσιοδότησης της μνήμης που ονομάζεται snake addressing mode για να διεγείρουν αποτελεσματικότερα ένα PDF. Για το κελί θύματος, το νέο σχήμα διευθυνσιοδότησης επιτρέπει τις λειτουργίες (μη διαδοχικής στο χρόνο) εγγραφής των τεσσάρων γειτονικών κελιών πριν από την κλήση της λειτουργίας ανάγνωσης της κελιού, καθιστώντας μια πιο αυστηρή συνθήκη ελέγχου για τα σφάλματα που προκύπτουν από τη θερμική αλληλεπίδραση μεταξύ των κελιών. Επιπλέον, παρουσιάζουν έναν αλγόριθμο δοκιμής τύπου πορείας με το snake addressing mode για να μειώσει την πολυπλοκότητα της δοκιμής σε $7 \cdot M \times N$ και ο αλγόριθμος έχει τη δυνατότητα να ελέγχει όλους τους τύπους σφαλμάτων σύμφωνα με την εργασία [24] και τα σφάλματα που προκύπτουν από τις παρασιτικές επιδράσεις [33].

Στο Σχήμα 3.9 παρουσιάζεται το snake addressing mode, που χρησιμοποιείται κατά τη διάρκεια του ελέγχου, και το οποίο ενεργοποιεί και τα τέσσερα γειτονικά κελιά του κελιού θύματος. Παρόμοια με τον αλγόριθμο σκακιέρας (checkerboard algorithm), το snake addressing mode διαιρεί τα κελιά μνήμης σε δύο ομάδες, οι οποίες φαίνονται ως μαύρα κελιά και λευκά κελιά στο Σχήμα 3.9. Για κάθε μαύρο κελί, και τα τέσσερα γειτονικά κελιά είναι λευκά και αντίστροφα για κάθε λευκό κελί. Το Σχήμα 3.9(a) και (b) απεικονίζει το snake addressing mode για τις λειτουργίες μαύρων και λευκών κελιών, αντίστοιχα, όπου οι συμπαγείς γραμμές υποδηλώνουν τις λειτουργίες σε μαυρα/λευκά κελιά και οι διακεκομμένες γραμμές υποδηλώνουν παράλειψη των αντίστοιχων λευκών/μαύρων κελιών. Εδώ, τα σύμβολα  και , που είναι παραλλαγές της σημειογραφίας που χρησιμοποιείται

από τον παραδοσιακό αλγόριθμο March, χρησιμοποιούνται για να υποδηλώσουν το snake addressing για τα μαύρα κελιά και τα λευκά κελιά, αντίστοιχα [33].



Σχήμα 3.9: Snake Addressing Mode (a) για Λευκά Κελιά και (b) Μαύρα Κελιά.

Για να γίνει ο έλεγχος για τα PDFs της συστοιχίας PCM, ένας αλγόριθμος πορείας τεσσάρων βημάτων προτείνεται στην εργασία [33]:

$$\text{March-PDF} = \{M1: \Downarrow (w0); M2: \swarrow (w0); M3: \searrow (r0, w0); M4: \nwarrow (r0)\}.$$

Ο αλγόριθμος αρχικοποιεί όλα τα κελιά στην κατάσταση RESET. Στη συνέχεια, εκτελεί λειτουργίες εγγραφής ‘0’ για κάθε λευκό και μαύρο κελί με το snake addressing mode στο Βήμα 2 και Βήμα 3 και διεγείρει τα PDFs με τις λειτουργίες RESET σε όλα τα γειτονικά κελιά. Για τα οριακά κελιά και τα γωνιακά κελιά, γίνεται πρόσβαση σε τρία και δύο γειτονικά κελιά, αντίστοιχα. Για τα άλλα κελιά της συστοιχίας, τέσσερα γειτονικά κελιά είναι προγραμματισμένα στην κατάσταση RESET. Τέλος, η κατάσταση κάθε κελιού θύματος διαβάζεται αναμένοντας την τιμή ‘0’ στο Βήμα 3 και στο Βήμα 4 για τα μαύρα κελιά και τα λευκά κελιά, αντίστοιχα. Αυτός ο αλγόριθμος ικανοποιεί πλήρως τη συνθήκη δοκιμής του PDF [33].

Επίσης, ένας αλγόριθμος ελέγχου έχει προταθεί, που ενσωματώνει το σχήμα snake addressing για να ελέγξει όλα τα σφάλματα που αναφέρονται στον Πίνακα

3.1. Ο προτεινόμενος αλγόριθμος πορείας με το snake addressing, δίνεται παρακάτω [33]:

$$\text{March-SA} = \{M1: \rightarrow(w0); M2: \rightarrow(w0, r0); M3: \rightarrow(r0, w0, r0); M4: \rightarrow(r0, w1, r1_m); M5: \rightarrow(r1)\},$$

όπου η κατάσταση 1_m είναι μια ασθενής κατάσταση 1, η αντίστοιχη αντίσταση του κελιού είναι συνήθως μεγαλύτερη από αυτή της κατάστασης 1.

Χρησιμοποιώντας τον προτεινόμενο αλγόριθμο March-SA, η κάλυψη σφαλμάτων για όλα τα σφάλματα που αναφέρονται στον Πίνακα 3.1 αναλύεται ως εξής:

- Σφάλμα SR: Η λειτουργία $w1$ στο $M4$ διεγείρει τα σφάλματα SR για όλα τα κελιά. Στη συνέχεια, τα σφάλματα ελέγχονται από τη λειτουργία $r1$ στο $M5$.
- Σφάλμα SS: Οι λειτουργίες $w0$ στα $M1$ και $M2$ επαναφέρουν (RESET) όλα τα κελιά μνήμης. Τα σφάλματα SS έπειτα ελέγχονται από τις λειτουργίες $r0$ στα $M2$ και $M3$.
- IPF0: Οι λειτουργίες $w0$ στο $M2$ και $M3$ διεγείρουν τα IPF0s για τα λευκά και τα μαύρα κελιά, αντίστοιχα. Τα IPF0s για τα λευκά κελιά ελέγχονται στη συνέχεια με το $r0$ στο $M2$ και το $r0$ στο $M4$, και τα IPF0s για τα μαύρα κελιά ελέγχονται από το δεύτερο $r0$ στο $M3$ και το $r0$ στο $M4$.
- WTF0: Η συνθήκη ελέγχου είναι η ίδια με αυτή ενός σφάλματος SS, δηλαδή ελέγχεται διεξοδικά από τον αλγόριθμο.
- WDF1: Τα σφάλματα διεγείρονται από τις λειτουργίες $w1$ στο $M4$ και στη συνέχεια ελέγχονται από το $r1$ στο $M5$.
- WWDF1: Ένα WWDF1 διεγείρεται και ελέγχεται από τα $w1$ και $r1_m$ στο $M4$, αντίστοιχα.
- PDF: Όλα τα μαύρα κελιά αρχικοποιούνται στην κατάσταση RESET στο $M1$ και τα PDF τους διεγείρονται στο $M2$ γράφοντας '0' στους γείτονές τους (λευκά κελιά). Στη συνέχεια, τα PDF των μαύρων κελιών ελέγχονται με την πρώτη λειτουργία $r0$ στο $M3$. Ομοίως, τα λευκά κελιά αρχικοποιούνται στην κατάσταση RESET στο $M2$, τα γειτονικά τους

κελιά (μαύρα κελιά) γράφονται στο '0' στο M3, και τα PDF ελέγχονται με τη λειτουργία r0 στο M4.

- Σφάλμα RRD: Τα σφάλματα RRD των λευκών κελιών διεγείρονται από τη λειτουργία w0 στο M2. Στη συνέχεια, τα σφάλματα ελέγχονται από τη λειτουργία r0 στο M2, η οποία ακολουθεί τη λειτουργία w0. Για τα μαύρα κελιά, η λειτουργία w0 στο M3 διεγείρει τα σφάλματα RRD και η δεύτερη λειτουργία r0 στο M3, ακολουθούμενη από τη λειτουργία w0, ελέγχει το σφάλμα.
- Σφάλμα RD: Τα σφάλματα RD των μαύρων και λευκών κελιών αρχικοποιούνται στο μηδέν με τη λειτουργία w0 στο M1 και M2, αντίστοιχα. Τα σφάλματα για τα λευκά κελιά διεγείρονται και ελέγχονται με το r0 στο M2 και το r0 στο M4, ενώ τα σφάλματα για τα μαύρα κελιά διεγείρονται και ελέγχονται από το r0 στο M3 και r0 στο M4, αντίστοιχα.
- Σφάλμα FWR: Η λειτουργία w1 στο M4 αρχικοποιεί όλα τα κελιά στην κατάσταση SET. Στη συνέχεια, τα σφάλματα FWR διεγείρονται από το r1_m στο M4 και ελέγχονται από τη λειτουργία r1 στο M5. Και οι δύο λειτουργίες r1^m και r1 εφαρμόζουν τον ίδιο παλμό ρεύματος στο ηλεκτρόδιο των κελιών PCM. Η μόνη διαφορά μεταξύ τους είναι ότι έχουν διαφορετικές τάσεις αναφοράς. Για το λόγο αυτό, το r1_m στο M4, το οποίο χρησιμοποιείται για τη διέγερση σφαλμάτων FWR, αντικαθιστά τη λειτουργία r1 στη συνθήκη ελέγχου [33].

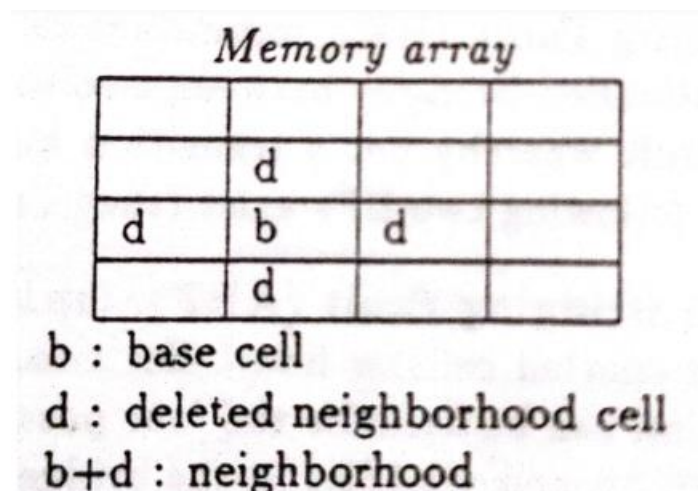
3.3.2 Μοντέλο Σφαλμάτων Ευαισθησίας στο Μοτίβο της Γειτονιάς (Neighborhood Pattern Sensitive Faults)

Δεδομένου του ότι η θερμική αλληλεπίδραση επηρεάζει τα γειτονικά κελιά κατά τη διαδικασία της εγγραφής, δύναται να χρησιμοποιηθούν μοτίβα ελέγχου που ευαισθητοποιούν τα σφάλματα γειτονιάς, δηλαδή το Μοντέλο Σφαλμάτων Ευαισθησίας στο Μοτίβο της Γειτονιάς (Neighborhood Pattern Sensitive Faults - NPSF). Ένα Μοτίβο Ευαίσθητης Γειτονιάς (Pattern Sensitive Fault-PSF) ορίζεται ως εξής: Τα περιεχόμενα ενός κελιού, ή η ικανότητα να αλλάζει τα περιεχόμενα, επηρεάζονται από τα περιεχόμενα όλων των άλλων κελιών στη μνήμη. Αυτά τα

περιεχόμενα αποτελούνται από ένα μοτίβο 0 και 1, ή αλλαγές σε αυτά τα περιεχόμενα.

Το PSF μπορεί να θεωρηθεί η πιο γενική περίπτωση του σφάλματος k-σύζευξης (k-coupling fault), δηλαδή η περίπτωση όπου το $k=n$ (όπου το n αντιπροσωπεύει όλα τα κελιά στη μνήμη). Το Σχήμα 3.10 δείχνει την ορολογία που χρησιμοποιείται με τα NPSF. Η γειτονιά είναι ο συνολικός αριθμός κελιών που εμπλέκονται σε ένα συγκεκριμένο μοντέλο σφάλματος. Το κελί βάσης (base cell) είναι το υπό δοκιμή κελί. Η γειτονιά με το βασικό κελί να εξαιρείται ονομάζεται διαγραμμένη γειτονιά (deleted neighborhood).

Το μοντέλο PSF επιτρέπει στη διαγραμμένη γειτονιά να πάρει οποιαδήποτε θέση στη συστοιχία μνήμης. Όταν η διαγραμμένη γειτονιά επιτρέπεται να λάβει μόνο μία θέση (έτσι ώστε να περιβάλλει το κελί βάσης), μιλάμε για ένα NPSF. Το μοντέλο NPSF είναι επομένως ένα υποσύνολο του μοντέλου PSF [46].



Σχήμα 3.10: Ορολογία Μοντέλου Σφαλμάτων Ευαισθησίας στο Μοτίβο της Γειτονιάς.

Όλοι οι γνωστοί αλγόριθμοι προσδιορίζουν ότι η διαγραμμένη γειτονιά πρέπει να βρίσκεται στη φυσική εγγύτητα του κελιού βάσης, επειδή αυτή η διαγραμμένη γειτονιά είναι πιο πιθανό να επηρεάσει το κελί βάσης.

Τρεις τύποι NPSF μπορούν να διακριθούν:

- Ενεργό NPSF (Active NPSF-ANPSF) που ονομάζεται επίσης Δυναμικό (Dynamic) NPSF: το κελί βάσης αλλάζει τα περιεχόμενά του λόγω μιας

αλλαγής στο μοτίβο της διαγραμμένης γειτονιάς. Αυτή η αλλαγή αποτελείται από μια μετάβαση σε ένα κελί της διαγραμμένης γειτονιάς, ενώ τα υπόλοιπα κελιά της διαγραμμένης γειτονιάς και το κελί βάσης περιέχουν ένα συγκεκριμένο μοτίβο. Ένας έλεγχος που πρέπει να ανιχνεύει και να εντοπίζει τα ANPSF θα πρέπει να ικανοποιεί την απαίτηση:

Κάθε κελί βάσης πρέπει να διαβάζεται στην κατάσταση 0 και στην κατάσταση 1, για όλα τις δυνατές αλλαγές στο μοτίβο της διαγραμμένης γειτονιάς.

Για μια γειτονιά Τύπου-1 (Type-1 neighborhood), η οποία αποτελείται από τέσσερα κελιά διαγραμμένης γειτονιάς, η γενική σύνταξη ενός NPSF είναι η εξής: $C_{ij} < d_0, d_1, d_2, d_3; b >$, όπου C_{ij} είναι η θέση του κελιού βάσης. Ένα ANPSF όπου το κελί βάσης γίνεται '1' όταν το d_1 κάνει μια μετάβαση Ψ (από 1 σε 0), ενώ τα υπόλοιπα κελιά της διαγραμμένης γειτονιάς d_0, d_2 και d_3 περιέχουν την τιμή 011, μπορεί να γραφτεί ως: $C_{ij} < 0, \Psi, 1, 1; 1 >$. Το ANPSF μπορεί να θεωρηθεί μια περιορισμένη έκδοση του σφάλματος αδυναμίας σύζευξης (idempotent coupling fault). Ομοίως, ο συμβολισμός για ένα ANPSF όπου το κελί βάσης γίνεται '0', υπό τις ίδιες συνθήκες για τα κελιά διαγραμμένης γειτονιάς, μπορεί να γραφτεί ως: $C_{ij} < 0, \Psi, 1, 1; 0 >$ και όταν το κελί βάσης είναι αντιστραμμένο, το ANPSF μπορεί να γραφτεί ως: $C_{ij} < 0, \Psi, 1, 1; \Phi >$ [46].

- Παθητικό NPSF (Passive NPSF-PNPSF): το περιεχόμενο του κελιού βάσης δεν μπορεί να αλλάξει (δεν μπορεί να κάνει μετάβαση) λόγω ενός συγκεκριμένου μοτίβου γειτονιάς. Ένας έλεγχος που πρέπει να ανιχνεύει και να εντοπίζει PNPSF θα πρέπει να ικανοποιεί την απαίτηση:

Κάθε κελί βάσης πρέπει να γράφεται και να διαβάζεται στην κατάσταση 0 και στην κατάσταση 1, για όλα τις μεταθέσεις στο μοτίβο της διαγραμμένης γειτονιάς.

Ο συμβολισμός για ένα PNPSF σε μια γειτονιά Τύπου-1, όπου το κελί βάσης δεν μπορεί να αλλάξει από την τιμή '0' στην τιμή '1' όταν τα κελιά της διαγραμμένης γειτονιάς περιέχουν την τιμή '0011', είναι $C_{i,j} < 0, 0, 1, 1; \uparrow/0 >$. Όταν το κελί βάσης δεν μπορεί να αλλάξει από την τιμή 1 δεδομένου του ίδιου μοτίβου διαγραμμένης γειτονιάς, το PNPSF μπορεί να γραφτεί ως: $C_{i,j} < 0, 0, 1, 1; \downarrow/1 >$, και όταν το κελί βάσης δεν μπορεί να αλλάξει, ανεξάρτητα από το περιεχόμενό του, αυτό το PNPSF μπορεί να γραφτεί ως: $C_{i,j} < 0, 0, 1, 1; \nabla/x >$ [46].

Στατικό NPSF (Static NPSF-SNPSF) το περιεχόμενο ενός κελιού βάσης αναγκάζεται σε μια συγκεκριμένη κατάσταση λόγω ενός συγκεκριμένου μοτίβου διαγραμμένης γειτονιάς. Ένας έλεγχος που πρέπει να ανιχνεύει και να εντοπίζει SNPSF θα πρέπει να ικανοποιεί την απαίτηση:

Κάθε κελί βάσης πρέπει να διαβάζεται στην κατάσταση 0 και στην κατάσταση 1, για όλα τις μεταθέσεις στο μοτίβο της διαγραμμένης γειτονιάς.

Ο συμβολισμός για ένα SNPSF σε μια γειτονιά Τύπου-2, όπου το κελί βάσης αναγκάζεται στην τιμή '0' όταν τα κελιά διαγραμμένης γειτονιάς d_0 έως d_7 περιέχουν την τιμή '01001111' μπορεί να γραφτεί ως: $C_{i,j} < 0, 1, 0, 0, 1, 1, 1, 1; -/0 >$. Όταν το κελί βάσης αναγκαστεί στην τιμή '1', το SNPSF μπορεί να γραφτεί ως: $C_{i,j} < 0, 1, 0, 0, 1, 1, 1, 1; -/1 >$ [46].

Για να ελεγχθεί μια γειτονιά για ένα συγκεκριμένο είδος NPSF, όλα τα απαιτούμενα μοτίβα ελέγχου πρέπει να εφαρμοστούν σε αυτήν τη γειτονιά και μετά από κάθε μοτίβο ελέγχου πρέπει να διαβάζεται το κελί βάσης. Με αυτόν τον τρόπο μπορούν να εντοπιστούν όλα τα NPSF, δηλαδή το μοτίβο ελέγχου που ευαισθητοποιεί το σφάλμα και η γειτονιά που είναι ελαττωματική μπορεί να προσδιοριστεί. Ο αριθμός των απαιτούμενων προτύπων δοκιμής και οι μηχανισμοί για μια ελάχιστη ακολουθία δοκιμών συζητούνται παρακάτω.

Για να εντοπιστούν ή να ανιχνευθούν όλα Active NPSF, όπου είναι και η κατηγορία που θα εξετάσουμε στις PCM, απαιτούνται $(k - 1) \cdot 2^k$ μοτίβα. Τα μοτίβα είναι: κελί βάσης 0 και 1, και για τα $k-1$ κελιά της διαγραμμένης γειτονιάς, ένα κελί με μία \uparrow και μια \downarrow μετάβαση, και τα άλλα $k-2$ κελιά, $2^{(k-2)}$ μοτίβα από 0 και 1. Άρα συνολικά $2 \cdot (k - 1) \cdot 2 \cdot 2^{(k-2)} = (k - 1) \cdot 2^k$ μοτίβα. Αυτά τα μοτίβα θα ονομάζονται Μοτίβα Ενεργής Γειτονιάς (Active Neighborhood Patterns-ANP). Όλα τα ANP για μια γειτονιά Τύπου 1 δίνονται στον Σχήμα 3.11 [46].

b	00000000000000001111111111111111
d ₁	↑↑↑↑↑↑↑↓↓↓↓↓↓↓↓↑↑↑↑↑↑↑↓↓↓↓↓↓↓↓
d ₂	00001111000011110000111100001111
d ₃	00110011001100110011001100110011
d ₄	01010101010101010101010101010101
b	00000000000000001111111111111111
d ₁	00001111000011110000111100001111
d ₂	↑↑↑↑↑↑↑↓↓↓↓↓↓↓↓↑↑↑↑↑↑↑↓↓↓↓↓↓↓↓
d ₃	00110011001100110011001100110011
d ₄	01010101010101010101010101010101
b	00000000000000001111111111111111
d ₁	00001111000011110000111100001111
d ₂	00110011001100110011001100110011
d ₃	↑↑↑↑↑↑↑↓↓↓↓↓↓↓↓↑↑↑↑↑↑↑↓↓↓↓↓↓↓↓
d ₄	01010101010101010101010101010101
b	00000000000000001111111111111111
d ₁	00001111000011110000111100001111
d ₂	00110011001100110011001100110011
d ₃	01010101010101010101010101010101
d ₄	↑↑↑↑↑↑↑↓↓↓↓↓↓↓↓↑↑↑↑↑↑↑↓↓↓↓↓↓↓↓

b = base cell

d₁, d₂, d₃, d₄ = deleted neighborhood cell

Σχήμα 3.11: Όλα τα Μοτίβα Ενεργής Γειτονιάς.

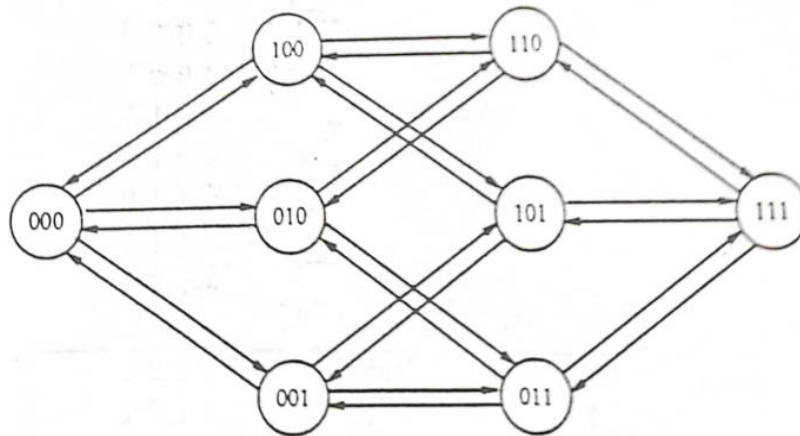
Όταν πρέπει να εφαρμοστεί ένας αριθμός μοτίβων σε μια γειτονιά, πρέπει να επιλεγεί μια ακολουθία για αυτά τα μοτίβα. Για να ληφθεί ένας ελάχιστος αριθμός λειτουργιών εγγραφής, η διαφορά μεταξύ ενός μοτίβου και του διαδόχου του πρέπει να είναι ελάχιστη. Ανάλογα με τη δοκιμή NPSF που θα πραγματοποιηθεί, θα πρέπει να χρησιμοποιείται η ακόλουθη σειρά μοτίβων δοκιμής:

1. Όταν τα μοτίβα περιέχουν μόνο 0 και 1, όπως συμβαίνει με τα SNPs, τότε θα πρέπει να χρησιμοποιείται μια ακολουθία Hamilton.
2. Όταν ένα από τα bit περιέχει μια μετάβαση \uparrow ή \downarrow , όπως συμβαίνει με το ANPs, PNPs και APNPs, τότε θα πρέπει να χρησιμοποιηθεί μια ακολουθία Euler.

Ακολουθία Hamilton. Αυτή είναι μια ακολουθία που χρησιμοποιείται για δοκιμές αναφορικά με τα SNPSFs. Μία k-bit ακολουθία Hamilton είναι μια ακολουθία k-bit μοτίβων στην οποία κάθε μοτίβο διαφέρει κατά ένα bit από το προηγούμενο μοτίβο, επειδή αυτό απαιτεί τον ελάχιστο αριθμό λειτουργιών εγγραφής για να πάει από το ένα μοτίβο στο επόμενο. Ο κώδικας Gray μπορεί να χρησιμοποιηθεί για μια ακολουθία Hamilton [46].

Ακολουθία Euler. Αυτή η ακολουθία χρησιμοποιείται για ANPSFs, PNPSFs και APNPSFs. Ένα k-bit γράφημα Euler ορίζεται ως ένα γράφημα στο οποίο υπάρχει ένας κόμβος για καθένα k-bit μοτίβο των 0 και 1 και υπάρχει μία ακμή μεταξύ δύο κόμβων, εάν και μόνο εάν διαφέρουν κατά ακριβώς ένα bit. Όταν συνδέονται δύο κόμβοι, τότε συνδέονται με ακριβώς δύο ακμές, μία ακμή από τον έναν κόμβο στον άλλο και μία ακμή πίσω. Στο Σχήμα 3.12 απεικονίζεται ένα γράφημα Euler 3-bit [46].

Οι ακμές σε ένα k-bit γράφημα Euler αντιστοιχούν στα ANPs, PNPs και τα APNPs μιας γειτονιάς k-bit. Οι κόμβοι αντιστοιχούν στα SNPs. Μια ακολουθία Euler είναι μια ακολουθία μέσω ενός γραφήματος Euler που επισκέπτεται κάθε ακμή ακριβώς μία φορά. Από την άλλη πλευρά, μια ακολουθία Hamilton είναι μια ακολουθία μέσω ενός γραφήματος Euler που επισκέπτεται κάθε κόμβο ακριβώς μία φορά [46].



Σχήμα 3.12: Γράφημα Euler για 3-bit Μοτίβα.

Όταν χρησιμοποιείται μια ακολουθία Hamilton, είναι δυνατή η εφαρμογή όλων των 2^k SNPs σε μια γειτονιά με $k + 2^k - 1$ πράξεις εγγραφής, με k αρχικές πράξεις εγγραφής για την πρώτο μοτίβο, και $2^k - 1$ για τα υπόλοιπα $2^k - 1$ μοτίβα, όπου k το μέγεθος της γειτονιάς.

Με μια ακολουθία Euler είναι δυνατή η εφαρμογή όλων των $k \cdot 2^k$ APNPs σε μια γειτονιά με $k + k \cdot 2^k$ πράξεις εγγραφής: k πράξεων εγγραφής για την αρχικοποίηση της γειτονιάς, και 1 πράξη εγγραφής για καθένα από $k \cdot 2^k$ APNPs.

Θα ήταν δυνατό να ελέγξουμε τη μνήμη για Ενεργά, Παθητικά ή Στατικά (A/P/S) NPSFs εφαρμόζοντας όλα τα (A/P/S) NPs σε κάθε γειτονιά ξεχωριστά. Όμως, όταν γράφεται ένα κελί, αλλάζουν k διαφορετικές γειτονιές (Τύπου-1 ή Τύπου-2). Είναι δυνατό να μειωθεί (ελαχιστοποιηθεί) ο αριθμός των πράξεων εγγραφής ελέγχοντας τις γειτονιές ταυτόχρονα, δηλαδή γράφοντας μοτίβα στις γειτονιές ταυτόχρονα. Οι μέθοδοι tiling και two-group το επιτρέπουν αυτό [46].

- **Η Μέθοδος Πλακιδίων (Tiling Method)**

Με τη μέθοδο πλακιδίων, η μνήμη καλύπτεται πλήρως από μια ομάδα γειτονιών που δεν επικαλύπτονται. Μια τέτοια ομάδα θα ονομάζεται 'ομάδα πλακιδίων' (tiling group). Το σύνολο όλων των γειτονιών σε μία ομάδα πλακιδίων θα ονομαστεί 'γειτονιές πλακιδίων' (tiling neighborhoods). Στο σχήμα 3.11 αυτό απεικονίζεται για μια γειτονιά Τύπου-1. Τα κελιά κάθε γειτονιάς αριθμούνται από το 0 έως το 4.

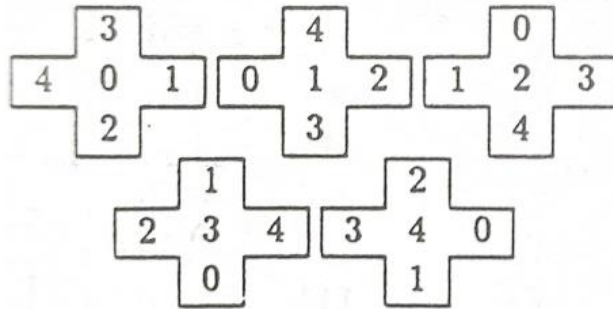
Οι γειτονιές πλακιδίων Τύπου-1 του σχήματος 3.11 έχουν όλες ένα κελί-2 ως κελί βάσης. Η διαγραμμένη γειτονιά όλων των κελιών-2 βάσης

σχηματίζεται από ένα κελί-0, ένα κελί-1, ένα κελί-3 και ένα κελί-4. Στο Σχήμα 3.14 οι γειτονιές των πέντε πιθανών κελιών βάσης, με αριθμό 0 έως 4, φαίνονται με τη δομή κελιών του Σχήματος 3.13.

0	1	2	3	4	0	1	2	3	4
2	3	4	0	1	2	3	4	0	1
4	0	1	2	3	4	0	1	2	3
1	2	3	4	0	1	2	3	4	0
3	4	0	1	2	3	4	0	1	2
0	1	2	3	4	0	1	2	3	4
2	3	4	0	1	2	3	4	0	1
4	0	1	2	3	4	0	1	2	3
1	2	3	4	0	1	2	3	4	0
3	4	0	1	2	3	4	0	1	2

Σχήμα 3.13: Γειτονιές Πλακιδίων Τύπου-1.

Όταν όλα τα SNPs εφαρμόζονται ταυτόχρονα στις γειτονιές όλων των κελιών-2 βάσης (αυτό σημαίνει το πρώτο μοτίβο σε όλες τις γειτονιές, μετά το δεύτερο μοτίβο σε όλες τις γειτονιές κ.λπ.), τότε επίσης όλα τα SNP εφαρμόζονται σε κάθε γειτονιά, που αποτελείται από ένα κελί-0, ένα κελί-1, ένα κελί-2, ένα κελί-3 και ένα κελί-4. Έτσι και στις γειτονιές όλων των κελιών-0 βάσης, όλων των κελιών-1 βάσης, όλων των κελιών-3 βάσης και όλων των κελιών-4 βάσης. Έτσι, όταν όλα τα SNPs εφαρμόζονται ταυτόχρονα στις γειτονιές όλων των κελιών-2 βάσης, εφαρμόζονται αυτόματα στις γειτονιές όλων των κελιών βάσης στη μνήμη. Με αυτόν τον τρόπο όλα τα SNPs μπορούν να εφαρμοστούν για να ελεγχθούν όλες οι γειτονιές για SNPSFs με $n/5 \cdot 2^5$ λειτουργίες εγγραφής (επειδή υπάρχουν $n/5$ κελιά-2 βάσης), αντί για $n \cdot 2^5$ λειτουργίες εγγραφής που θα απαιτούνταν εάν δεν εφαρμοζόταν η μέθοδος tiling. Γενικά ο αριθμός των πράξεων εγγραφής μειώνεται από το $n \cdot 2^k$ σε $n/k \cdot 2^k$.



Σχήμα 3.14: Γειτονίες Πλακιδίων Τύπου-1 για τα Κελιά Βάσης 0, 1, 2, 3 και 4.

Παρόμοια επιχειρήματα ισχύουν για τη γειτονιά Τύπου-2. Τα ίδια επιχειρήματα ισχύουν επίσης για τα Ενεργά και Παθητικά Μοτίβα Γειτονιάς (Active and Passive Neighborhood Patterns-APNPs) αντί για τα SNPs. Με τα APNPs χρειάζονται $n/k \cdot k \cdot 2^k$ λειτουργίες για τον έλεγχο όλων των κελιών για APNPSFs.

- **Η Μέθοδος των Δύο Ομάδων (Two-group Method)**

Η μέθοδος των δύο ομάδων βασίζεται στη δυαδικότητα των κελιών: ένα κελί είναι ένα κελί βάσης σε μια ομάδα και ένα κελί διαγραμμένης γειτονιάς στην άλλη ομάδα και αντίστροφα, βλέπε Σχήμα 3.15. Με τη μέθοδο των δύο ομάδων τα κελιά χωρίζονται σε δύο ομάδες, την ομάδα-1 (group-1) και την ομάδα-2 (group-2), με ένα μοτίβο σκακιέρας. Τα κελιά βάσης μιας ομάδας είναι τα κελιά της διαγραμμένης γειτονιάς της άλλης ομάδας. Η διαγραμμένη γειτονιά αποτελείται από ένα κελί-A, ένα κελί-B, ένα κελί-C και ένα κελί-D, έτσι ώστε κάθε κελί της διαγραμμένης γειτονιάς να είναι μέλος τεσσάρων γειτονιών. Κάθε ομάδα (ομάδα-1 ή ομάδα-2) αποτελείται από $n/2$ κελιά βάσης και $n/2$ κελιά διαγραμμένης γειτονιάς που σχηματίζονται από τέσσερις υποομάδες. Κάθε υπο-ομάδα αποτελείται από $n/8$ κελιά που σχηματίζονται από όλα τα κελιά-A, όλα τα κελιά-B, όλα τα κελιά-C ή όλα τα κελιά-D. Ένα νέο πρότυπο ελέγχου μπορεί να εφαρμοστεί σε όλες τις $n/2$ γειτονίες μιας ομάδας γράφοντας σε όλα τα $n/8$ κελιά μιας υποομάδας, μειώνοντας έτσι τον αριθμό των πράξεων εγγραφής κατά έναν παράγοντα του 4 [46].

Αυτή η μέθοδος μπορεί να χρησιμοποιηθεί μόνο για γειτονιές Τύπου-1. Η επέκτασή του σε γειτονιές Τύπου-2 (δηλ. γειτονιές που αποτελούνται από οκτώ γειτονικά κελιά, συμπεριλαμβάνοντας και τα διαγώνια κελιά) δεν είναι δυνατή, λόγω της έννοιας της δυαδικότητας. Με τις γειτονιές Τύπου-2, υπάρχουν περισσότεροι από ένας διαφορετικοί τύποι κελιών διαγραμμένης γειτονιάς [46].

A	b	B	b	A	b	B	b	b	A	b	B	b	A	b	B
b	C	b	D	b	C	b	D	C	b	D	b	C	b	D	b
B	b	A	b	B	b	A	b	b	B	b	A	b	B	b	A
b	D	b	C	b	D	b	C	D	b	C	b	D	b	C	b
A	b	B	b	A	b	B	b	b	A	b	B	b	A	b	B
b	C	b	D	b	C	b	D	C	b	D	b	C	b	D	b
B	b	A	b	B	b	A	b	b	B	b	A	b	B	b	A
b	D	b	C	b	D	b	C	D	b	C	b	D	b	C	b

(a) labels of cells of group-1

(b) labels of cells of group-2

Σχήμα 3.15: Επιγραφές Κελιών με τη Μέθοδο των Δύο Ομάδων.

ΚΕΦΑΛΑΙΟ 4

ΑΛΓΟΡΙΘΜΟΙ ΓΙΑ ΤΗ ΔΟΚΙΜΗ ΤΗΣ ΘΕΡΜΙΚΗΣ ΑΛΛΗΛΕΠΙΔΡΑΣΗΣ ΣΤΙΣ ΜΝΗΜΕΣ PCM

4.1 Τυπική Δοκιμή Μνημών PCM

4.2 Ενισχυμένη Δοκιμή Μνημών PCM

4.3 Προηγμένη Δοκιμή Μνημών PCM

Προκειμένου να ανιχνευθεί το φαινόμενο της θερμικής αλληλεπίδρασης στις μνήμες PCM είναι απαραίτητο τα κελιά μιας γειτονιάς να μεταβούν από κατάσταση SET σε κατάσταση RESET διαδοχικά, με το κελί βάσης να είναι σε κατάσταση RESET ή SET και έπειτα να διαβασθεί το κελί βάσης. Αν το κελί βάσης έχει αλλάξει κατάσταση τότε αυτό σημαίνει ότι η θερμική αλληλεπίδραση επηρέασε τα δεδομένα στο κελί βάσης. Σε μία γειτονιά Τύπου-1, στην πιο ακραία περίπτωση απαιτείται να ενεργοποιηθούν διαδοχικά τουλάχιστον τέσσερα κελιά της γειτονιάς (δηλ. να μεταβούν από κατάσταση SET σε κατάσταση RESET), έτσι ώστε να εκκληθεί η απαραίτητη θερμότητα και να αλλοιωθεί η τιμή που είναι αποθηκευμένη στο κελί βάσης. Σε σπάνιες περιπτώσεις ακόμα και το RESET ενός μόνο κελιού στη γειτονιά αρκεί για επηρεάσει το κελί βάσης. Ακολούθως, παραθέτουμε νέους αλγορίθμους για τον έλεγχο των κελιών που έχουν επηρεαστεί από το φαινόμενο της θερμικής αλληλεπίδρασης στις μνήμες PCM, με την ενεργοποίηση ενός ή δύο

κελιών (τυπική δοκιμή - typical testing), τριών κελιών (ενισχυμένη δοκιμή - enhanced testing) και τεσσάρων κελιών (προηγμένη δοκιμή - advanced testing). Για τους αλγορίθμους που παρουσιάζουμε παρακάτω θεωρούμε συστοιχία μνήμης $M \times N$, M γραμμών και N στηλών.

4.1 *Τυπική Δοκιμή Μνημών PCM*

Για την ανίχνευση της θερμικής αλληλεπίδρασης στην περίπτωση που ένα ή δύο κελιά μεταβαίνουν από κατάσταση SET σε κατάσταση RESET στη γειτονιά, μπορούμε να χρησιμοποιήσουμε είτε την μέθοδο πλακιδίων, είτε τη μέθοδο των δύο ομάδων, τις οποίες παρουσιάσαμε στο προηγούμενο κεφάλαιο.

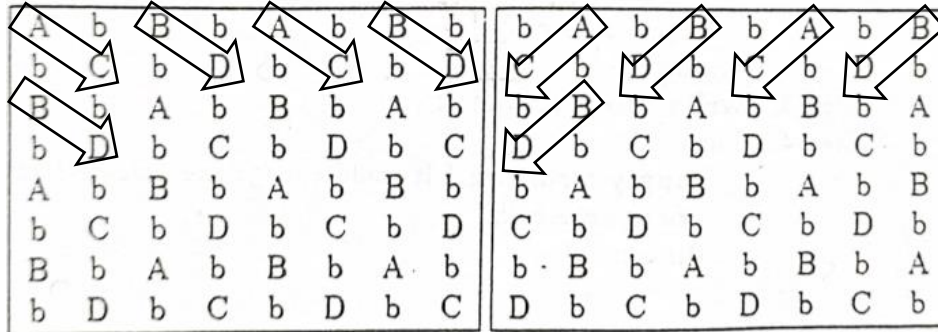
Πιο συγκεκριμένα, για την εφαρμογή των μεθόδων αρχικά θέτουμε σε κατάσταση SET όλα τα κελιά στη μνήμη και ακολούθως μεταθέτουμε ανάλογα με την περίπτωση σε κατάσταση RESET ένα ή δύο (διαδοχικά το ένα μετά το άλλο) κελιά σε κάθε γειτονιά. Στην περίπτωση της μεθόδου των πλακιδίων, η γειτονιά είναι τα κελιά 0, 1, 3, 4 όπως φαίνονται στο Σχήμα 3.13, ενώ στην περίπτωση της μεθόδου των δύο ομάδων η γειτονιά είναι τα κελιά A, B, C, D όπως φαίνονται στο Σχήμα 3.15. Τέλος, διαβάζουμε όλα τα κελιά βάσης (base cells) και αν η κατάστασή τους έχει αλλάξει από SET σε RESET τότε το κελί έχει επηρεαστεί από τη θερμική αλληλεπίδραση. Στην περίπτωση της δοκιμής ενός κελιού θα πραγματοποιηθούν αρχικά $M \times N$ διεργασίες SET στη μνήμη, έπειτα θα γίνουν $0.2 \cdot M \times N$ διεργασίες RESET, εφόσον κάνουμε RESET ένα μόνο κελί στην κάθε γειτονιά, και μετά $0.5 \cdot M \times N$ αναγνώσεις (READ) για να διαβάσουμε όλα τα κελιά βάσης. Συνεπώς η πολυπλοκότητα του σχετικού αλγορίθμου δοκιμής θα είναι $(2 \cdot 1.7) \cdot (M \times N)$ προσβάσεις στη μνήμη (και για τα δύο group). Ομοίως, για την περίπτωση που κάνουμε δοκιμή δύο κελιών στη γειτονιά, η πολυπλοκότητα θα είναι $M \times N$ SET και $0.4 \cdot M \times N$ RESET καθώς και $0.5 \cdot M \times N$ READ, άρα συνολικά $(2 \cdot 1.9) / 5 \cdot (M \times N)$ προσβάσεις στη μνήμη.

4.2 Ενισχυμένη Δοκιμή Μνημών PCM

Στην περίπτωση διαδοχικής ενεργοποίησης τριών κελιών στη γειτονιά, προτείνουμε τη χρήση της μεθόδου των δύο ομάδων που παρουσιάστηκε στο προηγούμενο κεφάλαιο και ένα νέο σχήμα διευθυνσιοδότησης για τη διενέργεια των μεταβάσεων σε κατάσταση RESET στα σχετικά κελιά, όπως παρουσιάζεται στο Σχήμα 4.1. Σε αυτό το σχήμα, τα κελιά θύματα (βάσης) είναι τα κελιά b, ενώ τα κελιά θύτες είναι τα κελιά A, B, C, D. Στόχος είναι να εκτελεστούν μεταβάσεις RESET σε τρία από τα κελιά της γειτονιάς και έπειτα να διαβαστούν όλα τα κελιά τύπου b, προκειμένου να ανιχνεύσουμε περιπτώσεις θερμικής αλληλεπίδρασης στη μνήμη.

Οι Αλγόριθμοι 4.1 και 4.2 παρουσιάζουν το ψευδοκώδικα της διαδικασίας που περιγράψαμε παραπάνω για το group-1 και το group-2 της μεθόδου των δύο ομάδων. Στους παρακάτω αλγορίθμους θεωρούμε μια συστοιχία μνήμης με M γραμμές και N στήλες. Η ακόλουθη περιγραφή του αλγορίθμου αφορά το group-1, ωστόσο με τις ίδιες ενέργειες πραγματοποιούνται οι διεργασίες και στο group-2. Έτσι, αρχικά θέτουμε σε κατάσταση SET όλα τα κελιά της μνήμης (γραμμές 1-5), και έπειτα θέτουμε διαδοχικά σε κατάσταση RESET τα κελιά κατά μήκος δύο διαδοχικών γραμμών (γραμμές 6-16), όπως φαίνεται στο Σχήμα 4.1 (δηλ. RESET A, C, B, D κτλ.). Ύστερα από τη μετάβαση σε κατάσταση RESET των υποδεικνυόμενων κελιών της πρώτης και της δεύτερης γραμμής, έχουν καλυφθεί όλα τα κελιά τύπου b της πρώτης γραμμής και όλα της δεύτερης γραμμής, εκτός του πρώτου (δηλ. του κελιού της δεύτερης γραμμής και της πρώτης στήλης), καθώς δεν έχουν μεταβεί σε κατάσταση RESET τρία κελιά στην γειτονιά του. Συνεπώς, έτσι έχουν καλυφθεί όλα τα b κελιά, εκτός της πρώτης και τελευταίας στήλης (έχει όμως καλυφθεί το τελευταίο κελί τύπου b της πρώτης στήλης και το πρώτο κελί τύπου b της τελευταίας στήλης, όντας γωνιακά). Ακολουθεί η ανάγνωση των κελιών τύπου b (γραμμές 17-34). Για τα εναπομείναντα μη δοκιμασμένα κελιά τύπου b (της πρώτης και της τελευταίας στήλης) θα πρέπει να τεθούν σε κατάσταση SET τα κελιά A, B, C, D των δύο πρώτων και των δύο τελευταίων στηλών (γραμμές 35-56) καθώς στην προηγούμενη φάση είχαν μεταβεί σε κατάσταση RESET. Τέλος, για να καλύψουμε τα κελιά τύπου b της πρώτης στήλης, θέτουμε σε κατάσταση RESET τα κελιά A, C, B, D της πρώτης και δεύτερης στήλης (γραμμές 57-78) εκτελώντας προσπέλαση κατά την κάθετη φορά. Όμοια για την τελευταία στήλη.

Ακολούθως, διαβάζουμε τα κελιά τύπου b της πρώτης και τελευταίας στήλης (γραμμές 78-84).



(a) labels of cells of group-1 (b) labels of cells of group-2

Σχήμα 4.1: Σχήμα Διευθυνσιοδότησης για τη Μετάβαση Τριών Κελιών.

Σε κάθε αλγόριθμο (δηλ. κάθε group) γίνονται $1.5 \cdot M \times N + 4 \cdot M$ λειτουργίες εγγραφής και $0.5 \cdot M \times N$ λειτουργίες ανάγνωσης για να διαβάσουμε όλα τα b κελιά οπότε η συνολική πολυπλοκότητα είναι $2 \cdot M \times N + 4 \cdot M$ για το κάθε ένα group. Η πολυπλοκότητα προκύπτει ως εξής: Αρχικά γίνονται $M \times N$ λειτουργίες SET για να κάνουμε SET όλα τα κελιά της μνήμης. Έπειτα γίνονται $0.5 \cdot M \times N$ λειτουργίες RESET για να κάνουμε RESET τα κελιά της γειτονιάς A, B, C, D, με τον τρόπο που περιγράψαμε παραπάνω, αφού αυτά κατέχουν τις μισές θέσεις στον πίνακα μνήμης. Απαιτούνται $2 \cdot M$ λειτουργίες SET για τα κελιά της γειτονιάς που περιβάλλουν τα b κελιά της πρώτης και τελευταίας στήλης και αντίστοιχα $2 \cdot M$ λειτουργίες RESET για τα ίδια κελιά. Τέλος, γίνονται $0.5 \cdot M \times N$ λειτουργίες ανάγνωσης στα κελιά b, αφού αυτά κατέχουν τις μισές θέσεις στον πίνακα μνήμης. Άρα συνολικά για τους αλγορίθμους και των δύο groups της μεθόδου two-group, η συνολική πολυπλοκότητα είναι $2 \cdot [2 \cdot M \times N + 4 \cdot M] = 4 \cdot M \times N + 8 \cdot M$. Πιο αναλυτικά απαιτούνται $2 \cdot M \times N + 4 \cdot M$ λειτουργίες SET, $1 \cdot M \times N + 4 \cdot M$ λειτουργίες RESET και $M \times N$ λειτουργίες ανάγνωσης συνολικά.


```
1: for ( $i = 0; i < M; i ++$ )
2:   for ( $j = 0; j < N; j ++$ )
3:     SET cell( $i, j$ )
4:   end for
5: end for
6: for ( $i = 0; i < M; i += 2$ )
7:   for ( $j = 0; j < N; j ++$ )
8:     if  $j$  is even then
9:       RESET cell( $i, j$ )
10:       $i \leftarrow i + 1$ 
11:    else
12:      RESET cell( $i, j$ )
13:       $i \leftarrow i - 1$ 
14:    end if
15:  end for
16: end for
17: for ( $i = 0; i < M; i ++$ )
18:   for ( $j = 1; j < N - 1; j += 2$ )
19:     if  $i$  is even then
20:       if  $j == N - 3$  and  $i == 0$  then
21:         READ cell( $i, j+1$ )
22:         READ cell( $i, j$ )
23:       else
24:         READ cell( $i, j$ )
25:       end if
26:     else
27:       if  $i == M - 1$  and  $j == 1$  then
28:         READ cell( $i, j-1$ )
29:         READ cell( $i, j+1$ )
30:       else
31:         READ cell( $i, j+1$ )
```

```
31:     end if
32:     end if
33:   end for
34: end for
35: for ( $i = 0; i < M - 1; i += 2$ )
36:   for ( $j = 0; j < 2; j ++$ )
37:     if  $j$  is even then
38:       SET cell( $i, j$ )
39:        $i \leftarrow i + 1$ 
40:     else
41:       SET cell( $i, j$ )
42:        $i \leftarrow i - 1$ 
43:     end if
44:   end for
45: end for
46: for ( $i = 1; i < M - 1; i += 2$ )
47:   for ( $j = N - 1; j > N - 3; j --$ )
48:     if  $j$  is odd then
49:       SET cell( $i, j$ )
50:        $i \leftarrow i + 1$ 
51:     else
52:       SET cell( $i, j$ )
53:        $i \leftarrow i - 1$ 
54:     end if
55:   end for
56: end for
57: for ( $i = 0; i < M - 1; i += 2$ )
58:   for ( $j = 0; j < 2; j ++$ )
59:     if  $j$  is even then
60:       RESET cell( $i, j$ )
61:        $i \leftarrow i + 1$ 
```

```

62:     else
63:         RESET cell(i, j)
64:          $i \leftarrow i - 1$ 
65:     end if
66: end for
67: end for
68: for ( $i = 1; i < M; i += 2$ )
69:     for ( $j = N - 1; j > N - 3; j --$ )
70:         if  $j$  is even then
71:             RESET cell(i, j)
72:              $i \leftarrow i + 1$ 
73:         else
74:             RESET cell(i, j)
75:              $i \leftarrow i - 1$ 
76:         end if
77:     end for
78: end for
79: for ( $i = 1; i < M - 1; i += 2$ )
80:     READ cell(i, 0)
81: end for
82: for ( $i = 2; i < M; i += 2$ )
83:     READ cell(i, c-1)
84: end for

```

Αλγόριθμος 4.2 3-cell transition PCM Testing (Group-2)

```

1: for ( $i = 0; i < M; i ++$ )
2:     for ( $j = 0; j < N; j ++$ )
3:         SET cell(i, j)
4:     end for
5: end for
6: for ( $i = 0; i < M; i += 2$ )
7:     for ( $j = N - 1; j > -1; j --$ )

```

```
8:   if  $j$  is odd then
9:     RESET cell( $i, j$ )
10:     $i \leftarrow i + 1$ 
11:   else
12:     RESET cell( $i, j$ )
13:     $i \leftarrow i - 1$ 
14:   end if
15: end for
16: end for
17: for ( $i = 0; i < M; i ++$ )
18:   for ( $j = 1; j < N - 1; j += 2$ )
19:     if  $i$  is even then
20:       if  $j = 1$  and  $i == 0$  then
21:         READ cell( $i, j+1$ )
22:         READ cell( $i, j-1$ )
23:       else
24:         READ cell( $i, j+1$ )
25:       end if
26:     else
27:       if  $i == M - 1$  and  $j == N - 3$  then
28:         READ cell( $i, j$ )
29:         READ cell( $i, j+1$ )
30:       else
31:         READ cell( $i, j$ )
32:       end if
33:     end if
34:   end for
35: end for
36: for ( $i = 0; i < M - 1; i += 2$ )
37:   for ( $j = N - 1; j > N - 3; j --$ )
38:     if  $j$  is odd then
```

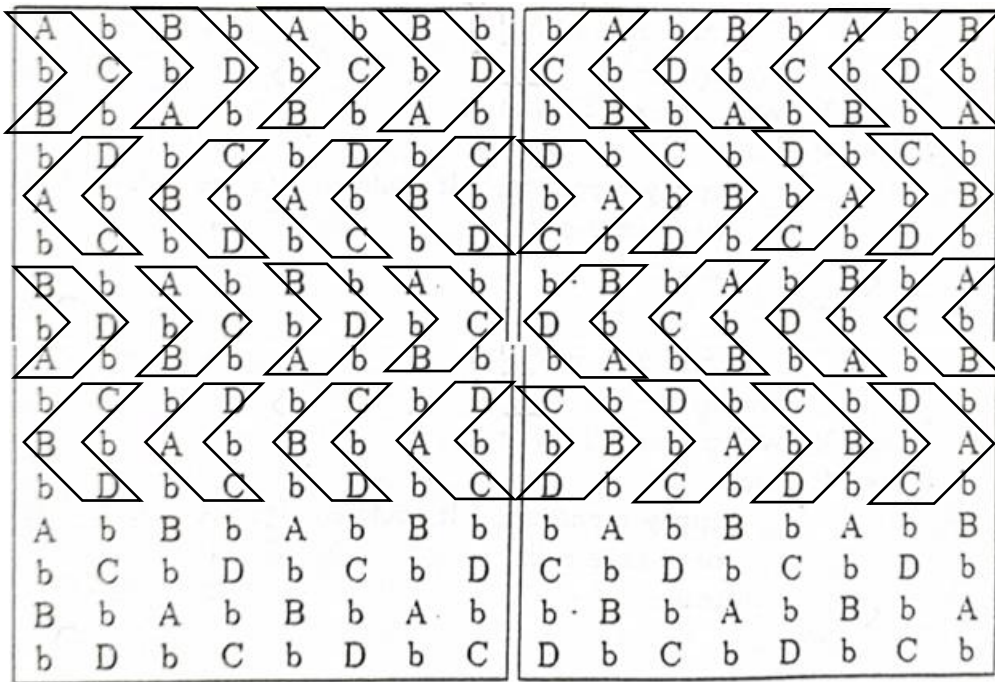
```
39:     SET cell(i, j)
40:      $i \leftarrow i + 1$ 
41:     else
42:     SET cell(i, j)
43:      $i \leftarrow i - 1$ 
44:     end if
45:   end for
46: end for
47: for ( $i = 1; i < M - 1; i += 2$ )
48:   for ( $j = 0; j < 2; j ++$ )
49:     if  $j$  is even then
50:       SET cell(i, j)
51:        $i \leftarrow i + 1$ 
52:     else
53:       SET cell(i, j)
54:        $i \leftarrow i - 1$ 
55:     end if
56:   end for
57: end for
58: for ( $i = 0; i < M - 1; i += 2$ )
59:   for ( $j = N - 1; j > N - 3; j --$ )
60:     if  $j$  is odd then
61:       RESET cell(i, j)
62:        $i \leftarrow i + 1$ 
63:     else
64:       RESET cell(i, j)
65:        $i \leftarrow i - 1$ 
66:     end if
67:   end for
68: end for
69: for ( $i = 1; i < M; i += 2$ )
70:   for ( $j = 0; j < 2; j ++$ )
```

```
71:   if  $j$  is even then
72:     RESET cell( $i, j$ )
73:      $i \leftarrow i + 1$ 
74:   else
75:     RESET cell( $i, j$ )
76:      $i \leftarrow i - 1$ 
77:   end if
78: end for
79: end for
80: for ( $i = 2; i < M - 1; i += 2$ )
81:   READ cell( $i, 0$ )
82: end for
83: for ( $i = 1; i < M; i += 2$ )
84:   READ cell( $i, N-1$ )
85: end for
```

4.3 Προηγμένη Δοκιμή Μνημών PCM

Για τη δοκιμή της μνήμης αναφορικά με τη θερμική αλληλεπίδραση με τη διενέργεια τεσσάρων μεταβάσεων σε κατάσταση RESET στα κελιά μιας γειτονιάς, θα αξιοποιήσουμε πάλι τη γενική μέθοδο των δύο ομάδων. Προτείνουμε εκ νέου ένα σχήμα διευθυνσιοδότησης που θα ενεργοποιεί διαδοχικά τα τέσσερα κελιά της γειτονιάς το οποίο παρουσιάζεται στο Σχήμα 4.2. Εποπτικά, εκτελούμε μεταβάσεις στα κελιά της γειτονιάς σε σχήμα «βέλους», οριζόντια σε έκταση τριών γραμμών. Πιο συγκεκριμένα, θέτουμε σε κατάσταση RESET το κελί A της πρώτης γραμμής, ακολούθως το κελί C της δεύτερης γραμμής, ύστερα το κελί B της τρίτης γραμμής και έπειτα συνεχίζουμε στο αμέσως επόμενο βέλος δεξιότερα. Με τη μέθοδο που περιγράψαμε ανωτέρω, εκτελούν διαδοχικά τις απαραίτητες μεταβάσεις και τα τέσσερα κελιά της γειτονιάς ώστε να μεγιστοποιηθεί η θερμική αλληλεπίδραση στο κελί βάσης. Στην επόμενη τριάδα γραμμών η φορά των βελών είναι αντίθετη και η διενέργεια των διεργασιών πραγματοποιείται από δεξιά προς τα αριστερά.

Προκειμένου να περιγραφεί με πληρότητα η λειτουργία των βελών παρουσιάζονται δύο συναρτήσεις `right_arrow(i)` και `left_arrow(i)`, όπου i είναι ο αύξων αριθμός της γραμμής όπου ξεκινάει το βέλος, και οι οποίες εντάσσονται στους Αλγορίθμους 4.3 και 4.4 αντίστοιχα.



(a) labels of cells of group-1 (b) labels of cells of group-2

Σχήμα 4.2: Σχήμα Διευθυνσιοδότησης για τη Μετάβαση Τεσσάρων Κελιών.

Αλγόριθμος 4.3 `right_arrow(i)`

```

1: counter ← 0
2: for (j = 0; j < N; j++)
3:   if j is even then
4:     RESET cell(i, j)
5:     i ← i + 1
6:     counter ← counter + 1
7:   if counter == 3 then
8:     i ← i - 3
9:     j ← j + 1
10:  counter ← 0

```

```
11:   end if
12:   else
13:     RESET cell(i, j)
14:      $i \leftarrow i + 1$ 
15:      $counter \leftarrow counter + 1$ 
16:      $j \leftarrow j - 2$ 
17:   end if
18: end for
```

Αλγόριθμος 4.4 left_arrow(i)

```
1:   $counter \leftarrow 0$ 
2:  for ( $j = 1; j < N; j ++$ )
3:    if  $j$  is odd then
4:      RESET cell(i, j)
5:       $i \leftarrow i + 1$ 
6:       $counter \leftarrow counter + 1$ 
7:       $j \leftarrow j - 2$ 
8:      if  $counter == 3$  then
9:         $i \leftarrow i - 3$ 
10:        $j \leftarrow j + 3$ 
11:        $counter \leftarrow 0$ 
12:     end if
13:   else
14:     RESET cell(i, j)
15:      $i \leftarrow i + 1$ 
16:      $counter \leftarrow counter + 1$ 
17:   end if
18: end for
```

Η πορεία ενεργοποίησης των τεσσάρων κελιών στη γειτονιά που αφορούν το group-1 και το group-2 παρουσιάζεται στους αλγορίθμους 4.5 και 4.6 αντίστοιχα. Στην περίπτωση του group-1, αρχικά θέτουμε σε κατάσταση SET όλα τα κελιά στη μνήμη (γραμμές 1-5). Ακολούθως, εκτελούμε τις απαραίτητες μεταβάσεις σε

κατάσταση RESET χρησιμοποιώντας της συναρτήσεις `left_arrow` και `right_arrow` (γραμμές 6-15). Χρησιμοποιείται ένας σημαφόρος (`right`) που θα προσδιορίσει αν θα κληθεί η `left_arrow` ή η `right_arrow` ανάλογα σε πια γραμμή βρισκόμαστε. Στις γραμμές 16-36 καθορίζονται οι διαδικασίες READ στα κελιά τύπου `b`. Για παράδειγμα, ξεκινώντας από τη πρώτη γραμμή, μεταθέτουμε σε κατάσταση RESET τα κελιά `A`, `B`, `C` και `D` με το σχήμα προσπέλασης που παρουσιάσαμε παραπάνω (δηλ. στην πρώτη γραμμή το κελί `A`, στη δεύτερη γραμμή το κελί `C`, στην τρίτη γραμμή το κελί `B` και ακολούθως στην πρώτη γραμμή το κελί `B`, κ.τ.λ.). Κατά αυτόν τον τρόπο έχουμε καλύψει όλα τα κελιά τύπου `b` της πρώτης και δεύτερης γραμμής, τα κελιά τύπου `b` της πέμπτης γραμμής κ.τ.λ. Τα κελιά τύπου `b` της τρίτης γραμμής για παράδειγμα δεν καλύπτονται και θα πρέπει να επαναλάβουμε την ίδια διαδικασία ξανά εφαρμόζοντας την ίδια λογική και ξεκινώντας από τη δεύτερη γραμμή. Ως εκ τούτου, πρέπει να γίνει εκ νέου SET σε όλα τα κελιά και έπειτα να προχωρήσουμε κάνοντας RESET τα κελιά χρησιμοποιώντας τα κατάλληλα βέλη ξεκινώντας από την επόμενη γραμμή κάθε φορά προκειμένου να καλύψουμε όλα τα `b` κελιά. Ο υπόλοιπος αλγόριθμος περιλαμβάνει την ίδια διαδικασία που περιγράψαμε παραπάνω και απαιτούνται συνολικά τρεις προσπελάσεις για να καλυφθεί όλοι η μνήμη. Συνεπώς, με την πρώτη προσπέλαση, καλύπτουμε τα `b` κελιά της πρώτης, της δεύτερης και της πέμπτης γραμμής. Με τη δεύτερη προσπέλαση, καλύπτουμε τα `b` κελιά της τρίτης και της έκτης γραμμής. Τέλος, με την τρίτη προσπέλαση, καλύπτουμε τα `b` κελιά της τέταρτης, της έβδομης και της όγδοης γραμμής. Γενικά, με την πρώτη προσπέλαση, καλύπτονται τα `b` κελιά της πρώτης, της δεύτερης, της πέμπτης γραμμής κτλ. έως τη γραμμή `M-3`. Με τη δεύτερη προσπέλαση, καλύπτουμε τα `b` κελιά της τρίτης, της έκτης γραμμής κλπ. έως τη γραμμή `M-2`. Με την τρίτη προσπέλαση, καλύπτουμε τα `b` κελιά της τέταρτης, της έβδομης κτλ. έως και τις γραμμές `M-1` και `M`. Για το `group-2` η διαδικασία είναι ίδια αλλά εφαρμόζεται κατά συμπληρωματικό τρόπο, δηλαδή ξεκινάμε την προσπέλαση με αριστερό βέλος αντί για δεξί και συνεχίζουμε με την ανωτέρω μέθοδο.

Για κάθε `group`, απαιτούνται $4.1 \cdot M \times N$ εγγραφές και $0.5 \cdot M \times N$ αναγνώσεις, συνεπώς η πολυπλοκότητα είναι $4.6 \cdot M \times N$ και προκύπτει ως εξής: Αρχικά γίνονται $M \times N$ λειτουργίες SET προκειμένου να γίνουν SET όλα τα κελιά στη μνήμη. Έπειτα

χρειαζόμαστε $0.37 \cdot M \times N$ λειτουργίες RESET για να κάνουμε RESET τα κελιά της γειτονιάς με το σχήμα διευθυνσιοδότησης που περιγράψαμε παραπάνω, δηλαδή με το σχήμα βέλους. Αυτή η διαδικασία γίνεται τρεις φορές για να καλύψουμε όλη τη μνήμη συνεπώς χρειάζονται $4.1 \cdot M \times N$ εγγραφές. Τέλος απαιτούνται $0.5 \cdot M \times N$ αναγνώσεις για να διαβάσουμε όλα τα b κελιά αφού καταλαμβάνουν τις μισές θέσεις της μνήμης. Συνεπώς η συνολική πολυπλοκότητα του αλγορίθμου και για τα δύο groups θα είναι $9.2 \cdot M \times N$. Πιο αναλυτικά, απαιτούνται $6 \cdot M \times N$ λειτουργίες SET, $2.2 \cdot M \times N$ λειτουργίες RESET και $M \times N$ λειτουργίες ανάγνωσης συνολικά.

Αλγόριθμος 4.5 4-cell transition PCM Testing (Group-1)

```

1:  for ( $i = 0; i < M; i ++$ )
2:    for ( $j = 0; j < N; j ++$ )
3:      SET cell( $i, j$ )
4:    end for
5:  end for
6:   $right \leftarrow 0$ 
7:  for ( $i = 0; i < M - 1; i += 3$ )
8:    if  $right == 0$  then
9:      right_arrow( $i$ )
10:      $right \leftarrow 1$ 
11:    else
12:      left_arrow( $i$ )
13:      $right \leftarrow 0$ 
14:    end if
15:  end for
16:  for ( $i = 1; i < M - 1; i += 3$ )
17:    if  $i == 1$  then
18:      for ( $j = 0; j < N; j ++$ )
19:        if  $j$  is even then
20:          READ cell( $i, j$ )
21:        else

```

```
22:     READ cell(i-1, j)
23:     end if
24:     end for
25: else
26:     if  $i$  is even then
27:         for ( $j = 1; j < N; j += 2$ )
28:             READ cell(i, j)
29:         end for
30:     else
31:         for ( $j = 0; j < N; j += 2$ )
32:             READ cell(i, j)
33:         end for
34:     end if
35: end if
36: end for
37: for ( $i = 0; i < M; i ++$ )
38:     for ( $j = 0; j < N; j ++$ )
39:         SET cell(i, j)
40:     end for
41: end for
42:  $right \leftarrow 0$ 
43: for ( $i = 1; i < M - 1; i += 3$ )
44:     if  $right == 0$  then
45:         left_arrow(i)
46:          $right \leftarrow 1$ 
47:     else
48:         right_arrow(i)
49:          $right \leftarrow 0$ 
50:     end if
51: end for
52: for ( $i = 2; i < M - 1; i += 3$ )
```

```
53:  if  $i$  is even then
54:    for ( $j = 1; j < N; j += 2$ )
55:      READ cell( $i, j$ )
56:    end for
57:  else
58:    for ( $j = 0; j < N; j += 2$ )
59:      READ cell( $i, j$ )
60:    end for
61:  end if
62: end for
63: for ( $i = 0; i < M; i ++$ )
64:   for ( $j = 0; j < N; j ++$ )
65:     SET cell( $i, j$ )
66:   end for
67: end for
68: end for
69:  $right \leftarrow 0$ 
70: for ( $i = 3; i < M - 1; i += 3$ )
71:   if  $right == 0$  then
72:     right_arrow( $i$ )
73:      $right \leftarrow 1$ 
74:   else
75:     left_arrow( $i$ )
76:      $right \leftarrow 0$ 
77:   end if
78: end for
79: end for
80: for ( $i = 2; i < M; i += 3$ )
81:   if  $i == M - 2$  then
82:     for ( $j = 0; j < N; j ++$ )
83:       if  $j$  is even then
84:         READ cell( $i+1, j$ )
85:       else
```

```
86:     READ cell(i, j)
87:     end if
88:     end for
89: else
90:     if  $i$  is even then
91:         for ( $j = 1; j < N; j += 2$ )
92:             READ cell(i, j)
93:         end for
94:     else
95:         for ( $j = 0; j < N; j += 2$ )
96:             READ cell(i, j)
97:         end for
98:     end if
99: end if
100: end for
```

Αλγόριθμος 4.6 4-cell transition PCM Testing (Group-2)

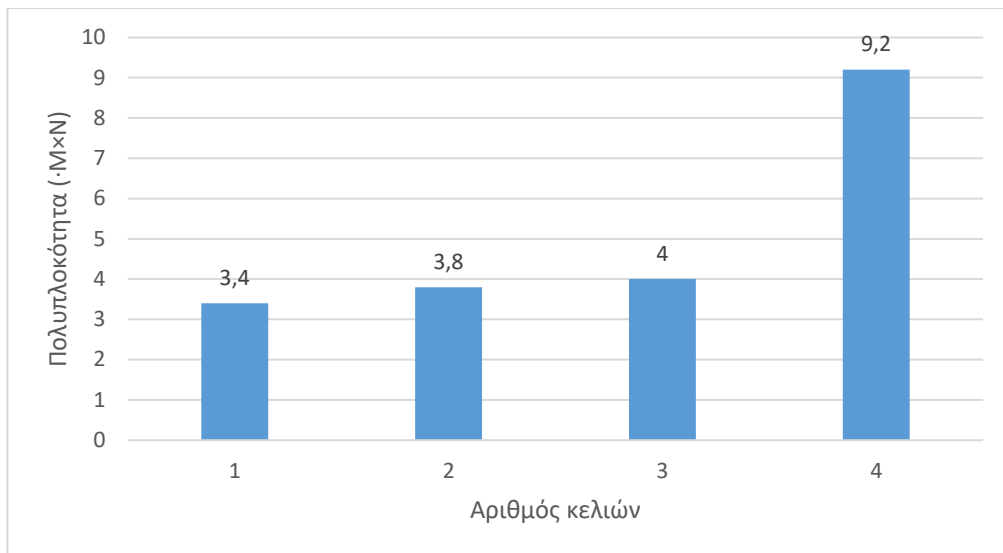
```
1: for ( $i = 0; i < M; i ++$ )
2:     for ( $j = 0; j < N; j ++$ )
3:         SET cell(i, j)
4:     end for
5: end for
6:  $right \leftarrow 0$ 
7: for ( $i = 0; i < M - 1; i += 3$ )
8:     if  $right == 0$  then
9:         left_arrow(i)
10:         $right \leftarrow 1$ 
11:    else
12:        right_arrow(i)
13:         $right \leftarrow 0$ 
14:    end if
```

```
15: end for
16: for ( $i = 1; i < M - 1; i += 3$ )
17:   if  $i == 1$  then
18:     for ( $j = 0; j < N; j ++$ )
19:       if  $j$  is even then
20:         READ cell( $i-1, j$ )
21:       else
22:         READ cell( $i, j$ )
23:       end if
24:     end for
25:   else
26:     if  $i$  is odd then
27:       for ( $j = 1; j < N; j += 2$ )
28:         READ cell( $i, j$ )
29:       end for
30:     else
31:       for ( $j = 0; j < N; j += 2$ )
32:         READ cell( $i, j$ )
33:       end for
34:     end if
35:   end if
36: end for
37: for ( $i = 0; i < M; i ++$ )
38:   for ( $j = 0; j < N; j ++$ )
39:     SET cell( $i, j$ )
40:   end for
41: end for
42:  $right \leftarrow 0$ 
43: for ( $i = 1; i < M - 1; i += 3$ )
44:   if  $right == 0$  then
45:     right_arrow( $i$ )
```

```
46:   right ← 1
47:   else
48:     left_arrow(i)
49:     right ← 0
50:   end if
51: end for
52: for (i = 2; i < M - 1; i += 3)
53:   if i is odd then
54:     for (j = 1; j < N; j += 2)
55:       READ cell(i, j)
56:     end for
57:   else
58:     for (j = 0; j < N; j += 2)
59:       READ cell(i, j)
60:     end for
61:   end if
62: end for
63: for (i = 0; i < M; i ++ )
64:   for (j = 0; j < N; j ++ )
65:     SET cell(i, j)
66:   end for
67: end for
68: end for
69: right ← 0
70: for (i = 2; i < M - 1; i += 3)
71:   if right == 0 then
72:     left_arrow(i)
73:     right ← 1
74:   else
75:     right_arrow(i)
76:     right ← 0
77:   end if
```

```
78: end for
80: for ( $i = 3; i < M; i += 3$ )
81:   if  $i == r - 2$  then
82:     for ( $j = 0; j < N; j ++$ )
83:       if  $j$  is even then
84:         READ cell( $i, j$ )
85:       else
86:         READ cell( $i+1, j$ )
87:       end if
88:     end for
89:   else
90:     if  $i$  is odd then
91:       for ( $j = 1; j < M; j += 2$ )
92:         READ cell( $i, j$ )
93:       end for
94:     else
95:       for ( $j = 0; j < N; j += 2$ )
96:         READ cell( $i, j$ )
97:       end for
98:     end if
99:   end if
100: end for
```

Στο Σχήμα 4.3 παρουσιάζονται συγκεντρωτικά οι πολυπλοκότητες των αλγορίθμων που παρουσιάσαμε παραπάνω, για την μετάβαση από την κατάσταση SET στην κατάσταση RESET, ενός, δύο, τριών ή τεσσάρων κελιών στη γειτονιά. Η περίπτωση του ενός κελιού αντιστοιχεί στην πολυπλοκότητα του κλασικού στη βιβλιογραφία αλγόριθμου για NPSF δοκιμή.



Σχήμα 4.3: Η Πολυπλοκότητα ως Συνάρτηση των Αριθμών των Κελιών που Διαδοχικά Γίνονται RESET σε μια Γειτονιά.

ΚΕΦΑΛΑΙΟ 5

ΣΥΜΠΕΡΑΣΜΑΤΑ

Το φαινόμενο της θερμικής αλληλεπίδρασης αποτελεί ένα σημαντικό ζήτημα αξιοπιστίας στις μνήμες PCM καθώς η τεχνολογία κλιμακώνεται στην περιοχή των νανομέτρων και τα κελιά βρίσκονται σε μικρές αποστάσεις μεταξύ τους. Είναι το φαινόμενο κατά το οποίο όταν προγραμματίζεται ένα κελί, τα γειτονικά κελιά μπορεί να υποστούν παρασιτική θέρμανση, η οποία μπορεί να οδηγήσει σε απώλεια δεδομένων. Αν και στη διεθνή βιβλιογραφία έχουν προταθεί αλγόριθμοι δοκιμής των μνημών PCM, η δοκιμή για την ανίχνευση μεταβάσεων ενός κελιού από την κατάσταση SET στην κατάσταση RESET εξαιτίας της θερμικής αλληλεπίδρασης δεν έχει μελετηθεί μέχρι τώρα. Μέχρι σήμερα έχουν προταθεί στη διεθνή βιβλιογραφία μόνο αλγόριθμοι πορείας (March) για τη δοκιμή των μνημών PCM. Στην παρούσα εργασία, παρουσιάζονται για πρώτη φορά αλγόριθμοι δοκιμής με αυξημένη κάλυψη σφαλμάτων βασισμένοι στη φιλοσοφία του NPSF μοντέλου σφαλμάτων για τις μνήμες PCM.

Η εγγύτητα των κελιών στον πίνακα της μνήμης και η ενδεχόμενη διαδοχική ενεργοποίηση πολλαπλών κελιών στη γειτονιά ενός κελιού μπορεί να οδηγήσει στη σημαντική έκκλιση θερμότητας, η οποία μέσω της θερμικής αλληλεπίδρασης ενδέχεται να προκαλέσει τη μετάβαση ενός κελιού θύματος από την κατάσταση SET στην κατάσταση RESET. Στην παρούσα διατριβή παρουσιάζουμε για πρώτη φορά αλγορίθμους δοκιμής με σκοπό την ανίχνευση των μεταβάσεων από την

κατάσταση SET στην κατάσταση RESET στα κελιά θύματα, ενεργοποιώντας αποδοτικά (διαδοχικά στο χρόνο) από δυο έως τέσσερα κελιά σε μια γειτονιά Τύπου-1 (σύμφωνα με το μοντέλο σφαλμάτων NPSF). Σε αντίθεση με του παραδοσιακούς αλγορίθμους δοκιμής NPSF, όπου εφαρμόζεται ένα μοτίβο και αλλάζει κατάσταση μόνο ένα κελί στη γειτονιά πριν διαβαστεί το κελί βάσης, στους αλγορίθμους NPSF που προτείναμε, γίνονται διαδοχικές στο χρόνο μεταβάσεις από την κατάσταση SET στην κατάσταση RESET σε δύο, τρία ή τέσσερα κελιά στην γειτονιά πριν διαβαστεί το κελί βάσης. Για την ενεργοποίηση ενός ή δύο κελιών στη γειτονιά, υιοθετούμε είτε τη μέθοδο πλακιδίων είτε τη μέθοδο των δύο ομάδων με πολυπλοκότητες $3.4 \cdot (M \times N)$ και $3.8 \cdot (M \times N)$ για ένα και δύο κελιά, αντίστοιχα, όπου $M \times N$ το μέγεθος της μνήμης σε γραμμές επί στήλες. Για την ενεργοποίηση τριών ή τεσσάρων κελιών στη γειτονιά, προτείνουμε δύο νέους αλγορίθμους βασισμένους στη μέθοδο των δύο ομάδων, παρουσιάζοντας δυο νέα σχήματα διευθυνσιοδότησης με πολυπλοκότητες $4 \cdot M \times N + 8M$ και $9.2 \cdot M \times N$ αντίστοιχα. Με την προσθήκη των δύο νέων αλγορίθμων για τη δοκιμή των μνημών PCM, πετυχαίνουμε να εκθέσουμε τα κελιά μιας μνήμης PCM στη μέγιστη παρασιτική θέρμανση και έτσι να ελέγξουμε όλα τα κελιά αποδοτικά ως προς την επιρροή του φαινομένου της θερμικής αλληλεπίδρασης.

ΒΙΒΛΙΟΓΡΑΦΙΑ

- [1] SR. Ovshinsky, 'Reversible electrical switching phenomena in disorder structures', *Phys. Rev. Lett.*, vol. 21, no. 20, pp. 1450-1453, 1968, doi: 10.1103/PhysRevLett.21.1450.
- [2] A.L. Lacaita, 'Phase change memories: State-of-the-art, challenges and perspectives', *Solid-State Electronics*, vol. 50, no. 1, pp. 24-31, 2006, doi: 10.1016/j.sse.2005.10.046.
- [3] S. Lai and T. Lowrey, 'OUM - A 180 nm nonvolatile memory cell element technology for stand alone and embedded applications', in *International Electron Devices Meeting. Technical Digest (Cat. No.01CH37224)*, 2001, pp. 36.5.1-36.5.4, doi: 10.1109/IEDM.2001.979636.
- [4] M. Gill, T. Lowrey and J. Park, 'Ovonic unified memory - a high-performance nonvolatile memory technology for stand-alone memory and embedded applications', in *2002 IEEE International Solid-State Circuits Conference. Digest of Technical Papers (Cat. No.02CH37315)*, 2002, pp. 202-459, vol.1, doi: 10.1109/ISSCC.2002.993006.
- [5] Y. N. Hwang et al., 'Full integration and reliability evaluation of phase-change RAM based on 0.24 μm -CMOS technologies', in *2003 Symposium on VLSI Technology. Digest of Technical Papers (IEEE Cat. No.03CH37407)*, 2003, pp. 173-174, doi: 10.1109/VLSIT.2003.1221141.
- [6] S. J. Ahn et al., 'Highly manufacturable high density phase change memory of 64Mb and beyond', in *IEDM Technical Digest. IEEE International Electron Devices Meeting*, 2004, pp. 907-910, doi: 10.1109/IEDM.2004.1419329.

- [7] F. Pellizzer et al., 'Novel /spl mu/trench phase-change memory cell for embedded and stand-alone non-volatile memory applications', in *Digest of Technical Papers. 2004 Symposium on VLSI Technology*, 2004, pp. 18-19, doi: 10.1109/VLSIT.2004.1345368.
- [8] N. Mielke et al., 'Reliability of ovonic unified memory', in *5th Top. Res. Conf. Reliability*, 2002.
- [9] A. Pirovano et al., 'Reliability study of phase-change nonvolatile memories', *IEEE Transactions on Device and Materials Reliability*, vol. 4, no. 3, pp. 422-427, 2004, doi: 10.1109/TDMR.2004.836724.
- [10] Kinarn Kim and Su Jin Ahn, 'Reliability investigations for manufacturable high density PRAM', in *2005 IEEE International Reliability Physics Symposium, 2005. Proceedings. 43rd Annual.*, 2005, pp. 157-162, doi: 10.1109/RELPHY.2005.1493077.
- [11] A. Pirovano, A. L. Lacaita, A. Benvenuti, F. Pellizzer, S. Hudgens and R. Bez, 'Scaling analysis of phase-change memory technology', in *IEEE International Electron Devices Meeting 2003*, 2003, pp. 29.6.1-29.6.4, doi: 10.1109/IEDM.2003.1269376.
- [12] M. Le Gallo and A. Sebastian, 'An overview of phase-change memory device physics', *Journal of Physics D: Applied Physics*, vol. 53, no. 21, 2020, doi: 10.1088/1361-6463/ab7794.
- [13] O. Mutlu et al., 'Processing data where it makes sense: Enabling in-memory computation', in *Proceedings of the 2019 on Great Lakes Symposium on VLSI (GLSVLSI '19)*, 2019, pp. 5-6, doi: 10.1145/3299874.3322805
- [14] G. W. Burr et al., 'Recent Progress in Phase-Change Memory Technology', *IEEE Journal on Emerging and Selected Topics in Circuits and Systems*, vol. 6, no. 2, pp. 146-162, 2016, doi: 10.1109/JETCAS.2016.2547718.
- [15] H. S. P. Wong et al., 'Phase Change Memory', *Proceedings of the IEEE*, vol. 98, no. 12, pp. 2201-2227, 2010, doi: 10.1109/JPROC.2010.2070050.

- [16] G. W. Burr et al., 'Recent Progress in Phase-Change Memory Technology', *IEEE Journal on Emerging and Selected Topics in Circuits and Systems*, vol. 6, no. 2, pp. 146-162, 2016, doi: 10.1109/JETCAS.2016.2547718.
- [17] C. Popescu, 'The effect of local non-uniformities on thermal switching and high field behaviour of structures with chalcogenide glasses', *Solid-State Electronics*, vol. 18, no. 7, pp. 671-681, 1975, doi: 10.1016/0038-1101(75)90139-2.
- [18] A. E. Owen, J. M. Robertson and C. Main, 'The threshold characteristics of chalcogenide-glass memory switches', *Journal of Non-Crystalline Solids*, vol. 32, no. 1, pp. 29-52, 1979, doi: 10.1016/0022-3093(79)90063-2.
- [19] D. Adler, H. K. Henisch and N. Mott, 'The mechanism of threshold switching in amorphous alloys', *Rev. Mod. Phys.*, vol. 50, no. 2, pp. 209-220, doi: 10.1103/RevModPhys.50.209.
- [20] M. G. Mohammad, L. Terkawi and M. Albasman, 'Phase change memory faults', in *19th International Conference on VLSI Design held jointly with 5th International Conference on Embedded Systems Design (VLSID'06)*, 2006, pp. 6, doi: 10.1109/VLSID.2006.134.
- [21] A. Pirovano et al., 'Electronic switching in phase-change memories', *IEEE Transactions on Electron Devices*, vol. 51, no. 3, pp. 452-459, 2004, doi: 10.1109/TED.2003.823243
- [22] M. Kastner, D. Adler and H. Fritzsche, 'Valence-Alternation Model for Localized Gap States in Lone-Pair Semiconductors', *Phys. Rev. Lett.*, vol. 37, no. 22, pp. 1504-1507, 1976, doi: 10.1103/PhysRevLett.37.1504.
- [23] M. Kastner, 'Bonding Bands, Lone-Pair Bands, and Impurity States in Chalcogenide Semiconductors', *Phys. Rev. Lett.*, vol. 28, no. 6, pp. 355-357, 1972, doi: 10.1103/PhysRevLett.28.355.
- [24] M. Alfailakawi, 'Fault model and test procedure for phase change memory', *Computers & Digital Techniques, IET.*, vol. 5, pp. 263-270, 2011, doi: 10.1049/iet-cdt.2010.0083.

- [25] W. Zhang and E. Ma, ‘Unveiling the structural origin to control resistance drift in phase-change memory materials’, *Materials Today*, vol. 41, pp. 156-176, 2020, doi: 10.1016/j.mattod.2020.07.016
- [26] A. Pirovano et al., ‘Scaling Analysis of Phase-Change Memory Technology’, in *Technical Digest - International Electron Devices Meeting*, 2004, pp. 29.6.1-29.6.4, doi: 10.1109/IEDM.2003.1269376.
- [27] P. Xiujuan et al., ‘Modeling and test for parasitic resistance and capacitance defects in PCM’, in *Non-Volatile Memory Technology Symposium (NVMTS)*, 2012, pp. 73-76, doi: 10.1109/NVMTS.2013.6632866.
- [28] Z. Xinxing et al., ‘Critical Parasitic Capacitance in nano-scale phase-change memory (PCM) Cell’, in *IEEE International Conference on Electron Devices and Solid-State Circuits, EDSSC 2014*, 2014, pp. 1-2, 10.1109/EDSSC.2014.7061264.
- [29] D. Ielmini, A. L. Lacaita and D. Mantegazza, ‘Recovery and Drift Dynamics of Resistance and Threshold Voltages in Phase-Change Memories’, *Electron Devices, IEEE Transactions on*, vol. 54, no. 2, pp. 308 – 315, 2007, doi: 10.1109/TED.2006.888752.
- [30] K. Osada et al., ‘Phase change RAM operated with 1.5-V CMOS as low cost embedded memory’, in *Proceedings of the IEEE 2005 Custom Integrated Circuits Conference*, 2005, pp. 431-434, doi: 10.1109/CICC.2005.1568698.
- [31] A. Pirovano et al., ‘Reliability Study of Phase-Change Nonvolatile Memories, Device and Materials Reliability, *IEEE Transactions on*, vol. 4, pp. 422 – 427, 2010, doi: 10.1109/TDMR.2004.836724.
- [32] J. D. Maimon, K. K. Hunt, L. Burcin and J. Rodgers, ‘Chalcogenide memory arrays: characterization and radiation effects’, *IEEE Transactions on Nuclear Science*, vol. 50, no. 6, pp. 1878-1884, 2003, doi: 10.1109/TNS.2003.821377.
- [33] C. Xiaole et al., ‘A snake addressing scheme for phase change memory testing’, *Science China Information Sciences*, vol. 59, no. 10, pp. 1-11, 2016, doi: 10.1007/s11432-015-5437-0.

- [34] A. J. van de Goor and Z. Al-Ars, 'Functional memory faults: a formal notation and a taxonomy', in *Proceedings 18th IEEE VLSI Test Symposium*, 2000, pp. 281-289, doi: 10.1109/VTEST.2000.843856.
- [35] D. Ielmini et al., 'Temperature acceleration of structural relaxation in amorphous $\text{Ge}_2\text{Sb}_2\text{Te}_5$ ', *Applied Physics Letters*, vol. 92, no. 19, pp. 193511 2008 doi: 10.1063/1.2930680.
- [36] S. Kim et al., 'Thermal disturbance and its impact on reliability of phase-change memory studied by the micro-thermal stage', in *2010 IEEE International Reliability Physics Symposium*, 2010, pp. 99-103, doi: 10.1109/IRPS.2010.5488847.
- [37] S. Kim and H. -S. P. Wong, 'Analysis of Temperature in Phase Change Memory Scaling', *IEEE Electron Device Letters*, vol. 28, no. 8, pp. 697-699, 2007, doi: 10.1109/LED.2007.901347.
- [38] U. Russo et al., 'Modeling of Programming and Read Performance in Phase-Change Memories—Part II: Program Disturb and Mixed-Scaling Approach', *IEEE Transactions on Electron Devices*, vol. 55, no. 2, pp. 515-522, 2008, doi: 10.1109/TED.2007.913573.
- [39] L. Jiang, Y. Zhang and J. Yang, 'Mitigating Write Disturbance in Super-Dense Phase Change Memories', in *2014 44th Annual IEEE/IFIP International Conference on Dependable Systems and Networks*, 2014, pp. 216-227, doi: 10.1109/DSN.2014.32.
- [40] S. M. Seyedzadeh, A. K. Jones and R. Melhem, 'Improving Sustainability Through Disturbance Crosstalk Mitigation in Deeply Scaled Phase-change Memory', in *2018 Ninth International Green and Sustainable Computing Conference (IGSC)*, 2018, pp. 1-8, doi: 10.1109/IGCC.2018.8752107.
- [41] U. Russo, D. Ielmini, A. Redaelli and A. L. Lacaita, 'Modeling of Programming and Read Performance in Phase-Change Memories—Part I: Cell Optimization

and Scaling', *IEEE Transactions on Electron Devices*, vol. 55, no. 2, pp. 506-514, 2008, doi: 10.1109/TED.2007.911630.

[42] B. Kim et al., 'Current status and future prospect of Phase Change Memory', in *Proceedings of International Conference on ASIC*, 2011, pp. 279-282, 10.1109/ASICON.2011.6157176.

[43] S. J. Ahn et al., 'Reliability perspectives for high density PRAM manufacturing', in *2011 International Electron Devices Meeting*, 2011, pp. 12.6.1-12.6.4, doi: 10.1109/IEDM.2011.6131542.

[44] Z. Zhang et al., 'Memory module-level testing and error behaviors for phase change memory', in *2012 IEEE 30th International Conference on Computer Design (ICCD)*, 2012, pp. 358-363, doi: 10.1109/ICCD.2012.6378664.

[45] S. Kannan et al., 'Sneak-Path Testing of Crossbar-Based Nonvolatile Random Access Memories', *IEEE Transactions on Nanotechnology*, vol. 12, no. 3, pp. 413-426, 2013, doi: 10.1109/TNANO.2013.2253329.

[46] A. J. van de Goor, *Testing semiconductor memories: theory and practice.*, John Wiley & Sons Inc., 1991.

ΕΥΡΕΤΗΡΙΟ

- Χαλκογονίδιο, 3
- Μνήμη Αλλαγής Φάσης, 3
- Κατάσταση SET, 5
- Κατάσταση RESET, 5
- Γραμμή bit (bitline-BL), 6
- Γραμμή λέξης (wordline-WL), 6
- Δομή έκκλισης θερμότητας (heater), 7
- Μοντέλα σφαλμάτων PCM, 20
- Θερμική αλληλεπίδραση, 21
- Διαταραχή εγγραφής, 23
- Μοντέλο Σφαλμάτων Ευαισθησίας στο Μοτίβο της Γειτονιάς (NPSF), 32
- Μέθοδος πλακιδίων, 38
- Μέθοδος των δύο ομάδων, 40

ΣΥΝΤΟΜΟ ΒΙΟΓΡΑΦΙΚΟ

Ο κ. Σπυρίδων Σπυρίδωνος αποφοίτησε από το Τμήμα Μηχανικών Η/Υ και Πληροφορικής του Πανεπιστημίου Ιωαννίνων το Σεπτέμβριο του 2020. Στο διάστημα από τον Μάρτιο του 2021 έως σήμερα εργάζεται ως προγραμματιστής σε εταιρεία πληροφορικής στο Τεχνολογικό Πάρκο Ηπείρου. Στα ερευνητικά του ενδιαφέροντα συμπεριλαμβάνονται τα φαινόμενα γήρανσης στα ολοκληρωμένα κυκλώματα των σύγχρονων νανοτεχνολογιών καθώς και η δοκιμή και αξιοπιστία μνημών σε αναδυόμενες τεχνολογίες.