

Εαρινό
Εξάμηνο
2022

Μελέτη των Διατάξεων Μνημών
Crossbar με Στοιχεία CBRAM &
Νευρομορφικές Εφαρμογές



Γεώργιος Κλεισιώτης
Εθνικό Μετσόβιο Πολυτεχνείο
Εαρινό Εξάμηνο 2022

Εθνικό
Μετσόβιο
Πολυτεχνείο



National
Technical
University of
Athens

*Μελέτη των Διατάξεων Μνημών Crossbar
με Στοιχεία CBRAM & Νευρομορφικές
Εφαρμογές*

Συγγραφέας:

Γεώργιος Κ. Κλειτσιώτης, MEng
Τομέας Φυσικής ΕΜΠ

Υπεύθυνος Καθηγητής:

Δημήτριος Τσουκαλάς
Καθηγητής
Τομέας Φυσικής ΕΜΠ

Η Διπλωματική Κατατίθεται για τη χορήγηση του Μεταπτυχιακού Τίτλου Σπουδών:

Μικροσυστημάτων και Νανοδιατάξεων, MSc

Εαρινό Εξάμηνο 2022

Ευχαριστίες

Η εργασία αυτή αποτέλεσε αποτέλεσμα μελέτης ενός χρόνου, ενός χρόνου επιτυχιών και αποτυχιών, αλλά πάνω από όλα καινούργιων εμπειριών. Είναι προφανές ότι, από τη στιγμή που μία ηλεκτρονική συσκευή 6 μηνών θεωρείται παλιά, η περαιτέρω ανάπτυξη νέων υπολογιστικών αρχιτεκτονικών αποτελεί μία εξαιρετικά δύσκολη διαδικασία. Είναι υποχρέωσή μου και τιμή μου λοιπόν να τονίσω ότι τα παρακάτω αποτελέσματα δεν θα ήταν δυνατό να προκύψουν χωρίς την ύπαρξη μίας δυνατής ομάδας.

Στο πνεύμα αυτό θέλω να ευχαριστήσω τους συναδέλφους μου, Παναγιώτη Μπούσουλα και Χάρη Τσιούστα για την βοήθειά τους και τη καθοδήγησή τους κατά την ερευνητική διαδικασία που παρουσιάζεται παρακάτω καθώς και τον υπεύθυνο καθηγητή, Δημήτριο Τσουκαλά για όλες τις γνώσεις και συμβουλές που μου μετέδωσε, τόσο σαν supervisor της εργασίας όσο και σαν διδάσκων του ΔΠΜΣ.

Τέλος, οι μεγαλύτερες ευχαριστίες πάνε στην οικογένεια μου, (τους γονείς μου και τα αδέρφια μου) που μου παρείχαν περίσσια ψυχική υποστήριξη, σε μία ιδιαίτερα απαιτητική περίοδο, τόσο λόγω φόρτου εργασίας, όσο και λόγω των δυσκολιών που προέκυψαν από την παρατεταμένη φάση πανδημίας, που δοκίμασε την αντοχή όλων μας.

Abstract-Περίληψη

Η συνεχώς αυξανόμενη αγορά διατάξεων μνήμης λόγω της ταχύτατης διάδοσης φορητών ηλεκτρονικών συσκευών (smartphones, tablets κ.α.) καθώς και η αύξηση των Κέντρων Δεδομένων (Data Centers), έχει οδηγήσει σε ραγδαία ανάπτυξη της τεχνολογίας των μη-πτητικών μνημών, δηλαδή των μνημών που διατηρούν την αποθηκευμένη πληροφορία ακόμα και με απώλεια τροφοδοσίας ισχύος. Οι μνήμες FLASH αποτελούν μέχρι σήμερα τον κυρίαρχο εκπρόσωπο των μη-πτητικών μνημών, ωστόσο φτάνουν τα φυσικά όρια σμίκρυνσής τους, λόγω κυρίως των μεγάλων ρευμάτων διαρροής που παρουσιάζουν σε μικρά πάχη οξειδίων και τα οποία καθιστούν τις καταστάσεις εγγραφής/διαγραφής μη διακρίσιμες. Επιπροσθέτως, η ανάπτυξη νέων διατάξεων μη-πτητικών μνημών δε λαμβάνει χώρα μόνο για να ικανοποιηθούν οι ανάγκες της τεχνολογίας για αυξημένη χωρητικότητα, αλλά αποτελούν αντικείμενο εντατικής έρευνας στο χώρο των λογικών κυκλωμάτων και των νευρομορφικών (neuromorphic) συστημάτων. Ορισμένες διεργασίες, όπως η εκμάθηση πραγματικών εικόνων, η αναγνώριση προτύπων και η λήψη αποφάσεων είναι εξαιρετικά δαπανηρές από πλευράς κατανάλωσης ενέργειας για τους συμβατικούς επεξεργαστές που στηρίζονται στη Boolean λογική, σε αντίθεση με τον ανθρώπινο εγκέφαλο, όπου οι παραπάνω διεργασίες λαμβάνουν χώρα αρκετά εύκολα.

Κάτω από αυτό το πλαίσιο, οι νέες διατάξεις μνήμης θα μπορούσαν να εκτελούν σύνθετες διεργασίες χαμηλής ισχύος, μιμούμενες τις αντίστοιχες διεργασίες του εγκεφάλου, με στόχο την τεχνητή εκμάθηση και την αναγνώριση προτύπων. Στην εργασία μελετήθηκε η χρήση των μνημών μεταβλητής αντίστασης αγωγίμου νήματος τύπου (conductive bridge random access memories – CBRAM) για καινοτόμες νευρομορφικές εφαρμογές. Αρχικά οι συγκεκριμένου τύπου διατάξεις χαρακτηρίστηκαν ως προς το παράθυρο μνήμης τους σε λειτουργία παροχής συνεχούς τάσης, όσο και στην ταχύτητα λειτουργίας τους, όταν η τάση εισόδου είναι παλμική. Έπειτα παρουσιάστηκε η διαδικασία ανάγνωσης μιας αρχιτεκτονική αλληλοδιασταυρωμένων δομών (crossbar structure), που κρίνεται ως ιδιαίτερης σημασίας τόσο για εφαρμογές αποθήκευσης πληροφορίας όσο και για νευρομορφικές εφαρμογές. Παράλληλα, αναπτύχθηκε μια καινούργια διάταξη χαρακτηρισμού και ελέγχου για την μέτρηση της ηλεκτρικής απόκρισης μιας τεχνητή σύναψη που δέχεται ερεθίσματα από έναν αισθητήριο νευρώνα. Τέλος εξετάστηκε η αξιοποίηση των CBRAM σε καινοτόμες αρχιτεκτονικές νευρομορφικών δικτυωμάτων.

The rapidly expanding market of memory devices due to the widespread use of portable devices (smartphones, tablets, etc.), as well as the increase of Data Centres, has led to the rapid development of non-volatile memories, i.e., memories capable of storing information even without power supply. FLASH memories are, as in the present, the main representative of non-volatile storage devices, however they are reaching their physical scaling limitations, due high leakage currents present in thin oxide devices, making the write/erase states indistinguishable. Additionally, the development of novel non-volatile memory devices is not only aimed on covering the demand for increased data storage, but also on covering the needs of logic circuitries and of neuromorphic systems. Certain processes like learning of real images, pattern recognition or decision making, are extremely demanding in terms of power consumption, when processed by modern computers based on Boolean logic, in direct contrast with human brain, where such tasks can be easily processed.

Under this framework, novel memory devices could handle complex, low power tasks, mimicking the respective functionalities of the human brain, the main goal being artificial learning and pattern recognition. In this work the use of conductive filament random access memories was investigated for neuromorphic applications. Initially, the devices were characterised when operating with direct or pulsating power supply in terms of their memory window and operating speed respectively. Up next, the read-out process of a CBRAM memory crossbar is presented, an architecture of great importance to both data storage and neuromorphic applications. At the same time a custom control and characterisation set-up was developed and used for measuring the electric response of the devices as terminal synapse substitutes of an artificial afferent neuron. Finally, the utilisation of the CBRAMs in novel neuromorphic applications was investigated.

Περιεχόμενα

.....	0
Ευχαριστίες	2
Abstract-Περίληψη	3
1 Εισαγωγή.....	7
1.1 Ο Νόμος του Moore.....	7
1.1.1 Προοπτικές του Νόμου του Moore;.....	8
1.2 Αρχιτεκτονική Υπολογιστών.....	9
1.3 Von Neumann και Μη Von Neumann Αρχιτεκτονική Υπολογιστών.....	10
1.4 Σκοπός της εργασίας.....	11
2 Μνήμες Τυχαίας Προσπέλασης Μεταβλητής Αντίστασης (RRAM).....	12
2.1 Εισαγωγή στις μνήμες RRAM.....	12
2.2 Αρχή λειτουργίας RRAM και Κατηγοριοποίηση.....	13
2.3 Κατασκευή και Χαρακτηρισμός των CBRAM.....	15
2.3.1 Κατασκευή Διατάξεων Μνήμης.....	15
2.3.2 Συσκευασία (packaging).....	16
2.3.3 Σύστημα Ηλεκτρικού Χαρακτηρισμού.....	17
2.3.4 Χαρακτηρισμός Συνεχούς Τάσης.....	17
2.3.5 Χαρακτηρισμός Παλμικής Τάσης.....	21
3 Σχεδιασμός Διάταξης Crossbar και Θεωρητικά Στοιχεία.....	24
3.1 Το πρόβλημα των κρυφών ρευμάτων.....	24
3.2 Λύσεις στο πρόβλημα των κρυφών ρευμάτων.....	27
3.2.1 Χρήση των transistors σαν διακόπτες επιλογής (Gating).....	28
3.2.2 Χρήση των ιδιοτήτων της CBRAM.....	29
3.3 Σχεδιασμός μία λειτουργικής διάταξης crossbar.....	30
3.3.1 Μέλος I: Το Crossbar.....	31
3.3.2 Μέλος II: Επιλογείς.....	32
3.3.3 Μέλος III: Ενισχυτής ανάγνωσης.....	35
3.4 Αποτελέσματα προσομοίωσης.....	37
3.5 Υλοποίηση του κυκλώματος.....	37
3.5.1 Λογισμικό.....	42
4 Χαρακτηρισμός CBRAM με Arduino.....	45
4.1 Χαρακτηρισμός συνεχούς τάσης.....	45
4.2 Χαρακτηρισμός παλμικής τάσης.....	48
5 Χρήση των CBRAM σε νευρομορφικές εφαρμογές.....	51

5.1	Έμπνευση από τη βιολογία	51
5.2	Λειτουργία τεχνητής σύναψης	51
5.2.1	Θεωρητικά στοιχεία	51
5.2.2	Πειραματικά στοιχεία – Παλμοί παραγόμενοι από ελεγχόμενη πηγή.....	52
5.2.3	Πειραματικά στοιχεία – Χαρακτηρισμός σύναψης με εξωτερικά ερεθίσματα..	53
5.3	Μάθηση.....	58
6	Νευρομορφικά δικτυώματα.....	60
6.1	Λειτουργία του δικτυώματος δεξαμενής δεδομένων (RC).....	61
6.2	Πειραματική διαδικασία.....	63
6.2.1	Ακρίβεια στον αριθμό των bits	63
6.2.2	Μετρήσεις.....	64
6.3	Προσομοίωση.....	67
7	Επίλογος - Συμπεράσματα.....	70
8	Ένθετο I: Σχέδιο Διάταξης crossbar χωρίς Selector Transistors	72
9	Ένθετο II: Σχέδιο Διάταξης crossbar με Transistors Επιλογής	73
10	Ένθετο III: Γενητρια Παλμών	74
11	Ένθετο IV: Παραγωγή παλμοσειράς 6 θέσεων στο Keithley 4200.....	75
	Αναφορές	89

1 Εισαγωγή

Είναι πλέον αδιαμφισβήτητο ότι η εποχή που ζούμε χαρακτηρίζεται από μια ευρείας κλίμακας συνδεσιμότητα των ηλεκτρονικών διατάξεων μέσω της χρήσης του διαδικτύου. Ακόμη και συσκευές, όπως αυτές που χρησιμοποιούμε στην καθημερινότητά μας μπορούν να ελεγχθούν μέσω του «Διαδικτύου των Πραγμάτων – Internet of Things (IoT)», ή ακόμη και τεχνολογίες τις οποίες τις θεωρούσαμε ανεπίτευκτες δύνανται πλέον να πραγματοποιηθούν (Τηλεϊατρική, Αυτόνομη Οδήγηση κ.λπ.)

Όλες αυτές οι διαδικασίες που έχουν συμβάλει σημαντικά στη βελτίωση της ποιότητας της ζωής μας συνεπάγονται την παραγωγή τεράστιου όγκου δεδομένων, ο οποίος με τη σειρά του απαιτεί την όσο το δυνατόν ταχύτερη επεξεργασία τους. Προς το παρόν φαίνεται να επιτυγχάνονται οι απαιτήσεις αυτές από τις σύγχρονες CMOS¹ (complementary metal oxide semiconductor) τεχνολογίες. Τα αυξημένα ρεύματα διαρροής σε συνδυασμό με τα φυσικά όρια σμίκρυνσης αυτών των διατάξεων εγείρουν σημαντικά ερωτήματα ως προς την αξιοπιστία τους. Επίσης, είναι επιβεβλημένο να εξερευνηθούν νέα σενάρια διατάξεων μνήμης ώστε να καλυφθεί το κενό που υπάρχει ανάμεσα στις υψηλής απόδοσης, χαμηλής πυκνότητας ολοκλήρωσης και δαπανηρές SRAM (static random access memories) και DRAM (dynamic random access memories), και στις χαμηλού κόστους, υψηλής πυκνότητας και χαμηλής απόδοσης HDD (hard drive disk). Για αυτό το λόγο τα τελευταία χρόνια βρίσκεται σε εξέλιξη μια εντατική προσπάθεια, τόσο σε ακαδημαϊκό όσο και σε βιομηχανικό επίπεδο, ούτως ώστε να ξεπεραστούν τα θέματα σμίκρυνσης των διατάξεων μνήμης, εξετάζοντας νέα υλικά και φυσικούς μηχανισμούς, με αυξημένη ικανότητα σμίκρυνσης. Αυτό το καινούργιο σενάριο μνήμης μπορεί να οριστεί ως Storage Class Memory (SCM). Εάν και φαίνεται δύσκολο ώστε να βρεθεί μια τεχνολογία που να μπορεί να ενσωματώσει όλα τα παραπάνω προαπαιτούμενα, ορισμένες καινοτόμες διατάξεις φαίνονται περισσότερο υποσχόμενες για μελλοντικές εφαρμογές μη-πτητικών μνημών.

1.1 Ο Νόμος του Moore

Τα παραπάνω συνοψίζονται στην παρατήρηση που έγινε το 1965 από τον Gordon Moore, συνιδρυτή των κολοσσών στον χώρο Fairchild Semiconductor και της Intel. Ο Moore με τον «νόμο» που φέρει το όνομά του προέβλεψε ότι ο αριθμός των transistors ενός πυκνού ολοκληρωμένου κυκλώματος διπλασιάζεται κάθε δύο χρόνια. Στο Σχήμα 1.1 μπορούμε να παρατηρήσουμε πως προέκυψε η παρατήρηση αυτή καθώς και κατά πόσο εξακολουθεί να επιβεβαιώνεται.

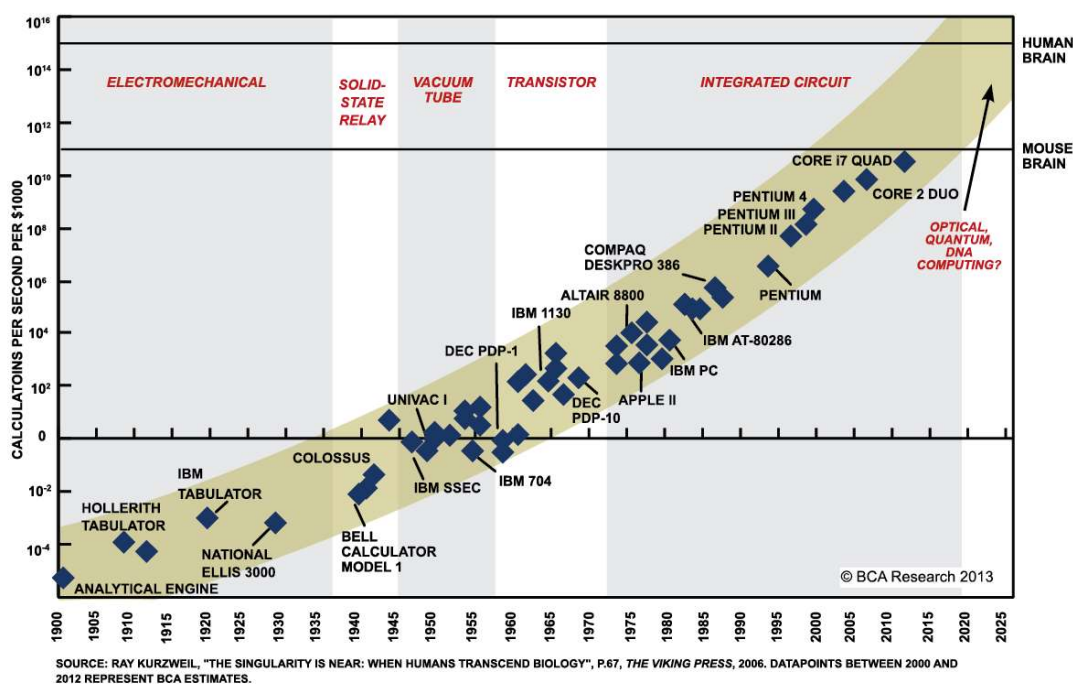
Με την πάροδο του χρόνου, πολλές λεπτομέρειες του νόμου αυτού τροποποιήθηκαν έτσι ώστε αυτός να ανταποκρίνεται καλύτερα στη «τρέχουσα» πυκνότητα ολοκλήρωσης των διατάξεων που βρίσκονταν την εκάστοτε χρονική περίοδο στην αγορά. Για παράδειγμα θεωρούμε πλέον πως ο διπλασιασμός επέρχεται κάθε 18 μήνες και όχι κάθε δύο χρόνια, όπως είχε αρχικά ισχυριστεί ο Moore. Το γεγονός ότι χρειάζεται μία τέτοιου είδους τροποποίηση εγείρει απορίες στο ως προς την ισχύ του νόμου [1].

¹ Complementary Metal Oxide Semiconductor

1.1.1 Προοπτικές του Νόμου του Moore;

Αναφερθήκαμε στην προηγούμενη παράγραφο στην δυσκολία αύξησης της πυκνότητας ολοκλήρωσης των transistor μέσα σε ένα ολοκληρωμένο κύκλωμα. Πράγματι, η εξέλιξη της τεχνολογίας, από φυσικής και μόνο άποψης, δεν μπορεί να συνεχίσει να εξελίσσεται με αυτόν τον τρόπο². Παρ' όλα αυτά, το γεγονός αυτό δεν σημαίνει και το τέλος της ισχύος του νόμου του Moore.

Αρκεί κάποιος να σκεφτεί ως καταναλωτής της τεχνολογίας και όχι απαραίτητα σαν φυσικός ή μηχανικός. Στην πραγματικότητα δεν ενδιαφερόμαστε μόνο για διατάξεις αυξημένης ολοκλήρωσης, αλλά για αύξηση στην ταχύτητα και την απόδοσή τους. Προκειμένου η βιομηχανία να πετύχει τις απαιτούμενες από την αγορά επιδόσεις εξέλιξε νέες τεχνολογίες σε επίπεδο μεθόδου υπολογισμού, όπως η επιστήμη των μεγάλων δεδομένων (big data science), η τεχνητή νοημοσύνη καθώς και η υπερσυνδεσιμότητα (δηλαδή η προσπάθεια όλο και περισσότερες συσκευές να επικοινωνούν μεταξύ τους), οι οποίες είναι σημαντικά πιο αποδοτικές από ότι χρησιμοποιούνταν στο παρελθόν. Οι νέες τεχνολογίες βέβαια απαιτούν και την ανάλογη ανάπτυξη νέων διατάξεων προκειμένου να επιτύχουν το 100% της δυναμικής τους.



Σχήμα 1.1 Σχηματική Απεικόνιση του Νόμου του Moore

² Τον καιρό της συγγραφής αυτής της εργασίας το τρέχον state-of-the-art ορίζεται από την IBM με μέγεθος της ελάχιστης διάστασης στα 2nm!

Νέες μορφές διατάξεων transistor, όπως τα FinFETs³, τα GAAFETs⁴ κλπ. εξακολουθούν να προσφέρουν λύσεις στην κατεύθυνση της αύξησης της πυκνότητας ολοκλήρωσης. Ωστόσο, η ανάλυση αυτών των τεχνολογιών ξεφεύγει από τα όρια αυτής της εργασίας. Έτσι συμπεραίνουμε πως παρά τις όποιες αναδιατυπώσεις, ο νόμος του Moore είναι ακόμη ζωντανός. Για να μείνει όμως έτσι, προβάλλεται επιτακτική η ανάγκη αλλαγής του τρόπου με τον οποίο σχεδιάζουμε ένα υπολογιστικό σύστημα (change of paradigm).

1.2 Αρχιτεκτονική Υπολογιστών

Στην μηχανική των υπολογιστών (computer engineering), ο όρος της αρχιτεκτονικής υπολογιστών (computer architecture) περιγράφει την λειτουργικότητα, την οργάνωση και την σχεδίαση ή υλοποίηση των υπολογιστικών συστημάτων.

Ορισμός: Η αρχιτεκτονική υπολογιστών διερευνά την εξισορρόπηση μεταξύ της απόδοσης (performance), της αποδοτικότητας (efficiency), του κόστους και της αξιοπιστίας (reliability) ενός υπολογιστικού συστήματος.

Για παράδειγμα ένας μηχανικός, χρησιμοποιώντας ένα πιο περίπλοκο σύνολο εντολών μπορεί να γράψει πιο γρήγορα προγράμματα τα οποία καταλαμβάνουν λιγότερο χώρο και καταναλώνουν λιγότερους πόρους επεξεργασίας και κατά συνέπεια ενέργεια. Το κόστος μία τέτοιας μελέτης είναι βέβαια σημαντικά υψηλότερο. Με ένα πιο απλό σύνολο εντολών, θα μπορούσε πιθανότατα να ελαττωθεί, έχοντας όμως αρνητικά αποτελέσματα για έναν ή περισσότερους από τους παραπάνω στόχους [2].

Η αρχιτεκτονική υπολογιστών μπορεί να χωριστεί στις παρακάτω υποκατηγορίες

- Η Αρχιτεκτονική Συνόλου εντολών, η οποία αφορά την διεπαφή μεταξύ του λογισμικού (Software) και του φυσικού μέρους (Hardware) ενός υπολογιστικού συστήματος. Τέτοια παραδείγματα είναι οι Compilers, οι οποίοι μεταφράζουν μία High Level γλώσσα προγραμματισμού (πχ C) σε γλώσσα μηχανής (binary)
- Η Μικρο-Αρχιτεκτονική (Οργάνωση Υπολογιστών), η οποία βοηθά στην βελτιστοποίηση υπολογιστικών προϊόντων
- Η Σχεδίαση - Υλοποίηση (implementation) των υπολογιστικών συστημάτων. Όταν το σύνολο των εντολών και η βελτιστοποίηση έχουν ολοκληρωθεί, θα πρέπει να σχεδιαστεί ένα φυσικό υπολογιστικό σύστημα. Η τελευταία μπορεί να χωριστεί σε περαιτέρω υποκατηγορίες όπως:
 - Λογική Σχεδίαση, δηλαδή η σχεδίαση ενός συστήματος σε επίπεδο λογικών πυλών
 - Σχεδίαση κυκλωμάτων: σχεδίαση βασικών συστημάτων σε επίπεδο transistor (gates, multiplexers, latches etc.)
 - Φυσικός σχεδιασμός ή αλλιώς η σχεδίαση των φυσικών συστημάτων και τέλος

³ Fin field-effect transistor

⁴ Gate-all-around field-effect transistor

- Η επικύρωση του σχεδιασμού, δηλαδή ο έλεγχος του υπολογιστικού συστήματος προκειμένου να διαπιστωθεί αν υπάρχουν σφάλματα σε κάποιο ενδεχόμενο λειτουργίας.

Στην εργασία αυτή θα εξετάσουμε την τεχνολογία και τη σχεδίαση υπολογιστικών συστημάτων με τη χρήση μνημών τυχαίας προσπέλασης αγωγίμου νήματος (Conductive Bridge Random Access Memories - CBRAM). Πριν όμως ξεκινήσουμε να αναλύουμε τις μνήμες αυτές καθ' αυτές και με ποιον τρόπο θα μπορέσουν να συμβάλουν στην περαιτέρω εξέλιξη του νόμου του Moore, Χρειαζόμαστε μερικές ακόμη θεωρητικές πληροφορίες πάνω στην αρχιτεκτονική

1.3 Von Neumann και Μη Von Neumann Αρχιτεκτονική Υπολογιστών

Η αρχιτεκτονική Von Neumann αποτελεί τον παραδοσιακό τρόπο σχεδιασμού ενός υπολογιστικού συστήματος, τον οποίο χρησιμοποιούμε στη καθημερινότητά μας. Πρόκειται για συστήματα στα οποία η μνήμη και η κεντρική μονάδα επεξεργασίας αποτελούν ξεχωριστές οντότητες. Τα βασικά μέρη ενός τέτοιου συστήματος φαίνονται στο Σχήμα 1.2.

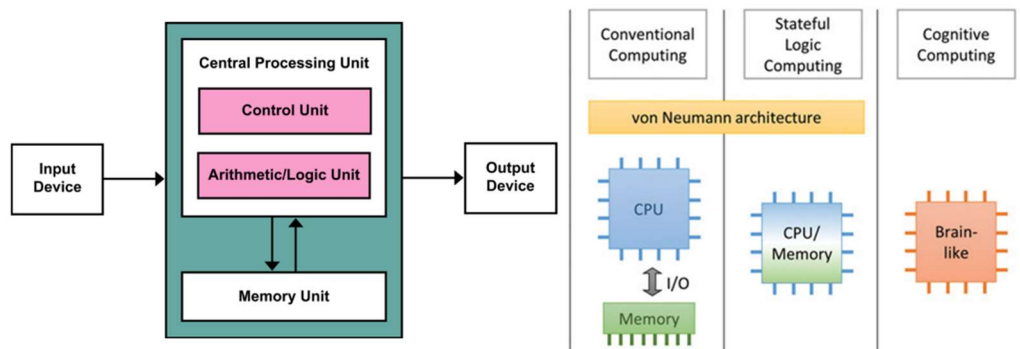
Εν συντομία ένας Von Neumann ηλεκτρονικός υπολογιστής αποτελείται από: [3]

- Μία κεντρική μονάδα επεξεργασίας (CPU), στην οποία εμπεριέχονται μία αριθμητική λογική (arithmetic logic) και οι καταχωρητές του επεξεργαστή (processor registers). Σε αυτή γίνεται η επεξεργασία των δεδομένων.
- Τη μονάδα ελέγχου (Control Unit), η οποία συμπεριλαμβάνει τον καταχωρητή εντολών και τον απαριθμητή των προγραμμάτων (programme counter). Ελέγχει τη λειτουργία της κεντρικής μονάδας επεξεργασίας
- Τη κύρια μνήμη, όπου αποθηκεύονται δεδομένα και εντολές (RAM, ROM)
- Τη δευτερεύουσα μνήμη μαζικής αποθήκευσης (SSD, HDD, External Storage)
- Μηχανισμούς εισαγωγής και εξαγωγής δεδομένων

Η αποδοτικότητα επεξεργασίας των δεδομένων είναι περιορισμένη εξ' ορισμού, λόγω των φυσικά διακριτών μονάδων επεξεργασίας και μνήμης. Για παράδειγμα, στη σύγχρονη τεχνολογία, οι ταχύτητες των κεντρικών μονάδων επεξεργασίας αυξάνονται ταχύτερα από ότι μπορεί να ακολουθήσει η αντίστοιχη αύξηση των μνημών, δημιουργώντας ένα τεχνολογικό «χάσμα» μεταξύ των δύο βασικών μελών της αρχιτεκτονικής Von Neumann. Επιπλέον ο περιορισμός της μεταφοράς των δεδομένων μεταξύ των δύο μέσω του διαύλου περιορίζει περαιτέρω την συνολική ταχύτητα του συστήματος καθώς και την κατανάλωση λόγω της ίδιας λειτουργίας του (data transfer bottleneck).

Προκειμένου να αντιμετωπιστούν αυτές οι προκλήσεις αναπτύσσονται νέες αρχιτεκτονικές, όπως ο «υπολογισμός-στη-μνήμη (in-memory-computing)» και οι νευρομορφικές αρχιτεκτονικές υπολογιστικών συστημάτων (neuromorphic computer architectures). Ο υπολογισμός-στη-μνήμη, σαν αρχιτεκτονική, ξεκινά από το επίπεδο των διατάξεων. Έχει την δυνατότητα πραγματοποίησης υπολογισμών ψηφιακής λογικής, όπως και νευρομορφικών υπολογισμών μέσω της χρήσης καινοτόμων μη-πτητικών μνημών, όπως είναι οι μνήμες μεταβλητής αντίστασης. Διαφέρει από την παραδοσιακή Von Neumann αρχιτεκτονική καθώς πραγματοποιεί τις απαραίτητες πράξεις (επεξεργασία δεδομένων) πάνω στην ίδια τη μνήμη,

περατώνοντας ταυτόχρονα και τις διεργασίες αποθήκευσης δεδομένων. Με αυτόν τον τρόπο εξαλείφεται η ανάγκη μεταφοράς των δεδομένων μεταξύ μνήμης και της κεντρικής μονάδας επεξεργασίας μέσω ενός διαύλου, εξοικονομώντας υπολογιστικούς πόρους και ενέργεια. Το δε «χάσμα» ταχύτητας μεταξύ επεξεργαστή και μνήμης δεν υπάρχει σε αυτή την αρχιτεκτονική, καθώς τα δύο στοιχεία δεν αποτελούν πλέον διακριτές οντότητες [4].



Σχήμα 1.2 Διάγραμμα Αρχιτεκτονικής Von Neumann και σύγκρισή της με μελλοντικές μη Von Neumann τεχνολογίες

1.4 Σκοπός της εργασίας

Τα επόμενα κεφάλαια θα τα αφιερώσουμε στη μελέτη των μνημών τυχαίας προσπέλασης αγωγικής γέφυρας (conductive bridge random access memories - CBRAM). Θα εξετάσουμε πως αυτές οι μνήμες μπορούν να βρουν εφαρμογή σε νευρομορφικά υπολογιστικά συστήματα, μελετώντας τα πλεονεκτήματα και τους περιορισμούς που εμφανίζονται στη χρήση τους για τον σκοπό αυτών.

Για τον σκοπό αυτό σχεδιάσαμε μία πρωτότυπη αρχιτεκτονική αλληλοδιασταυρούμενων δομών (crossbar), οδηγούμενη με τη χρήση μικρο-ελεγκτή Arduino, του προγράμματος Simulink και διαφόρων περιφερειακών ηλεκτρονικών στοιχείων, τα οποία θα αναλύσουμε και διεξοδικά στη συνέχεια.

Έγινε επίσης σύγκριση των αποτελεσμάτων της ανάγνωσης ενός memristive στοιχείου που προσφέρει το παραπάνω σύστημα, με τα αντίστοιχα αποτελέσματα που προκύπτουν από τη χρήση επαγγελματικού εξοπλισμού χαρακτηρισμού ημιαγωγικών διατάξεων.

Τέλος γίνεται μία αρχική πειραματική μελέτη πάνω στη χρήση των CBRAM του εργαστηρίου σε νευρομορφικές εφαρμογές όπως τα Recurrent Neural Networks και το Reservoir Computing, και προτείνουμε πιθανές αλλαγές στη διάταξή μας ώστε να μπορούν να υλοποιηθούν τα συστήματα αυτά.

2 Μνήμες Τυχαίας Προσπέλασης Μεταβλητής Αντίστασης (RRAM)

Η συνεχώς αυξανόμενη ζήτηση, κυρίως από τον τομέα των φορητών ηλεκτρονικών συσκευών έχει οδηγήσει την έρευνα και τη βιομηχανία στην κατασκευή μνημών υψηλής ταχύτητας, όμοιες με αυτές της πτητικής δυναμικής μνήμης τυχαίας προσπέλασης (Dynamic Random Access Memories – DRAM), αλλά με την δυνατότητα να αποθηκεύουν πληροφορίες δίχως απαίτηση συνεχούς παροχής ρεύματος. Με άλλα λόγια έχουν ως απαίτηση τη μη πτητικότητα.

Παράδειγμα μη πτητικής μνήμης είναι ο σκληρός δίσκος, ή οι SSDs.

Είναι η απαίτηση αυτή που μας οδήγησε στην κατασκευή μη πτητικών μνημών τυχαίας προσπέλασης (Non-volatile RAM – NVRAM) όπως οι διαδομένες μνήμες Flash ή οι V-NAND⁵ τεχνολογίες. Το τελευταίο παράδειγμα επιτρέπει την εγγραφή και ανάγνωση πληροφορίας σε ταχύτητες συγκρίσιμες με αυτές μίας DDR3 SDRAM⁶.

Άλλες τέτοιες πολλά υποσχόμενες τεχνολογίες είναι οι μνήμες αλλαγής φάσης (Phase Change Memories), οι σιδηρο-ηλεκτρικές μνήμες (Ferro-electric Memory), οι μνήμες μεταφοράς Ροπής (Spin Torque Transfer Memories) [5] που αποτελούν και μαγνητική τεχνολογία με πιθανές εφαρμογές στην αεροδιαστημική. Η ανάλυση αυτών ξεφεύγει από τα όρια της εργασίας αυτής. Η τεχνολογία στην οποία θα επικεντρωθούμε είναι αυτή των μνημών εναλλαγής αντίστασης (Resistive Memories).

2.1 Εισαγωγή στις μνήμες RRAM

Η κατηγορία των μνημών εναλλαγής αντίστασης (Resistive RAMs – RRAM) αποτελεί έναν ελκυστικό τύπο μη πτητικής μνήμης για πολλούς λόγους, ο βασικότερος εκ των οποίων η ευκολία κατασκευής, καθώς πρόκειται για “sandwich materials” αποτελούμενα από ένα μεταλλικό κάτω ηλεκτρόδιο (πχ Πλατίνα - Pt ή Νιτρίδιο του Τιτανίου - TiN), ένα διηλεκτρικό υλικό συνήθως από διοξείδιο του Πυριτίου (SiO₂) και ένα ακόμη μεταλλικό άνω ηλεκτρόδιο, σε μία δομή που θυμίζει έντονα αυτή ενός πυκνωτή μετάλλου – οξειδίου – μετάλλου (metal – oxide – metal MOS). Λόγω αυτής της αρχιτεκτονικής, η κατασκευή των μνημών αυτών μπορεί να λάβει χώρα στα τελευταία στάδια των διεργασιών CMOS (back-end CMOS line), ελαττώνοντας σημαντικά το κόστος παραγωγής και την ενεργή επιφάνεια κάλυψης, οδηγώντας σε μεγαλύτερη πυκνότητα ολοκλήρωσης [5].

Μία μνήμη RAM, για την αποθήκευση ψηφιακών δεδομένων, απαιτείται να έχει δύο διακριτές καταστάσεις, οι οποίες να μπορούν να αντιστοιχηθούν σε 0 (digital LOW) και 1 (digital HIGH). Στις RRAM ο διαχωρισμός αυτός γίνεται μετρώντας την αγωγιμότητα. Σε χαμηλή αγωγιμότητα θα λέμε ότι η μνήμη βρίσκεται στην κατάσταση **HRS (High Resistance State)** ή **Off**, και συνήθως αντιστοιχούμε την κατάσταση αυτή σε **digital LOW**. Η αντίθετη περίπτωση της υψηλής αγωγιμότητας θα λέμε ότι αντιστοιχεί στην **LRS (Low Resistance**

⁵ Samsung® V-NAND®

⁶ Σύγχρονες δυναμικές μνήμες τυχαίας προσπέλασης διπλής ταξινόμησης δεδομένων 3^{ης} γενιάς (Double Data Rate 3 – Dynamic RAM)

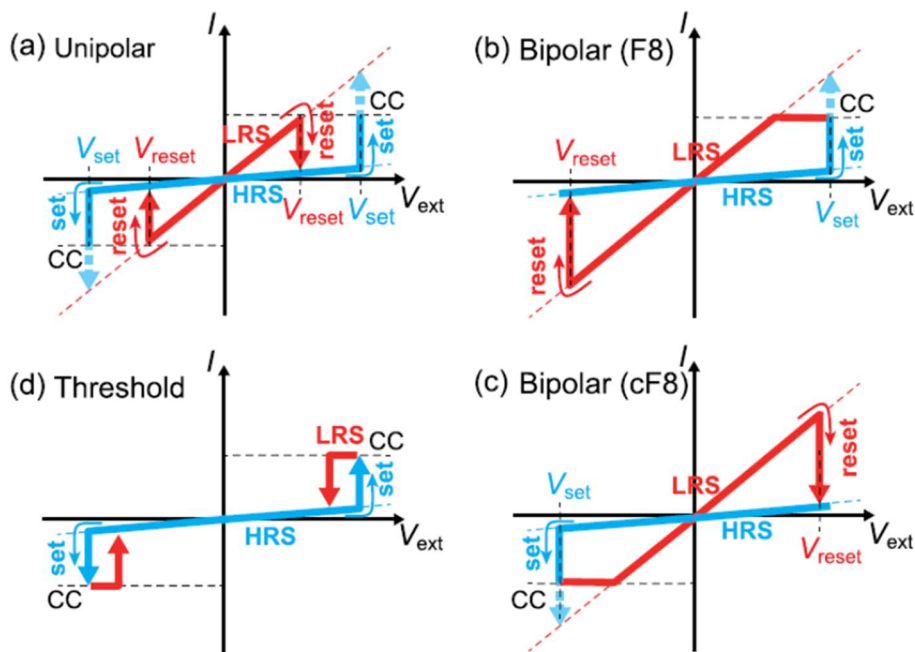
State) ή **On**. Η μετάβαση από την κατάσταση HRS στην LRS ονομάζεται **SET**. Η αντίστροφη διαδικασία (από LRS σε HRS) ονομάζεται **RESET**.

2.2 Αρχή λειτουργίας RRAM και Κατηγοριοποίηση

Η φυσική αρχή λειτουργίας του φαινομένου αποτελεί ακόμη αντικείμενο έρευνας στον χώρο της ακαδημίας. Γενικά, ανάλογα με την διαφορά δυναμικού που εφαρμόζεται στα ηλεκτρόδια προκύπτει η μεταβολή από τη μία κατάσταση στην άλλη. Με βάση τα χαρακτηριστικά που προκύπτουν από την εφαρμογή της τάσης αυτής, μπορούμε να χωρίσουμε τις μνήμες σε δύο κατηγορίες:

- Μονοπολικές (Unipolar Switching) για τις οποίες οι μεταβάσεις SET και RESET προκύπτουν για την ίδια πολικότητα εφαρμοζόμενης τάσης
- Διπολικές (Bipolar Switching) όπου απαιτείται εναλλαγή της πολικότητας για τη διαδικασία RESET.

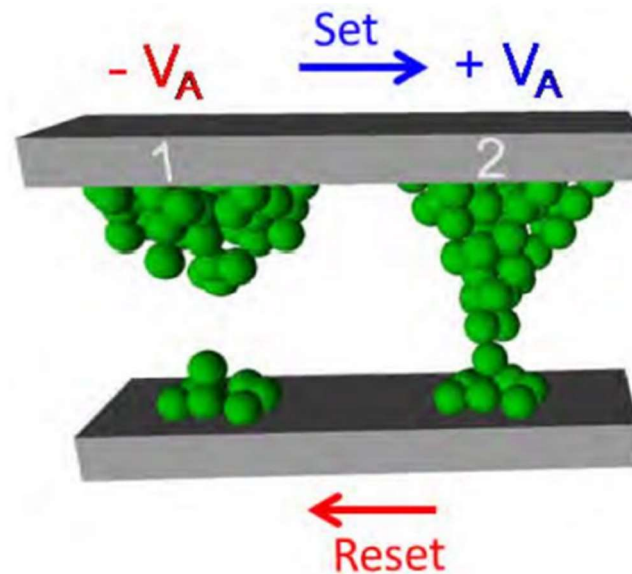
Μία άλλη κατηγοριοποίηση που μπορεί να γίνει με βάση στην πτητικότητα μίας μνήμης. Οι μνήμες που παρουσιάζουν τουλάχιστον δύο σταθερές καταστάσεις αντίστασης, ανεξάρτητα από την εφαρμογή ή μη κάποιου εξωτερικού ηλεκτρικού πεδίου χαρακτηρίζονται ως μη πτητικές. Εν αντιθέσει, οι μνήμες μεταβολής κατωφλίου (threshold switching) παρουσιάζουν μία μόνο σταθερή κατάσταση, χαμηλής αγωγιμότητας, όταν εφαρμόζεται κάποιο σχετικά μικρότερο εξωτερικό πεδίο, η οποία αυξάνεται, με μη μόνιμο τρόπο, όταν η τάση ξεπεράσει κάποιο κατώφλι.



Σχήμα 2.1 Σχηματική Απεικόνιση των διαφορετικών ειδών εναλλαγής αντίστασης σε μνήμες RRAM [5]

Συνήθως, μεγαλύτερο ερευνητικό ενδιαφέρον παρατηρούμε στις μνήμες διπολικής μεταβολής λόγω της καλύτερης ηλεκτρικής συμπεριφοράς σε εφαρμογές αποθήκευσης, σε σχέση με τις μονοπολικές. Για τη λειτουργία μίας διπολικής μεταβολής μνήμης είναι συνήθως απαραίτητη

μία πρότερη διαδικασία ηλεκτροσχηματισμού (electroforming). Κατά τη διαδικασία αυτή, προκαλείται μία ήπια κατάρρευση του ενδιάμεσου διηλεκτρικού, ελαττώνοντας τοπικά την συγκέντρωση των ιόντων οξυγόνου και αυξάνοντας την συγκέντρωση των ατελειών, όπως οι κενές θέσεις οξυγόνου. Το ρεύμα αυξάνεται συνεχώς μέχρι που προκύπτει το τελικό αποτέλεσμα, η ένωση των δύο ηλεκτροδίων μέσω ενός αγώγιμου νήματος (Conductive Filament – CF) – διαδικασία SET. Αν εφαρμόσουμε μία τάση αντίθετης πολικότητας, παρατηρούμε την μετακίνηση των κενών οξυγόνου προς το CF, τα οποία το «κόβουν». Έτσι χάνεται η επαφή μεταξύ των δύο ηλεκτροδίων προκαλώντας την επαναφορά στη κατάσταση HRS – διαδικασία RESET.



Σχήμα 2.2 Σχηματική αναπαράσταση της διαδικασίας εναλλαγής [5]

Με παρόμοια λογική μπορούμε να ορίσουμε τις μνήμες αγώγιμου νήματος ή αγώγιμης γέφυρας (Conductive Bridge Memories). Οι μνήμες αυτές αποτελούνται από ένα ενεργό άνω ηλεκτρόδιο (συνήθως Χάλκινο Cu ή ασημένιο Ag) από το οποίο ιόντα διεισδύουν στο διηλεκτρικό κατά τη διαδικασία SET, δημιουργώντας ένα μεταλλικό αγώγιμο νήμα (conductive filament - CF) που με τη σειρά του γεφυρώνει τα δύο ηλεκτρόδια. Στη μετάβαση RESET τα ιόντα ολισθαίνουν προς το άνω ηλεκτρόδιο, «κόβοντας» έτσι το αγώγιμο νήμα. Είναι υποχρεωτικό σε αυτή την περίπτωση το κάτω ηλεκτρόδιο να αποτελείται από ένα χημικά αδρανές υλικό [5].

Στην εργασία αυτή θα χρησιμοποιήσουμε στοιχεία CBRAM κατασκευασμένα από το εργαστήριο μικροηλεκτρονικής της Σχολής Εφαρμοσμένων Μαθηματικών και Φυσικών Επιστημών – Εθνικό Μετσόβιο Πολυτεχνείο. Οι μνήμες αυτές, έπειτα από πειράματα χαρακτηρισμού που θα αναλύσουμε εκτενώς στο επόμενο κεφάλαιο, παρουσιάζουν ένα μεγάλο παράθυρο μνήμης, το οποίο μπορεί να φτάσει το μέγεθος των 10^6 Siemens⁷. Επιπλέον πρόκειται για διατάξεις μικρού ενεργειακού αποτυπώματος, καθώς δουλεύουν ικανοποιητικά σε αρκετά μικρές τάσεις, περί το 1V και μικρό ρεύμα. Το κυριότερο όμως χαρακτηριστικό

⁷ Το παράθυρο μνήμης ορίζεται ως η διαφορά της αγωγιμότητας στην HRS και την LRS για την τάση όπου προκύπτει η εναλλαγή. Θα οριστεί αναλυτικότερα στο κεφάλαιο 3

τους είναι πως ανάλογα με την εφαρμοζόμενη τάση, μπορούν να παρουσιάσουν τόσο διπολική όσο και συμπεριφορά κατωφλίου. Όπως θα δούμε στη συνέχεια της εργασίας, ένας νευρομορφικός υπολογιστής, καθώς προσομοιάζει τη λειτουργία των νευρικών συνάψεων, χρειάζεται τόσο πτητικά όσο και μη πτητικά στοιχεία, ρόλοι που καλύπτονται από τα ίδια στοιχεία CBRAM που κατασκευάσαμε.

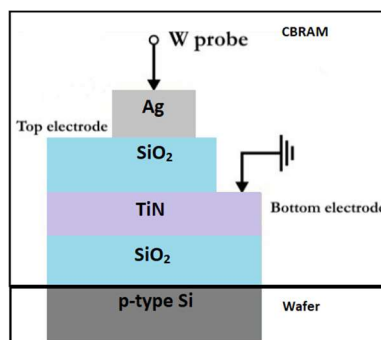
2.3 Κατασκευή και Χαρακτηρισμός των CBRAM.

Ισχυριστήκαμε στη προηγούμενη παράγραφο ότι οι CBRAM του εργαστηρίου μας, μπορούν να δουλέψουν σε δύο διαφορετικές λειτουργίες (από εδώ και πέρα “modes”), διπολικής μεταβολής και μεταβολής κατωφλίου. Στην εργασία αυτή έλαβαν χώρα, πειράματα χαρακτηρισμού των διατάξεων μνήμης, τόσο με συνεχές ρεύμα (DC – IV χαρακτηρισμός) όσο και παλμικής απόκρισης. Πριν μιλήσουμε όμως για τον χαρακτηρισμό των διατάξεων, πρέπει πρώτα να περιγράψουμε την διαδικασία κατασκευής τους.

2.3.1 Κατασκευή Διατάξεων Μνήμης

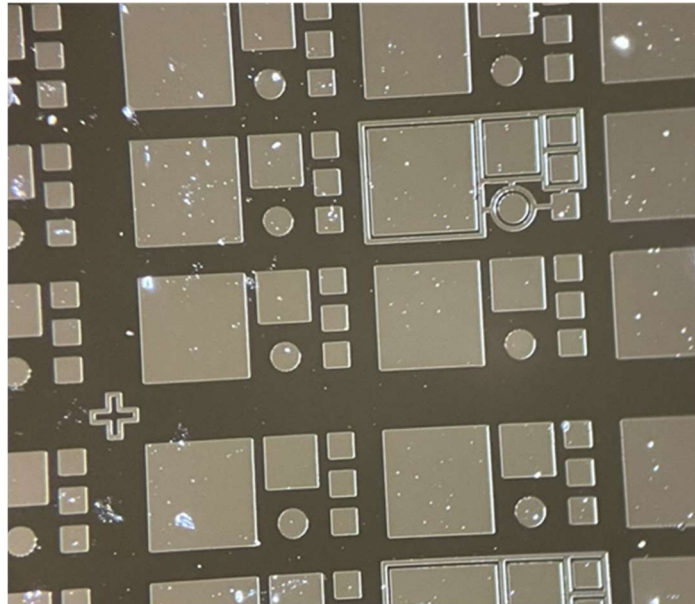
Οι μνήμες μας αποτελούνται από μία sandwich διάταξη με 40nm TiN ως κάτω ηλεκτρόδιο, 20nm SiO₂ ως ενεργό διηλεκτρικό και τέλος το ενεργό υλικό 40nm Ag στο άνω ηλεκτρόδιο.

Όλα τα λεπτά φιλμ εναποτέθηκαν πάνω σε ένα wafer από SiO₂ με την τεχνική RF magnetron sputtering, σε θερμοκρασία δωματίου. Η ισχύς της RF γεννήτριας ήταν, για τους κεραμικούς στόχους των TiN και SiO₂ ρυθμισμένη στα 140W, ενώ για τον μεταλλικό στόχο Ag στα 200W. Ο θάλαμος της εναπόθεσης, πριν τη διαδικασία, ήταν σε συνθήκες υψηλού κενού ($2 \cdot 10^{-6} mbar$), πριν την ανάφλεξη πλάσματος λόγω της εισαγωγής Αργού (Ar gas injection). Κατά τη διάρκεια της εναπόθεσης, η πίεση του θαλάμου ήταν περί τα $4 \cdot 10^{-3} mbar$. Η ροή του Ar κατά την εναπόθεση είχε οριστεί ως σταθερή στα 20 sccm, ενώ για το O₂ χρησιμοποιήσαμε δύο διαφορετικές ροές (1 και 5 sccm). Έγινε αρχικά εναπόθεση 10nm SiO₂ με ροή O₂ 1 sccm η οποία αυξήθηκε στα 5 sccm, μέχρι το τελικό πάχος. Για την εναπόθεση χρησιμοποιήθηκε στόχος SiO₂ υψηλής καθαρότητας που είχε αναπτυχθεί με ξηρή οξείδωση.



Σχήμα 2.3 Σχηματική αναπαράσταση ενός στοιχείου CBRAM Ag

Τα μεταλλικά ηλεκτρόδια εναποτέθηκαν χρησιμοποιώντας τους κατάλληλους στόχους από TiN και Ag, όπως φαίνεται και στο Σχήμα 2.3. Η τελική μορφή των διατάξεων ήταν τετραγωνική, με διαστάσεις 100 μm x 100 μm , 200 μm x 200 μm , 400 μm x 400 μm και ϕ 112.85 μm , αντίστοιχες ενός πυκνωτή μετάλλου-μονωτή-μετάλλου (MOS) (Σχήμα 2.4) [6] [7].



Σχήμα 2.4 Φωτογραφία της τελικής μορφής μίας διάταξης μνήμης CBRAM. Το κάθε τετράγωνο ηλεκτρόδιο αντιστοιχεί σε ένα top ηλεκτρόδιο, ο ενδιάμεσος χώρος μεταξύ των τετραγώνων, για τη συγκεκριμένη διάταξη αποτελεί το κοινό για κάθε στοιχείο bottom ηλεκτρόδιο. Χρησιμοποιούμε τα μικρά τετράγωνα 100 μm x 100 μm .

2.3.2 Συσκευασία (packaging)

Έγιναν προσπάθειες συσκευασίας των διατάξεων σε υποδοχείς ολοκληρωμένου κυκλώματος (IC connectors) μέσω της διαδικασίας συγκόλλησης λεπτών καλωδίων (στο εξής: wire-bonding). Κατά τη διαδικασία αυτή, χρησιμοποιώντας ένα νήμα από 25 μm Au, επιχειρούμε να ενώσουμε τις επαφές της μνήμης μας με τους ακροδέκτες του υποδοχέα. Η ολοκλήρωση της διαδικασίας συνεπάγεται τη διευκόλυνση χρήσης των CBRAMs σε ολοκληρωμένα κυκλώματα χωρίς την χρήση κάποιου εξωτερικού σταθμού εξέτασης (probe station).

Οι προσπάθειές μας δεν είχαν θετικό αποτέλεσμα. Αρχικά, συμπεράναμε πως δεν είναι δυνατή η συγκόλληση (adhesion) του νήματος πάνω στα υλικά των ηλεκτροδίων. Για τον λόγο αυτό εναποθέσαμε επιπλέον 40nm Ti και 40nm Au στα πάνω ηλεκτρόδια (με αυτή τη σειρά) και επιχειρήσαμε κόλληση τόσο με νήμα Au όσο και Al, ίδιου πάχους. Μία τέτοια διαδικασία εναπόθεσης δεν έχει κάποια επίδραση στη φυσική ή στη λειτουργία της μνήμης, φαίνεται όμως να διευκολύνει τη συγκόλληση των ηλεκτροδίων με τα νήματα. Οι προσπάθειες με το νήμα χρυσού, παρά την επιτυχία τους για τα κάτω ηλεκτρόδια, φαίνεται να είναι ατελέσφορες όσον αφορά τα πάνω κάτι που αποδόθηκε στην τραχύτητα του υποστρώματος από Ag. Στην

περίπτωση του νήματος Al και του επιχρυσωμένου ηλεκτροδίου παρουσιάζεται μία σημαντική αύξηση στο ποσοστό επιτυχίας της συγκόλλησης. Παρ' όλα αυτά το ποσοστό αυτό δεν είναι αρκετό ώστε να προχωρήσουμε στη σχεδίαση πιο περίπλοκων αρχιτεκτονικών (όπως τα crossbars που θα εξετάσουμε αναλυτικά στο επόμενο κεφάλαιο).

Κατά την περίοδο συγγραφής αυτής της εργασίας, προκύπτουν πολλά υποσχόμενα αποτελέσματα για την περίπτωση συγκόλλησης νήματος Al σε επιφάνεια Al. Η κατασκευή όμως τέτοιων διατάξεων και η βελτιστοποίηση της διαδικασίας της κόλλησης αποτελεί αντικείμενο μελέτης που δυστυχώς δεν μπορεί να συμπεριληφθεί στη εργασία αυτή. Για τον λόγο αυτό και στα πειράματα που ακολουθούν είτε υποκαταστήσαμε τις CBRAM με κάποιο άλλο στοιχείο ώστε να εξετάσουμε τα πρωτότυπα συστήματα που σχεδιάστηκαν σε αυτό το πλαίσιο, είτε χρησιμοποιήσαμε σταθμό εξέτασης και προσομοίωση της συμπεριφοράς των στοιχείων, όπου αυτό ήταν δυνατό.

2.3.3 Σύστημα Ηλεκτρικού Χαρακτηρισμού

Οι μετρήσεις συνεχούς τάσης (I-V measurements) πραγματοποιήθηκαν με τη χρήση του Keithley 4200[®] Semiconductor Parameter Analyzer (4200-SCS) σε έναν σταθμό εξέτασης SUSS MicroTech[®] probe station. Ο σταθμός παρέχει θωράκιση από διάφορες παραμέτρους που μπορεί να επηρεάσουν μία τόσο ευαίσθητη μέτρηση, όπως οι δονήσεις ή η το υπόβαθρο ηλεκτρομαγνητικής ακτινοβολίας στον χώρο.

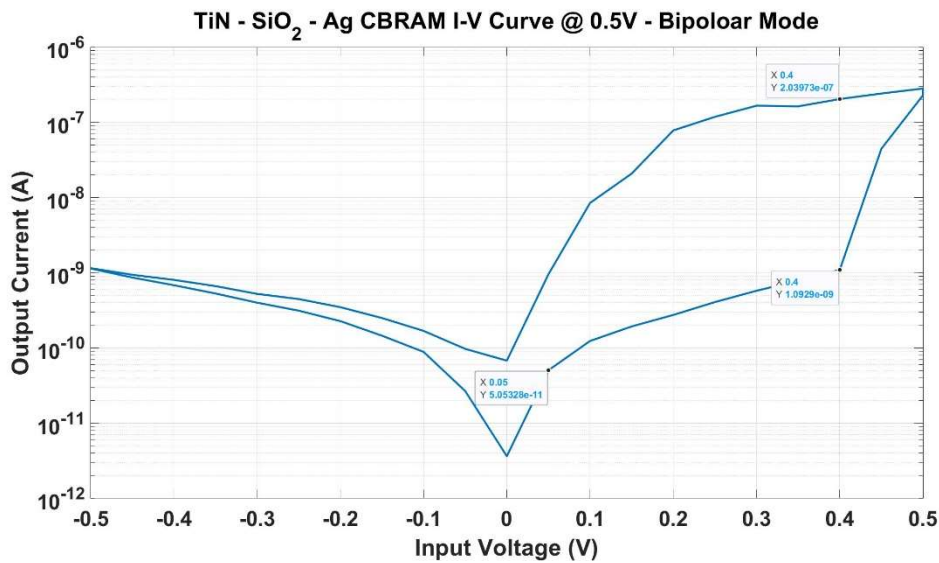
Για τις παλμικές μετρήσεις χρησιμοποιήσαμε το ίδιο σύστημα μαζί με το Keithley 4225-PMU[®] (Pulsed Measurement Unit) και τον ίδιο θάλαμο θωράκισης.

2.3.4 Χαρακτηρισμός Συνεχούς Τάσης

Πραγματοποιήθηκαν μετρήσεις εξαγωγής καμπύλης I-V σε συνεχή τάση. Με τις μετρήσεις αυτές προκύπτουν τα περισσότερα από τα χαρακτηριστικά ενός στοιχείου μνήμης, όπως αυτά του παραθύρου μνήμης, της τάσης εναλλαγής καθώς και το καθεστώς στο οποίο λειτουργεί (διπολική ή λειτουργία κατωφλίου). Θα εξάγουμε τα συμπεράσματά μας ελέγχοντας έναν επαρκή αριθμό από καμπύλες⁸.

Στο Σχήμα 2.5 φαίνονται οι έννοιες που αναφερθήκαμε παραπάνω. Ορίζουμε ως παράθυρο μνήμης τον λόγο, σε τάξεις μεγέθους, του ρεύματος για την ίδια διαφορά δυναμικού μεταξύ του άνω και του κάτω κλάδου της καμπύλης, ή αλλιώς τον λόγο ρεύματος για την ίδια τάση στους κλάδους της LRS και της HRS. Η τάση εναλλαγής είναι η τάση για την οποία πραγματοποιείται, όπως δηλώνει ο όρος, η εναλλαγή από τη μία κατάσταση στην άλλη.

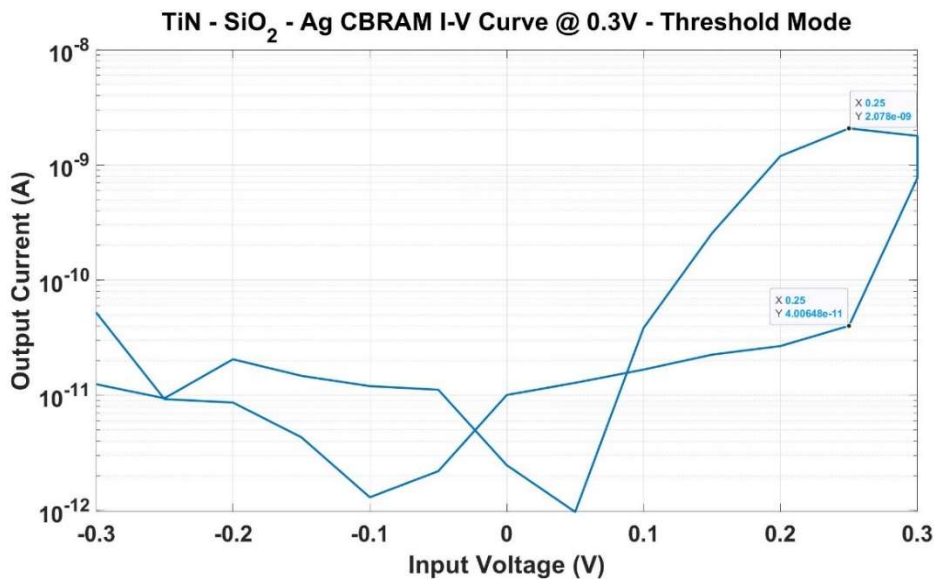
⁸ Για όλες τις επόμενες καμπύλες, είχαμε ορίσει μέγιστο ρεύμα (compliance current) $I_{cc} = 10^4 A$. Η συνθήκη δεν ενεργοποιήθηκε καθώς οι μνήμες, σε αυτά τα πειράματα, λειτουργήσαν με σημαντικά μικρότερα ρεύματα.



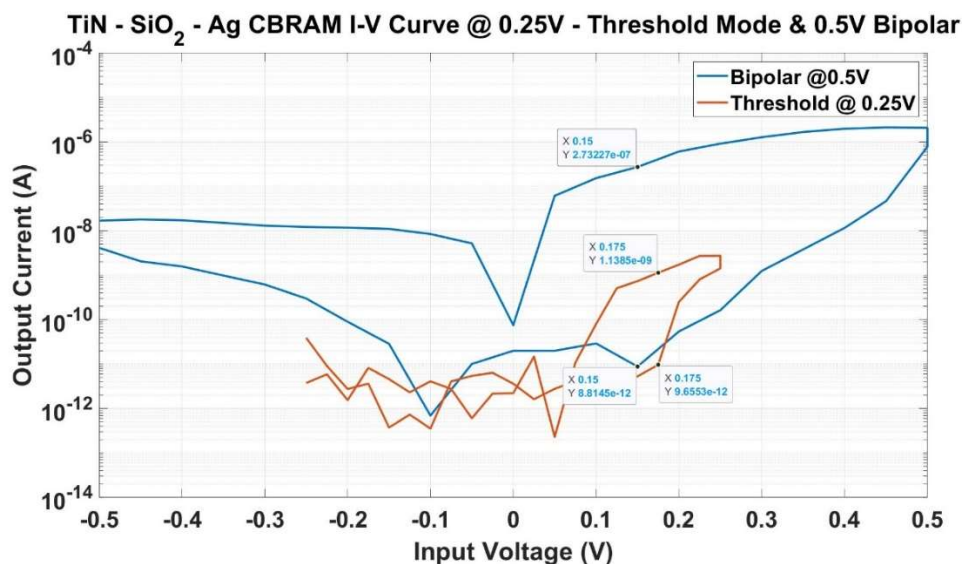
Σχήμα 2.5 Καμπύλη Χαρακτηρισμού I-V ενός στοιχείου CBRAM (sample No. 210-12-3). Bipolar Mode @ 0.5V

Για τη μέτρηση του Σχήματος 2.5, το παράθυρο μνήμης ορίζεται στα $\frac{2.04 \cdot 10^{-7} A}{1.1 \cdot 10^{-9} A} \cong 10^2$, για την τάση εναλλαγής στα 0.4V. Παρατηρούμε επίσης πως χρειάζεται εναλλαγή της πολικότητας της τάσης, προκειμένου να επανέλθει το δείγμα στην αρχική του κατάσταση (HRS), γεγονός που ταυτίζεται με τον ορισμό της διπολικής μεταβολής που δώσαμε στην παράγραφο 2.2.

Στο Σχήμα 2.6 παρατηρούμε την απόκριση σε χαρακτηρισμό συνεχούς τάσης ενός διαφορετικού στοιχείου μνήμης, αυτή τη φορά με μέγιστη τάση 0.3V. Παρατηρούμε ότι στο τέλος του διπλού περάσματος από θετικές τάσεις, η μνήμη βρίσκεται στην ίδια κατάσταση και άρα, σύμφωνα με τον ορισμό της παραγράφου 2.2 βρισκόμαστε σε λειτουργία κατωφλίου. Παρατηρούμε επίσης ότι για μία τέτοια κατάσταση το παράθυρο μνήμης είναι σημαντικά μικρότερο, συγκεκριμένα στα $\frac{2.08 \cdot 10^{-9} A}{4.01 \cdot 10^{-11} A} \cong 10$ για μία τάση των 0.25V. Επίσης το παράθυρο μνήμης αυτό παρατηρείται για πολύ μικρότερα ρεύματα, λόγω της μικρής εφαρμοζόμενης τάσης αλλά και κυρίως λόγω της μικρής διαφοράς αγωγιμότητας ανάμεσα στην LRS και HRS. Μπορούμε έτσι να συμπεράνουμε ότι η ανάγνωση για αυτές τις συνθήκες είναι σημαντικά πιο δύσκολη καθώς τα μικρότερα ρεύματα είναι και πιο επιρρεπή σε σφάλματα θορύβου.



Σχήμα 2.6 Καμπύλη Χαρακτηρισμού I-V ενός στοιχείου CBRAM (sample No. 210-13-1). Threshold Mode @ 0.3V



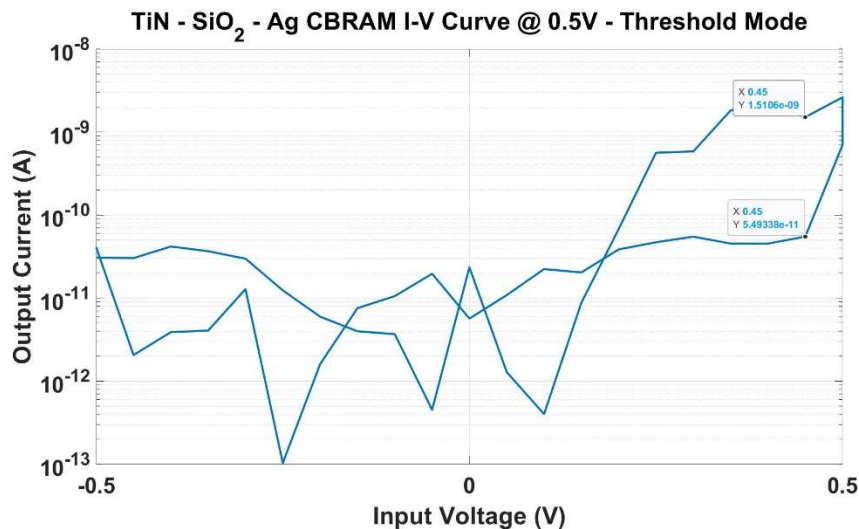
Σχήμα 2.7 Σύγκριση Καμπυλών I-V σε Bipolar (μπλε) και Threshold (κόκκινο) modes για το ίδιο στοιχείο μνήμης (sample No. 210-21-3)

Στο Σχήμα 2.7 βλέπουμε πως μπορεί το ίδιο στοιχείο μνήμης να παρουσιάσει διαφορετικό τρόπο εναλλαγής, ανάλογα με το μέγεθος της τάσης που του ασκούμε. Σε μικρότερη τάση παρουσιάζει λειτουργία κατωφλίου ενώ σε μεγαλύτερη διπολική λειτουργία. Παρατηρούμε επίσης αρκετά μεγαλύτερο παράθυρο μνήμης από αυτό του Σχήματος 2.6 και μάλιστα για μικρότερη τάση εναλλαγής. Πέραν του γεγονότος ότι αυτό το στοιχείο έχει καλύτερη απόδοση από το αντίστοιχο του Σχήματος 2.6, ως προς το παράθυρο μνήμης και την τάση μεταβολής, παρατηρούμε πως για ακριβώς τις ίδιες παραμέτρους χαρακτηρισμού, οι CBRAMs συμπεριφέρονται διαφορετικά. Ένα ακραίο παράδειγμα του παραπάνω ισχυρισμού φαίνεται στο Σχήμα 2.8. Με μέγιστη τάση τα 0.5V, παρά το γεγονός ότι για όλες τις προηγούμενες περιπτώσεις προέκυψε διπολική λειτουργία, το συγκεκριμένο στοιχείο λειτουργεί σε λειτουργία κατωφλίου.

Τέλος, παρατηρούμε ότι σε όλες τις περιπτώσεις ο αρνητικός κλάδος παρουσιάζει πάντα πολύ μικρή αγωγιμότητα, με ρεύματα που σπανίως ξεπερνάνε την τάξη των 10^{-8} A. Αυτό μας παραπέμπει σε μία συμπεριφορά αυτοανόρθωσης⁹ (self-rectifying) η οποία μπορεί να βρει χρήσεις σε εφαρμογές αλληλοδιασταυρούμενων δικτυωμάτων (crossbars), όπως θα δούμε στα επόμενα κεφάλαια.

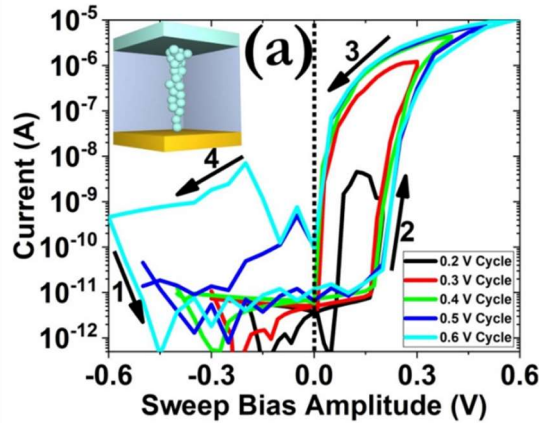
Συμπεραίνουμε:

- Οι μνήμες όπου κατασκευάσαμε μπορούν να λειτουργήσουν σε διπολική ή λειτουργία κατωφλίου για μεγαλύτερη ($\sim 0.5V$) ή μικρότερη ($\sim 0.3V$) μέγιστη τάση αντίστοιχα.
- Στη μεταβολή κατωφλίου το παράθυρο μνήμης είναι μεγαλύτερο και προκύπτει για μεγαλύτερα ρεύματα
- Οι CBRAMs δεν είναι πανομοιότυπες. Κάθε στοιχείο παρουσιάζει διαφορετική συμπεριφορά (δηλαδή τάση εναλλαγής, παράθυρο μνήμης ακόμη και τρόπο εναλλαγής) ακόμη και με πανομοιότυπες συνθήκες.
- Οι μνήμες παρουσιάζουν μία συμπεριφορά αυτό ανόρθωσης. Το ρεύμα έχει προτίμηση διάδοσης από το άνω προς το κάτω ηλεκτρόδιο.



Σχήμα 2.8 Καμπύλη Χαρακτηρισμού I-V ενός στοιχείου CBRAM (sample No. 210-11-3). Threshold Mode @ 0.5V

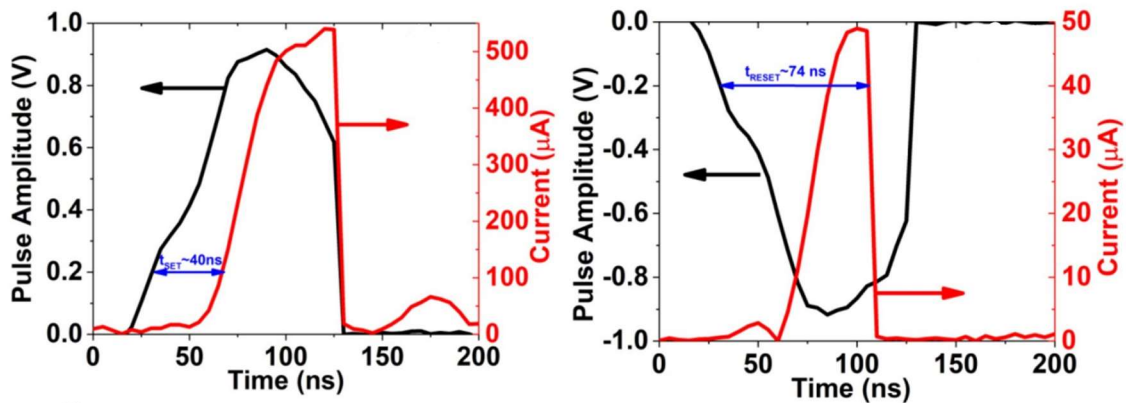
⁹ Φαίνεται πως στα CBRAM όπου εξετάσαμε, πως το ρεύμα έχει προτίμηση διάδοσης από το top στο bottom ηλεκτρόδιο, και όχι αντίθετη.



Σχήμα 2.9 Καμπύλες I-V για διάφορους κύκλους τάσης από 0.3V ως 0.6V σε μνήμη TiN – SiO₂ – Ag [6]

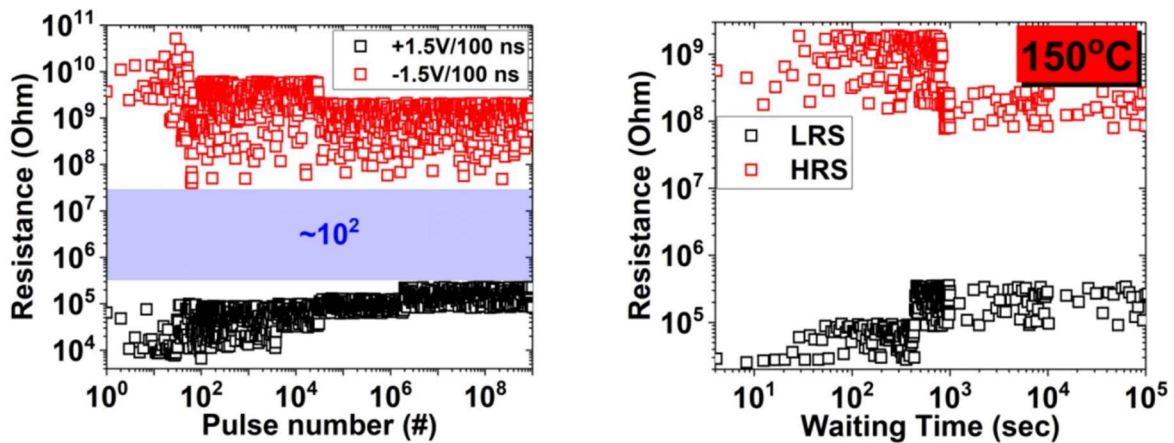
2.3.5 Χαρακτηρισμός Παλμικής Τάσης

Οι Μπούσουλας, Σακελλαρόπουλος και λοιποί συνεργάτες [6], πραγματοποίησαν πειράματα χαρακτηρισμού της «κινητικής μεταβολής» (kinetic switching), εφαρμόζοντας τετραγωνικούς παλμούς πλάτους $\pm 1V$ και χρονικής διάρκειας 100 ns. Παρατηρήθηκε, όπως φαίνεται στο Σχήμα 2.10 πως μία μνήμη TiN – SiO₂ – Ag, σε εφαρμογή παλμικής τάσης χρειάζεται για τις διαδικασίες SET και RESET 40 και 74 ns αντίστοιχα, καθιστώντας τις ταχύτερες σε εφαρμογές εγγραφής σε πραγματικό χρόνο.



Σχήμα 2.10 Διαδικασίες SET (αριστερά) και RESET (δεξιά) για μνήμη TiN - SiO₂ – Ag [6]

Επιπλέον πραγματοποιήθηκαν πειράματα αντοχής σε παλμικές τάσεις, πάνω σε αχρησιμοποίητες μνήμες, προκειμένου να διαπιστωθεί η συμπεριφορά των CBRAM σε συνθήκες κανονικής – παρατεταμένης λειτουργίας (όπως για παράδειγμα ως RAM ενός υπολογιστή, ή ακόμη και σε νευρομορφικές εφαρμογές). Παρόλου που διαπιστώθηκαν μεταβολές στα αποτελέσματα από στοιχείο σε στοιχείο, κάτι που οφείλεται στη στοχαστικότητα του φαινομένου της ελεγχόμενης κατάρρευσης του διηλεκτρικού (δημιουργία αγωγίου νήματος – Σχήμα 2.2), μετρήθηκε ένας λόγος μεταβολής $\frac{Resistance_{off}}{Resistance_{on}} = 10^2$ για 10^9 κύκλους εγγραφής (Σχήμα 2.11 - αριστερά).



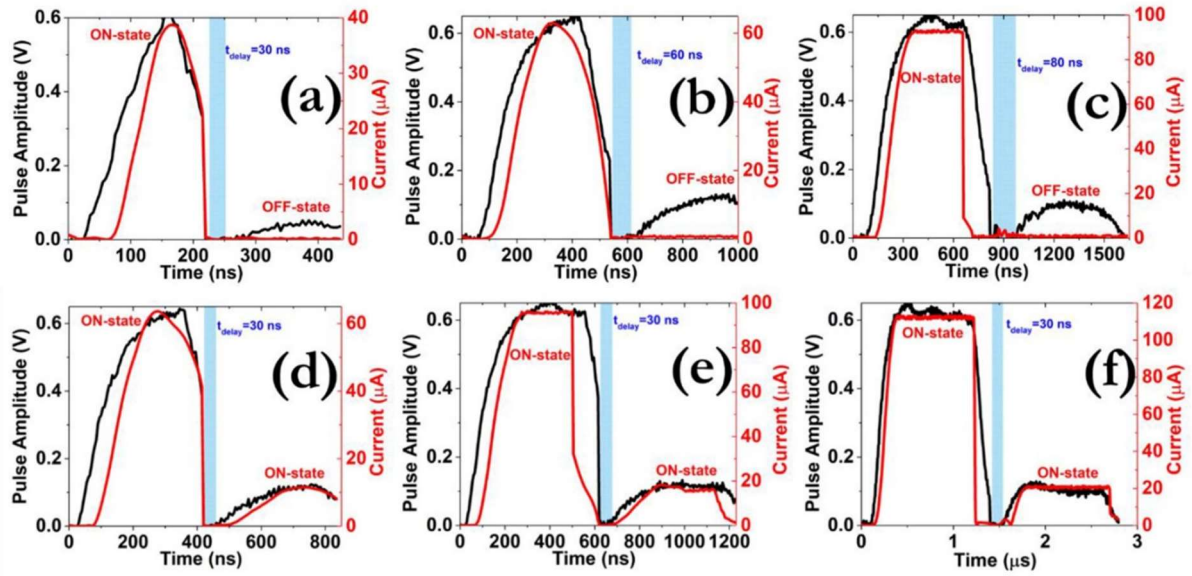
Σχήμα 2.11 Πείραμα Παλμικής Αντοχής σε θερμοκρασία δωματίου (αριστερά) και Retention στους 150 °C (δεξιά)

Όσον αφορά την μνημονική συμπεριφορά (retention) και με βάση το δεξί μέρος του Σχήματος 2.11, παρά την αρχική αλλοίωση της κατάστασης της μνήμης, τόσο στην LRS όσο και στην HRS, παρατηρούμε πως διατηρείται ένας λόγος αντιστάσεων περί τα 10^3 , μετά από 10^5 δευτερόλεπτα, ακόμη και για υψηλές θερμοκρασίες κοντά στους 150 °C. Έτσι μπορούμε να θεωρήσουμε ότι οι μνήμες αυτές λειτουργούν άριστα και σε συνθήκες μακροχρόνιας αποθήκευσης.

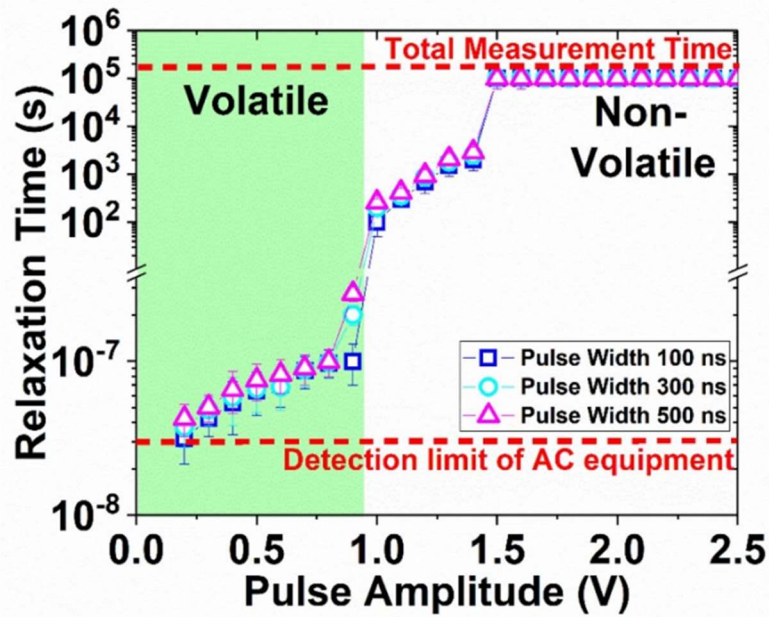
Το τελευταίο πείραμα που πραγματοποιήθηκε είναι αυτό της χαλάρωσης (relaxation). Ορίστηκε ως χρόνος χαλάρωσης η χρονική διάρκεια που θα χρειαστεί το αγώγιμο νήμα προκειμένου να μεταπέσει στην OFF κατάστασή του (δηλαδή να «κοπεί») όταν η μνήμη λειτουργεί σε καθεστώς μεταβολής κατωφλίου. Ένας μικρός χρόνος χαλάρωσης θεωρείται επιτακτική ανάγκη όσον αφορά εφαρμογές πάνω σε αρχιτεκτονική Crossbar, η οποία απαιτεί έναν μικρό χρόνο χαλάρωσης προκειμένου να αποφευχθούν σφάλματα κατά τη διαδικασία ανάγνωσης της συστοιχίας της μνήμης. Επιπλέον η δυναμική του χρόνου χαλάρωσης αποτελεί σημαντικό χαρακτηριστικό όσον αφορά νευρωνικές εφαρμογές όπως θα δούμε σε επόμενο κεφάλαιο.

Η μέθοδος όπου χρησιμοποιήθηκε για τα πειράματα της χαλάρωσης ήταν, για διαφορετικές διάρκειες παλμού εγγραφής (100ns – 1μs) και τάσεις (0.3V – 1V), εφαρμόζαμε έναν δεύτερο παλμό ανάγνωσης, πολύ μικρότερης τάσης, και μεταβάλλαμε το κενό μεταξύ των δύο αυτών παλμών, με σταθερό βήμα, μέχρι ο δεύτερος παλμός να μη προκαλέσει ροή ρεύματος μέσα από τη μνήμη. Η μέθοδος αυτή αναπαρίσταται πολύ καλύτερα σχηματικά στο Σχήμα 2.12. Στο Σχήμα 2.13 βλέπουμε μία γραφική αναπαράσταση των αποτελεσμάτων.

Παρατηρούμε επίσης ότι μετά από κάποιες συνθήκες η μνήμη εξακολουθεί να βρίσκεται στην ON κατάσταση (LRS) ανεξάρτητα από την απόσταση των παλμών εγγραφής και ανάγνωσης. Στις περιπτώσεις αυτές με ασφάλεια να θεωρήσουμε ότι λειτουργεί πλέον στη μη πτητική της λειτουργία (διπολική μεταβολή).



Σχήμα 2.12 Αναπαράσταση της μεθόδου μέτρησης του χρόνου χαλάρωσης. Παρατηρούμε ότι στα Σχήματα a-c δεν περνάει ρεύμα κατά την ανάγνωση, και έτσι θεωρούμε την αυτές τις χρονικές αποστάσεις ως έγκυρους χρόνους χαλάρωσης. Στα Σχήματα d-f παιρνάει ρεύμα κατά την ανάγνωση, επιβάλλοντας περαιτέρω διερεύνηση για τον σωστό υπολογισμό του χρόνου χαλάρωσης.

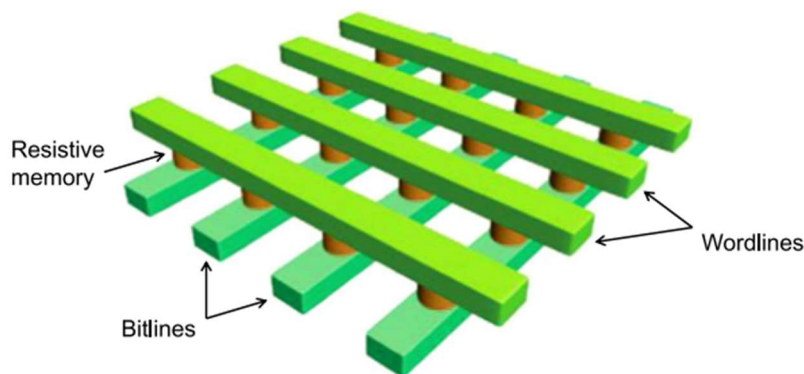


Σχήμα 2.13 Αποτελέσματα των πειραμάτων μέτρησης του χρόνου χαλάρωσης.

3 Σχεδιασμός Διάταξης Crossbar και Θεωρητικά Στοιχεία

Οι RRAM, όπως και οι υπόλοιπες τεχνολογίες που αναφέραμε στο πρώτο κεφάλαιο ανήκουν σε μία ευρύτερη κατηγορία μνημών όπου η πληροφορία αποθηκεύεται μέσω μεταβολής της αντίστασης. Για παράδειγμα στις PCMs η αντίσταση αλλάζει λόγω αλλαγής φάσης του ενεργού υλικού, στις STT-RAMs το spin των ηλεκτρονίων μπορεί να προσανατολίσει τη μαγνητική δίοδο (Magnetic Tunnel Junction) προκαλώντας αύξηση της αγωγιμότητας ή όπως αναλύσαμε στο προηγούμενο κεφάλαιο, με τη δημιουργία ενός αγώγιμου νήματος στις RRAM.

Το κοινό χαρακτηριστικό των παραπάνω μνημών, σε σύγκριση με τις ήδη εν χρήση DRAM ή τις FLASH είναι πως έχουν δύο τερματικά (top και bottom ηλεκτρόδια) αντί για τρία. Το γεγονός αυτό επιτρέπει στις μνήμες αυτές να συμπεριληφθούν εύκολα σε διατάξεις crossbar, κατασκευασμένες από τη λιθογραφία. Η μεγάλη πυκνότητα των οριζοντίων και κάθετων γραμμών (word lines και bit lines¹⁰), επιτρέπει την δημιουργία στοιχείων εξαιρετικά μικρού εμβαδού (bit area), μόλις $4F^2$, όπου F είναι το ελάχιστο μήκος ενός χαρακτηριστικού προσπελάσιμου από τη λιθογραφία [8]. Στο Σχήμα 3.1 μπορούμε να δούμε μία βασική απεικόνιση αυτής της αρχιτεκτονικής.



Σχήμα 3.1 Αρχιτεκτονική Crossbar για RRAMs [8]

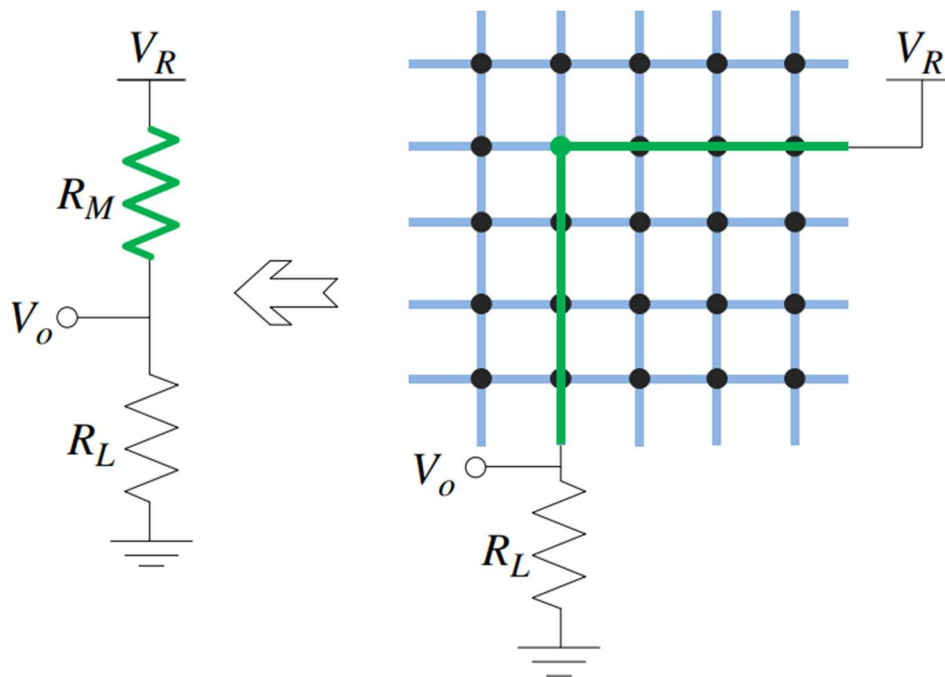
3.1 Το πρόβλημα των κρυφών ρευμάτων

Για να καταλάβουμε τη φύση αυτού του προβλήματος των κρυφών ρευμάτων, αρκεί πρώτα να καταλάβουμε πως δουλεύει η πιο απλή διάταξη crossbar. Το crossbar αποτελείται από πολλές παράλληλες εισόδους και εξόδους, οι οποίες συνδέονται μεταξύ τους μόνο στα σημεία διασύνδεσης, τα οποία στην περίπτωσή μας αποτελούνται από τα ίδια τα στοιχεία μνήμης. Μπορεί κάποιος να φανταστεί τις θέσεις των στοιχείων διασύνδεσης, όπως φαίνονται στο Σχήμα 3.1, σαν μία αναπαράσταση σημείων σε ένα καρτεσιανό σύστημα συντεταγμένων. Στο σύστημα αυτό οι γραμμές εισόδου (word lines – WLs) αναπαρίστανται από την τεταγμένη (X – άξονας) ενώ οι γραμμές εξόδου (bit lines – BLs) από τη τεταγμένη (Y – άξονας). Επιλέγουμε

¹⁰ Συνήθως θεωρούμε WLs τις οριζόντιες και BLs τις κάθετες γραμμές του Crossbar. Ο ορισμός αυτός ενδέχεται να εναλλάσσεται στη βιβλιογραφία.

με πιο στοιχείο θέλουμε να αλληλοεπιδράσουμε απλά εφαρμόζοντας την επιθυμητή τάση (εγγραφής, ανάγνωσης ή και διαγραφής) στην αντίστοιχη WL και γειώνοντας την αντίστοιχη BL, με άλλα λόγια δίνοντας τις συντεταγμένες του σημείου αυτού. Στο Σχήμα 3.2 μπορούμε να δούμε και γραφικά τη διαδικασία αυτή.

Κατά τη λειτουργία εγγραφής, επιλέγουμε το επιθυμητό στοιχείο της μνήμης και «γράφουμε» την πληροφορία, απλώς περνώντας μία τάση αρκετά μεγάλη, έτσι ώστε το στοιχείο να αυξήσει την αγωγιμότητά του, δηλαδή να μεταβεί στην LRS – 1 ή αντιστρόφως μία αρκούντως μεγάλη αρνητική τάση προκειμένου το στοιχείο να βρεθεί στην HRS – 0 (erase).

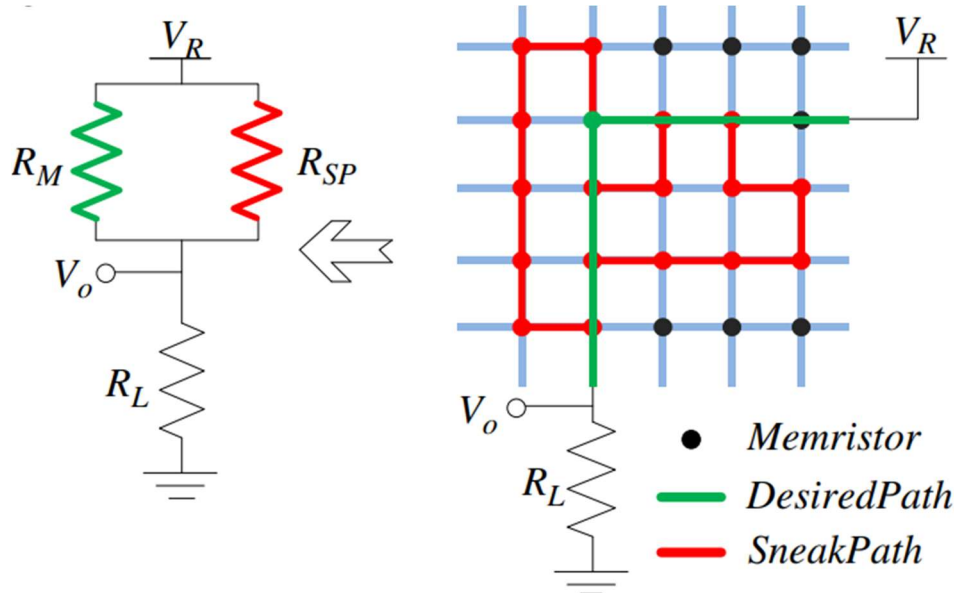


Σχήμα 3.2 Επιλογή και Εγγραφή - Ανάγνωση ενός Στοιχείου μνήμης σε Crossbar [9]

Η ανάγνωση κρίνεται ως πιο δύσκολη διαδικασία, καθώς καλούμαστε να μετρήσουμε την αγωγιμότητα ενός στοιχείου, μέσα σε μία δομή επί της ουσίας αποκλειστικά αποτελούμενη από παράλληλα συνδεδεμένους αντιστάτες. Η επιλογή του σωστού στοιχείου γίνεται με τον ίδιο τρόπο, αλλά η τάση που εφαρμόζεται στη WL αυτή τη φορά πρέπει να είναι μικρότερη, έτσι ώστε να μην προκαλέσει κάποια εναλλαγή στο επιλεγμένο (ή στα γειτονικά) στοιχείο, καταστρέφοντας τη πληροφορία.

Το παραπάνω μας εισάγει κατευθείαν στο πρόβλημα των κρυφών ρευμάτων. Πρόκειται για ρεύματα τα οποία ρέουν από μονοπάτια παράλληλα στο επιθυμητό, γεγονός αναπόφευκτο καθώς μία διάταξη αποτελούμενη μόνο από RRAMs δεν δύναται να ανακόψει τη διαρροή αυτή. Η περίπτωση του Σχήματος 3.2 αποτελεί μία ιδανική περίπτωση εγγραφής και ανάγνωσης στα οποία για χάρη του παραδείγματος θεωρήσαμε ότι τα κρυφά ρεύματα διαρροής είναι μηδενικά. Μία πιο ρεαλιστική απεικόνιση φαίνεται στο Σχήμα 3.3. Η προστιθέμενη

αντίσταση λόγω των κρυφών ρευμάτων, στη καλύτερη περίπτωση θα μας δώσει απλώς λάθος αποτελέσματα, με πολύ πιθανή την πρόκληση μη επιθυμητής μεταβολής της κατάστασης σε κάποιο γειτονικό στοιχείο. Στις περισσότερες περιπτώσεις τα ρεύματα αυτά μικραίνουν σημαντικά το παράθυρο θορύβου και περιορίζουν το μέγιστο μέγεθος που μπορεί να έχει ένα τέτοιο δίκτυωμα (μεγαλύτερη διάταξη σημαίνει και περισσότερα κρυφά μονοπάτια ρεύματος).



Σχήμα 3.3 Επιλογή και Εγγραφή - Ανάγνωση ενός στοιχείου μνήμης Crossbar. Συμπεριλαμβάνεται το πρόβλημα των κρυφών ρευμάτων [9]

Το σοβαρότερο μέρος του προβλήματος είναι πως η έντασή του δεν είναι σταθερή για δεδομένη διάταξη καθώς παρουσιάζει άμεση εξάρτηση από τη πληροφορία η οποία εμπεριέχεται μέσα στη διάταξη. Για παράδειγμα, ένα crossbar με όλα τα υπόλοιπα στοιχεία πέρα από το επιλεγμένο στην HRS θα έχει πολύ λίγα κρυφά μονοπάτια ρεύματος, καθώς οι πολλές υψηλές αντιστάσεις που προκύπτουν για την HRS δρουν σαν εμπόδια στη ροή των κρυφών ρευμάτων. Τα κρυφά ρεύματα μεγιστοποιούνται στην άλλη ακραία περίπτωση όπου όλα τα στοιχεία βρίσκονται στην LRS, καθώς δεν υπάρχει κάποιο σημαντικό εμπόδιο στη διάδοση ενός σήματος από οποιοδήποτε μονοπάτι [9].

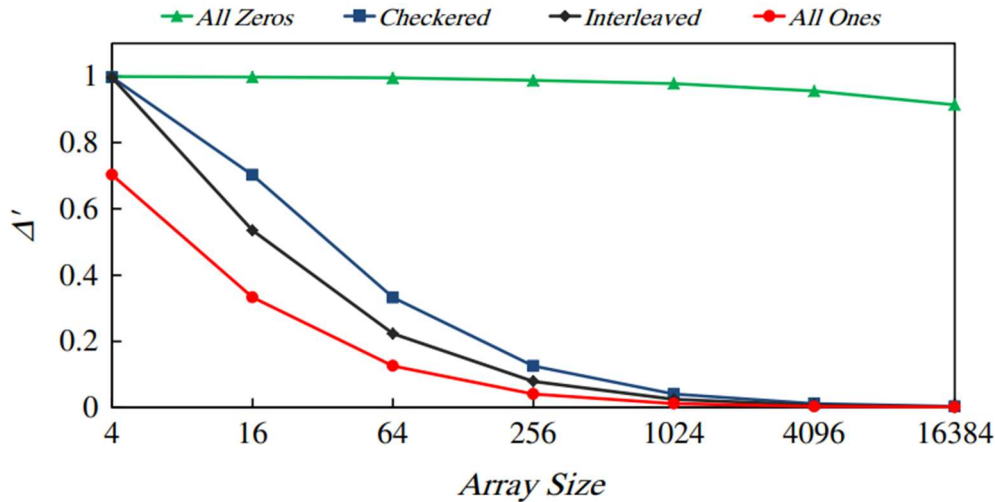
Σε αυτό το σημείο είναι σημαντικό να ορίσουμε το παράθυρο θορύβου όπου προαναφέραμε. Το ολικό παράθυρο θορύβου (noise margin) ορίζεται ως:

$$\Delta = V_{one} - V_{zero} \quad (3.1.1)$$

Όπου V οι τάσεις που μετράμε στον αντιστάτη αίσθησης στην έξοδο του crossbar. Ορίζουμε επίσης τον λόγο θορύβου ως:

$$\Delta' = \frac{\Delta_{array}}{\Delta_{device}} \quad (3.1.2)$$

Η τιμή στον παρονομαστή αντιστοιχεί στο παράθυρο ενός μεμονωμένου στοιχείου μνήμης, ενός η τιμή του αριθμητή στο ίδιο στοιχείο, όταν αυτό είναι κομμάτι ενός crossbar. Στο γράφημα του Σχήματος 3.4 βλέπουμε την επίδραση, τόσο του μεγέθους της διάταξης, όσο και του περιεχομένου της στο παράθυρο θορύβου.



Σχήμα 3.4 Παράθυρο θορύβου σε διάταξη crossbar συναρτήσει του μεγέθους και του περιεχομένου της διάταξης. Όλες οι γραμμές πλην των επιλεγμένων είναι σε υψηλή εμπέδηση (floating)

3.2 Λύσεις στο πρόβλημα των κρυφών ρευμάτων

Ένα ακόμη μέγεθος που μπορεί να ποσοτικοποιήσει την επίδοση μίας διάταξης crossbar κατά τη διαδικασία της ανάγνωσης είναι το λεγόμενο παράθυρο ανάγνωσης (Read Margin). Το μέγεθος αυτό ορίζεται ως:

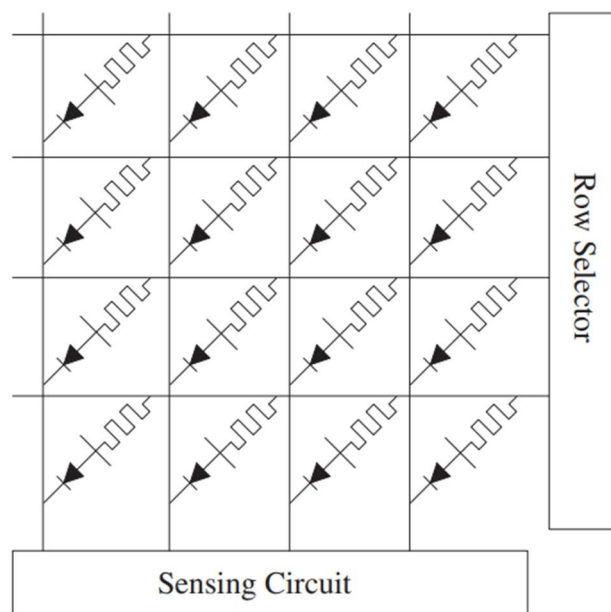
$$RM = \frac{\Delta V_{out}}{V_{read}} = \frac{V_{out}^{LRS} - V_{out}^{HRS}}{V_{WS}} \quad (3.2.1)$$

Όπου, V_{WS} είναι η τάση όπου ασκούμε στην επιλεγμένη WL και V_{out} η τάση στην αντίσταση εξόδου. Κατανοούμε πως κατά την διαδικασία ανάγνωσης του στοιχείου μνήμης, το παραπάνω μέγεθος θα πρέπει ιδανικά να μεγιστοποιείται (όσο το δυνατόν) [10]. Με μερικές απλές πράξεις μπορούμε να διαπιστώσουμε ότι σε μία μεγαλύτερη διάταξη, με αυξημένο παράγοντα κρυφών ρευμάτων, οι τάσεις εξόδου στον αριθμητή της εξίσωσης (3.2.1) θα είναι μικρότερες, λόγω των παράλληλων παρασιτικών αντιστάσεων που λειτουργούν ως διαιρέτες τάσης.

Προκειμένου να αντιμετωπιστεί αυτό το πρόβλημα θα πρέπει με κάποιο τρόπο να ανακόψουμε τα κρυφά ρεύματα. Αρκεί να εφαρμόσουμε κάποιο εμπόδιο στην πορεία τους.

3.2.1 Χρήση των transistors σαν διακόπτες επιλογής (Gating)

Με τον όρο Gating ή αλλιώς «εφαρμογή πυλών» στα crossbars εννοούμε την εισαγωγή «φυσικών» εμποδίων, τα οποία απαγορεύουν ροή του ρεύματος προς κάποια συγκεκριμένη κατεύθυνση. Ένα τέτοιο παράδειγμα είναι η χρήση μίας αρχιτεκτονικής **μίας μνήμης – μίας διόδου** (1 Diode 1 Memristor – 1D1M), όπως φαίνεται στο Σχήμα 3.5. Η εισαγωγή των διόδων στο σύστημα θα αυξήσει την καθυστέρηση και τη χωρητικότητα (capacitance) του συστήματος, αποκόποντας έτσι τα κρυφά ρεύματα.



Σχήμα 3.5 Crossbar με χρήση της αρχιτεκτονικής 1D1M [9]

Μία τέτοια αρχιτεκτονική παράγει άριστα αποτελέσματα κατά τη διαδικασία της ανάγνωσης ή της εγγραφής. Στο Σχήμα 3.5, κατανοούμε πως το ρεύμα δύναται μόνο να κινηθεί από τους επιλογής γραμμής εισόδου – WL (Row Selectors στο Σχήμα 3.5) προς το κύκλωμα ανάγνωσης (Sensing circuit), επιτρέποντας την ελεύθερη από κρυφά ρεύματα διεξαγωγή των παραπάνω διεργασιών. Για την σωστή λειτουργία της διάταξης όμως χρειάζεται και η λειτουργία διαγραφής, η οποία στην περίπτωση των μνημών διπολικής μεταβολής (σαν τις υπό μελέτη CBRAM) απαιτεί και την αντίθετη πολικότητα. Είναι προφανές πως η ύπαρξη και μόνο της διόδου απαγορεύει τη διεξαγωγή μίας τέτοιας διαδικασίας, καθιστώντας την αρχιτεκτονική αυτή ακατάλληλη.

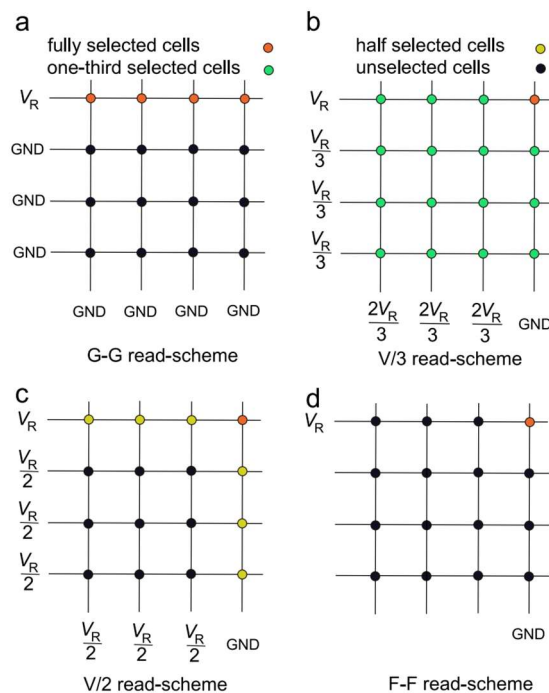
Στην ίδια λογική μπορούμε να συνδέσουμε το κάθε στοιχείο μνήμης σε σειρά με την πηγή ενός transistor (1 Transistor 1 Memristor – 1T1M). Τα transistors μπορούν να λειτουργήσουν ως διακόπτες, ανοίγοντας το κύκλωμα σε μη επιλεγμένες γραμμές, αποκόποντας σχεδόν ολοκληρωτικά τα κρυφά ρεύματα. Το πρόβλημα με αυτή τη λύση είναι ότι χάνεται ένα από

τα βασικά πλεονεκτήματα της διάταξης, η υψηλή πυκνότητα ολοκλήρωσης. Μπορούμε εναλλακτικά να χρησιμοποιήσουμε μικρότερα transistors στη κατεύθυνση αυτή (όπως αυτά που χρησιμοποιούνται σε state-of-the-art διατάξεις), τα οποία όμως δρουν ως «βαλβίδες με απώλειες», περιορίζοντας τα κρυφά ρεύματα αλλά όχι αποκόποντάς τα. Τέλος πρέπει να σημειωθεί ότι για τη χρήση αυτής της αρχιτεκτονικής χρειαζόμαστε δύο παροχές προκειμένου να επιλέξουμε τη επιθυμητή γραμμή εισόδου. Η μία για τη εγγραφή ή ανάγνωση πληροφορίας στη μνήμη, όπως ακριβώς οι επιλογείς της WL στο προηγούμενο παράδειγμα. Η δεύτερη αφορά τον έλεγχο των πυλών των transistor επιλογής κάτι που καθιστά το κύκλωμα πιο περίπλοκο και ενεργειακά πιο κοστοβόρο [9] [11].

3.2.2 Χρήση των ιδιοτήτων της CBRAM

Στη βιβλιογραφία υπάρχει μία πληθώρα μεθόδων για την ανάγνωση δικτυωμάτων crossbar. Αυτές οι μέθοδοι έχουν ως σκοπό να περιορίσουν τις πηγές των κρυφών ρευμάτων μέσα στο crossbar ή αλλιώς τις πηγές διαταραχής, αντί για τα ρεύματα αυτά καθ' αυτά, περιορίζοντας έτσι την αρνητική τους επίπτωση στο Read Margin. Προσπαιτούμενο για να έχει νόημα η εφαρμογή αυτών είναι ο αυτό-ανορθωτικός χαρακτήρας των στοιχείων όπου χρησιμοποιούμε. Οι CBRAM του εργαστηρίου μας, όπως φαίνεται στο Σχήμα 2.7, πληρούν αυτή τη προϋπόθεση.

Οι βασικότερες μέθοδοι είναι οι $V/2$, $V/3$, GN-GN και FL-FL. Το πώς εφαρμόζονται φαίνεται γραφικώς στο Σχήμα 3.6.



Σχήμα 3.6 Πρωτόκολλα ανάγνωσης διατάξεων crossbar. A) GN-GN, b) V/3, c) V/2, d) FL-FL [11]

Κατά τη μέθοδο GN-GN (Σχήμα 3.6a), η τάση που ασκείται στα στοιχεία της επιλεγμένης γραμμής είναι ίση με την τάση ανάγνωσης V_R , ενώ τα υπόλοιπα, μη επιλεγμένα στοιχεία είναι γειωμένα. Κατανοούμε πως στην περίπτωση αυτή πηγές κρυφών ρευμάτων είναι αυτά τα επιλεγμένα στοιχεία.

Στα Σχήματα 3.6b και 3.6c βλέπουμε τις μεθόδους $V/3$ και $V/2$ αντίστοιχα (όπου V η τάση ανάγνωσης). Στη πρώτη περίπτωση, ασκούμε στα μη επιλεγμένα στοιχεία μία τάση ίση με $V/3$ (εμφανίζονται με πράσινο χρώμα), τα οποία είναι και υπεύθυνα για την παραγωγή κρυφών ρευμάτων. Στη δεύτερη περίπτωση, τα κρυφά ρεύματα οφείλονται στα ημι-επιλεγμένα στοιχεία των WL και BL (Σχήμα 3.6c – κίτρινο χρώμα), υπό τάση $V/2$.

Στη τελευταία μέθοδο, η παραγωγή των κρυφών ρευμάτων οφείλεται σε όλα τα μη επιλεγμένα στοιχεία που βρίσκονται σε καθεστώς υψηλής εμπέδησης.

Η παραπάνω ταξινόμηση των μεθόδων ανάγνωσης δεν έγινε τυχαία. Κατά τον σχεδιασμό ενός δικτύωματος crossbar πρέπει να έχουμε στο νου πέρα από τη μεγιστοποίηση του Read Margin, όσο το δυνατόν γίνεται, και την κατανάλωση ισχύος. Ο J. Zhou (και συνεργάτες) έδειξαν με τις προσομοιώσεις τους ότι το RM μεγιστοποιείται χρησιμοποιώντας τη μέθοδο GN-GN. Το ίδιο όμως και η κατανάλωση. Στο αντίποδα η κατανάλωση μειώνεται δραστικά χρησιμοποιώντας τη μέθοδο FL-FL, αλλά με μέγιστες διαταραχές λόγω κρυφών ρευμάτων [10], [12]. Κατανοούμε πως υπάρχει μία αντιστρόφως ανάλογη σχέση (trade-off) μεταξύ των δύο παραγόντων, την οποία πρέπει να λάβουμε οπωσδήποτε υπόψη.

Τέλος, θα πρέπει να σημειώσουμε ότι το παράθυρο ανάγνωσης βελτιστοποιείται αν χρησιμοποιηθεί μία αντίσταση ίση με $R_{sense} = \sqrt{R_{HRS} \cdot R_{LRS}}$ σε σειρά με την έξοδο της επιλεγμένης BL, και μετρήσουμε την πτώση τάσης (ή το ρεύμα) στην αντίσταση αυτή.

3.3 Σχεδιασμός μία λειτουργικής διάταξης crossbar

Στα πλαίσια της εργασίας αυτής σχεδιάστηκε με το λογισμικό Proteus[®] ένα δικτύωμα crossbar, μεγέθους 8×8 , με τη χρήση αντιστατών για προσομοίωση των στοιχείων μνήμης. Επιλέχθηκαν αντιστάτες, των οποίων την αντίσταση ορίζαμε ως την LRS και HRS των CBRAM κατά βούληση, καθώς το εν λόγω λογισμικό δεν υποστηρίζει (την ημερομηνία εγγραφής της εργασίας αυτής) προσομοίωση RRAMs. Παρόλα αυτά η χρήση του κρίνεται απαραίτητη για να προσομοιώσουμε τη διαδικασία ανάγνωσης.

Το κύκλωμα όπως φαίνεται στα σχέδια των ενθέτων I & II, αποτελείται από 3 διακριτά μέλη. Το ένα από αυτά είναι το ίδιο το crossbar (Σχήματα 3.7 & 3.8). Το δεύτερο μέλος αποτελείται από τους επιλογείς WL και BL, καθώς και το σύστημα επιβολής των μεθόδων ανάγνωσης

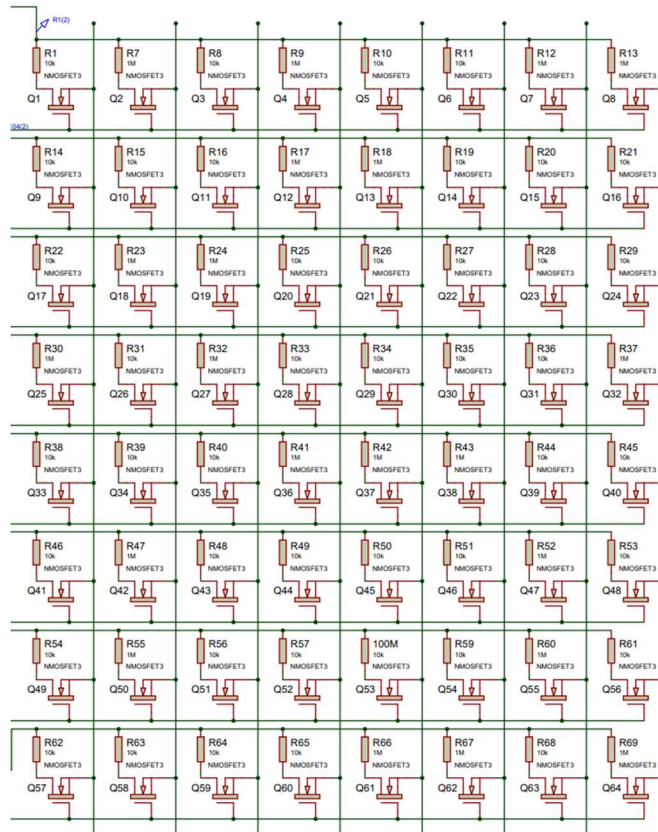
(Σχήμα 3.9), όπως αυτές ορίστηκαν στη προηγούμενη παράγραφο. Το τελευταίο μέλος περιλαμβάνει το σύστημα ανάγνωσης, δηλαδή την αντίσταση ανάγνωσης και έναν ενισχυτή (προκειμένου να γίνει πιο ευδιάκριτο το σήμα ανάγνωσης) (Σχήμα 3.10). Προαιρετικά συνδέουμε στην έξοδο το ενισχυτή ανάγνωσης μία φωτοδίοδο (LED) η οποία ενεργοποιείται ή παραμένει σβηστή για τις καταστάσεις HIGH και LOW στο επιλεγμένο στοιχείο μνήμης αντίστοιχα.

3.3.1 Μέλος I: Το Crossbar

Το ενεργό μέλος της διάταξης είναι και το πιο απλό στον σχεδιασμό του. Αποτελείται από τα WLs και BLs τα οποία συνδέονται μεταξύ τους μόνο μέσω των δύο τερματικών των στοιχείων μνήμης ή και του transistor στην περίπτωση της αρχιτεκτονικής 1T1M.



Σχήμα 3.7 Selectorless crossbar memory array



Σχήμα 3.8 1T1M crossbar memory array

Παρατηρούμε, όπως προαναφέραμε, ότι στην περίπτωση της αρχιτεκτονικής 1T1M απαιτείται μία παραπάνω παροχή, η οποία συνδέεται με τις πύλες των transistors. Τη γραμμή αυτή στη συνέχεια της εργασία αυτής θα τη λέμε γραμμή επιλογής (select line – SL).

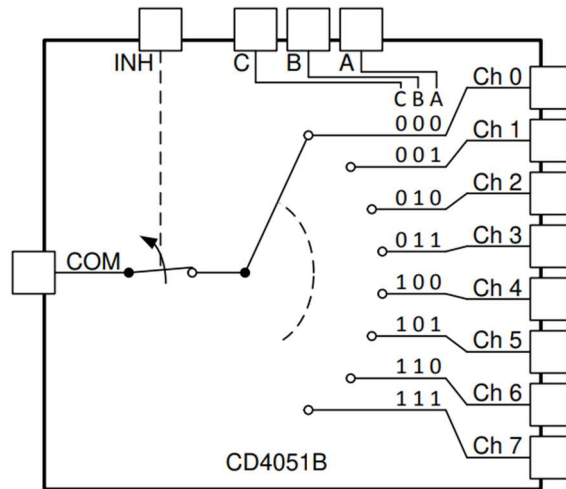
3.3.2 Μέλος II: Επιλογείς

Στα Σχήματα 3.10 και 3.11 μπορούμε να δούμε το σύστημα επιλογής WL, SL και BL. Μία μικρή διάταξη θα μπορούσε να ελεγχθεί (δηλαδή να επιλογή το επιθυμητό ενεργό στοιχείο) κατευθείαν από έναν μικρο-ελεγκτή¹¹. Γίνεται όμως εύκολα κατανοητό πως κάτι τέτοιο είναι αδύνατο να πραγματοποιηθεί για μεγαλύτερες διατάξεις, ειδικά αν πρέπει να συμπεριλάβουμε στο κύκλωμα και κάποιο σύστημα επιλογής μεθόδου ανάγνωσης ($V/3$ ή $V/2$).

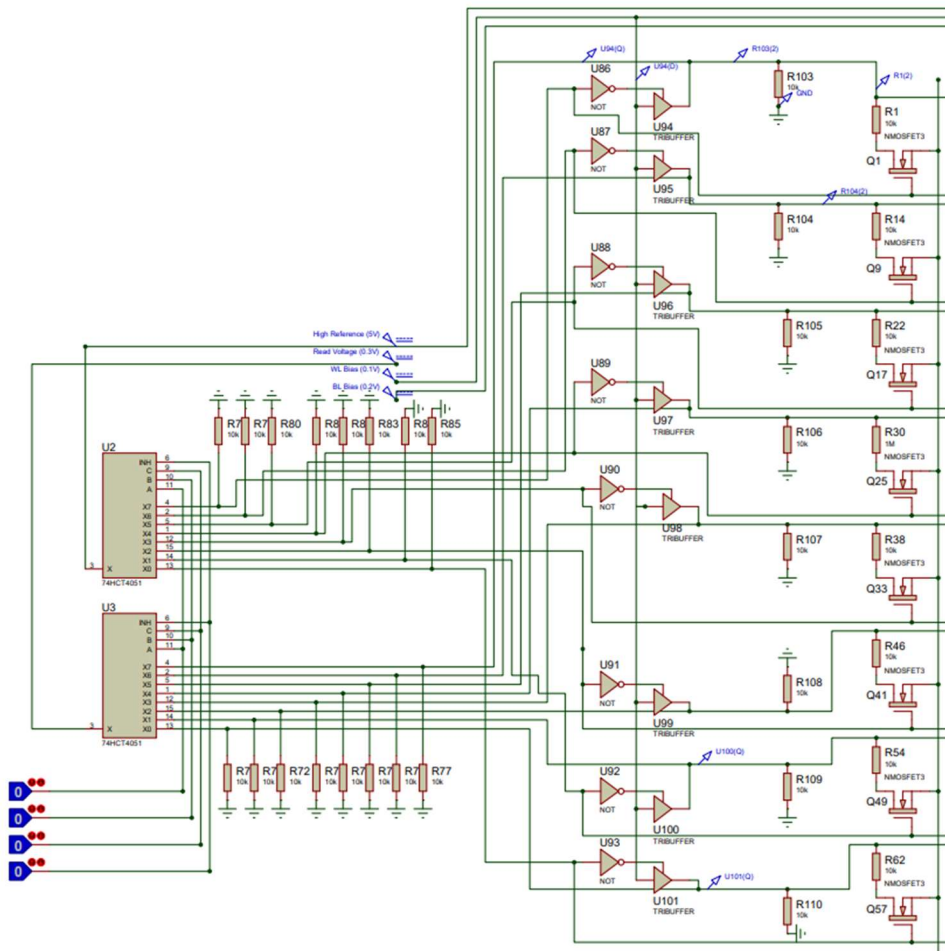
Γενικότερα, σκοπός μας είναι να ελέγξουμε μία διάταξη με όσο το δυνατό λιγότερες πύλες διεπαφής. Για να επιτευχθεί ο σκοπός αυτός, χρησιμοποιήθηκε ένα σύστημα με πολυπλέκτες – από-πολυπλέκτες (DEMUX). Επιλέξαμε για κάθε περίπτωση (WL – SL & BL) ένα ζεύγος παράλληλα συνδεδεμένων πολυπλεκτών τύπου CD4051 της Texas Instruments. Ένας τέτοιος πολυπλέκτης, χρησιμοποιώντας 3εις ψηφιακές πύλες μπορεί να στείλει κάποιο σήμα, αναλογικό ή και ψηφιακό, από μία κοινή πύλη (COM) σε ένα από τα 8 κανάλια (Ch0-Ch7) και αντίστροφα. Το σήμα αυτό μπορεί να είναι οποιασδήποτε τάσης στο εύρος των $V_{DD} - V_{EE}$ και

¹¹ Για την εργασία αυτή χρησιμοποιείται πάντα ο Arduino DUE

$V_{DD} - V_{SS}$, αναλογικής ή ψηφιακής μορφής. Επιπλέον, ο πολυπλέκτης, μπορεί να απενεργοποιηθεί (δηλαδή να βρεθούν όλα τα κανάλια σε υψηλή εμπέδηση) αν εφαρμόσουμε μία digital HIGH τάση στην πόλη INH (Σχήμα 3.9) [13].

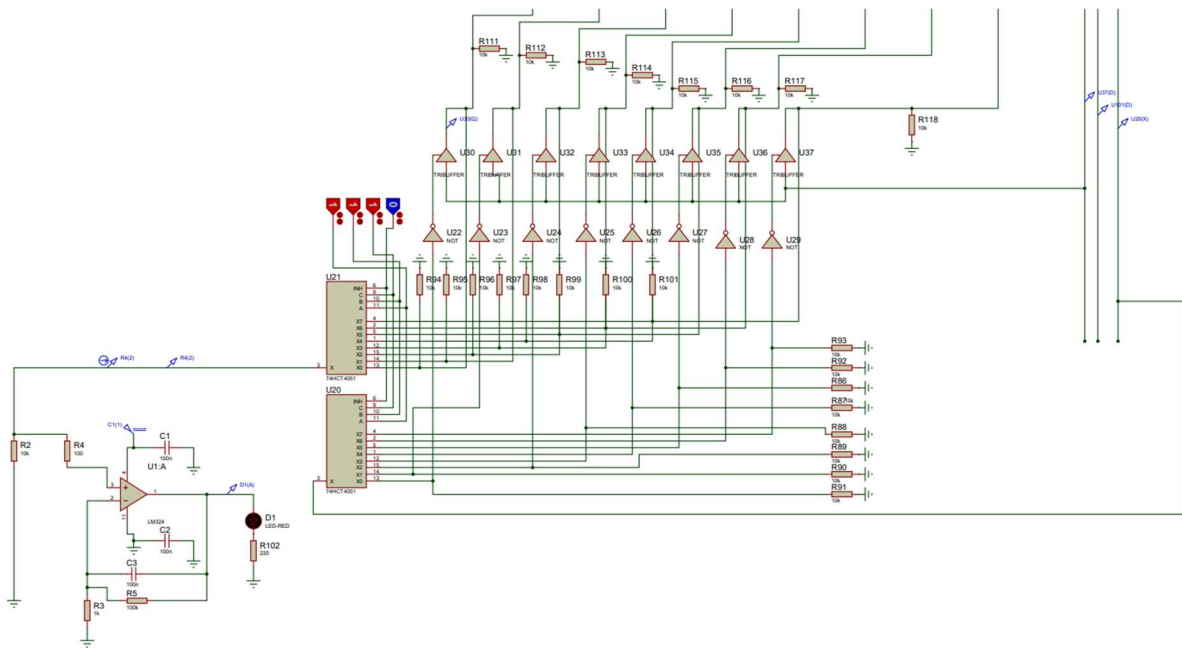


Σχήμα 3.9 CD4051B - Functional Diagram



Σχήμα 3.10 Σύστημα επιλογής WL & SL

Στο Σχήμα 3.10 ο κάτω πολυπλέκτης επιλέγει την επιθυμητή WL και στέλνει την τάση εγγραφής ή ανάγνωσης στη γραμμή αυτή. Ταυτόχρονα ο άνω πολυπλέκτης, ο οποίος βρίσκεται σε ψηφιακή λειτουργία ευθύνεται για την εφαρμογή τάσης μεθόδου ανάγνωσης (εδώ $V/3$) σε όλες τις υπόλοιπες γραμμές. Η τελευταία διαδικασία επιτυγχάνεται μέσω της χρήσης λογικής τριών καταστάσεων ως εξής. Ο πολυπλέκτης στέλνει ένα HIGH σήμα στα tribuffers¹² της επιλεγμένης γραμμής και LOW στα αντίστοιχα των μη επιλεγμένων γραμμών¹³. Τα tribuffers με την σειρά τους αφήνουν την τάση μεθόδου ανάγνωσης να περάσει σε όλες τις μη επιλεγμένες WLs ενώ αφήνουν το κανάλι επιλογής μεθόδου ανάγνωσης της επιλεγμένης γραμμής σε υψηλή εμπέδηση. Έτσι στην επιλεγμένη γραμμή περνάει πάντα το σήμα V ενώ ένα σήμα $V/3$ στις μη επιλεγμένες. Τέλος παρατηρούμε την ύπαρξη ορισμένων αντιστάσεων pull-down, μία για κάθε κανάλι του πολυπλέκτη, και άλλη μία για κάθε γραμμή μέσα στο crossbar. Οι πρώτες χρησιμοποιούνται για να αποφύγουμε τις καταστάσεις υψηλής εμπέδησης στις μη επιλεγμένες γραμμές, λόγω της λειτουργίας του CD4051B. Οι δεύτερες χρησιμοποιούνται για την περίπτωση που επιλέξουμε να εφαρμόσουμε τη μέθοδο GN-GN (αν δεν εφαρμόσουμε τάση στην είσοδο των tribuffers, οι αντιστάσεις αυτές ρίχνουν την έξοδο αυτών στα $0V - LOW$). Επιπλέον στην περίπτωση χρήσης transistors επιλογής, αρκεί να συνδέσουμε την κάθε SL με την αντίστοιχη έξοδο του άνω πολυπλέκτη.



Σχήμα 3.11 Σύστημα επιλογής BL

Ένα αντίστοιχο κύκλωμα χρησιμοποιείται για την επιλογή BL. Η βασική διαφορά είναι ότι στην περίπτωση αυτή, ο αναλογικός πολυπλέκτης (επιλογής γραμμής) μεταφέρει το σήμα ανάγνωσης (ή εγγραφής) από τα κανάλια στην κοινή πύλη, η οποία συνδέεται με τον ενισχυτή ανάγνωσης, με το σύστημα του ψηφιακού πολυπλέκτη (εφαρμογής bias) και τα tribuffers να εφαρμόζουν τη αντίστοιχη μέθοδο ανάγνωσης (εδώ: $2V/3$).

¹² Επιλογείς λογικής τριών καταστάσεων

¹³ Οι tribuffers του λογισμικού Proteus ενεργοποιούνται με τάση LOW. Απαιτείται η αντιστροφή του σήματος με πύλες NOT.

3.3.3 Μέλος III: Ενισχυτής ανάγνωσης

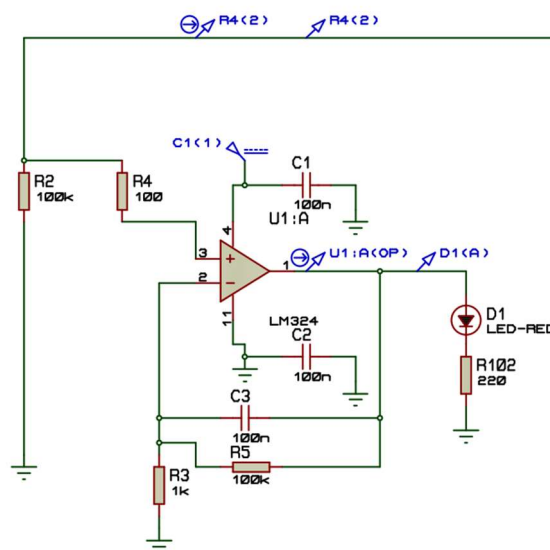
Στις προηγούμενες παραγράφους αναφέραμε ότι, για δεδομένη μέθοδο ανάγνωσης και δεδομένη την ύπαρξη ή μη transistor επιλογής, το μέγιστο παράθυρο ανάγνωσης επιτυγχάνεται για $R_{sense} = \sqrt{R_{HRS} \cdot R_{LRS}}$. Ας μελετήσουμε την περίπτωση του ενθέτου II, του οποίου η διάταξη είναι πιο σταθερή, λόγω εφαρμογής της αρχιτεκτονικής 1T1M. Για να προσομοιάσουμε τη διαδικασία του read-out επιλέξαμε αντιστάτες των $1M\Omega$ και $10k\Omega$, τιμές που αντιστοιχούν στις HRS και LRS των υπό μελέτη CBRAM.

$$R_{sense} = \sqrt{10 \cdot 10^3 \times 1 \cdot 10^6} = \sqrt{10^{10}} = 10^5 \Omega = 100k\Omega$$

Η μέτρηση πραγματοποιείται μετρώντας το ρεύμα που διέρχεται από αυτή την αντίσταση, μέσω της χρήσης ενός τελεστικού ενισχυτή, ο οποίος «μετράει» και «ενισχύει» την πτώση τάσης πάνω της.

Για έναν διαφορικό ενισχυτή σε μη αναστρέφουσα συνδεσμολογία, σαν και αυτόν του Σχήματος 3.12, γνωρίζουμε ότι η τάση στην έξοδο δίνεται από την σχέση (3.3.1)

$$V_{out} = \left(1 + \frac{R_5}{R_3}\right) V_{in} \quad (3.3.1)$$



Σχήμα 3.12 Ενισχυτής ανάγνωσης (1T1M)

Πολύ εύκολα μπορούμε να υπολογίσουμε ότι η ενίσχυση για τον ενισχυτή του Σχήματος 3.12 θα είναι $A = 1 + \frac{100k\Omega}{1k\Omega} \cong 100$. Άρα στην περίπτωση που το στοιχείο το οποίο διαβάζουμε βρίσκεται στην HRS, θα περνάει αμελητέο ρεύμα από την αντίσταση ανάγνωσης, και ο ενισχυτής θα επιστρέφει μία έξοδο κοντά στα 0V. Αντίστοιχα στην περίπτωση που βρισκόμαστε στην LRS, ο ενισχυτής θα επιστρέψει τη μέγιστη τάση ή οποία προσομοιώνεται

στα 3.7V ($V_{cc} = 5V - LM324^{14}$). Επιλέξαμε να οπτικοποιήσουμε το αποτέλεσμα της ανάγνωσης με τη χρήση ενός απλού LED. Το LED θα παραμένει σβηστό ή θα εκπέμπει φως για κάθε μία από της παραπάνω περιπτώσεις αντίστοιχα.

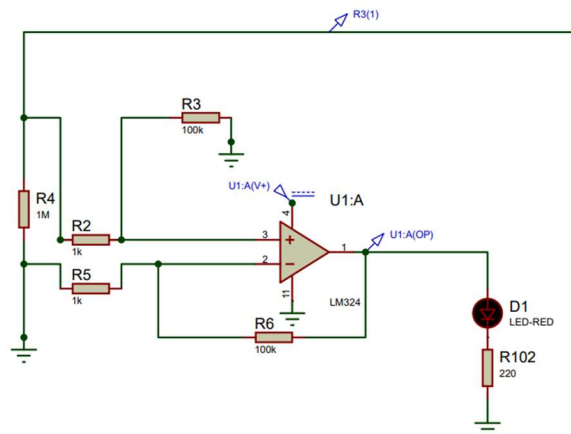
Μία εναλλακτική μορφή συνδεσμολογίας ενισχυτή μπορούμε να δούμε στην περίπτωση του ενθέτου I και του Σχήματος 3.13 (διάταξη χωρίς transistors επιλογής). Στην περίπτωση αυτή η τάση στην έξοδο θα δίνεται από τη σχέση:

$$V_{out} = \left(1 + \frac{R_6}{R_5}\right) \left(\frac{R_3}{R_2 + R_3}\right) V_{in} \quad (3.3.2)$$

Αν όμως επιλέξουμε $R_2 = R_5$ & $R_3 = R_6$, όπως φαίνεται στο Σχήμα 3.13, τότε η τάση εξόδου θα δίνεται από:

$$V_{out} = \frac{R_6}{R_5} V_{in} \quad (3.3.3)$$

και έτσι για τον ενισχυτή του Σχήματος, το gain θα είναι πάλι $A \cong 100$.



Σχήμα 3.13 Ενισχυτής Ανάγνωσης (Selectorless)

Το gain του δεύτερου ενισχυτή, όπως βλέπουμε και από την εξίσωση (3.3.2) επιδέχεται, λόγω της εξάρτησης από περισσότερες αντιστάσεις, περισσότερες μικρο-ρυθμίσεις στο gain. Για τον ίδιο λόγο είναι πιο επιρρεπής σε αποκλίσεις κατά την ανάγνωση. Αρκεί να αναλογιστούμε ότι ο κάθε διαθέσιμος αντιστάτης έχει από τον κατασκευαστή του μία απόκλιση από $\pm 1\% - 5\%$. Για τον λόγο αυτό επιλέξαμε να χρησιμοποιήσουμε στα πραγματικά κυκλώματα των επόμενων

¹⁴ Η μέγιστη τάση εξόδου αποτελεί χαρακτηριστικό του εκάστοτε ενισχυτή.

κεφαλαίων την πιο απλή και ελεύθερη μεγάλων σφαλμάτων, αλλά αντίστοιχα αποτελεσματική διάταξη του Σχήματος 3.12.

3.4 Αποτελέσματα προσομοίωσης

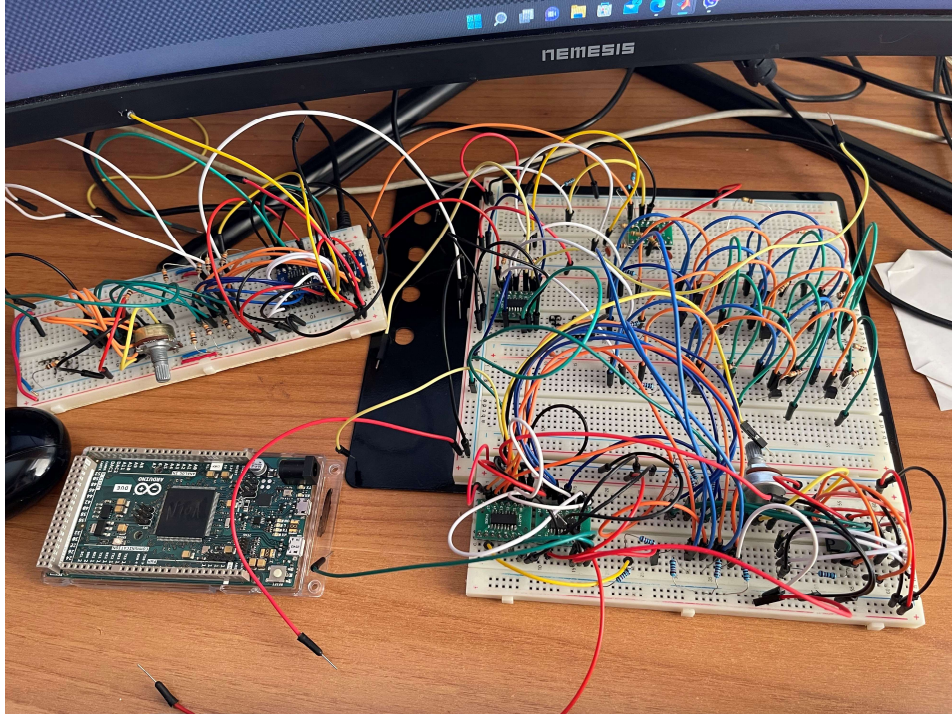
Στο σημείο αυτό κρίνεται δόκιμο να γίνει μία σύγκριση στα των δύο διατάξεων στα πλαίσια του προγράμματος προσομοίωσης Proteus. Για τον σκοπό αυτό επιλέξαμε να συγκρίνουμε το αποτέλεσμα ανάγνωσης των δύο διατάξεων για την χειρότερη δυνατή περίπτωση, δηλαδή όλα τα στοιχεία στην LRS εκτός από το άνω δεξιά το οποίο θα βρίσκεται στην HRS.

Παρατηρήσαμε ότι και στις δύο περιπτώσεις η ανάγνωση γίνεται κανονικά, με κάποιες μικρές διαφορές. Στην περίπτωση που δεν χρησιμοποιήσουμε transistors επιλογής, έχουμε στην έξοδο $V_{LRS} = 3.2V$ και $V_{HRS} = 0.2V$. Στην περίπτωση που τα συμπεριλάβουμε, το σήμα ανάγνωσης στην LRS κυμαίνεται στα $V_{LRS} = 3.7V$. Αντίστοιχα στην HRS παραμένει αμετάβλητη στα $V_{HRS} = 0.2V$. Η μικρή πτώση τάσης που βλέπουμε στην πρώτη περίπτωση οφείλεται στα κρυφά ρεύματα που δημιουργούνται λόγω των ανοιχτών παράλληλων δρόμων όπως ακριβώς έχουμε προαναφέρει. Στη δεύτερη περίπτωση, με πανομοιότυπες συνθήκες, η αρχιτεκτονική 1T1M απαγορεύει τη διέλευση ρευμάτων σε μονοπάτια παράλληλα από το επιθυμητό. Ως αποτέλεσμα παρατηρούμε αύξηση στο παράθυρο ανάγνωσης, η οποία ερμηνεύεται σε μεγαλύτερο εύρος τάσεων εξόδου, περίπου $0.5V$.

3.5 Υλοποίηση του κυκλώματος

Κλείνουμε το κεφάλαιο των crossbar παρουσιάζοντας τα αποτελέσματα για ένα 3×3 δικτύωμα, με ωμικούς αντιστάτες ως ενεργό στοιχείο μνήμης. Στη διάταξη εφαρμόζουμε τόσο την αρχιτεκτονική 1T1M όσο και την εφαρμογή τάσης μεθόδου ανάγνωσης στις μη επιλεγμένες γραμμές για την αντιμετώπιση των κρυφών ρευμάτων. Επιλέξαμε να τοποθετήσουμε τους αντιστάτες μέσα στο crossbar όπως φαίνεται στον παρακάτω πίνακα. Θεωρούμε το πείραμα επιτυχημένο αν καταφέρουμε να καταγράψουμε διαφορετικό ρεύμα εξόδου για κάθε διαφορετικής τιμής αντιστάτη. Στο Σχήμα 3.14 βλέπουμε μία φωτογραφία του κυκλώματος. Ο έλεγχος της διάταξης γίνεται με τη χρήση ενός μικρό-ελεγκτή Arduino DUE, ενώ η διεπαφή γίνεται μέσω του προγράμματος Simulink[®]. Η τάση ανάγνωσης ορίστηκε στα $0.55V$.

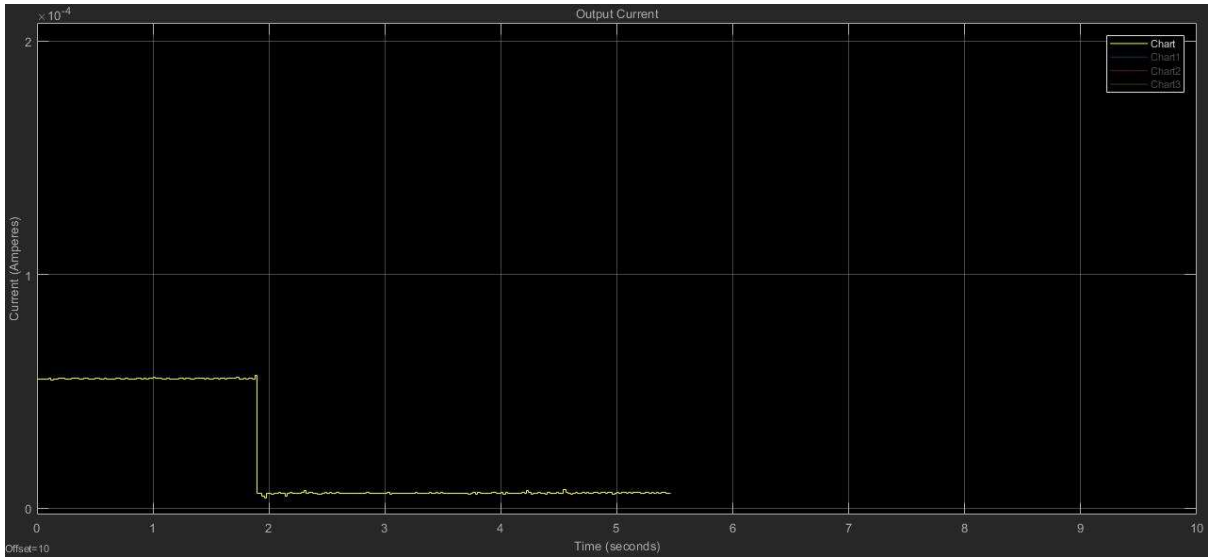
(1,1) = 10kΩ	(1,2) = 100kΩ	(1,3) = 1.2kΩ
(2,1) = 100kΩ	(2,2) = 10kΩ	(2,3) = 100kΩ
(3,1) = 10kΩ	(3,2) = 100kΩ	(3,3) = 6.8kΩ



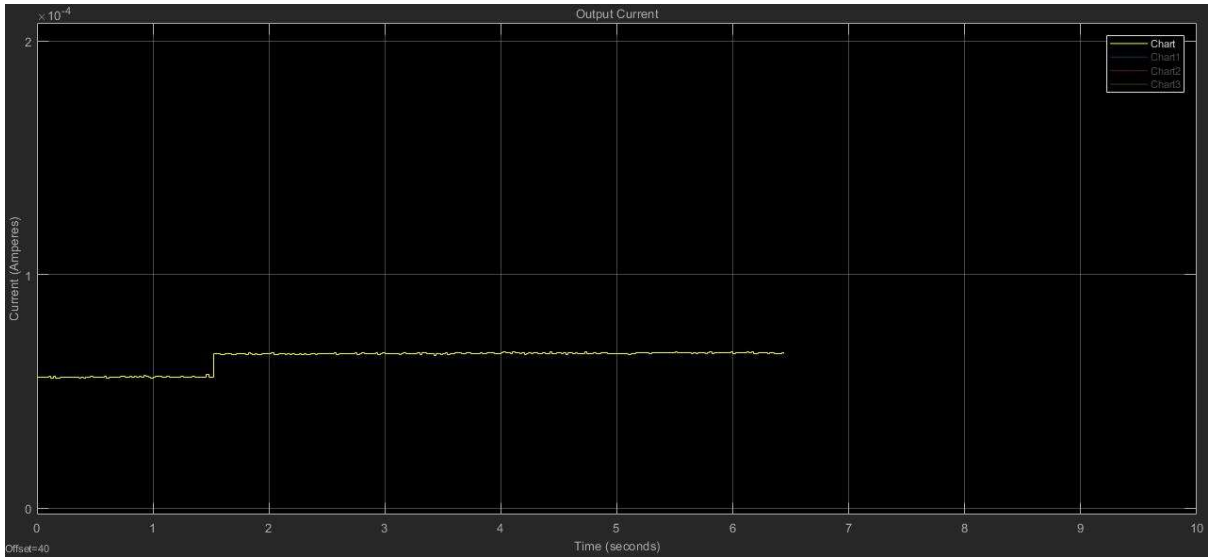
Σχήμα 3.14 Φωτογραφία διάταξης Crossbar με Selectors

Στο Σχήμα μπορούμε εύκολα να ξεχωρίσουμε με μπλε χρώμα τη SL, με πράσινο τη BL και με πορτοκαλί τη WL. Επιλέξαμε να χρησιμοποιήσουμε ξεχωριστό πολυπλέκτη για την επιλογή της SL για λόγους καθαρά και μόνο διαχείρισης χωροταξίας στο breadboard.

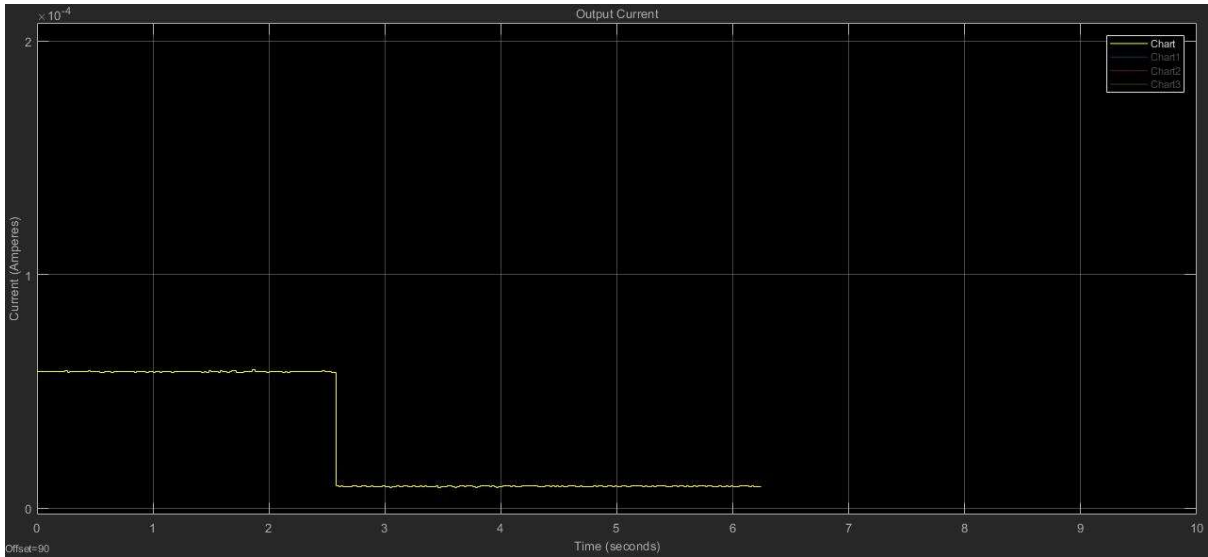
Στα επόμενα Σχήματα μπορούμε να δούμε τα αποτελέσματα.



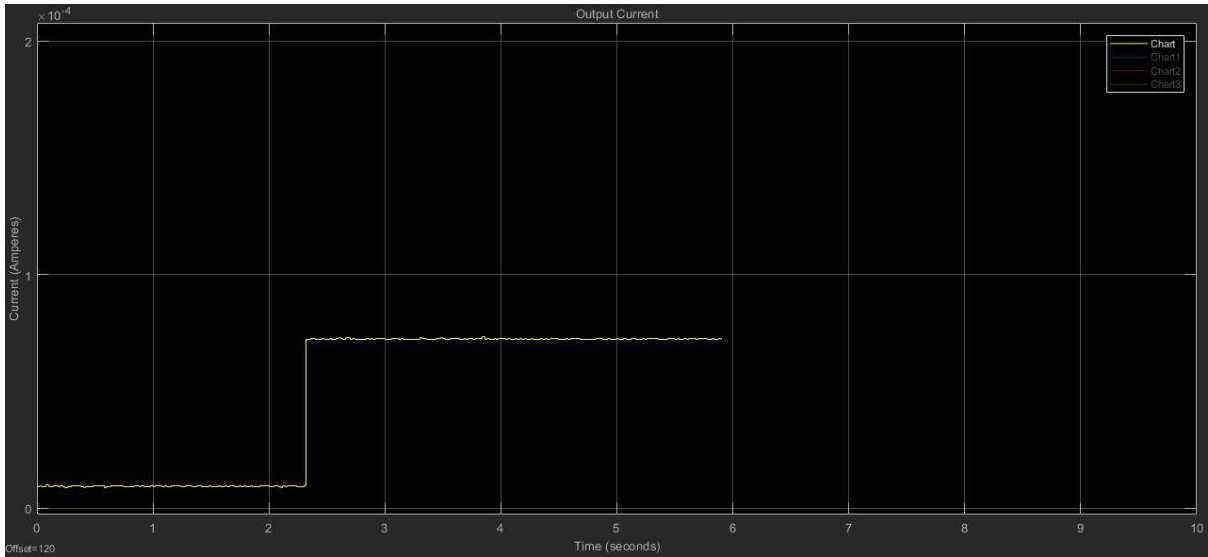
Σχήμα 3.15 Read-Out 1,1(10kΩ) → 1,2(100kΩ)



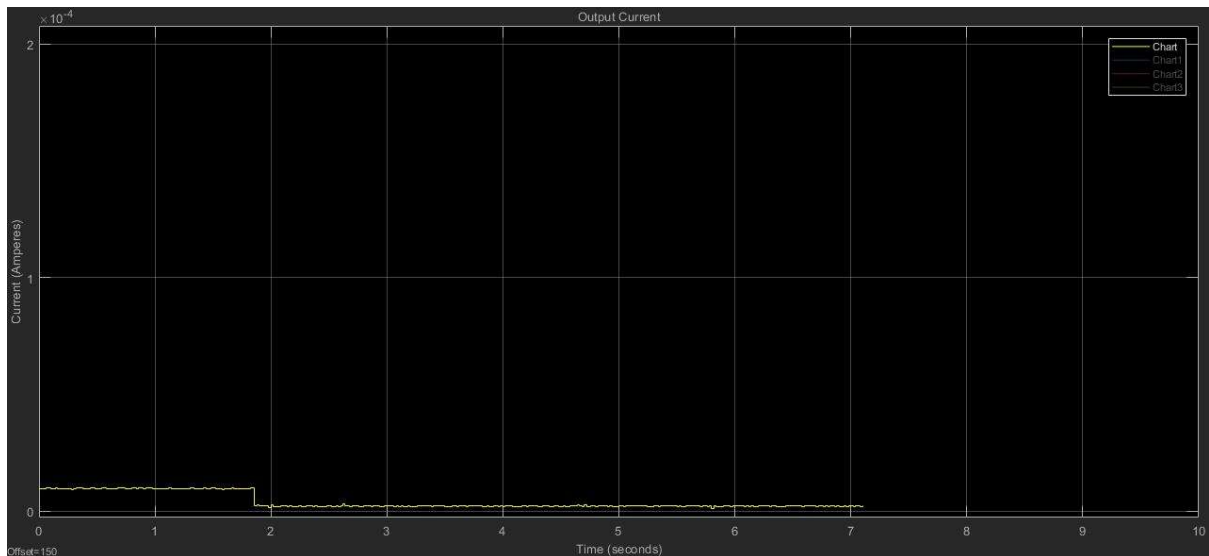
Σχήμα 3.16 Read-Out 1,1(10kΩ) → 1,3(1.2kΩ)



Σχήμα 3.17 Read-Out 1,1(10kΩ) → 2,1(100kΩ)

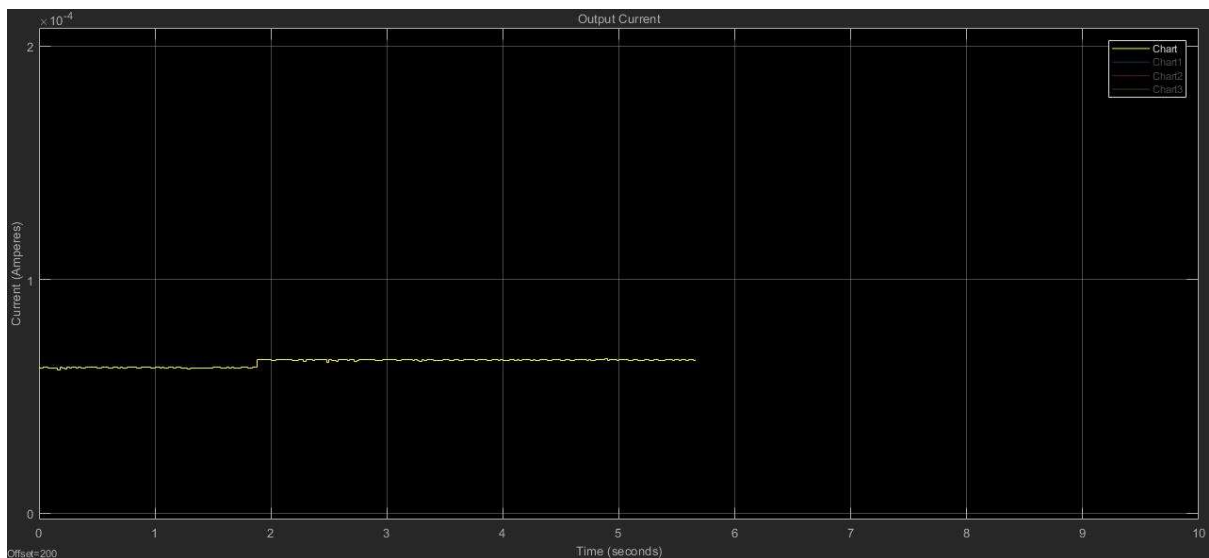


Σχήμα 3.18 Read-Out 2,1(100kΩ) → 2,2(10kΩ)

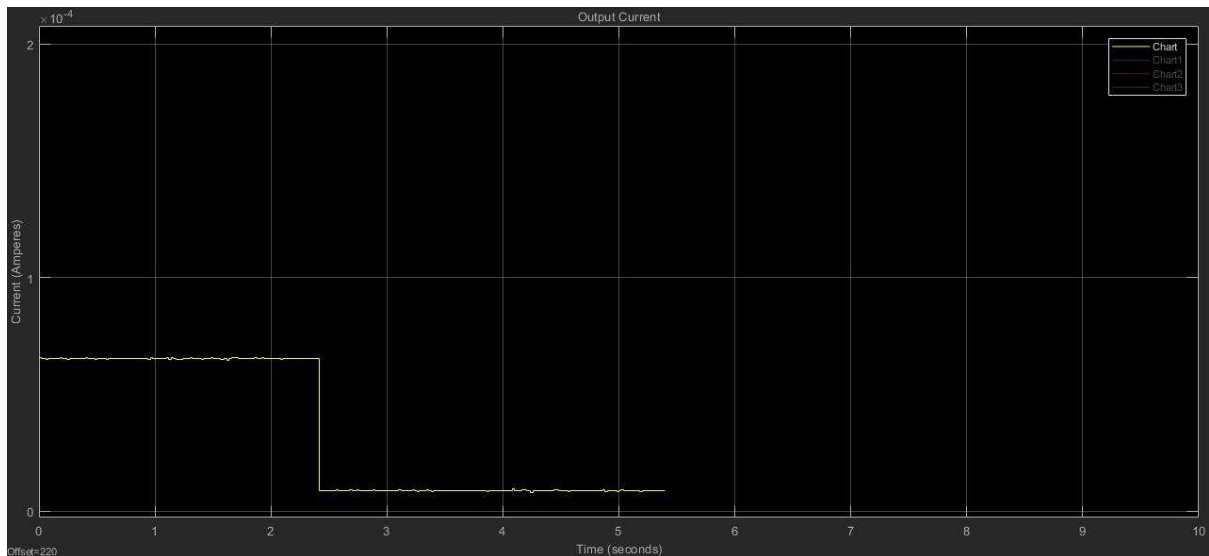


Σχήμα 3.19 Read-Out 2,1(100k Ω) \rightarrow 2,3(~100k Ω)

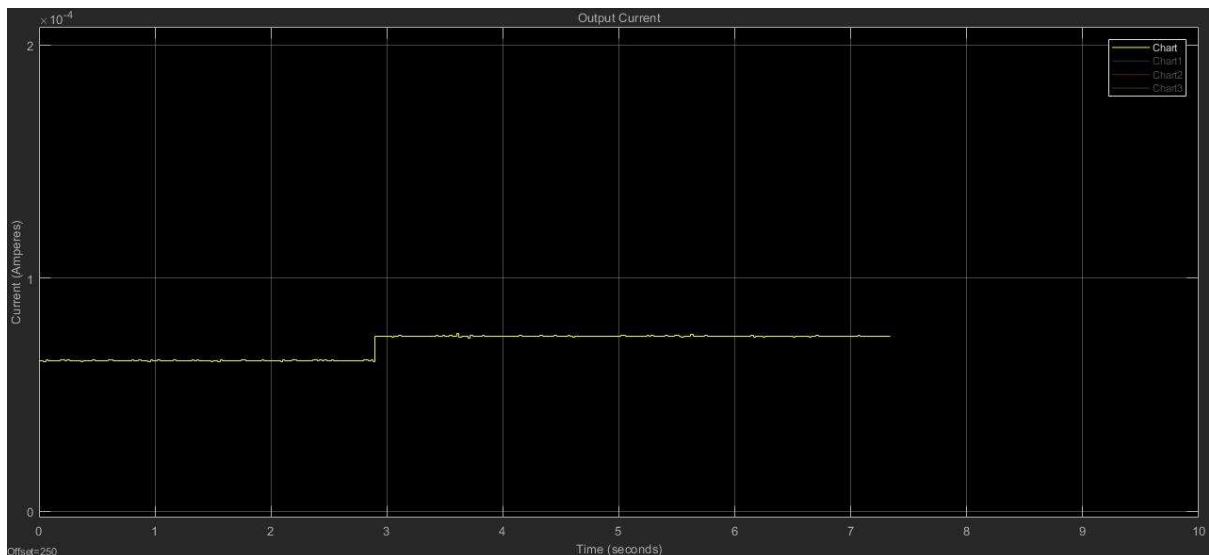
Στο Σχήμα 3.19 παρατηρούμε πως παρόλο που γίνεται αλλαγή από ένα στοιχείο με ονομαστική αντίσταση 100k Ω σε ένα άλλο ονομαστικά πανομοιότυπο, υπάρχει μία μικρή διαφορά στο ρεύμα εξόδου. Αυτό μπορεί να οφείλεται σε σφάλμα στη συνδεσμολογία του *bradboard*, κάτι που μπορεί να δημιουργήσει ανεπιθύμητα ρεύματα διαρροής, αλλά και στο σφάλμα της τιμής του αντιστάτη, του οποίου ο κατασκευαστής το ορίζει ως $\geq 5\%$. Αντίστοιχο αποτέλεσμα παρατηρούμε και στο Σχήμα 3.20.



Σχήμα 3.20 Read-Out 1,1(10k Ω) \rightarrow 3,1(10k Ω)



Σχήμα 3.21 Read-Out 3,1(10kΩ) → 3,2(100kΩ)



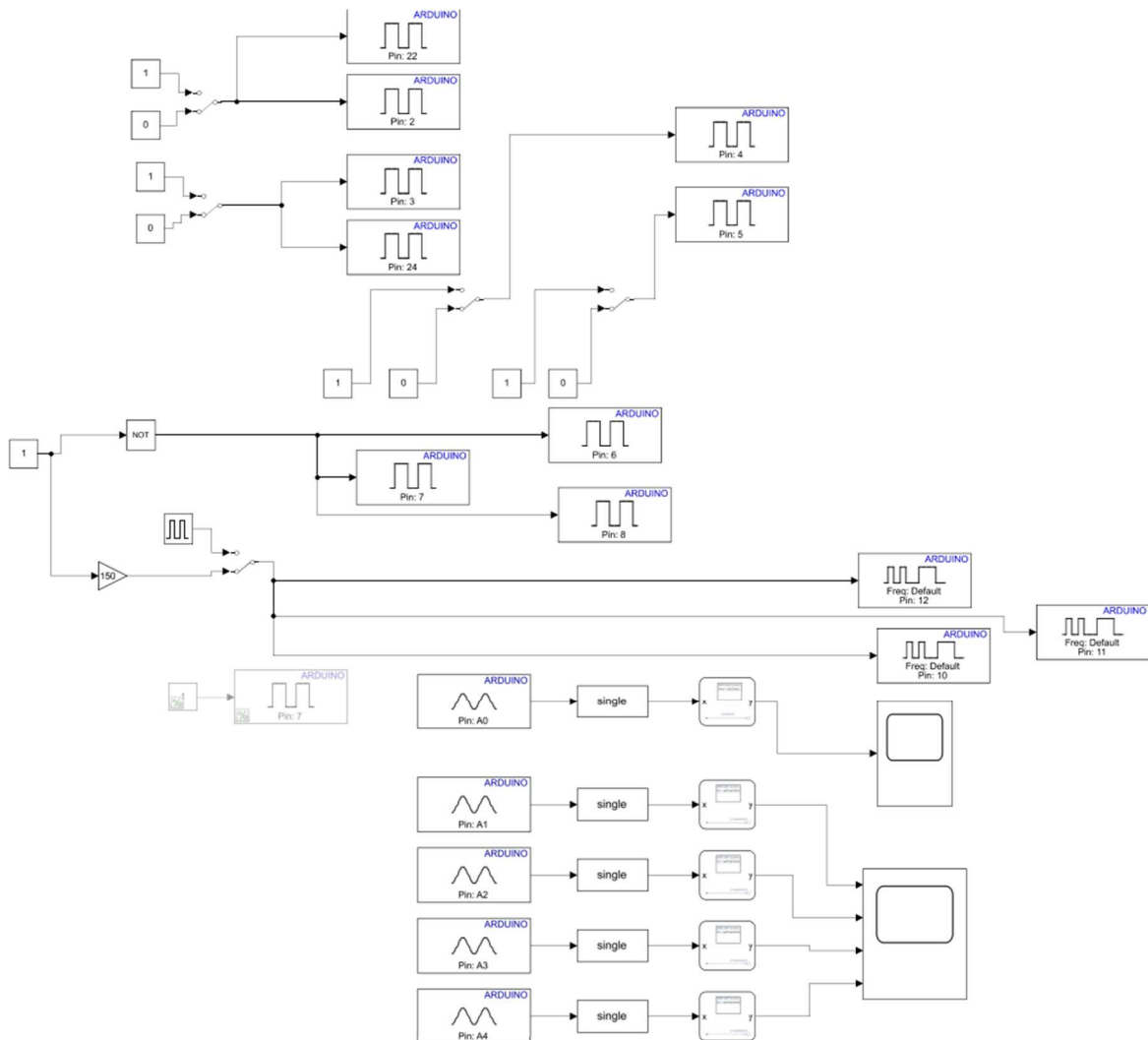
Σχήμα 3.22 Read-Out 3,1(100kΩ) → 3,3(6.8kΩ)

Εν τέλει παρατηρούμε πως το κύκλωμα που σχεδιάσαμε μπορεί να διακρίνει, μαζί με τον κατάλληλα σχεδιασμένο ενισχυτή ανάγνωσης τις διαφορές μεταξύ των αντιστάσεων, ή σε επίπεδο εφαρμογής αποθήκευσης δεδομένων, δύναται να διακρίνει μεταξύ HRS και LRS, ακόμη και για ένα μικρό παράθυρο ανάγνωσης, όπως μεταξύ των $10\text{k}\Omega - 100\text{k}\Omega$ που φαίνεται στα παραπάνω Σχήματα.

3.5.1 Λογισμικό

Για τον έλεγχο της διάταξης, χρησιμοποιήθηκε το λογισμικό Simulink[®] σε απευθείας σύνδεση με τον μικρό-ελεγκτή, χρησιμοποιώντας το πρωτόκολλο XCP on Serial [14]. Αυτή η μορφή επικοινωνίας μας επιτρέπει να τρέχουμε κάποιο μοντέλο του Simulink σε πραγματικό χρόνο σε κάποιο εξωτερικό σύστημα (εδώ: Arduino DUE).

Στο Σχήμα 3.21 μπορούμε να δούμε το μοντέλο Simulink όπου χρησιμοποιήθηκε.



Σχήμα 3.23 Μοντέλο Simulink για έλεγχο της διάταξης crossbar

Το μοντέλο αποτελείται από τρία διακριτά μέλη, τα οποία μπορούμε να συμπεριλάβουμε ή όχι κατά βούληση στο σχεδιασμό μας, ανάλογα με τη μέτρηση – πείραμα που έχουμε σκοπό να πραγματοποιήσουμε. Ξεκινώντας από το πάνω μέρος του Σχήματος προς τα κάτω βλέπουμε αρχικά τις εντολές ελέγχου των πολυπλεκτών. Τα pins¹⁵ 2 & 3 και τα pins 4 & 5 ελέγχουν τα κανάλια των 4-bit πολυπλεκτών της WL και της BL αντίστοιχα. Τα pins 22 & 24 ελέγχουν τα αντίστοιχα κανάλια της SL. Το μοντέλο αυτό ελέγχει μία διάταξη μεγέθους 4 × 4 με την αναβάθμιση σε μεγαλύτερες διαστάσεις (προσθέτοντας τις ίδιες εντολές σε περισσότερα pins) να αποτελεί εύκολη διαδικασία.

¹⁵ Κάθε ακροδέκτης του μικρό-ελεγκτή Arduino ονομάζεται Pin και χαρακτηρίζεται από μία αλφαριθμητική τιμή. Πχ ο αναλογικός ακροδέκτης #2 θα ονομάζεται Pin: A2.

Το μεσαίο κομμάτι πραγματοποιεί δύο λειτουργίες. Ενεργοποιεί και απενεργοποιεί την διάταξη, κατά βούληση, και παράγει τα σήματα εγγραφής και ανάγνωσης. Αποτελείται από μία ψηφιακή είσοδο και 6 εξόδους, 3εις ψηφιακές και 3εις μεταβολής κύκλου λειτουργίας (PWM – duty cycle) υπεύθυνες για τη δεύτερη λειτουργία. Στην περίπτωση που η τιμή εισόδου είναι 0, τα digital pins 6,7 & 8 απενεργοποιούν τους πολυπλέκτες ενώ η τάση στα PWM pins θα είναι μηδέν. Στην αντίθετη περίπτωση οι πολυπλέκτες είναι ενεργοποιημένοι και τα σήματα εγγραφής ή ανάγνωσης παράγονται κανονικά. Τα PWM pins δίνουν σήμα 0V & 3.3V¹⁶ (LOW & HIGH) με μεταβλητό duty cycle¹⁷. Προκειμένου να ορίσουμε την τάση στα επίπεδα που προβλέπονται για την εγγραφή ή την ανάγνωση χρησιμοποιούμε κατάλληλα διαιρέτες τάσης πάνω στο κύκλωμα.

Το τελευταίο μέλος λαμβάνει και εκτυπώνει σε ένα score¹⁸ τα σήματα εισόδου και εξόδου, χρησιμοποιώντας τους 12-bit μετατροπείς αναλογικού σήματος σε ψηφιακό (ADCs) του Arduino DUE. Τα σήματα αυτά είναι μορφής Unsigned Integer 16-bit. Για αυτόν το λόγο κρίνεται απαραίτητη η μετατροπή τους σε μορφή single ή double ώστε να πραγματοποιηθεί η επεξεργασία τους σε δεύτερο χρόνο. Το pin A0 συνδέεται με ένα probe στην είσοδο του crossbar, ανιχνεύοντας το σήμα εισόδου. Τα υπόλοιπα pins μπορούν συνδέονται με τους ενισχυτές στην έξοδο της διάταξης επιστρέφοντας τα ρεύματα εξόδου – ανάγνωσης.

Το θετικό της χρήσης αυτού του λογισμικού για τον έλεγχο της διάταξης είναι η προφανής ευκολία στην χρήση καθώς και στην εφαρμογή αλλαγών στη μορφή του μοντέλου, όταν αυτές κρίνονται απαραίτητες. Ο περιορισμός της είναι η αργή επικοινωνία μεταξύ Arduino και υπολογιστή μέσω σειριακών πρωτοκόλλων επικοινωνίας. Αυτό θεωρείται εμπόδιο σε ενδεχόμενα πειράματα χαρακτηρισμού, καθώς θα περιόριζε σημαντικά τη συχνότητα δειγματοληψίας άρα και τη διακριτική ικανότητα της διάταξης. Στα πλαίσια της διακριτικής ικανότητας που απαιτούνται κατά τα πειράματα αυτής της εργασίας, τόσο σε συνθήκες εγγραφής όσο και ανάγνωσης, ο παραπάνω σχεδιασμός ανταποκρίνεται άριστα.

¹⁶ Arduino DUE

¹⁷ Το κλάσμα μία δεδομένης περιόδου για το οποίο ο παλμός δεδομένου ψηφιακού σήματος βρίσκεται στην κατάσταση HIGH

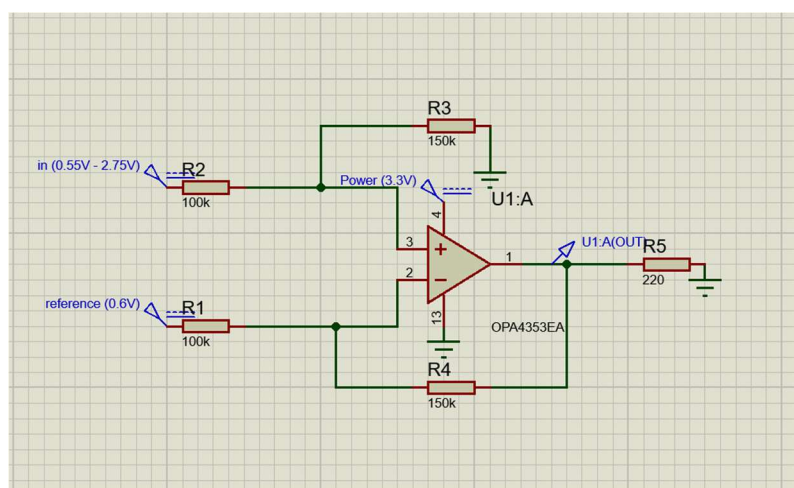
¹⁸ Υπολογιστική μονάδα εκτύπωσης των δεδομένων των ADC του Arduino στο SIMULINK.

4 Χαρακτηρισμός CBRAM με Arduino

Στο Κεφάλαιο 2 αναφέραμε ότι λόγω τεχνικών προβλημάτων στη διαδικασία της συσκευασίας (packaging) των CBRAM δεν ήταν δυνατή η δημιουργία μίας λειτουργικής διάταξης crossbar με RRAMs. Για αυτόν τον λόγο παρουσιάσαμε τη λειτουργία της με την αντικατάσταση των υπό διερεύνηση στοιχείων από απλούς αντιστάτες, παρουσιάζοντας έτσι τη συμπεριφορά ενός δικτύωματος σε λειτουργία ανάγνωσης. Απαιτείται όμως και η επίδειξη της λειτουργίας εγγραφής, για να ολοκληρώσουμε την απόδειξη λειτουργίας της βασικής ιδέας μας. Έτσι πραγματοποιήσαμε μετρήσεις συνεχούς (IV χαρακτηρισμός) και παλμικής τάσης, με τον Arduino DUE να παράγει τόσο τις τάσεις εισόδου, όσο και να πραγματοποιεί την ανάγνωση (monitoring). Επιπλέον, για να έχουμε ένα σύστημα αναφοράς, οι έγινε σύγκριση των μετρήσεων αυτών με τις αντίστοιχες, ακριβέστερες, μετρήσεις του συστήματος ηλεκτρικού χαρακτηρισμού ημιαγωγών Keithley 4200®.

4.1 Χαρακτηρισμός συνεχούς τάσης

Ο Arduino DUE μας παρέχει δύο μετατροπείς ψηφιακού σήματος σε αναλογικό ακρίβειας 12-bit (DACs – Digital to Analogue Converters), οι οποίοι παράγουν μία οποιαδήποτε τάση – σήμα, οποιασδήποτε μορφής στο εύρος των 0.55V – 2.75V. Κατανοούμε πως μία τάση τέτοιου εύρους είναι ανίκανη να χρησιμοποιηθεί για να ληφθεί μία μέτρηση καμπύλης χαρακτηρισμού ρεύματος προς εφαρμοζόμενη τάση (I-V), για την οποία επιθυμούμε ένα εύρος από 0V – 3.3V. Το εμπόδιο αυτό ξεπεράστηκε με τη χρήση ενός κυκλώματος τάσης αναφοράς (reference circuit), όπως φαίνεται στο Σχήμα 4.1.

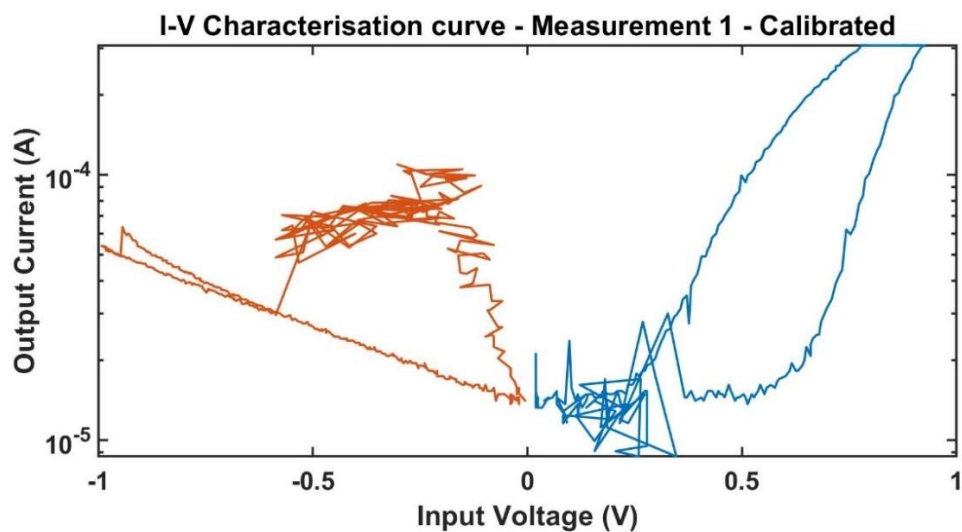


Σχήμα 4.1 Κύκλωμα reference

Πρόκειται για έναν μη αναστρέφοντα διαφορικό ενισχυτή, στου οποίου την αρνητική είσοδο εφαρμόζουμε μία τάση αναφοράς 0.6V, ενώ στην θετική την τάση εξόδου από το DAC. Ο ενισχυτής υπολογίζει την διαφορά αυτή και την ενισχύει με ένα κέρδος ενίσχυσης (gain)

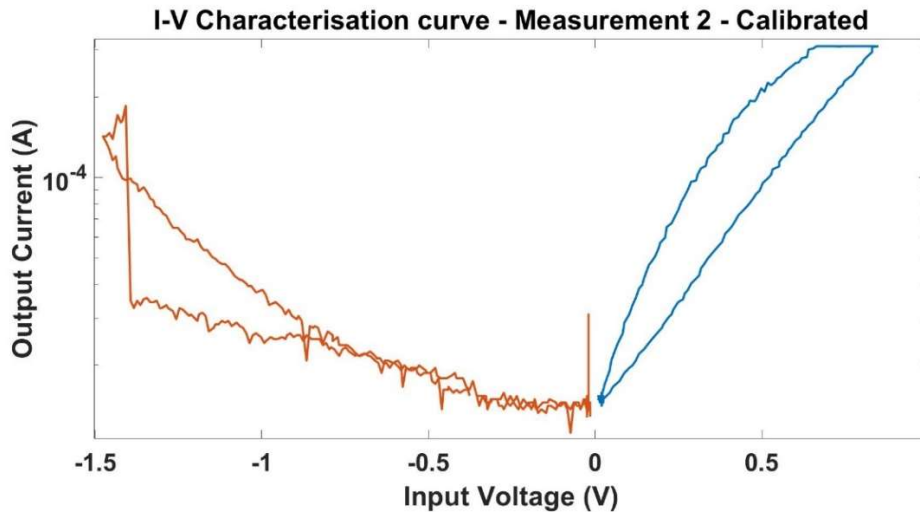
$A = \frac{R_3}{R_2} = 1.5$. Αποτέλεσμα είναι η παραγωγή μίας τάσης από 0V έως και 3V, εύρος που μας επιτρέπει να προχωρήσουμε στη εξαγωγή της καμπύλης I-V.

Μία ολοκληρωμένη καμπύλη χαρακτηρισμού I-V χρειάζεται επίσης και τον αρνητικό κλάδο, για τη διαδικασία RESET, εφόσον μελετάμε μνήμες διπολικής μεταβολής. Ο Arduino δυστυχώς, δεν έχει τη δυνατότητα παραγωγής αρνητικών τάσεων. Το πρόβλημα αυτό λύθηκε επίσης με την χειροκίνητη αλλαγή της πολικότητας στα ηλεκτρόδια της CBRAM και συνδυάζοντας δύο μετρήσεις (μία για SET και άλλη μία για RESET) σε ένα γράφημα. Στα Σχήματα 4.2 και 4.3 βλέπουμε το αποτέλεσμα δύο τέτοιων μετρήσεων.



Σχήμα 4.2 Μέτρηση IV με Arduino (10-bit ADCs)

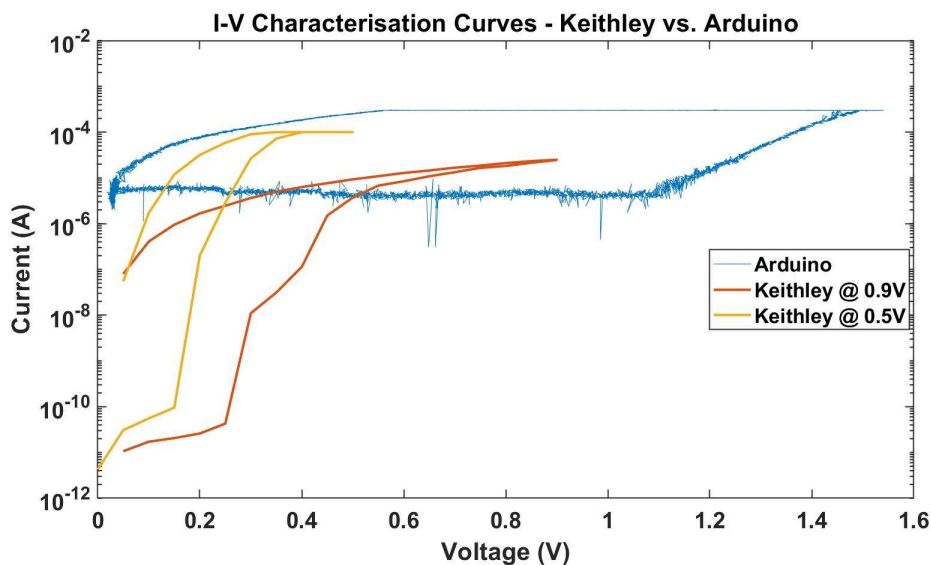
Στο Σχήμα 4.2 βλέπουμε ξεκάθαρα τις διαδικασίες SET και RESET της μνήμης, παρά την ύπαρξη ασταθειών για μικρές τάσεις. Παρατηρούμε ότι η μεταβολή λαμβάνει χώρα περίπου στα 0.4V, με τις HRS ($10^{-5}A$) και LRS ($> 10^{-4}$) εύκολα διαχωρίσιμες. Για τη μέτρηση αυτή χρησιμοποιήθηκαν ADCs ακρίβειας 10-bit.



Σχήμα 4.3 Μέτρηση IV με Arduino (12-bit ADCs)

Στο Σχήμα 4.3 βλέπουμε αντίστοιχα αποτελέσματα με αυτά του Σχήματος 4.2. Παρατηρούμε ότι η χρήση καλύτερης ανάλυσης στους μετατροπείς αναλογικού σήματος σε ψηφιακό (εδώ: 12-bit) μείωσε σημαντικά τον θόρυβο.

Τέλος μένει να συγκρίνουμε τα αποτελέσματα των μετρήσεων μέσω Arduino με αυτών έγιναν με την εξειδικευμένη διάταξη χαρακτηρισμού (Keithley 4200®).



Σχήμα 4.4 Σύγκριση μετρήσεων IV Custom διάταξη & Dedicated

Προφανώς η αυτοσχέδια διάταξη δεν έχει τη δυνατότητα να μας δώσει μέτρηση στα μικρότερα ρεύματα, όπως κάνει η εξειδικευμένη διάταξη. Παρατηρούμε επίσης πως η μνήμη αργεί να φτάσει στην τάση εναλλαγής στη πρώτη περίπτωση. Επιπλέον η αυτοσχέδια διάταξη παρουσιάζει αυξημένο θόρυβο, κάτι που είναι αναπόφευκτο όταν χρησιμοποιούνται απλότερα συστήματα ελέγχου και ανάγνωσης, όπως είναι ο επιλεγμένος μικρό-ελεγκτής.

Το πρώτο πρόβλημα εξηγείται αν σκεφτούμε πως δουλεύει ο ενισχυτής ανάγνωσης. Ορίσαμε μία αντίσταση ανάγνωσης $R_{sense} = \sqrt{R_{HRS} \cdot R_{LRS}}$ και μετράμε την πτώση τάσης πάνω σε αυτή. Για μεγάλα ρεύματα χρειαζόμαστε όσο το δυνατόν μικρότερη αντίσταση και το αντίθετο για τα πάρα πολύ μικρά. Έτσι, προκειμένου να καταφέρουμε να πραγματοποιήσουμε μία επιτυχή μέτρηση σε όλο το εύρος του παραθύρου μνήμης, θα έπρεπε να είμασταν σε θέση να μεταβάλλουμε την τιμή της αντίστασης ανάγνωσης σε πραγματικό χρόνο. Αυτό λύνεται με την αντικατάσταση την αντίστασης ανάγνωσης με ένα transistor τύπου MOSFET, του οποίου την αντίσταση μεταξύ πηγής και απαγωγού μπορούμε να ελέγξουμε μέσω της εφαρμοζόμενης τάσης στη πύλη αυτού, προϋποθέτοντας και τις αντίστοιχες αλλαγές στον κώδικα ελέγχου είτε πρόκειται για MATLAB – Simulink είτε για γλώσσα προγραμματισμού C κατευθείαν στον Arduino (firmware)¹⁹.

Το δεύτερο πρόβλημα οφείλεται στο γεγονός ότι η κινητήριος δύναμη για την εναλλαγή αγωγιμότητας στη μνήμη δεν είναι η τάση, αλλά το ρεύμα την διαρρέει. Ο Arduino αποτελεί μία πολύ πιο αδύναμη πηγή ρεύματος από οποιαδήποτε εξειδικευμένη διάταξη χαρακτηρισμού. Έτσι χρειάζονται μεγαλύτερες τάσεις προκειμένου να προκληθεί η εναλλαγή.

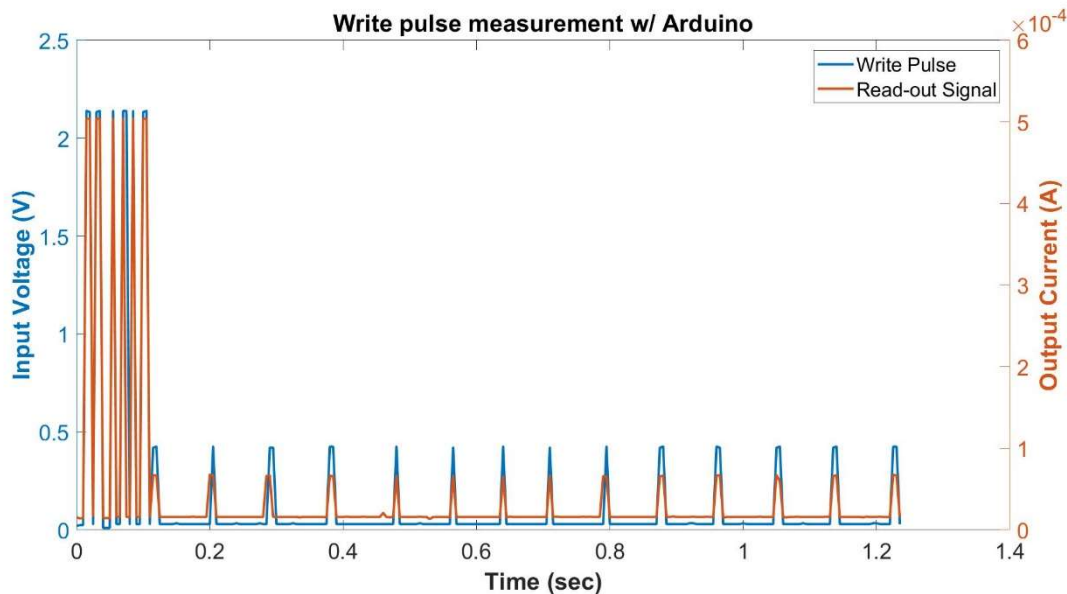
Συμπεραίνουμε πως η custom διάταξη είναι ακατάλληλη, στην τρέχουσα μορφή της, για τον χαρακτηρισμό διατάξεων μνήμης. Παρ' όλα αυτά φαίνεται να μπορεί να «ξεχωρίσει» τις δύο καταστάσεις HRS και LRS. Καταλήγοντας, μπορεί να χρησιμοποιηθεί για εγγραφή και ανάγνωση, αλλά όχι για χαρακτηρισμό.

4.2 Χαρακτηρισμός παλμικής τάσης

Στο Ένθετο I βλέπουμε ένα πρόγραμμα σε γλώσσα C στο Arduino IDE (Integrated Development Environment). Ένας Arduino με αυτό πρόγραμμα μπορεί παράγει μία custom παλμοσειρά, ρυθμιζόμενου χρονικού πλάτους με βάσει τη καθυστέρηση λειτουργίας (μέθοδος `delay()` στη γλώσσα C, στη βιβλιοθήκη Arduino) που εφαρμόζουμε κάθε φορά, απλά χρησιμοποιώντας το ακρίβειας 12-bit DAC. Όταν τελειώσει η παλμοσειρά, συνεχίζουμε να στέλνουμε παλμούς μικρότερου πλάτους ώστε να μπορούμε να διαβάσουμε τα αποτελέσματα της προηγούμενης παλμοσειράς.

Στο Σχήμα 4.5 βλέπουμε το αποτέλεσμα μίας τέτοιας διαδικασίας.

¹⁹ Η μελέτη ενός τέτοιου κυκλώματος ξεπερνάει τα όρια αυτής της εργασίας



Σχήμα 4.5 4εις παλμοί εγγραφής και παλμοί ανάγνωσης

Εφαρμόζουμε τέσσερις παλμούς εγγραφής περί τα 2.1V και έπειτα παρακολουθούμε το αποτέλεσμα της διαδικασίας. Η παλμοσειρά προκάλεσε εγγραφή, την οποία επαληθεύουμε καθώς, διέρχεται από τη μνήμη ρεύμα «ανάγνωσης» όταν εφαρμόζουμε μικρότερους παλμούς.

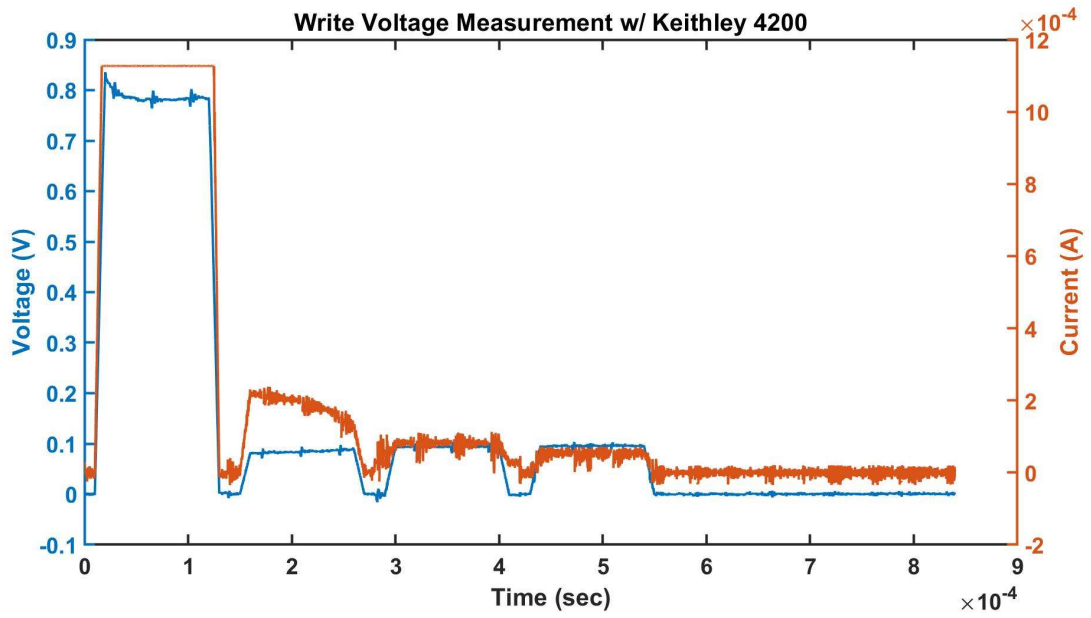
Στο Σχήμα 4.6 μπορούμε να δούμε και την αντίστοιχη σύγκριση της παραπάνω μέτρησης με αυτή που προκύπτει με χρήση του Keithley 4200. Αξίζει να σημειωθεί ότι στην παρακάτω μέτρηση χρησιμοποιήθηκε το μεγαλύτερο δυνατό χρονικό εύρος για τη παραγωγή παλμών (~ 100 μs). Με μία πρώτη σύγκριση γίνεται κατανοητό ότι οι παλμοί που παράγονται από τον Arduino είναι πολύ μεγαλύτεροι σε εύρος. Η παραγωγή παλμών της τάξης μερικών ns είναι δυνατή με χρήση του μικρό-ελεγκτή αν χρησιμοποιήσουμε γλώσσα C χαμηλού επιπέδου ώστε να ελέγξουμε απευθείας της πύλες του (BareMetal programming). Για τον σκοπό της εργασίας αυτής η τρέχουσα διάταξη φαίνεται να πραγματοποιεί ικανοποιητική, αν και πιο αργή, εγγραφή. Έτσι θεωρούμε τα αποτελέσματα αποδεκτά.

Παρατηρούμε επίσης ότι τα ρεύματα εξόδου (για τους παλμούς μετά την εγγραφή και την ανάγνωση) είναι συγκρίσιμα και για τις δύο περιπτώσεις. Έτσι θεωρούμε την αυτοσχέδια διάταξη κατάλληλη για λειτουργία, και περιορισμένο παλμικό χαρακτηρισμό δικτυωμάτων crossbar.

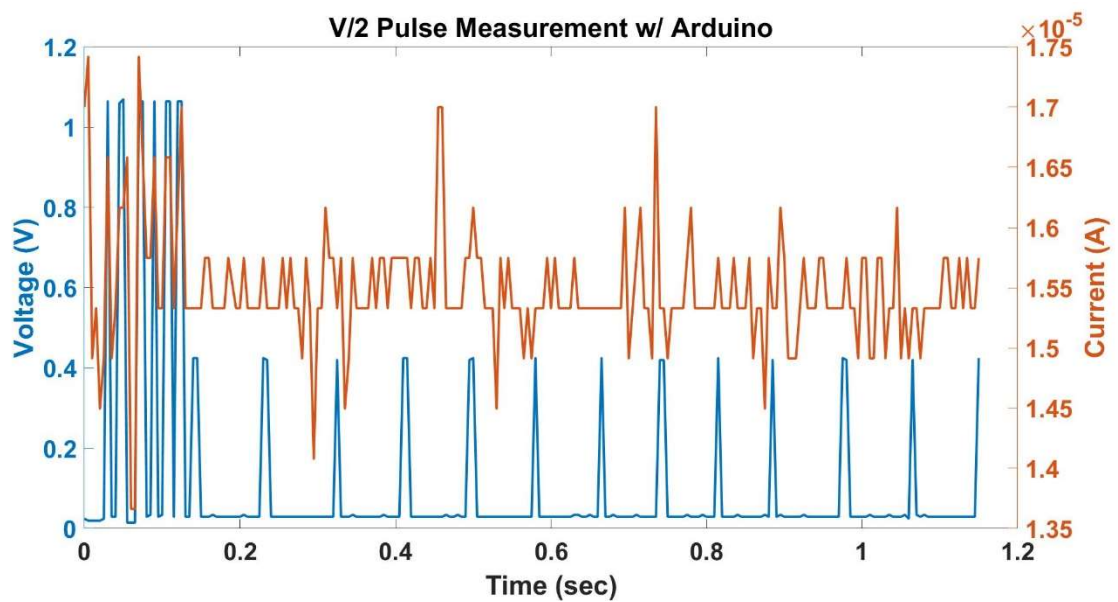
Πραγματοποιήσαμε επίσης μέτρηση με τη μισή τάση εγγραφής από αυτή της περίπτωσης του Σχήματος 4.5. Τα αποτελέσματα της μέτρησης αυτής φαίνονται στο Σχήμα 4.7. Παρατηρούμε μία μικρή ροή ρεύματος, περίπου $1.7 \cdot 10^{-5} A$, δηλαδή ~15 φορές μικρότερη από αυτή στο Σχήμα 4.5. Έπειτα βλέπουμε δεν περνάει ρεύμα από τη CBRAM κατά τη διάρκεια της ανάγνωσης²⁰, σε αντίθεση με την προηγούμενη περίπτωση. Ο συνδυασμός αυτός σημαίνει πως βρισκόμαστε σε καθεστώς λειτουργίας κατωφλίου, δηλαδή η CBRAM μας λειτουργεί ως

²⁰ Ορισμένα spikes οφείλονται σε θόρυβο. Δεν αποτελούν πραγματική ροή ρεύματος.

πτητική μνήμη, επαληθεύοντας άλλη μία φορά τα πειραματικά δεδομένα της παραγράφου 2.3.4.



Σχήμα 4.6 Παλμικός χαρακτηρισμός με χρήση του Keithley 4200



Σχήμα 4.7 5 παλμοί V/2 και παλμοί ανάγνωσης

5 Χρήση των CBRAM σε νευρομορφικές εφαρμογές

Στα τελευταία δύο κεφάλαια θα παρουσιάσουμε μία σειρά πειραμάτων, που πραγματοποιήθηκαν με χρήση τόσο της αυτοσχέδιας διάταξης με Arduino όσο και με τη χρήση της εξειδικευμένης διάταξης χαρακτηρισμού ημιαγωγών. Τα πειράματα αυτά αφορούν εφαρμογές νευρομορφικού χαρακτήρα των CBRAM.

5.1 Έμπνευση από τη βιολογία

Ο όρος «νευρομορφικά υπολογιστικά συστήματα» περιλαμβάνει όλες τις διατάξεις οι οποίες πραγματοποιούν περίπλοκους υπολογισμούς μιμούμενες τις λειτουργίες του ανθρώπινου εγκεφάλου. Η κύρια διαφορά του εγκεφάλου, σαν υπολογιστικό σύστημα, σε σύγκριση με έναν von-Neumann υπολογιστή είναι ο συντριπτικά μεγαλύτερος όγκος πληροφοριών που μπορεί να επεξεργαστεί παράλληλα, και μάλιστα με ελάχιστη κατανάλωση ενέργειας. Την μίμηση αυτή έχουμε ήδη ξεκινήσει να τη βλέπουμε στη καθημερινότητα μας, με τη χρήση GPUs των οποίων οι πολλοί πυρήνες προσφέρονται για παράλληλη επεξεργασία, με μεγάλο κόστος στην κατανάλωση ενέργειας [15].

Στον εγκέφαλο, η αποθήκευση δεδομένων δεν είναι ξεχωριστή από την επεξεργασία δεδομένων (μη von-Neumann αρχιτεκτονική). Τα ίδια στοιχεία, συγκεκριμένα οι νευρώνες (10^{11} στον αριθμό) και οι συνάψεις (10^{15} στον αριθμό) εκτελούν και τις δύο λειτουργίες και μάλιστα σε ένα συντριπτικά παράλληλο και προσαρμοστικό περιβάλλον, με ελάχιστη κατανάλωση ενέργειας (περί τα 20W) [15]. Οι συνάψεις αποσκοπούν στον έλεγχο της ροής της πληροφορίας από στοιχείο σε στοιχείο, ενώ οι νευρώνες συγκεντρώνουν την πληροφορία και την απελευθερώνουν, εφόσον αυτή ξεπεράσει κάποιο κατώφλι.

Οι διατάξεις εναλλαγής αντίστασης, όπως οι CBRAM που μελετάμε στα πλαίσια της εργασίας αυτής, λόγω της πληθώρας δυναμικών ιδιοτήτων που παρουσιάζουν, φαίνεται να είναι ο κύριος υποψήφιος για την υλοποίηση τέτοιων υπολογιστικών συστημάτων.

5.2 Λειτουργία τεχνητής σύναψης

5.2.1 Θεωρητικά στοιχεία

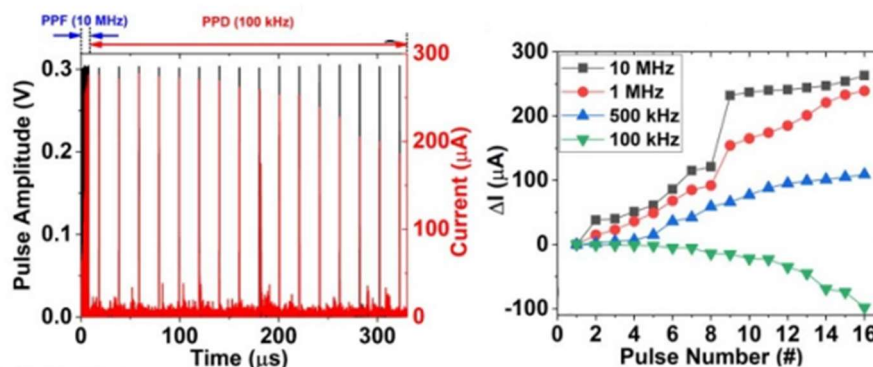
Παράλληλα με την κατανομή των νευρώνων στον εγκέφαλο, το τεράστιο πλέγμα των συνάψεων είναι υπεύθυνο για την ανθρώπινη αντίληψη, μνήμη, εκμάθηση και τα συναισθήματα. Όπως προαναφέραμε, στους νευρώνες οφείλουμε τη δημιουργία και τη μεταφορά ενός σήματος πληροφορίας (action potential), ενώ η σύναψη πραγματοποιεί επεξεργασία και αποθήκευση του σήματος αυτού.

Διακρίνουμε δύο φαινόμενα:

- Το paired-pulse-facilitation (PPF)
- Το paired-pulse-depression (PPD)

Στο πρώτο από αυτά τα δύο φαινόμενα οφείλουμε την εκπαίδευση (ή με όρους νευρωνικών δικτύων, τον καθορισμό των βαρών) μίας σύναψης, ενώ το δεύτερο σχετίζεται με τους μηχανισμούς μόνιμης μνήμης.

Κατά τη διαδικασία PPF, το κάθε επόμενο σήμα ρεύματος εξόδου είναι μεγαλύτερης έντασης από το προηγούμενο, με την πηγή να παράγει ταυτόσημους παλμούς τάσης σε σταθερή συχνότητα. Αντίθετα το ρεύμα εξόδου μειώνεται για κάθε επόμενο παλμό κατά τη διαδικασία PPD. Η τελευταία καθορίζεται κυρίως από τη συχνότητα των παλμών διέγερσης και από το πλάτος. Στο Σχήμα 5.1 βλέπουμε μία σύνοψη αυτών των φαινομένων. Στο αριστερό σχήμα βλέπουμε την αύξηση της αγωγιμότητας της διάταξης καθώς εφαρμόζουμε μία παλμοσειρά υψηλής συχνότητας, και έπειτα τη μείωσή της με την εφαρμογή μίας αντίστοιχης, ίδιου πλάτους αλλά 10 φορές μικρότερης συχνότητας, επαληθεύοντας τον προηγούμενο ισχυρισμό. Δεξιά απεικονίζεται σχηματικά η διαφορά ρεύματος εξόδου για διαφορετικές συχνότητες [7].



Σχήμα 5.1 PPF & PPD (αριστερά). Διαφορά ρεύματος εξόδου ως προς τον αριθμό των παλμών για διαφορετικές συχνότητες (δεξιά) [7]

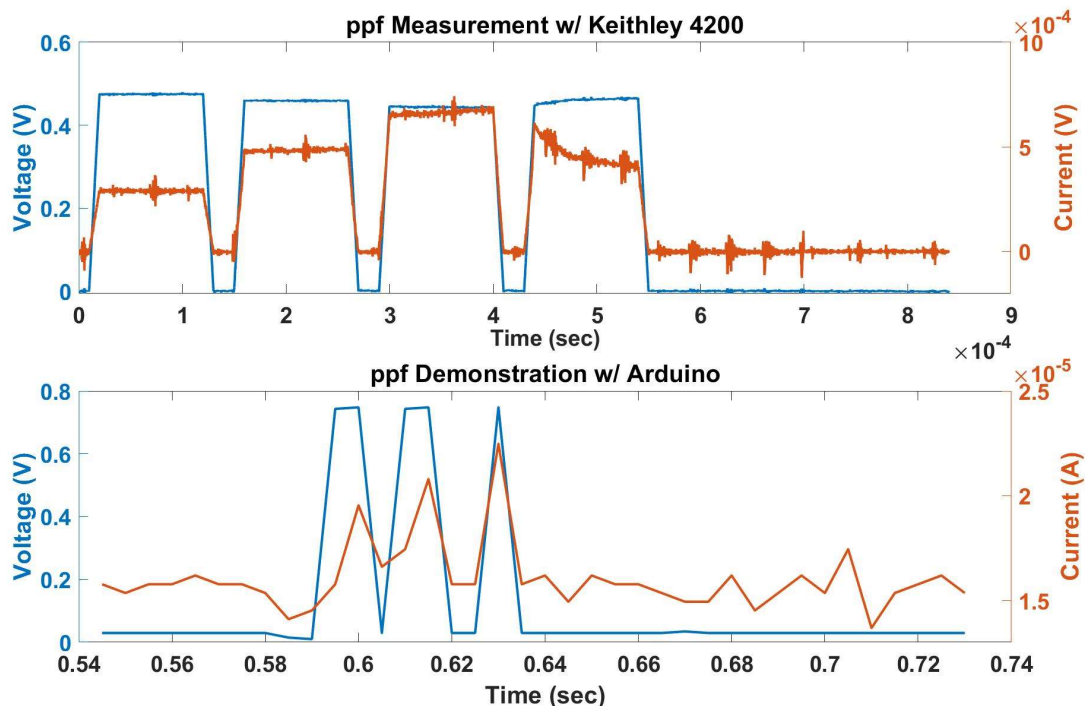
5.2.2 Πειραματικά στοιχεία – Παλμοί παραγόμενοι από ελεγχόμενη πηγή.

Ο σκοπός της μέτρησης που παρουσιάζουμε σε αυτή την παράγραφο ήταν να επαληθεύσουμε ότι η αυτοσχέδια διάταξη είναι ικανή αναπαραγωγής των συναπτικών ιδιοτήτων της μνήμης. Για να το αποδείξουμε αυτό αρκεί να παρατηρήσουμε ένα από τα φαινόμενα PPF/PPD. Πράγματι στο Σχήμα 5.2 βλέπουμε το φαινόμενο PPF, με πηγή παλμικής τάσης τον μικρό-ελεγκτή σε PWM λειτουργία. Επιπλέον συμπεριλαμβάνεται μία σύγκριση με μία αντίστοιχη μέτρηση με τον εξειδικευμένο εξοπλισμό.

Παρατηρούμε ότι το PPF που προκαλείται από το Arduino είναι μικρότερης «έντασης» παράγοντας μικρότερα ρεύματα από την αντίστοιχη τάση του Keithley 4200, κάτι που μπορεί

να εξηγηθεί τόσο λόγω του μικρότερου ρεύματος οδήγησης της αυτοσχέδιας διάταξης (driving current), όσο και από τις διαφορετικές ιδιότητες των διαφορετικών στοιχείων μνήμης.

Το τελευταίο χαρακτηριστικό είναι αυτό που κάνει τις CBRAM έναν τέλειο υποψήφιο για νευρομορφικές εφαρμογές. Αρκεί να σκεφτούμε ότι κάθε σύναψη του νευρικού συστήματος έχει διαφορετική ευαισθησία για δεδομένα ερεθίσματα. Έτσι μία διάταξη με μνήμες οι οποίες παρουσιάζουν διαφορετικές συναπτικές ιδιότητες αποτελεί άριστη προσομοίωση ενός βιολογικού εγκεφάλου.



Σχήμα 5.2 Paired Pulse Facilitation. Keithley 4200 (άνω) Custom Διάταξη (κάτω)

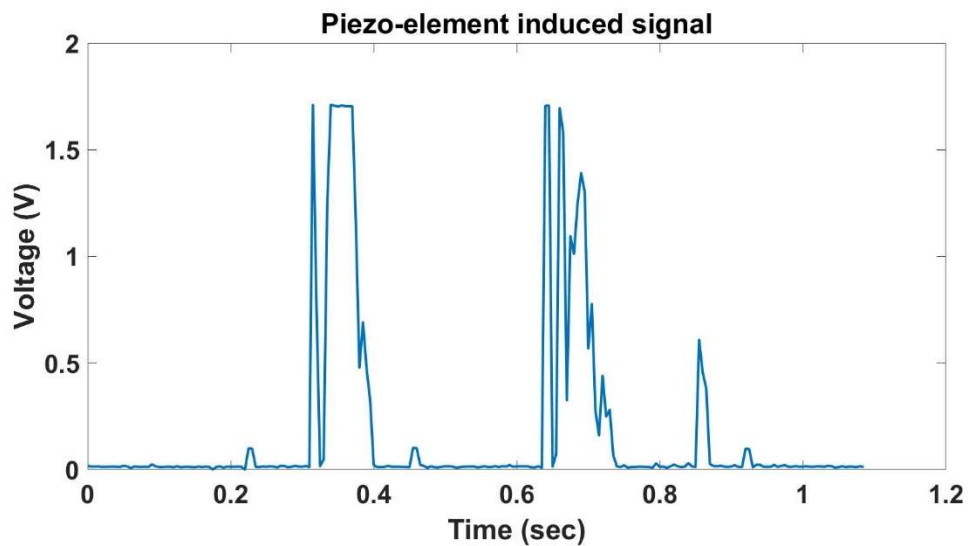
Εφόσον αποδείξαμε ότι μπορεί να γίνει χαρακτηρισμός συναπτικών ιδιοτήτων με χρήση της αυτοσχέδιας διάταξης, στην επόμενη παράγραφο προχωρούμε στην αναπαραγωγή των ίδιων φαινομένων, εφαρμόζοντας εξωτερικά ερεθίσματα.

5.2.3 Πειραματικά στοιχεία – Χαρακτηρισμός σύναψης με εξωτερικά ερεθίσματα.

Η σύναψη έχει τη δυνατότητα, όπως έχουμε προαναφέρει, να «επεξεργαστεί» και να «αποθηκεύσει» μέσω της πλαστικότητας της τα δεδομένα που παράγονται από τους νευρώνες.

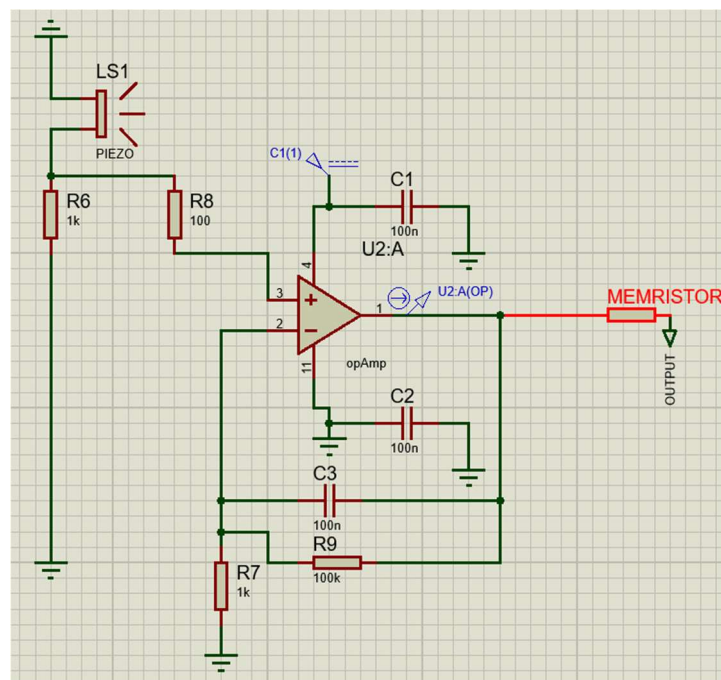
Μία καλή προσομοίωση τέτοιων νευρωνικών σημάτων είναι τα κρουστικά σήματα (spikes) παραγόμενα από ένα απλό πιεζοηλεκτρικό στοιχείο περοβσκίτη (PZT). Στο Σχήμα 5.3 μπορούμε να δούμε τη μορφή του σήματος που παράγεται από ένα τέτοιο πιεζοηλεκτρικό

στοιχείο (piezo-element), όταν η μία επαφή του είναι γειωμένη και η άλλη συνδεδεμένη με την είσοδο ενός ADC (ή ενός ενισχυτή αν απαιτείται ενίσχυση του σήματος πριν το ADC).



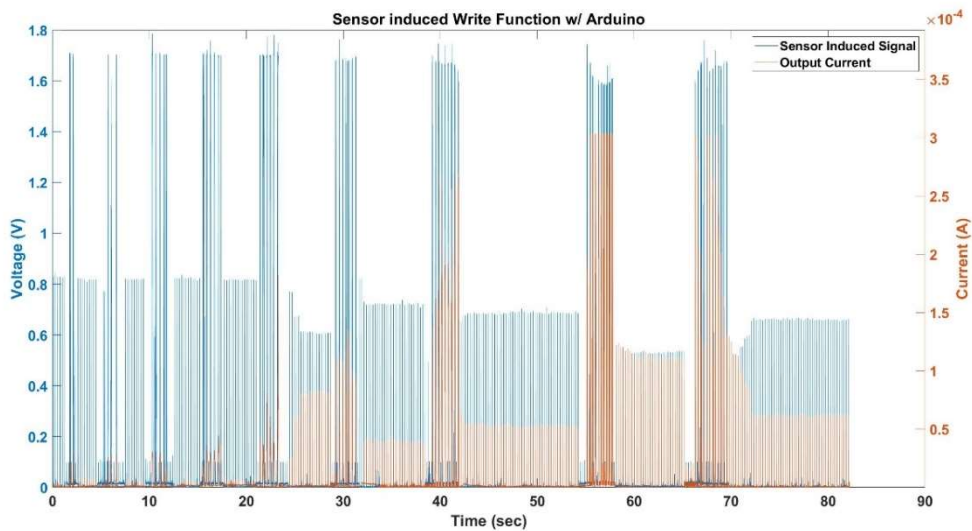
Σχήμα 5.3 Σήμα πιεζοηλεκτρικού στοιχείου, μετά από ενίσχυση

Εφαρμόζοντας αυτό το σήμα στη σύνδεση, μέσω της διάταξης που φαίνεται στο Σχήμα 5.4, έχουμε μία απλή, αλλά χρήσιμη προσομοίωση της αίσθησης της αφής, κάτι που παρουσιάζει μεγάλο ενδιαφέρον στον τομέα της ρομποτικής.

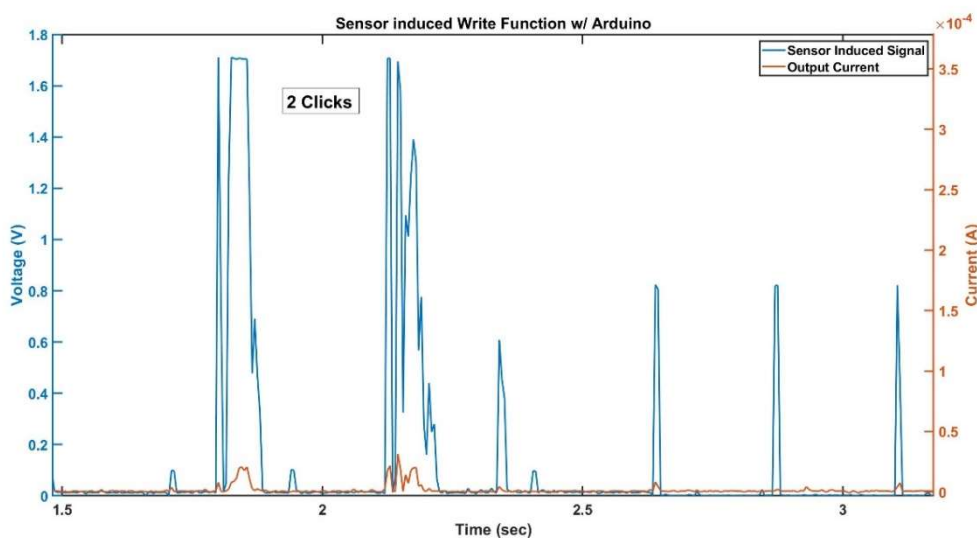


Σχήμα 5.4 Διάταξη "τεχνητής αφής" με χρήση Piezo και Memristive σύνδεσης(εδώ: CBRAM Ag-SiO₂-TiN) στοιχείο

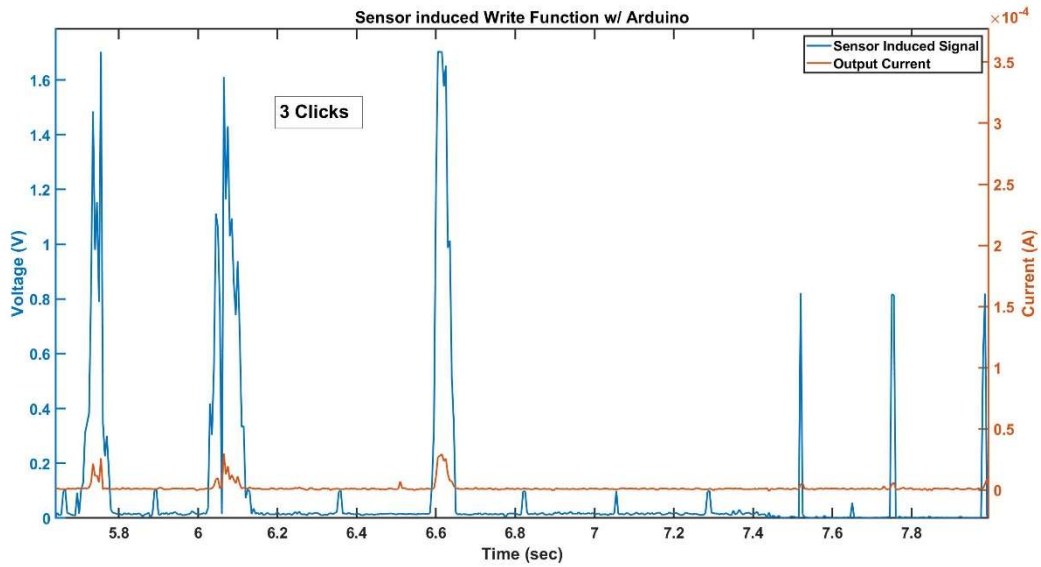
Για να προχωρήσουμε στον χαρακτηρισμό, ορίσαμε το παρακάτω πειραματικό πρωτόκολλο εφαρμογής παλμών. Τα δεδομένα συλλέγονταν μέσω του Arduino Due και του λογισμικού Simulink. Μέσω ενός απλού διακόπτη μπορούσαμε να επιλέξουμε αν η εισαγωγή παλμών γινόταν από το πιεζοηλεκτρικό ή από το DAC του Arduino. Στη πρώτη περίπτωση «γράφουμε» πληροφορία στη σύναψη μέσω μεμονωμένων clicks, ίδιας δύναμης και διάρκειας. Όταν ο διακόπτης έδειχνε προς το Arduino, εφαρμόζαμε παλμούς στο 50% του μέγιστου παραγόμενου από το ριεζο πλάτους, συχνότητας $f \cong 4.3\text{Hz}$. Εφαρμόσαμε κύκλους εκπαίδευσης με 2, 3, 4, ..., 12 clicks ανά κύκλο στο πιεζοηλεκτρικό στοιχείο, και παλμούς ενδιάμεσης ανάγνωσης για παρακολούθηση των αποτελεσμάτων της εκπαίδευσης. Τα αποτελέσματα αυτών των μετρήσεων φαίνονται στο Σχήμα 5.5.



Σχήμα 5.5 Πειραματικά αποτελέσματα εγγραφής με πιεζοηλεκτρικό στοιχείο (2, 3, 4, 5, 6, 8, 9, 10 & 12 Clicks – left to right)

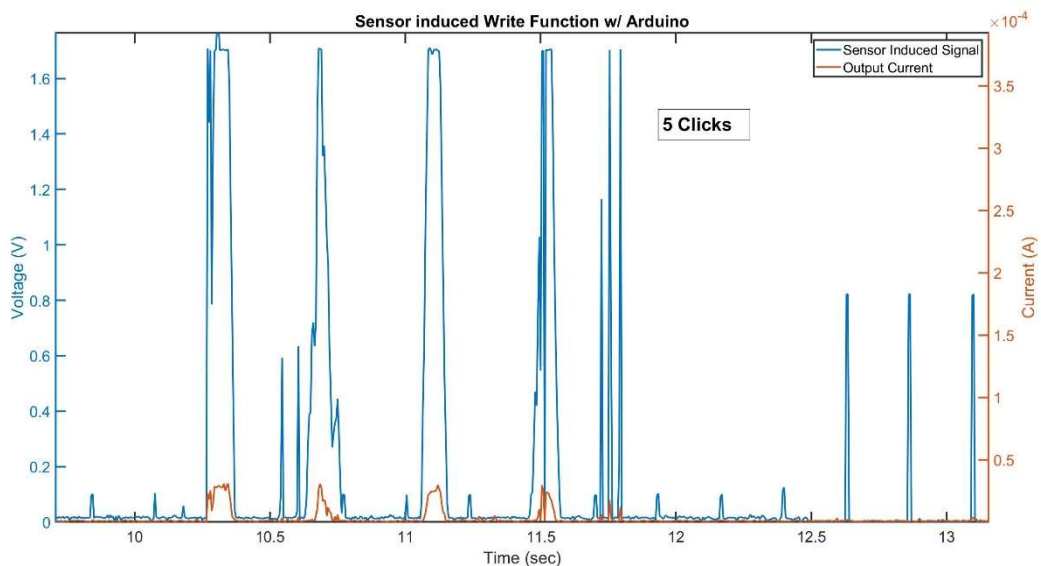


Σχήμα 5.6 Πειραματικά αποτελέσματα: 2 Clicks

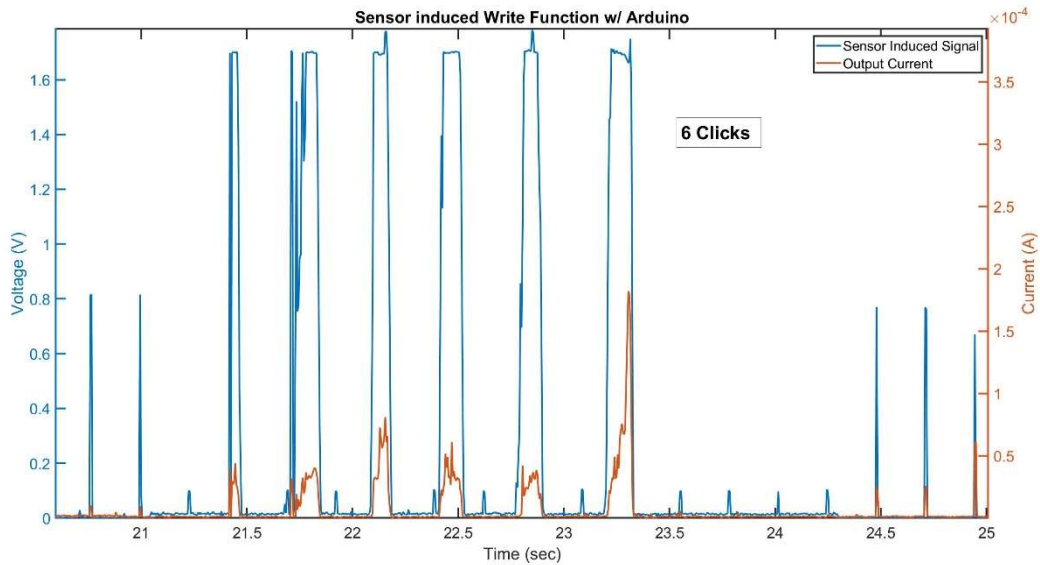


Σχήμα 5.7 Πειραματικά αποτελέσματα: 3 Clicks

Στα Σχήματα 5.6 και 5.7 βλέπουμε τα αποτελέσματα των μετρήσεων για 2 και 3 clicks στον αισθητήρα αντίστοιχα. Παρατηρούμε ότι δεν έχει λάβει χώρα κάποιο φαινόμενο PPF σε αυτή τη φάση της εκπαίδευσης. Κατά την εφαρμογή των παλμών του πιεζοηλεκτρικού βλέπουμε μία μικρή αύξηση στο ρεύμα εξόδου, το οποίο μηδενίζεται κατά την εφαρμογή των παλμών $V_{\text{piezo}}/2$, μαρτυρώντας μεταβολή κατωφλίου, σε αντιστοιχία με τις μετρήσεις του Σχήματος 4.7.

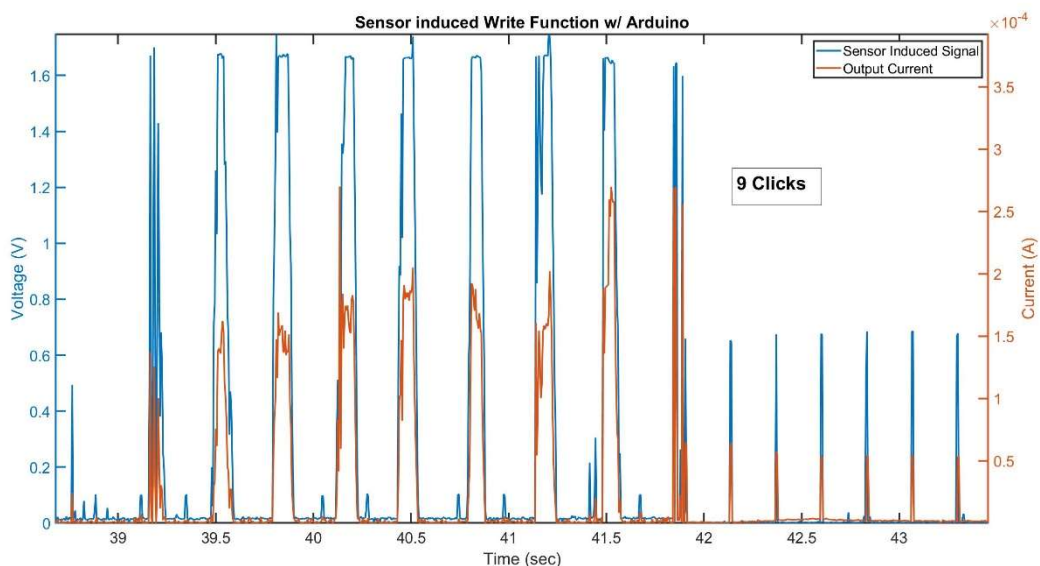


Σχήμα 5.8 Πειραματικά αποτελέσματα: 5 Clicks



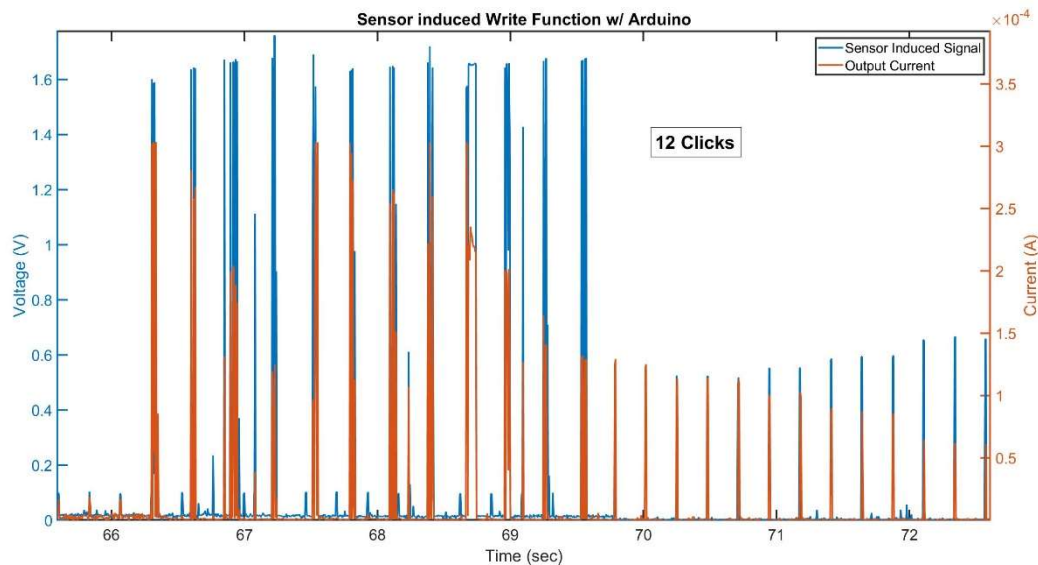
Σχήμα 5.9 Πειραματικά αποτελέσματα: 6 Clicks

Τα πειράματα έδειξαν ότι τα ίδια αποτελέσματα αναπαράγονται μέχρι και τον κύκλο των 5 clicks (Σχήμα 5.8). Στα 6 clicks (Σχήμα 5.9) ξεκινάμε να έχουμε τις πρώτες ενδείξεις για PPF και κατά συνέπεια για την πλαστικότητα της μνήμης. Πράγματι, παρατηρούμε αύξηση στην ένταση του ρεύματος εξόδου με την εφαρμογή κάθε επόμενου πιεζοηλεκτρικού παλμού και ροή κατά την εφαρμογή της τάσης ανάγνωσης $V_{\text{piezo}}/2$ (διπολική εναλλαγή) επαληθεύοντας τον ισχυρισμό μας. Το ίδιο φαινόμενο είναι πλέον ξεκάθαρο και κατά τη διάρκεια του κύκλου των 9 clicks (Σχήμα 5.10).



Σχήμα 5.10 Πειραματικά αποτελέσματα: 9 Clicks

Τέλος ενδιαφέρον παρουσιάζουν τα αποτελέσματα του Σχήματος 5.11, όπου αφορούν την εφαρμογή 12 clicks. Παρατηρούμε πως η σύναψη είναι πιο ευαίσθητη στην εγγραφή, καθώς χρειάζονται πολύ λιγότεροι παλμοί για να πάρουμε το μέγιστο ρεύμα εξόδου δηλαδή έχουμε κατά μεγάλο βαθμό ταχύτερο PPF. Στη συνέχεια της μέτρησης, στη φάση εφαρμογής των παραγόμενων από τον Arduino παλμών ανάγνωσης, βλέπουμε σταδιακή μείωση του ρεύματος εξόδου, έως ότου αυτό σταθεροποιηθεί περίπου στα $I_{out} = 0.5 \cdot 10^{-4} A$, μέτρηση συμβατή με το φαινόμενο το PPD, όπως το αναφέραμε στην παράγραφο 5.2.1 και το είδαμε στο Σχήμα 5.1.



Σχήμα 5.11 Πειραματικά αποτελέσματα: 12 Clicks

5.3 Μάθηση

Για την τελευταία παρατήρηση ας επικεντρωθούμε στα Σχήμα 5.5 και 5.11. Η πρώτη μόνιμη εγγραφή στη σύναψη παρατηρείται στον κύκλο των 6 clicks. Το ρεύμα εξόδου κατά την εφαρμογή των $V_{pulse}/2$ παλμών κυμαίνεται στα $I_{out} \cong 0.8 \cdot 10^{-4} A$. Μετά τον κύκλο των 8 clicks, έπειτα από ένα σύντομο φαινόμενο PPD κατά την εφαρμογή των παλμών διέγερσης, το ρεύμα ανάγνωσης πέφτει στα $\cong 0.4 \cdot 10^{-4} A$, και αυξάνεται σε κάθε επόμενο κύκλο, μέχρι να προκύψει το έντονο PPD τους Σχήματος 5.11. Παρατηρούμε επίσης ότι το PPF κάθε επόμενης περίπτωσης είναι πιο έντονο – ευαίσθητο από την προηγούμενη. Στο σημείο αυτό είναι σημαντικό να οριστεί η έννοια της μάθησης.

Ορισμός: Ως μάθηση (*learning*) ορίζεται η απόκτηση γνώσεων ή δεξιοτήτων που είναι αποτέλεσμα εμπειριών, οδηγιών ή και των δύο [16].

Ως μνήμη ορίζεται η αποθήκευση της αποκτηθείσας γνώσης, ώστε να είναι διαθέσιμη σε μελλοντική ανάκληση. Η νευρική αλλαγή που ευθύνεται για την διατήρηση ή την αποθήκευση

της γνώσης ονομάζεται μνημονικό ίχνος. Γενικά στη μνήμη αποθηκεύονται έννοιες και όχι λεκτική πληροφορία. Με άλλα λόγια ένας νευρομορφικός υπολογιστής αποθηκεύει «εμπειρική πληροφορία» την ώρα που ένας συμβατικός αποθηκεύει δυαδικά ψηφία.

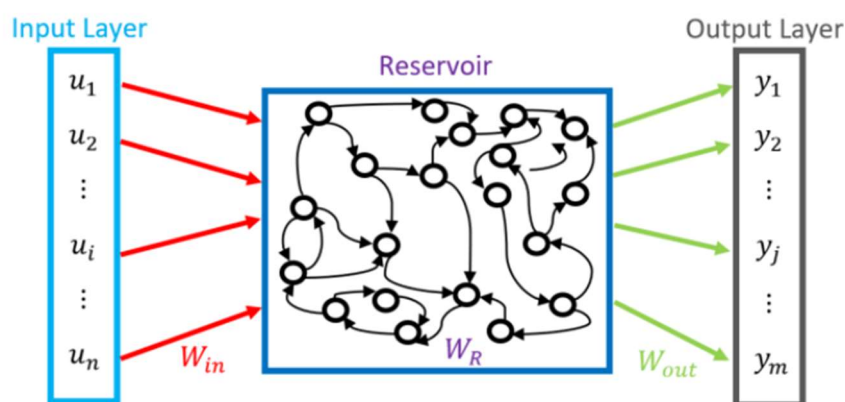
Η αποθήκευση της αποκτηθείσας πληροφορίας γίνεται μέσω δύο βασικών σταδίων την βραχυπρόθεσμη μνήμη και τη μακροπρόθεσμη (STM & LTM αντίστοιχα). Η πρώτη στον οργανισμό μπορεί να διαρκέσει από μερικά δευτερόλεπτα ως και μερικές ώρες. Η δεύτερη διαρκεί από ημέρες ως και χρόνια. Η διαδικασία μετατροπής και παγίωσης των ιχνών της STM σε αποθήκες LTM ονομάζεται εδραίωση μνήμης (consolidation) [16].

Ας επιστρέψουμε τώρα στο Σχήμα 5.5. Πριν τη πρώτη μόνιμη εγγραφή παρατηρούμε μόνο ένα ασθενές PPF, το οποίο με βάσει τους παραπάνω ορισμούς μπορούμε εύκολα να συσχετίσουμε με την STM. Στη συνέχεια παρατηρούμε τα φαινόμενα που περιγράψαμε στην αρχή αυτής της παραγράφου. Μετά την εφαρμογή των 12 clicks και το τελευταίο PPD – η σύναψη φαίνεται να ισορροπεί κατά την εφαρμογή της $V_{pulse}/2$, επιστρέφοντας $I_{out} \cong 0.6 \cdot 10^{-4} A$, τιμή που μεταφράζεται σε σημαντικά μικρότερη αγωγιμότητα από αυτή των προηγούμενων περιπτώσεων. Το τελευταίο εύρημα είναι αντίστοιχο της προσαρμογής μίας βιολογικής σύναψης σε δεδομένο εξωτερικό ερέθισμα (habituation). Όταν εφαρμόζεται ένα επαναλαμβανόμενο σήμα σε μία σύναψη, η σύναψη προσαρμόζεται σε περαιτέρω εφαρμογή του σήματος αυτού ρίχνοντας την απόκρισής της ή αλλιώς την αγωγιμότητά της. Το φαινόμενο αυτό αποτελεί και την πλέον ισχυρότερη απόδειξη της εδραίωσης της μνήμης.

Γενικότερα η όλη πειραματική διαδικασία είναι μία τέλεια προσομοίωση της διαδικασίας εκμάθησης, με ξεκάθαρα ορατά τα στάδια της STM και της LTM και τέλεια απεικόνιση της σταδιακής εδραίωσης της μνήμης. Στην περίπτωση αυτή, το κύκλωμα μαθαίνει να ανταποκρίνεται ταχύτερα στα ερεθίσματα – μνημονικά ίχνη που εφαρμόζονται από τον πιεζοηλεκτρικό αισθητήρα, μία διαδικασία που προσομοιάζει την μυϊκή μνήμη και τη δεξιοτεχνία. Η πλήρης προσομοίωση αποτελεί αντικείμενο εκτεταμένης μελέτης τόσο της επιστήμης υπολογιστών όσο και πιο «εφαρμοσμένων - applied» επιστημών όπως η ρομποτική και η βιο-ιατρική. Η βάση πλέον δίνεται επίσης στη δημιουργία ενός κυκλώματος πλήρους προσομοίωσης νευρικού κυττάρου (νευρώνας, δενδρίτες και σύναψη), που θα μπορεί να φέρει εις πέρας πιο περίπλοκους νευρομορφικούς υπολογισμούς, όντας πιο κοντά στο ανθρώπινο νευρικό σύστημα [17]. Η μελέτη τέτοιων εννοιών ξεπερνά τα όρια της εργασίας αυτής.

6 Νευρομορφικά δικτυώματα

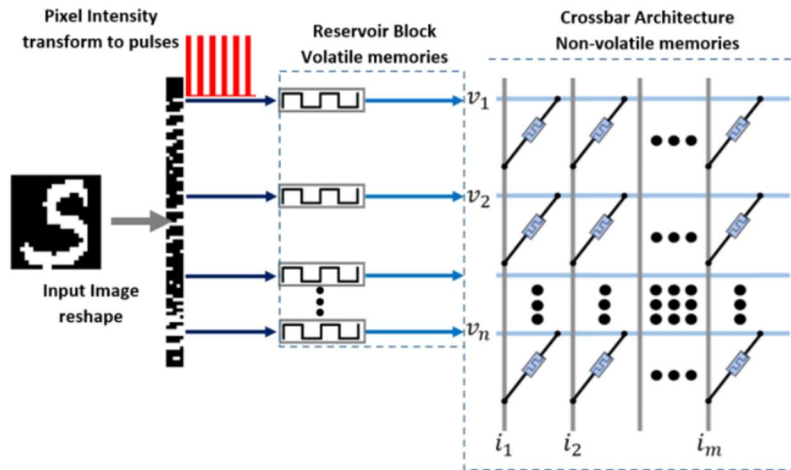
Στο τελευταίο κεφάλαιο θα παρουσιάσουμε μία προτεινόμενη εφαρμογή των νευρομορφικών δικτύων αυτή των υπολογιστικών συστημάτων δεξαμενής δεδομένων (Reservoir Computers – στο εξής: RCs). Αυτά τα δίκτυα ανήκουν στην ευρύτερη κατηγορία των «επαναλαμβανόμενων» νευρωνικών δικτύων (Recurrent Neural Networks – στο εξής RNNs). Πρόκειται για μία κλάση νευρωνικών δικτύων στα οποία οι συνδέσεις μεταξύ των κόμβων τους σχηματίζουν ένα, κατευθυνόμενο ή μη κατευθυνόμενο «γράφημα» κατά τη διάρκεια μίας χρονικής ακολουθίας. Αυτή η ιδιότητα τα καθιστά άριστα για την επίλυση υπολογιστικών προβλημάτων όπως η αναγνώριση χαρακτήρων, φωνής, ή άλλων δυναμικώς εξαρτώμενων διεργασιών. Στο Σχήμα 6.1 βλέπουμε μία αναπαράσταση ενός τέτοιου δικτύου.



Σχήμα 6.1 Αναπαράσταση ενός δικτύου Reservoir Computing

Ως RNN το RC μπορεί να μετατρέψει τα δεδομένα στην εισαγωγή σε έναν διαστατικά ανώτερο χώρο, μέσω των δεξαμενών πληροφορίας (στο εξής: reservoir blocks). Για να το πετύχουμε αυτό στη προτεινόμενη εφαρμογή, το reservoir block εκμεταλλεύεται τις νευρομορφικές ιδιότητες των CBRAM, πιο συγκεκριμένα το φαινόμενο PPF, αυστηρά για την περίπτωση πτητικής λειτουργίας του στοιχείου μνήμης, δηλαδή σε συνθήκες βραχυπρόθεσμης μνήμης. Έπειτα η έξοδος του reservoir block οδηγείται σε ένα δίκτυωμα crossbar με μη πτητικά ενεργά στοιχεία μνήμης (bipolar mode CBRAMs). Η διάταξη αυτή απεικονίζεται καλύτερα στο Σχήμα 6.2.

Μπορούμε να φανταστούμε τις δεξαμενές πληροφορίας σαν δεξαμενές αποθήκευσης νερού, με μία καταβόθρα εκτόνωσης. Κάθε φορά που προστίθεται νερό στη δεξαμενή η στάθμη του νερού ανεβαίνει, αλλά λόγω της ύπαρξης της καταβόθρας, όταν δεν προστίθεται νερό, η στάθμη μειώνεται συνεχώς. Στη υπολογιστική εφαρμογή μας το νερό αντιστοιχείται στη θετική τάση ενός παλμού, ενώ η μεταβαλλόμενη στάθμη στην αγωγιμότητα του στοιχείου μνήμης.



Σχήμα 6.2 Reservoir Computer Concept

6.1 Λειτουργία του δικτύωματος δεξαμενής δεδομένων (RC)

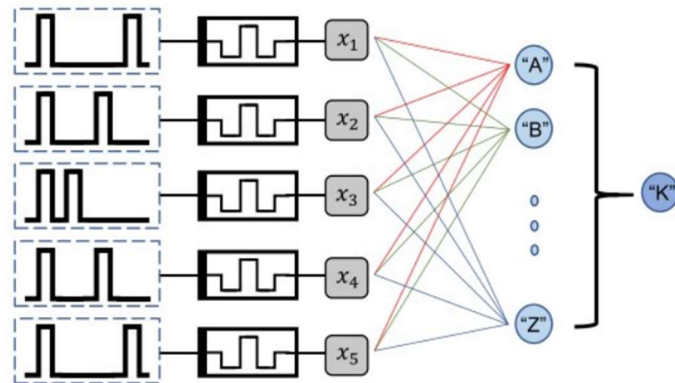
Για να καταλάβουμε τις αρχές λειτουργίας μία τέτοιας διάταξης, αρκεί να εξηγήσουμε πως αυτή θα λειτουργεί. Σκοπεύουμε να εκπαιδεύσουμε το RC του Σχήματος 6.2 έτσι ώστε να μπορεί να πραγματοποιεί αναγνώριση χειρόγραφων λατινικών χαρακτήρων, όπως αυτοί ορίζονται από τη βάση δεδομένων MNIST [18].

Το RC όπως προαναφέραμε αποτελεί ένα δίκτυωμα το οποίο στοχεύει στην πραγματοποίηση παράλληλων υπολογισμών, όταν η πληροφορία στην είσοδο είναι χρονικά εξαρτώμενη. Έτσι μπορούμε να κωδικοποιήσουμε – ψηφιοποιήσουμε τον MNIST χαρακτήρα, ανάλογα με το «χρώμα» του κάθε pixel του, όπως φαίνεται στο Σχήμα 6.3 για την απλοποιημένη περίπτωση του χαρακτήρα «K».

0	1	1	0
0	1	0	1
0	0	1	1
0	1	0	1
0	1	1	0

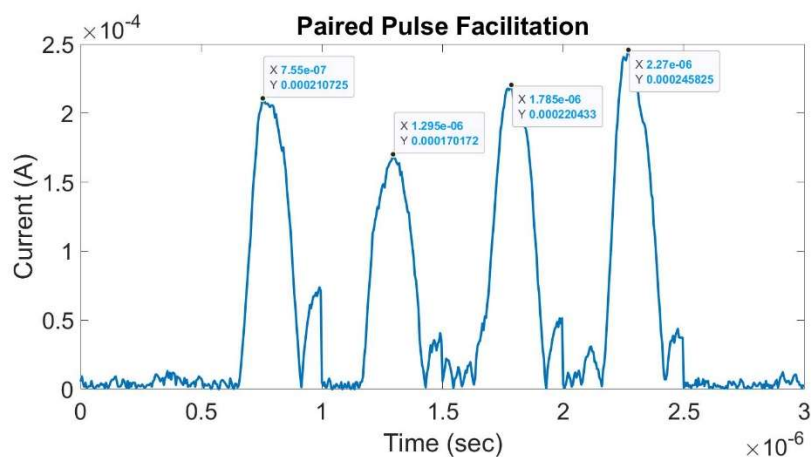
Σχήμα 6.3 Ψηφιοποίηση του "K"

Κάθε μαύρο pixel θα αναπαρίσταται από το λογικό 0, ενώ ένα λευκό από το λογικό 1. Έτσι ο χαρακτήρας μπορεί να αναπαρασταθεί από πέντε 4-bit μεταβλητές, οι οποίες με τη σειρά τους θα μετατραπούν σε σήμα εισόδου μορφής παλμοσειράς, με παλμούς δεδομένου χρονικού εύρους, πλάτους καθώς και χρονικής απόστασης μεταξύ τους. Η λογική τιμή 1 αναπαρίσταται στη παλμοσειρά με παρουσία ενός παλμού. Απουσία του παλμού μεταφράζεται ως λογικό 0. Η διαδικασία αυτή ονομάζεται αποκωδικοποίηση (decoding) και αποτελεί το πρώτο μέρος του RC. Οι πέντε μεταβλητές, στη συνέχεια εισέρχονται στο σύστημα παράλληλα (Σχήματα 6.2 και 6.4).



Σχήμα 6.4 Εισαγωγή μεταβλητών σε RC

Μετατρέπονται σε αναλογικό ερέθισμα, μέσω του φαινομένου PPF, στο δεύτερο μέρος της διάταξης, το reservoir block. Με άλλα λόγια ο κάθε κόμβος (CBRAM) του reservoir block αποτελεί μία σύναψη βραχυπρόθεσμης μνήμης. Μία τέτοια μετατροπή, για την περίπτωση της μεταβλητής «1111» φαίνεται στο Σχήμα 6.5, όπου έχουμε ορίσει HIGH@ 0.1V και χρονικό εύρος 100 ns.



Σχήμα 6.5 Μετατροπή της μεταβλητής "1111" σε αναλογικό σήμα

Το σήμα που θα εισέλθει στο crossbar είναι διαστατικά ανώτερο, καθώς πέρα από τη χρονική εξάρτηση της παλμοσειράς εισόδου στο reservoir block, δηλαδή της απουσίας ή παρουσίας

παλμού, παρουσιάζει και μεταβολή στην ένταση του σήματος εξόδου. Με άλλα λόγια καταφέραμε να μετατρέψουμε μία χρονικά εξαρτώμενη πληροφορία (temporal) σε σήμα εισόδου το οποίο παρουσιάζει μεταβολές και στο πλάτος του με την πάροδο του χρόνου (spatio-temporal), μορφή πιο εύκολη να επεξεργαστεί από το crossbar.

Το τελευταίο μέρος αποτελείται από το ίδιο το crossbar, αποτελούμενο από συνάψεις μακροχρόνιας μνήμης. Τα βάρη αυτών των συνάψεων (δηλαδή η αγωγιμότητα κάθε CBRAM) μπορούν να οριστούν μέσω μία διαδικασίας επιβλεπόμενης μάθησης. Λόγω των δυσκολιών που αναφέρθηκαν στο Κεφάλαιο 2 ως προς την υλοποίηση ενός crossbar, η διαδικασία αυτή προσομοιώθηκε υπολογιστικά με βάσει τα πειραματικά δεδομένα που προέκυψαν κατά τη διάρκεια αυτής της εργασίας [19].

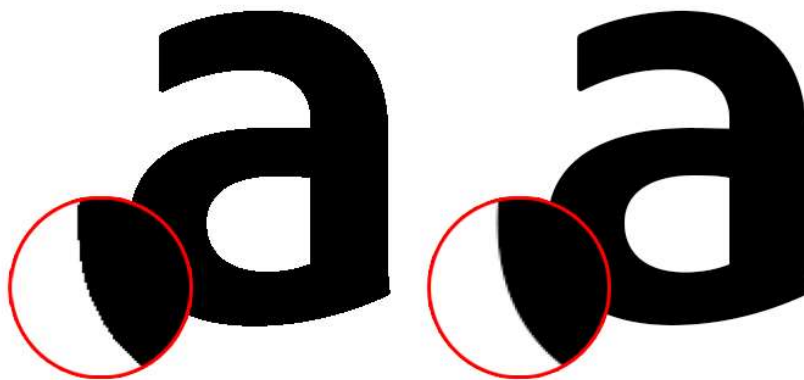
6.2 Πειραματική διαδικασία

6.2.1 Ακρίβεια στον αριθμό των bits

Για να καταφέρουμε να προχωρήσουμε σε έναν ολοκληρωμένο χαρακτηρισμό του reservoir block πρέπει πρώτα να ορίσουμε την «ακρίβεια των bits» (bit precision). Όπως συμβαίνει με κάθε μετατροπή αναλογικού σήματος σε ψηφιακό (και αντίστροφα) όσο περισσότερες (και μεγαλύτερες) ψηφιακές μεταβλητές μπορούμε να χρησιμοποιήσουμε για την αναπαράσταση μίας πληροφορίας (εδώ: ενός χαρακτήρα MNIST) τόσο καλύτερη και πιο αναλυτική θα είναι η τελική αναπαράσταση. Μία τέτοια μετατροπή μπορεί να έχει μεγάλο υπολογιστικό κόστος στις δυνατότητες του συστήματος.

Μικρότερες πιο απλές μεταβλητές κοστίζουν λιγότερο στο σύστημα προκειμένου να τις επεξεργαστεί. Υπάρχει όμως ο κίνδυνος να εμφανιστεί το φαινόμενο aliasing, δηλαδή η αναπαράσταση του αρχικού σήματος να προκαλέσει «παραμορφώσεις» (artifacts) οι οποίες στην καλύτερη περίπτωση θα προσθέσουν σφάλματα κατά την επεξεργασία και την εκπαίδευση του δικτύου, με την καταστροφή της αρχικής πληροφορίας πιθανή στη χειρότερη περίπτωση. Ένα τέτοιο παράδειγμα παρουσιάζουμε στο Σχήμα 6.6.

Είναι σημαντικό να βρεθεί η «χρυσή τομή» μεταξύ ανάλυσης και κόστους υπολογισμού για δεδομένο κατά τον σχεδιασμό οποιουδήποτε υπολογιστικού συστήματος. Επιπλέον πρέπει να έχουμε στο νου μας, όπως αναφέρθηκε στο Κεφάλαιο 1, ότι η χρήση περισσότερων στοιχείων μνήμης αυξάνουν και την κατανάλωση ισχύος. Για την σχεδίαση ενός ολοκληρωμένου συστήματος πρέπει οπωσδήποτε να ληφθούν οι προαναφερθείσες παράμετροι υπόψη.



Σχήμα 6.6 Aliasing

6.2.2 Μετρήσεις

Πραγματοποιήθηκαν μετρήσεις για μεταβλητές 3-bit, 4-bit και 5-bit προκειμένου να εκτιμηθεί η απόδοση των Ag-CBRAM για χρήση ως μεμονωμένες δεξαμενές (reservoir nodes). Μία τέτοια χρήση, όπως είδαμε σε προηγούμενη παράγραφο απαιτεί τα στοιχεία να βρίσκονται σε λειτουργία κατωφλίου. Για αυτό το λόγο επιλέγουμε παλμούς μικρής τάσης και χρονικού εύρους, αδύναμους να προκαλέσουν διπολική εναλλαγή.

Συγκεκριμένα οι παλμοί που εφαρμόζονται έχουν:

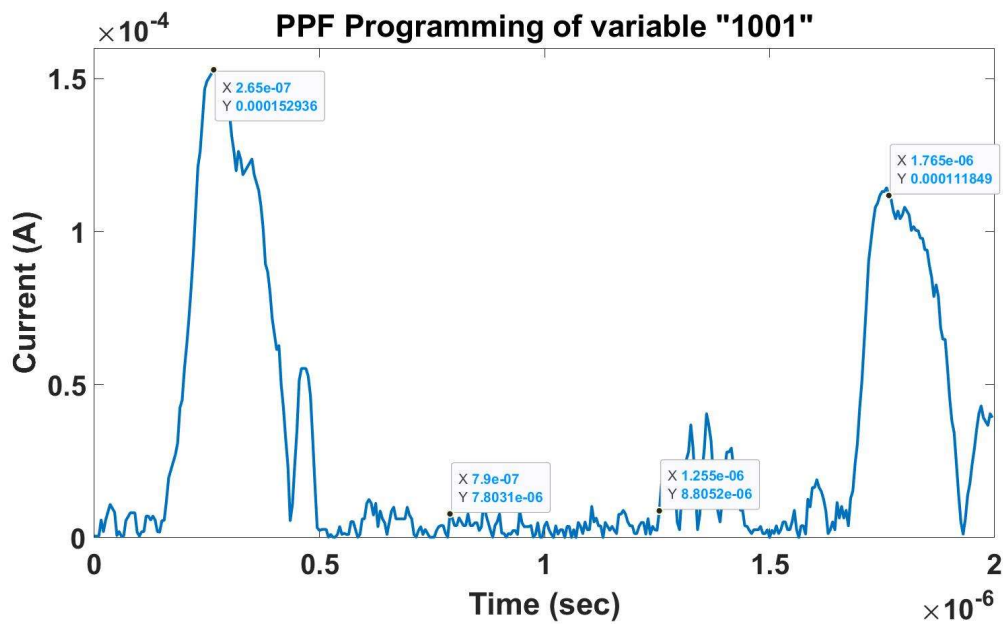
- 0.1V HIGH
- 0.0V LOW
- 100ns εύρος (width) – 100ns χρόνος ανάβασης και πτώσης της τάσης των παλμών (rising & falling time) – 200 ns κενό μεταξύ παλμών (delay)

Οι μετρήσεις έγιναν μέσω του συστήματος χαρακτηρισμού Keithley 4200, χρησιμοποιώντας μία πρωτότυπη βιβλιοθήκη για την παραγωγή των απαιτούμενων σημάτων – μεταβλητών. Ο κώδικας για αυτή τη διαδικασία φαίνεται στο Ένθετο IV.

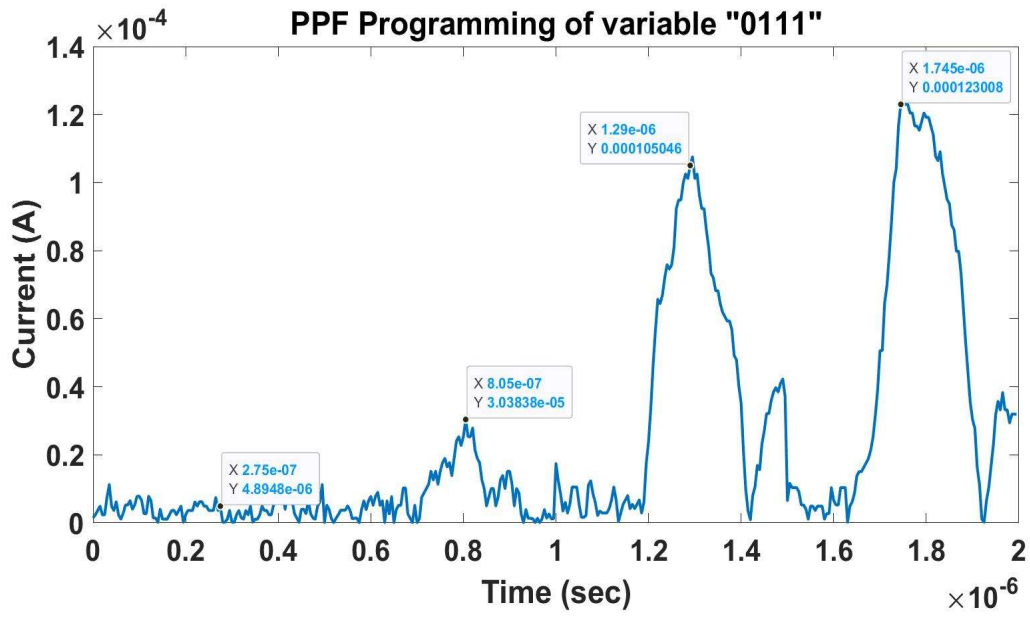
Στα Σχήματα 6.7 και 6.8 βλέπουμε τη μηχανική προγραμματισμού δύο διαφορετικών μεταβλητών στην ίδια σύναψη. Σε όλες τις περιπτώσεις βασιζόμαστε στο φαινόμενο PPF, όπως ορίστηκε στα προηγούμενα κεφάλαια, έτσι ώστε να προκαλέσουμε αλλαγή στην αγωγιμότητα της σύναψης. Απουσία ή παρουσία του παλμού (σε αντιστοιχία με το στοιχείο 0 ή 1), καθώς και ο συνολικός αριθμός των παλμών, αλλάζουν τη αποτελεσματικότητα του PPF, κάτι που αντικατοπτρίζεται στην ένταση του ρεύματος στο τελευταίο bit κάθε μεταβλητής. Τα συνολικά αποτελέσματα παρουσιάζονται στα παρακάτω Σχήματα.

Ορίζουμε ως ποσοστό επιτυχίας της εγγραφής, το σύνολο (τοίς εκατό) των μεταβλητών που ξεχωρίζουν προς τον αριθμό όλων των μεταβλητών για κάθε περίπτωση ακρίβειας των bits. Παρατηρούμε ότι η διάταξη είναι λιγότερο από ιδανική για τις περιπτώσεις των 5- και 4-bit,

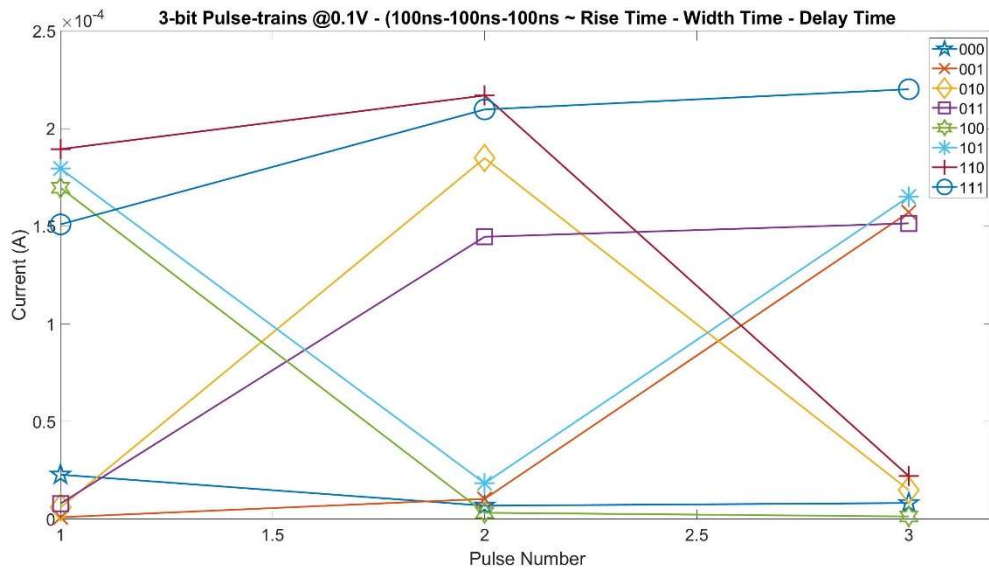
με ποσοστό επιτυχίας 62.5% και 68.75% αντίστοιχα. Το αντίθετο ισχύει για την περίπτωση των 3 bits, με τέλεια αποτελέσματα εγγραφής (100%) (Σχήματα 6.9 – 6.11). Τα παραπάνω επιβεβαιώνουν ότι περιγράψαμε για την ακρίβεια των bits. Ένα τέτοιο σύστημα μπορεί να επεξεργαστεί, ή αλλιώς να μετατρέψει άριστα δεδομένα από χρονικώς εξαρτώμενα σε σήματα ανώτερης διάστασης, σε λειτουργία ακρίβειας 3-bit. Ο «πήγης» αυτός μπορεί να υπερκεραστεί χρησιμοποιώντας είτε διαφορετικό σχήμα ανάγνωσης, (εγγραφή στα 200mV έδειξε μία αύξηση 10% στην επιτυχία εγγραφής) ή ακόμη και τη χρήση αυστηρά πτητικών συνάψεων που λειτουργούν σε υψηλότερες τάσεις εναλλαγής, όπως αυτές της αρχιτεκτονικής Pt-SiO₂-Ag, μέθοδοι οι οποίες δείχνουν πειραματικά πολλά υποσχόμενα αποτελέσματα, η ανάλυση των οποίων ξεπερνά τα όρια αυτής της εργασίας.



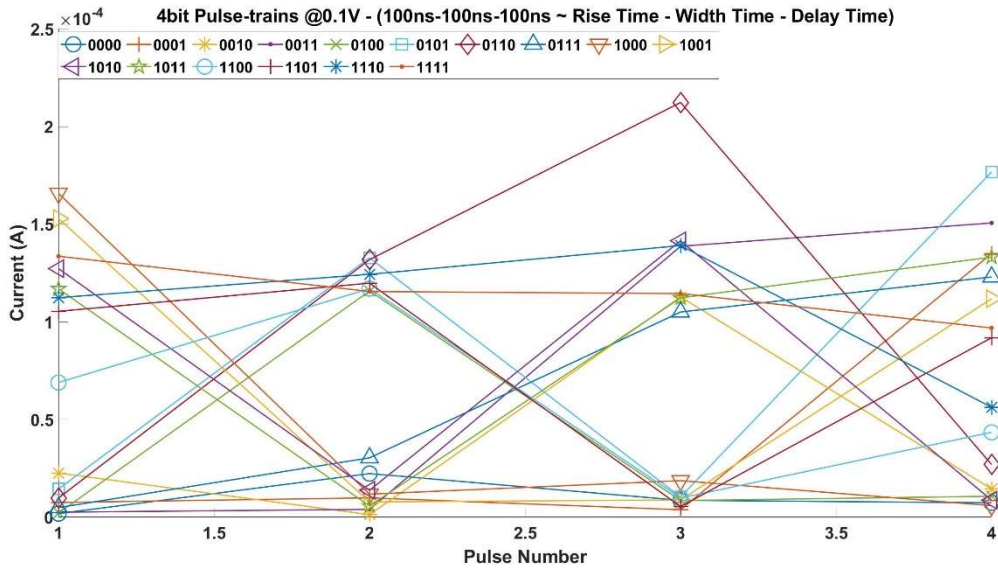
Σχήμα 6.7 Προγραμματισμός σύνταξης - Μεταβλητή "1001"



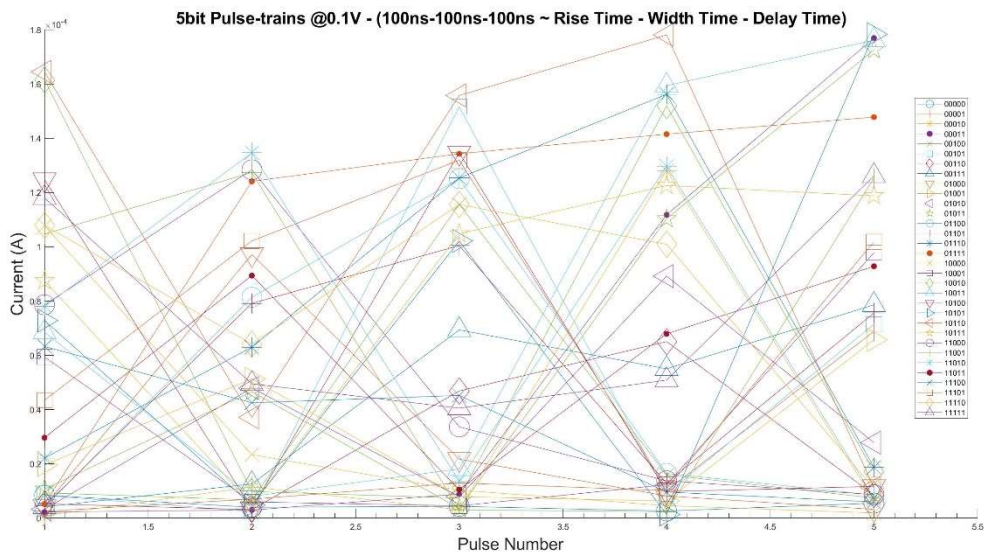
Σχήμα 6.8 Προγραμματισμός σύνταξης - Μεταβλητή "0111"



Σχήμα 6.9 Αποτελέσματα Μεταβλητής 3-bit



Σχήμα 6.10 Αποτελέσματα Μεταβλητής 4-bit



Σχήμα 6.11 Αποτελέσματα Μεταβλητής 5-bit

6.3 Προσομοίωση

Λόγω αδυναμίας ολοκλήρωσης της διάταξης Crossbar με μη πτητικές μνήμες, πραγματοποιήθηκαν πειράματα προσομοίωσης, βάσει των αποτελεσμάτων της προηγούμενης παραγράφου, προκειμένου να κατανοήσουμε κατά πόσο το προτεινόμενο σύστημα ανταποκρίνεται επιτυχώς στις προσδοκίες μας.

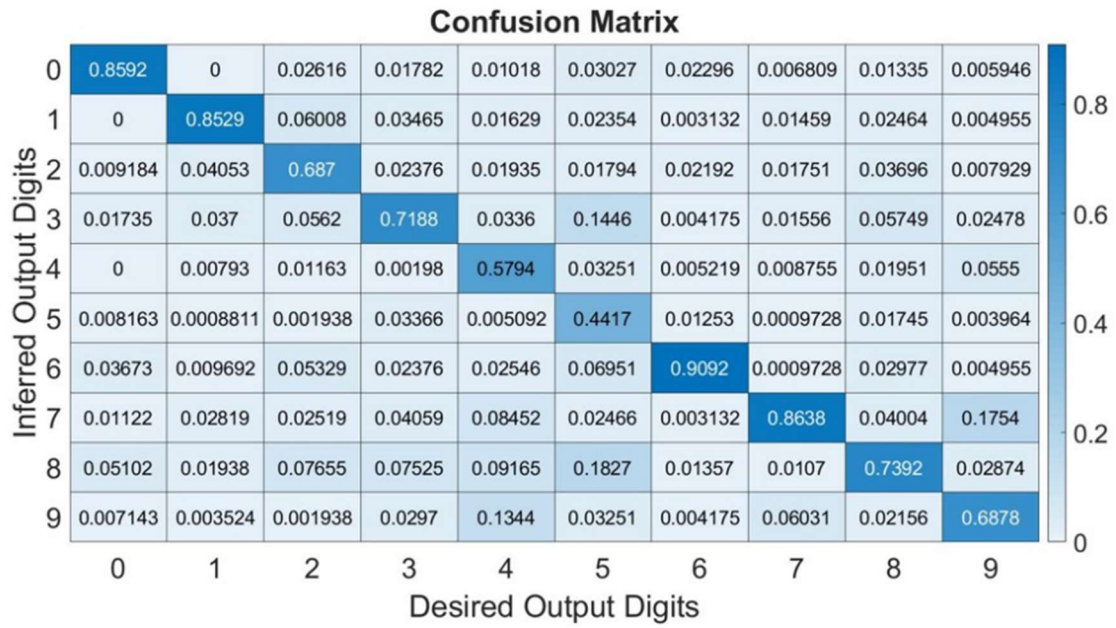
Αρχικά είδαμε πως η εγγραφή με μικρότερες μεταβλητές σημαίνει μεγαλύτερη ακρίβεια στη φάση των υπολογισμών. Μεγαλύτερο bit precision θα σήμαινε πως τα ίδια δεδομένα μπορούν να εισέλθουν στη διάταξη με χρήση λιγότερων στοιχείων στο crossbar και πτητικών συνάψεων στο reservoir block, μειώνοντας σημαντικά την ολική κατανάλωση του συστήματος. Προσομοιώσεις πραγματοποιήθηκαν για τις περιπτώσεις 2-, 3-, 4- και 5-bit precision. Τα αποτελέσματα φαίνονται στον παρακάτω πίνακα.

Bit Precision	2	3	4	5
Reservoir Synapses	210	140	105	84
Crossbar Neurons	4200	2800	2100	1680
Test Accuracy (%)	83.62	82.68	77.83	76.63
Total Power Consumption (μW)	429	287	215	172

Πίνακας 6-1 Ακρίβεια και κατανάλωση ισχύος κατά την εφαρμογή μεταβλητών

Προκύπτει ότι η βέλτιστη περίπτωση είναι αυτή των 3-bit, με πτώση ακρίβειας περί τα ~1.2% σε σχέση με την περίπτωση των 2-bits, αλλά με τη μισή κατανάλωση ισχύος. Για την ακραία περίπτωση των 5-bits, αν και ακατάλληλη ώστε να προταθεί για χρήση σε πραγματικές εφαρμογές, τα αποτελέσματα προδίδουν ένα ποσοστό επιτυχίας υπολογισμού περί τα 76% με κατανάλωση 172 μW , μετά από $50 \cdot 10^3$ εποχές εκπαίδευσης, με τους χαρακτήρες να κωδικοποιούνται σε $10 \cdot 10^3$ κομμάτια εκπαίδευσης (training batches) ανά εποχή. Τα παραπάνω φαίνονται πιο αναλυτικά στον πίνακα σύγχυσης (confusion matrix) του Σχήματος 6.12. Το RC κλήθηκε να αναγνωρίσει αριθμητικούς χαρακτήρες. Στον οριζόντιο άξονα τοποθετούνται οι χαρακτήρες στην είσοδο του συστήματος, ενώ στον κάθετο οι χαρακτήρες στην έξοδο. Το κάθε στοιχείο του πίνακα αποτελεί τον λόγο των περιπτώσεων που προέκυψε ο αντίστοιχος χαρακτήρας ως αποτέλεσμα, προς τον συνολικό αριθμό των προσπαθειών να προκύψει ο επιθυμητός χαρακτήρας εισόδου (εξίσωση 6.3.1). Στην ιδανική περίπτωση όλα τα στοιχεία της διαγώνιου θα πρέπει να είναι 1 με τα λοιπά στοιχεία του πίνακα να είναι 0. Το γεγονός ότι τα αποτελέσματα απέχουν από αυτό μας δείχνει προς ποια κατεύθυνση πρέπει να κινηθούμε κατά τη βελτιστοποίηση ενός τέτοιου συστήματος.

$$\text{σύγχυση} = \frac{\text{\#αποτελέσματα εξόδου}}{\text{\#δεδομένα εισόδου}} \quad (6.3.1)$$



Σχήμα 6.12 Πίνακας σύγκρισης υπολογισμών στα 5-bits - 76% ποσοστό επιτυχίας

7 Επίλογος - Συμπεράσματα

Στην εργασία αυτή έγινε μία σφαιρική μελέτη των μνημών εναλλαγής αντίστασης τύπου CBRAM που παράγονται από το εργαστήριό μας. Αρχικά μελετήσαμε – χαρακτηρίσαμε τη λειτουργία τους τόσο σε συνθήκες συνεχής όσο και σε συνθήκες παλμικής παροχής, αποδεικνύοντας τη δυνατότητά τους για χρήση τόσο σε εφαρμογές πτητικής όσο και σε εφαρμογές μη πτητικής μνήμης.

Παρουσιάσαμε τη βασική σκέψη λειτουργίας μίας διάταξης crossbar, τόσο σε επίπεδο προσομοίωσης και σχεδιασμού όσο και στο επίπεδο της υλοποίησής της. Με τις απαιτήσεις της σύγχρονης τεχνολογίας τόσο σε αποθηκευτικό χώρο (όσον αφορά τις εφαρμογές μνήμης) όσο και σε νευρομορφικές εφαρμογές καθώς αλλά και τις πρόσθετες απαιτήσεις λόγω απαίτησης πυκνότερων συστημάτων (scaling), είναι επιτακτική ανάγκη δημιουργίας και βελτιστοποίησης τέτοιων διατάξεων, καθώς πάνω σε αυτές βασίζεται το μεγαλύτερο πλεονέκτημα των CBRAM, η πυκνότητα ολοκλήρωσης τέτοιων διατάξεων. Παρατηρήσαμε και στις δύο περιπτώσεις την αντίστροφη σχέση μεγέθους της διάταξης σε σχέση με το μέγεθος του παραθύρου ανάγνωσης, λόγω της ύπαρξης παρασιτικών ρευμάτων, και προτείναμε τρόπους αντιμετώπισης αυτών, τόσο με τη χρήση τάσεων bias όσο και με τη χρήση transistor επιλογής.

Στη συνέχεια παρουσιάσαμε τη δυνατότητα της αυτοσχέδιας, πρωτότυπης διάταξης να προχωρήσει σε χαρακτηρισμό ενός μεμονωμένου στοιχείου μνήμης, σε σύγκριση με τα αντίστοιχα αποτελέσματα μίας εξειδικευμένης διάταξης χαρακτηρισμού ημιαγωγών. Αποδείξαμε ότι διάταξή μας λειτουργεί άρτια σε παλμική λειτουργία, αλλά χρειάζεται περαιτέρω ανάπτυξη σε συνεχή λειτουργία.

Έπειτα αφοσιωθήκαμε στις νευρομορφικές ιδιότητες. Με τη χρήση της αυτοσχέδιας διάταξης προγραμματίσαμε μία μη πτητική μνήμη η οποία προσομοίαζε μία σύναψη ενός αισθητήριου νευρώνα. Για μεγαλύτερο επίπεδο ρεαλισμού, χρησιμοποιήθηκε ένας πιεζοηλεκτρικός αισθητήρας για την επικοινωνία της σύναψης με το περιβάλλον της. Μπορέσαμε να παρακολουθήσουμε και να αναλύσουμε τα διαφορετικά στάδια της διαδικασίας εκμάθησης – εδραίωσης μνήμης στη σύναψη, από τη φάση της βραχυπρόθεσμης (STM) μέχρι της μακροπρόθεσμης (LTM) μνήμης, επιβεβαιώνοντας την αντιστοιχία με τις βιολογικές συνάψεις.

Στο τελευταίο μέρος προτείναμε τη χρήση των CBRAM του εργαστηρίου μας σε μία διάταξη Reservoir Computing, η οποία απαιτεί πτητική λειτουργία σε επίπεδο σύναψης εισόδου όσο και μη πτητική σε επίπεδο crossbar. Έγινε μία διεξοδική μελέτη της επιτυχίας εγγραφής στη πτητική σύναψη εισόδου, με διάφορες παλμοσειρές εισόδου και με βάσει τα αποτελέσματα αυτών προσομοιάστηκε η λειτουργία μίας τέτοιας διάταξης για διεργασίες αναγνώρισης χαρακτήρων.

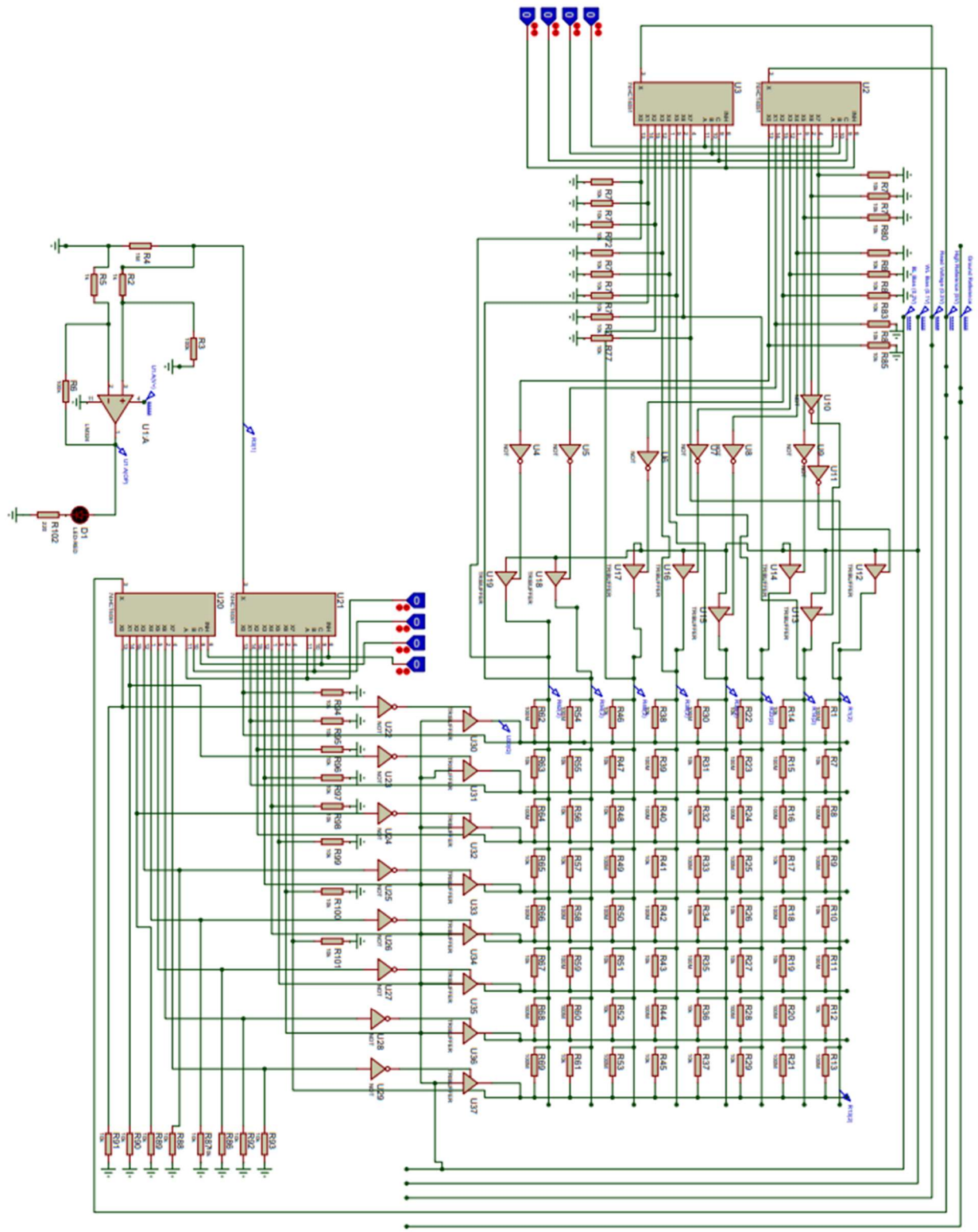
Βεβαίως υπάρχουν περιθώρια βελτίωσης σε πολλά σημεία από αυτά που παρουσιάστηκαν. Όσον αφορά τη custom διάταξη η χρήση Low Level - BareMetal προγραμματισμού στον μικρό-ελεγκτή, ή η αντικατάστασή του από καταλληλότερες λύσεις (όπως τα FPGAs) θα μας επιτρέψει να αυξήσουμε τόσο την ανάλυση όσο και τη ταχύτητα λειτουργίας αντίστοιχων διατάξεων. Επιπλέον συστήματα όπως οι αισθητήρες ανάγνωσης χρήζουν βελτιστοποίησης, έτσι ώστε να διευρυνθεί το παράθυρο ανάγνωσης της διάταξης, πλησιάζοντας τα επίπεδα του Keithley 4200.

Το βασικότερο πρακτικό πρόβλημα που έχουμε να αντιμετωπίσουμε είναι αυτό της αδυναμίας wire-bonding κατά το packaging των μνημών στα πλαίσια του εργαστηρίου. Χωρίς wire-bonding αδυνατούμε να προχωρήσουμε στην υλοποίηση μίας αρχιτεκτονικής crossbar με τις CBRAM ως ενεργά στοιχεία. Πρέπει να γίνουν όλες οι απαραίτητες αλλαγές τόσο στη διαδικασία της λιθογραφίας όσο και στη παραμετροποίηση του συγκολλητή ώστε να αντιπαρέλθουμε και αυτό το εμπόδιο.

8 Ένθετο I: Σχέδιο Διάταξης crossbar χωρίς Selector Transistors

Υψηλότερη ανάλυση διαθέσιμη στο link:

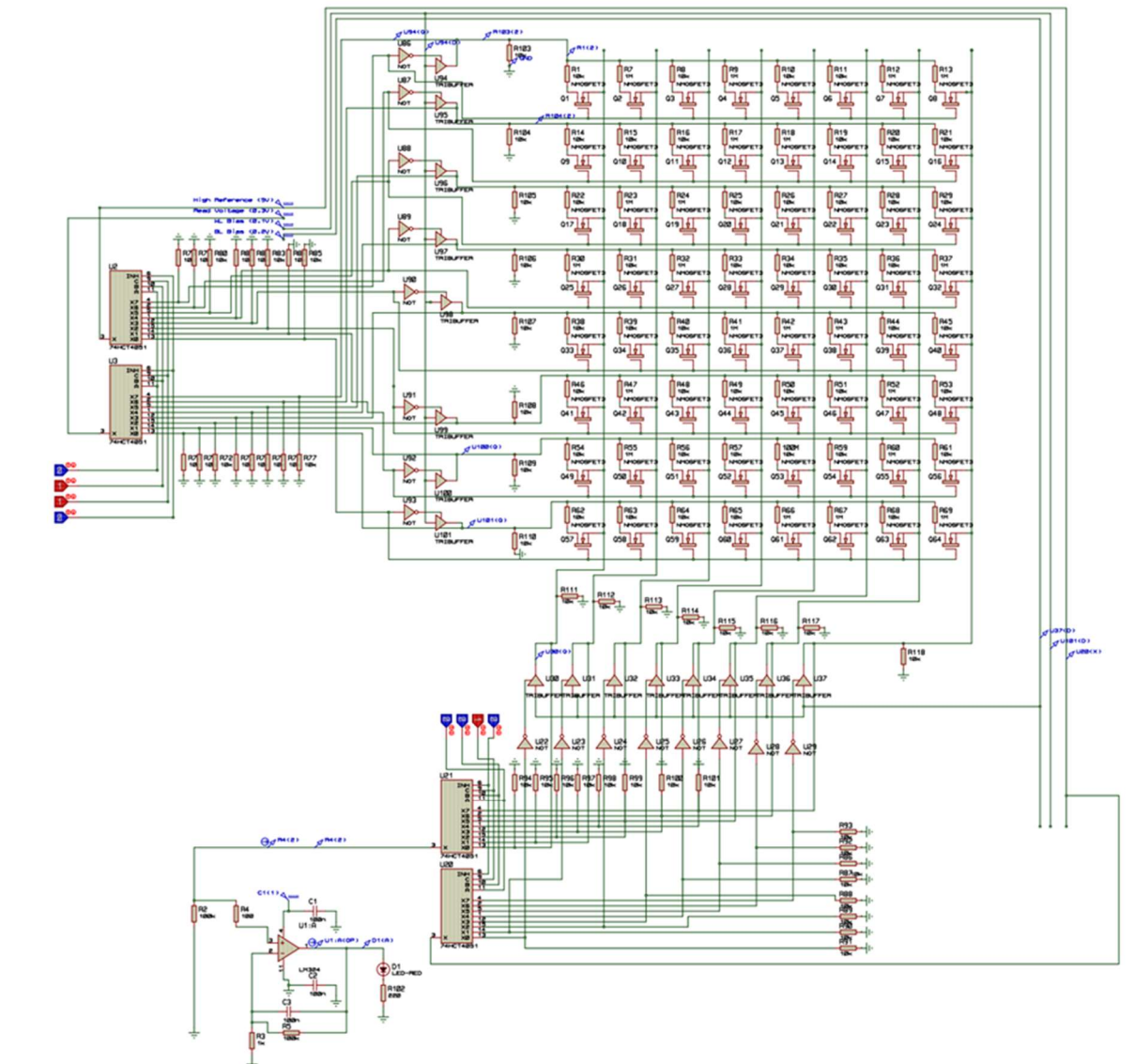
<https://1drv.ms/b/s!Ar4lztMoEvi9g8UPavblWf1wkv5gWw?e=DTj45a>



9 Ένθετο II: Σχέδιο Διάταξης crossbar με Transistors Επιλογής

Υψηλότερη ανάλυση διαθέσιμη στο

link: <https://1drv.ms/b/s!Ar4IztMoEvi9hJc4FdBL8K7ofmqOWw?e=AnQHD3>



10 Ένθετο III: Γενητρια Παλμών

```
/*
  This programme outputs a pulse-train of *uint16* voltage on DAC0 followed by a
  smaller read voltage
*/

int counter = 0;
int i = 0;
int var[4] = {1400,1400,1400,1400}; // any value 0-4095
int a=0;

// the setup routine runs once when you press reset:
void setup() {
  // initialize serial communication at 9600 bits per second:
  Serial.begin(9600);
  pinMode(9,OUTPUT);
}

// the loop routine runs over and over again forever:
void loop() {
  // read the input on analogue pin 0:
  // delay defines the pulse width
  analogReadResolution(12);
  analogWriteResolution(12);
  if(counter<1)
  {
    for(i=0;i<4;i++)
    {
      analogWrite(DAC0,var[i]);
      a = analogRead(A0);
      Serial.println(a);
      delay(50);
      analogWrite(DAC0,0);
      a = analogRead(A0);
      Serial.println(a);

      delay(50);
    }
    counter = counter+1;
  }
  analogWrite(DAC0,200);
  a = analogRead(A0);
  Serial.println(a);
  delay(100);
  analogWrite(DAC0,0);
  delay(100);
  a = analogRead(A0);
  Serial.println(a);
}
```

11 Ένθετο IV: Παραγωγή παλμοσειράς 6 θέσεων στο Keithley 4200

```
/* USRLIB MODULE INFORMATION
  Edited By: George "Blaze" Kleitsiotis
  MODULE NAME: chris_man_reramSweep_6 pulses
  MODULE RETURN TYPE: int
  NUMBER OF PARMS: 25
  ARGUMENTS:
    riseTime,      double,      Input, 1e-4, 20e-9,
    widthTime,     double,      Input, 1e-6, 20e-9,
    delayTime,     double,      Input, 1e-4, 20e-9,
    complianceCH,  int,         Input, 2,    1,    2
    resetV,        double,      Input, -2.5, -20, 20
    setV,          double,      Input, 2.5, -20, 20
    Irange,        double,      Input, 1e-2, 0.0, 1
    resetIcomp,    double,      Input, 0.0, -0.1, 0.1
    setIcomp,      double,      Input, 1e-3, -0.1, 0.1
    resTestV,      double,      Input, 2,    -10, 10
    takeRmeas,     int,         Input, 1,    0,    1
    useSmu,        int,         Input, 0,    0,    1
    numIter,       double,      Input, 1,    1,
    Vforce,        D_ARRAY_T,   Output,    ,    ,
    Vforce_size,   int,         Input, 1000, 10, 10000
    Imeas, D_ARRAY_T, Output,    ,    ,
    Imeas_size,    int,         Input, 1000, 10, 10000
    Time, D_ARRAY_T, Output,    ,    ,
    Time_size,     int,         Input, 1000, 10, 10000
    pts,           int *,      Output,    ,    ,
    resetResistance, double *,   Output,    ,    ,
    setResistance,  double *,   Output,    ,    ,
    initResistance, double *,   Output,    ,    ,
    loopNum,       int,         Input, 1,    1,
    stepIncrement, double,      Input, 0,    -1, 1

  INCLUDES:
#include "keithley.h"
#include "nvm.h"

  END USRLIB MODULE INFORMATION
*/
/* USRLIB MODULE HELP DESCRIPTION
DESCRIPTION:
-----
The routine is used to apply a custom train of up to 6 pulses, with a flat section
at the peak of each pulse. To test a reram device, choose appropriate values for
the two peaks, either positive or negative, and then set the timing you would like
to implement.

Note: In the test it is assumed that RPM1 is linked with SMU1 and RPM2 is linked
with SMU2. It is assumed that RPM1 (Channel 1) is connected to the side of the
dut with higher capacitance, such as chuck, substrate, which is usually a
*lower/bottom side*. RPM2 (Channel 2) should be connected on the opposite side,
which is usually its *top side* to minimize parasitic current transients. Channel
2 forces 0 V and is used to measure current. If useSmu = 1, SMU1 and SMU2 are
used for force voltage and measure current. If useSmu = 0, pulse
force/measurement is performed with PMU. Voltage bias polarities should be
applied, as if bias is applied from the top to simulate standard SMU/DC testing.
```

Polarities of the forced bias and measured current inverted in the code if you are using the PMU instead of SMU (useSmu = 0).

INPUTS:

riseTime (double) The time it takes for voltage to ramp to the final value. This only applies if using PMU (useSmu = 0) if using SMU there is a fixed ramp rate. Rise time should be longer than 5% of the widthTime.

widthTime (double) The time to wait at the top of the pulse at full voltage

delayTime (double) This is the delay time at the before and after each set or reset pulse. The time between a set and reset pulse will therefore be double this time.. This only applies when the PMU is being used instead of the SMU (useSmu = 0)

complianceCH (integer) On which SMU channel to enforce current compliance.

resetV (double) The peak voltage of the reset pulse. For ReRAM devices, this value should be negative

setV (double) The peak voltage of the set pulse. For ReRAM devices, this value should be positive

Irange (double) The range at which to measure current.

resetIcomp (double) Current compliance value to enforce during the reset pulse.

Please note that this variable is used both for PMU and SMU control during RESET. If it is set to 0, than no current limit is applied. If useSmu = 1, and sweep is performed with SMU, and current limit is 0, autorange is used. With current limit not zero and useSmu = 1, SMU is set in the fixed current range.

setIcomp (double) Current compliance value to enforce during the set pulse.

Please note that this variable is used both for PMU and SMU control during SET. If it is set to 0, than no current limit is applied. If useSmu = 1, and sweep is performed with SMU, and current limit is 0, autorange is used. With current limit not zero and useSmu = 1, SMU is set in the fixed current range.

Polarities for PMU compliances are selected, based on the a) inversion of the voltage on low side b) polarity of the intended voltage (setV and resetV - are biases of top/PRM2 to the bottom/RPM1), and c) used channel (2 or 1).

resTestV (double) The voltage at which to measure the resistance of the DUT. This should be much less than the set and reset voltages as to not set or reset the DUT.

takeRmeas (integer) Whether or not to take resistance measurements. 1 means yes, take resistance measurements, while 0 means no, do not take resistance measurements.

useSmu (integer) Whether or not to use a SMU to take DC measurements instead of using the PMU for pulsing measurements. A 1 means yes, use the SMU and a 0 means no, don't use the SMU, use the PMU.

numIter (integer) When in pulse mode, number of pulses to perform total. For instance, a value of 5 would execute 4 pulses without measuring

resistance. Then on the fifth pulse, resistance measurements would be taken. A value on 1 simply executes the sweep once.

Vforce_size (Integer)
Imeas_size (Integer)
Time_size (Integer) These three values should be the same and represent the number of items in the output arrays.

OUTPUTS:

Vforce (double) Array of forced voltages
Imeas (double) Array of measured currents
Time (double) Array of measured times
resetResistance (double) Resistance of DUT after the reset pulse
setResistance(double) Resistance of DUT after the set pulse
initResistance (double) Resistance of DUT before any pulse

Function returns error status, where:

- 1 : success [TEST_SUCCESS]
 - 10 : cannot initialize nvm structure, [NVM_INITIALIZE_ERROR]
 - 20 : array sizes are not the same [ARRAY_SIZES_DIFFERENT]
 - 30 : array size is too small, < 10, [ARRAY_SIZE_ERROR]
 - 200 : invalid number of points returned, [NUMBER_OF_POINTS_ERROR]
 - 210 : main, pulse_test function failed, [PULSE_TEST_FAILED]
 - 240 : used rate is less than minimum allowed, [MIN_RATE_ERROR]
 - 250 : dcSweep failed, [DC_SWEEP_FAILED]
 - 260 : riseTime is shorter than 5% of the width time [RISE_WIDTH_ERROR]
- END USRLIB MODULE HELP DESCRIPTION */

/* USRLIB MODULE PARAMETER LIST */

#include "keithley.h"

#include "nvm.h"

```
int chris_man_reramSweep( double riseTime, double widthTime, double delayTime, int
complianceCH, double resetV1, double setV1, double resetV2, double setV2, double
resetV3, double setV3, double Irange, double resetIcomp, double setIcomp, double
resTestV, int takeRmeas, int useSmu, double numIter, double *Vforce, int
Vforce_size, double *Imeas, int Imeas_size, double *Time, int Time_size, int *pts,
double *resetResistance, double *setResistance, double *initResistance, int
loopNum, double stepIncrement )
```

```
{
```

```
/* USRLIB MODULE CODE */
```

```
int stat;
```

```
double time_offset=0; // offset to be added to each element of timearr
```

```
//we add desc in comments
```

```
resetV1 *= -1.0;
```

```
setV1 *= -1.0;
```

```
int measCH = 2;
```

```
double initResetV1 = resetV1;
```

```
double initSetV1 = setV1;
```

```
resetV2 *= -1.0;
```

```
setV2 *= -1.0;
```

```
double initResetV2 = resetV2;
```

```
double initSetV2 = setV2;
```

```
resetV3 *= -1.0;
```

```

setV3 *= -1.0;
double initResetV3 = resetV3;
double initSetV3 = setV3;

double set = setV1;//Initialisation of "In-Loop" Parameters
double reset = resetV1;

double initReset = initResetV1;
double initSet = initSetV1;
for(int p = 0;p<loopNum;p++){

    if (p==1)
    {
        set = setV1;
        reset = resetV1;

        initReset = initResetV1;
        initSet = initSetV1;
    }
    else if (p == 2)
    {
        set = setV2;
        reset = resetV2;

        initReset = initResetV2;
        initSet = initSetV2;
    }

    else if (p == 3)
    {
        set = setV3;
        reset = resetV3;

        initReset = initResetV3;
        initSet = initSetV3;
    }

    }

    char mod[] = "chris_man_reramSweep";
stat = -1;
int j;
double resetDVForce[MAX_OUT_PTS];
double setDVForce[MAX_OUT_PTS];
double resetIMeas[MAX_OUT_PTS];
double setIMeas[MAX_OUT_PTS];
double timeArr[MAX_OUT_PTS];
double timeArr2[MAX_OUT_PTS];
double voffset;
double ioffset;
double initr;
double resetr;
double setr;
double ttime;
int numsetpts;
int numresetpts;
double toffset, dt;
double reset_sign, set_sign;

//Setup NVM structures

```

```

NVMS *nvm;
pmu *pmu1;
pmuch *ch1, *ch2;
seg *seg1, *seg2;
int maxpts = MAX_OUT_PTS, max_rate;

if(reset > 0)
    reset_sign = 1.0;
else
    reset_sign = -1.0;

if(set > 0)
    set_sign = 1.0;
else
    set_sign = -1.0;

//set increment
reset = initReset - (2*p)* stepIncrement;
set = initSet - (2*p+1)* stepIncrement;
//initialization:
*resetResistance = 0.0;
*setResistance = 0.0;
*initResistance = 0.0;

if(riseTime/widthTime < 0.05)
{
    nlog("%s: Rise time (%g) is too short relative to the width (%g) !\n",
        mod, riseTime, widthTime);
    stat = RISE_WIDTH_ERROR;
    goto RETURN;
}

//Check if output arrays are the same size
if(Vforce_size != Imeas_size || Vforce_size != Time_size)
{
    nlog("%s: Make sizes of all output arrays equal!\n", mod);
    stat = ARRAY_SIZES_DIFFERENT;
    goto RETURN;
}

//Reset max points according to output array sizes
//NOTE: max points are only half of the size of the array
if(maxpts > (Vforce_size/2)) maxpts = Vforce_size/2;

//make sure minimum width is 20 ns
if(widthTime < 2e-8) widthTime = 2e-8;

//Check if output arrays are too small
if(Vforce_size < 10)
{
    nlog("%s: Make sizes of all output arrays at least 20 or more!\n", mod);
    stat = ARRAY_SIZE_ERROR;
    goto RETURN;
}

//-----Completely branch off if we are doing more than one loop -----
-----
if(numIter > 1)

```

```

{
    if(useSmu==0)
    {

        //Initialize NVM using 1 PMU
        nvm = initNVMST(1);
        nvm->init = 0;

        //Check if initialization is successful
        if(NULL == nvm)
        {
            nlog("%s: Cannot initialize NVM structure!\n", mod);
            stat = NVM_INITIALIZE_ERROR; goto RETURN;
        }
        nlog("%s: starts\n", mod);

        //Find PMU
        pmu1 = getPMU(1);

        //Find Channels 1 and 2
        ch1 = getCH(pmu1,1);//ch1
        ch2 = getCH(pmu1,2);//ch2

        //Set measurement mode for both channels
        ch1->mode = PULSE_MEAS_NONE;
        ch2->mode = PULSE_MEAS_NONE;

        ch1->irange = Irange;
        ch2->irange = Irange;

        ch1->vrage = 10.0;
        ch2->vrage = 10.0;

        //Initialize segments
        seg1 = initSEGS(ch1,10); //ch1 8 segments
        seg2 = initSEGS(ch2,10); //ch2 8 segments

        if(fabs(resetIcomp) > 0)
        {
            if(complianceCH == 1)
                ch1->ilimit = reset_sign * fabs(resetIcomp);
            if(complianceCH == 2)
                ch2->ilimit = reset_sign * (-1.0) * fabs(resetIcomp);
        }

        //overwrite by set compliance if not ZERO
        if(fabs(setIcomp) > 0)
        {
            if(complianceCH == 1)
                ch1->ilimit = set_sign * fabs (setIcomp);
            if(complianceCH == 2)
                ch2->ilimit = set_sign * (-1.0) * fabs (setIcomp);
        }

        /*~~~~~Beginning of Setup for Ch1 Segments~~~~~*/
        seg1[0].segtime = delayTime;

        seg1[1].segtime = riseTime;
        seg1[1].stopv = reset;
    }
}

```



```

seg1[2].segtime = widthTime;
seg1[2].stopv = reset;

seg1[3].segtime = riseTime;
seg1[4].segtime = delayTime;
seg1[5].segtime = delayTime;

seg1[6].segtime = riseTime;
seg1[6].stopv = set;

seg1[7].segtime = widthTime;
seg1[7].stopv = set;

seg1[8].segtime = riseTime;
seg1[9].segtime = delayTime;

/*~~~~~End of Setup for Ch1 Segments~~~~~*/

//Set the total time and maximum rate
ttime = getTotalTime(seg1, 20);
nvm->total_time = ttime;

//max_rate = calc_rate (ttime, maxpts);
//pmu1->rate = max_rate;

//Setup Ch2 segment
copyTime(seg1, seg2, 20);

nlog("%s: total time:%g\n", mod, nvm->total_time);
nlog("%s: init: %g\n", mod, nvm->init);

pmu1->loops = numIter-1;

// DO THE TEST
stat = pulse_test();
if(0 > stat){stat = PULSE_TEST_FAILED; goto RETURN;};

devint();
}
else //do PULSE sweep //if(useSmu==1)
{

j = 0;
//while(j++ <= numIter -1)//PR47385
while(j++ <= numIter - 2)
{
//running Reset
stat = dcSweep("SMU1", "SMU2",
complianceCH, measCH,
Irange, resetIcomp,
0.002, widthTime,
(-1.0)*reset, (Vforce_size/2),
resetDVForce, (Vforce_size/2),
resetIMeas, (Vforce_size/2),
timeArr, (Vforce_size/2));

if(stat < 0)
{

```

```

        nlog("%s: error in dcSweep:%d\n", mod, stat);
        stat = DC_SWEEP_FAILED;
        goto RETURN;
    }

    //runnig set
    stat = dcSweep("SMU1", "SMU2",
        complianceCH, measCH,
        Irange, setIcomp,
        0.002, widthTime,
        (-1.0)*set, (Vforce_size/2),
        setDVForce, (Vforce_size/2),
        setIMeas, (Vforce_size/2),
        timeArr2, (Vforce_size/2));
    if(stat < 0)
    {
        nlog("%s: error in dcSweep:%d\n", mod, stat);
        stat = DC_SWEEP_FAILED;
        goto RETURN;
    }
}

}
}

//-----MEASURE PART-----
-----

// Get an initial Resistance measurement
if(takeRmeas == 1)
{
    stat = getRes2( "SMU2", "SMU1", resTestV, &initr );
    *initResistance = initr;
    PostDataDouble("initResistance", initr);
}

if(useSmu==0)
{
    //Initialize NVM using 1 PMU
    nvm = initNVMST(1);

    nvm->init = 0;

    //Check if initialization is successful
    if(NULL == nvm)
    {
        nlog("%s: Cannot initialize NVM structure!\n", mod);
        stat = NVM_INITIALIZE_ERROR; goto RETURN;
    }
    nlog("%s: starts\n", mod);

    //Find PMU
    pmu1 = getPMU(1);

    //Find Channels 1 and 2
    ch1 = getCH(pmu1,1);//ch1
    ch2 = getCH(pmu1,2);//ch2

```

```

//Set measurement mode for both channels
ch1->mode = PULSE_MEAS_WFM_PER;
ch2->mode = PULSE_MEAS_WFM_PER;

ch1->irange = Irange;
ch2->irange = Irange;

ch1->vrange = 10.0;
ch2->vrange = 10.0;

// // DO THE TEST

//Initialize segments
seg1 = initSEGS(ch1,5); //ch1 5 segments
seg2 = initSEGS(ch2,5); //ch2 5 segments

if(fabs(resetIcomp) > 0)
{
    if(complianceCH == 1)
        ch1->ilimit = reset_sign * fabs(resetIcomp);
    if(complianceCH == 2)
        ch2->ilimit = reset_sign * (-1.0) * fabs(resetIcomp);
}

/*~~~~~Beginning of Setup for Ch1 Segments~~~~~*/
seg1[0].segtime = delayTime;

seg1[1].segtime = riseTime;
seg1[1].stopv = reset;

seg1[2].segtime = widthTime;
seg1[2].stopv = reset;

seg1[3].segtime = riseTime;
seg1[4].segtime = delayTime;

/*~~~~~End of Setup for Ch1 Segments~~~~~*/

//Set the total time and maximum rate
ttime = getTotalTime(seg1, 4);
nvm->total_time = ttime;

max_rate = calc_rate (ttime, maxpts);
if(max_rate < NVM_MIN_RATE)
{
    nlog("%s: calculated rate (%d) is smaller than minimum allowed rate
(%d)\n",
        mod, max_rate, NVM_MIN_RATE);
    stat = MIN_RATE_ERROR;
    goto RETURN;
}
pmu1->rate = max_rate;

//Setup Ch2 segment
copyTime(seg1, seg2, 5);

nlog("%s: total time:%g\n", mod, nvm->total_time);
nlog("%s: init: %g\n", mod, nvm->init);

```

```

stat = pulse_test();
if(0 > stat){stat = PULSE_TEST_FAILED; goto RETURN;};

//Determine the number of points from pulse_test()
*pts = ch1->out_pts;
PostDataInt("pts", ch1->out_pts);
nlog("%s: number of points:%d\n", mod, ch1->out_pts);
if(*pts < 1)
{
    nlog("%s: Wrong number of points! %d\n", mod, *pts);
    stat = NUMBER_OF_POINTS_ERROR;
    goto RETURN;
}

//Get current/voltage offsets
if(2 == measCH)
    ioffset = getPulseI(ch2, 0);
else
    ioffset = getPulseI(ch1, 0);

voffset = getPulseV(ch2, 0) - getPulseV(ch1, 0);

nlog("%s: Current Offset is: %g Voltage offset: %g\n", mod,
    ioffset, voffset);

//Correct data with voltage/current offsets,
for(j = 0; j < ch1->out_pts; j++)
{
    if(2 == measCH)
    {
        ch2->pulseI[j] -= ioffset;
        resetIMeas[j] = ch2->pulseI[j];
    }
    else
    {
        ch1->pulseI[j] -= ioffset;
        resetIMeas[j] = ch1->pulseI[j];
    }

    resetDVForce[j] = ch2->pulseV[j] - ch1->pulseV[j] - voffset;

    // Get the time
    timeArr[j] = ch1->pulseT[j];
}
numresetpts = ch1->out_pts;

devint();
}
else //do PULSE sweep //if(useSmu==1)
{
    // DC option
    stat = dcSweep("SMU1", "SMU2",
        complianceCH, measCH,
        Irange, resetIcomp,
        0.002, widthTime,
        (-1.0)*reset, (Vforce_size/2),
        resetDVForce, (Vforce_size/2),
        resetIMeas, (Vforce_size/2),

```

```

        timeArr, (Vforce_size/2));
if(stat < 0)
{
    nlog("%s: error in dcSweep:%d\n", mod, stat);
    stat = DC_SWEEP_FAILED;
    goto RETURN;
}

    numresetpts = Vforce_size/2;
    // set up time array
    // toffset = timeArr[Vforce_size/2-2]; // edit
    // for(j = (Vforce_size/2); j<Vforce_size; j++){
    // timeArr[j] = timeArr2[j-(Vforce_size/2)] + toffset;
    //}
}

//Determine the resistance of the device
if(takeRmeas == 1)
{
    stat = getRes2( "SMU2", "SMU1", resTestV, &resetr );
    *resetResistance = resetr;
    PostDataDouble("resetResistance", resetr);
}

if(useSmu==0)
{
    // SET PULSE -----
    // force init
    nvm->init = 0;

    if(fabs(setIcomp) > 0)
    {
        if(complianceCH == 1)
            ch1->ilimit = set_sign * fabs(setIcomp);
        if(complianceCH == 2)
            ch2->ilimit = set_sign * (-1.0) * fabs(setIcomp);
    }

    //set Set Voltages
    seg1[1].stopv = set;
    seg1[2].stopv = set;

    stat = pulse_test();
    if(0 > stat){stat = PULSE_TEST_FAILED; goto RETURN;};

    //Determine the number of points from pulse_test()
    *pts = ch1->out_pts;
    PostDataInt("pts", ch1->out_pts);
    nlog("%s: number of points:%d\n", mod, ch1->out_pts);
    if(*pts < 1)
    {
        nlog("%s: Wrong number of points! %d\n", mod, *pts);
        stat = NUMBER_OF_POINTS_ERROR;
        goto RETURN;
    }

    //Get current/voltage offsets
    if(2 == measCH)

```

```

        ioffset = getPulseI(ch2, 0);
else
        ioffset = getPulseI(ch1, 0);

voffset = getPulseV(ch2, 0) - getPulseV(ch1, 0);

nlog("%s: Current Offset is: %g Voltage offset: %g\n", mod,
        ioffset, voffset);

//Correct data with voltage/current offsets,
dt = ch1->pulseT[1] - ch1->pulseT[0]; //one extra time step
for(j = 0; j < ch1->out_pts; j++)
{
        if(2 == measCH)
        {
                ch2->pulseI[j] -= ioffset;
                setIMeas[j] = ch2->pulseI[j];
        }
        else
        {
                ch1->pulseI[j] -= ioffset;
                setIMeas[j] = ch1->pulseI[j];
        }

        //disable voltage offset:
        //voffset = 0.0;

        setDVForce[j] = ch2->pulseV[j] - ch1->pulseV[j] - voffset;
        // Do time
        //timeArr[j + ch1->out_pts] = ch1->pulseT[j] + timeArr[ch1->out_pts-
1];
        timeArr[j + ch1->out_pts] = ch1->pulseT[j] + timeArr[ch1->out_pts-1] +
dt;
}
numsetpts = ch1->out_pts;

devint();

}
else//use_smu = 1
{
        // DC option
        stat = dcSweep("SMU1", "SMU2",
                complianceCH, measCH,
                Irange, setIcomp,
                0.002, widthTime,
                (-1.0)*set, (Vforce_size/2),
                setDVForce, (Vforce_size/2),
                setIMeas, (Vforce_size/2),
                timeArr2, (Vforce_size/2));

        if(stat < 0)
        {
                nlog("%s: error in dcSweep:%d\n", mod, stat);
                stat = DC_SWEEP_FAILED;
                goto RETURN;
        }
}

```

```

    numsetpts = Vforce_size/2;
    // set up the time array
    toffset = timeArr[Vforce_size/2-2];

    for(j = (Vforce_size/2); j<Vforce_size; j++){
        timeArr[j] = timeArr2[j-(Vforce_size/2)] + toffset ;
    }

}

if(takeRmeas == 1)
{
    stat = getRes2( "SMU2", "SMU1", resTestV, &setr );
    *setResistance = setr;
    PostDataDouble("resetResistance", setr);
}

//Print data to the appropriate buffers
PostDataDoubleBuffer("Vforce", resetDVForce, numresetpts);
PostDataDoubleBuffer("Imeas", resetIMeas, numresetpts);
// We add time_offset to each element of array before printing.This ensues
linear time results.
for(int k=0; k<numresetpts*2; ++k){
    timeArr[k] += time_offset;
}
time_offset= timeArr[numresetpts*2-1] + timeArr[numresetpts*2-1] -
timeArr[numresetpts*2-2];

PostDataDoubleBuffer("Time", timeArr, (numresetpts*2));

PostDataDoubleBuffer("Vforce", setDVForce, numresetpts);
PostDataDoubleBuffer("Imeas", setIMeas, numresetpts);

//Sweep Completed Successfully
stat = TEST_SUCCESS;

RETURN:
//printNVMST();
nlog("%s: exiting with status:%d\n", mod, stat);
}

return stat;
/* USRLIB MODULE END */
}          /* End routine */

```


Αναφορές

- [1] “What is Moore’s Law? | Is Moore’s Law Dead? | Synopsys.” <https://www.synopsys.com/glossary/what-is-moores-law.html> (accessed May 19, 2022).
- [2] “Computer architecture - Wikipedia.” https://en.wikipedia.org/wiki/Computer_architecture (accessed May 19, 2022).
- [3] “Von Neumann architecture - Wikipedia.” https://en.wikipedia.org/wiki/Von_Neumann_architecture (accessed May 19, 2022).
- [4] Z. Y. He *et al.*, “CMOS back-end compatible memristors for: In situ digital and neuromorphic computing applications,” *Materials Horizons*, vol. 8, no. 12, pp. 3345–3355, Dec. 2021, doi: 10.1039/d1mh01257f.
- [5] Π. Μπούσουλας, «Νανοηλεκτρονικές Διατάξεις Μνήμης», Διδακτορική Διατριβή, Σχολή Εφαρμοσμένων Μαθηματικών και Φυσικών Επιστημών, Εθνικό Μετσόβιο Πολυτεχνείο, Αθήνα 2017
- [6] P. Bousoulas *et al.*, “Investigating the origins of ultra-short relaxation times of silver filaments in forming-free SiO₂-based conductive bridge memristors,” *Nanotechnology*, vol. 31, no. 45, p. 454002, Aug. 2020, doi: 10.1088/1361-6528/ABA3A1.
- [7] P. Bousoulas, M. Panagopoulou, N. Boukos, and D. Tsoukalas, “Emulating artificial neuron and synaptic properties with SiO₂-based memristive devices by tuning threshold and bipolar switching effects,” *Journal of Physics D: Applied Physics*, vol. 54, no. 22, p. 225303, Mar. 2021, doi: 10.1088/1361-6463/ABEA3B.
- [8] D. Ielmini, “Resistive switching memories based on metal oxides: Mechanisms, reliability and scaling,” *Semiconductor Science and Technology*, vol. 31, no. 6, May 2016, doi: 10.1088/0268-1242/31/6/063002.
- [9] M. A. Zidan, H. A. H. Fahmy, M. M. Hussain, and K. N. Salama, “Memristor-based memory: The sneak paths problem and solutions,” *Microelectronics Journal*, vol. 44, no. 2, pp. 176–183, Feb. 2013, doi: 10.1016/J.MEJO.2012.10.001.
- [10] J. Zhou, K. H. Kim, and W. Lu, “Crossbar RRAM arrays: Selector device requirements during read operation,” *IEEE Transactions on Electron Devices*, vol. 61, no. 5, pp. 1369–1376, 2014, doi: 10.1109/TED.2014.2310200.
- [11] Y. Gao, O. Kavehei, S. F. Al-Sarawi, D. C. Ranasinghe, and D. Abbott, “Read operation performance of large selectorless cross-point array with self-rectifying memristive device,” *Integration*, vol. 54, pp. 56–64, Jun. 2016, doi: 10.1016/J.VLSI.2016.02.002.
- [12] S. Kim, J. Zhou, and W. D. Lu, “Crossbar RRAM arrays: Selector device requirements during write operation,” *IEEE Transactions on Electron Devices*, vol. 61, no. 8, pp. 2820–2826, 2014, doi: 10.1109/TED.2014.2327514.
- [13] “CD4051B data sheet, product information and support | TI.com.” <https://www.ti.com/product/CD4051B> (accessed Jun. 03, 2022).

- [14] “Signal Monitoring and Parameter Tuning Over XCP on Serial - MATLAB & Simulink.”
<https://www.mathworks.com/help/supportpkg/texasinstrumentsc2000/ug/signal-monitoring-paramter-tune-xcp-on-serial.html> (accessed Jun. 07, 2022).
- [15] A. Mehonic and A. J. Kenyon, “Brain-inspired computing needs a master plan,” *Nature* 2022 604:7905, vol. 604, no. 7905, pp. 255–260, Apr. 2022, doi: 10.1038/s41586-021-04362-w.
- [16] Lauralee. Sherwood, “Introduction to human physiology,” 2013.
- [17] X. Li *et al.*, “Power-efficient neural network with artificial dendrites,” *Nature Nanotechnology* 2020 15:9, vol. 15, no. 9, pp. 776–782, Jun. 2020, doi: 10.1038/s41565-020-0722-5.
- [18] “MNIST database - Wikipedia.” https://en.wikipedia.org/wiki/MNIST_database (accessed Jun. 14, 2022).
- [19] C. Tsiouostas, P. Bousoulas, G. Kleitsiotis, and D. Tsoukalas, “Pulse-stream impact on recognition accuracy of reservoir computing from SiO₂-based low power memory devices”.