



**ΠΑΝΕΠΙΣΤΗΜΙΟ ΘΕΣΣΑΛΙΑΣ**  
**ΣΧΟΛΗ ΘΕΤΙΚΩΝ ΕΠΙΣΤΗΜΩΝ**  
**ΤΜΗΜΑ ΠΛΗΡΟΦΟΡΙΚΗΣ ΜΕ ΕΦΑΡΜΟΓΕΣ**  
**ΣΤΗ ΒΙΟΙΑΤΡΙΚΗ**

**«Ανάπτυξη νευρομορφικού κυκλώματος σε  
αναδιατασσύμενη τεχνολογία για προσομοίωση συνάψεων»**

**Κομπογιάννης Χρήστος**

**ΠΤΥΧΙΑΚΗ ΕΡΓΑΣΙΑ**  
**Υπεύθυνος**  
**Κακαρούντας Αθανάσιος**

**Λαμία, 2021**





**ΠΑΝΕΠΙΣΤΗΜΙΟ ΘΕΣΣΑΛΙΑΣ**  
**ΣΧΟΛΗ ΘΕΤΙΚΩΝ ΕΠΙΣΤΗΜΩΝ**  
**ΤΜΗΜΑ ΠΛΗΡΟΦΟΡΙΚΗΣ ΜΕ ΕΦΑΡΜΟΓΕΣ ΣΤΗ**  
**ΒΙΟΙΑΤΡΙΚΗ**

**«Ανάπτυξη νευρομορφικού κυκλώματος σε  
αναδιατασσομένη τεχνολογία για προσομοίωση συνάψεων»**

**Κομπογιάννης Χρήστος**

**ΠΤΥΧΙΑΚΗ ΕΡΓΑΣΙΑ**  
**Υπεύθυνος**  
**Κακαρούντας Αθανάσιος**

**Λαμία, 2021**

Με ατομική μου ευθύνη και γνωρίζοντας τις κυρώσεις (1), που προβλέπονται από της διατάξεις της παρ. 6 του άρθρου 22 του Ν. 1599/1986, δηλώνω ότι:

1. Δεν παραθέτω κομμάτια βιβλίων ή άρθρων ή εργασιών άλλων αυτολεξεί χωρίς να τα περικλείω σε εισαγωγικά και χωρίς να αναφέρω το συγγραφέα, τη χρονολογία, τη σελίδα. Η αυτολεξεί παράθεση χωρίς εισαγωγικά χωρίς αναφορά στην πηγή, είναι λογοκλοπή. Πέραν της αυτολεξεί παράθεσης, λογοκλοπή θεωρείται και η παράθεση εδαφίων από έργα άλλων, συμπεριλαμβανομένων και έργων συμφοιτητών μου, καθώς και η παράθεση στοιχείων που άλλοι συνέλεξαν ή επεξεργάστηκαν, χωρίς αναφορά στην πηγή. Αναφέρω πάντοτε με πληρότητα την πηγή κάτω από τον πίνακα ή σχέδιο, όπως στα παραθέματα.
2. Δέχομαι ότι η αυτολεξεί παράθεση χωρίς εισαγωγικά, ακόμα κι αν συνοδεύεται από αναφορά στην πηγή σε κάποιο άλλο σημείο του κειμένου ή στο τέλος του, είναι αντιγραφή. Η αναφορά στην πηγή στο τέλος π.χ. μιας παραγράφου ή μιας σελίδας, δεν δικαιολογεί συρραφή εδαφίων έργου άλλου συγγραφέα, έστω και παραφρασμένων, και παρουσιάσή τους ως δική μου εργασία.
3. Δέχομαι ότι υπάρχει επίσης περιορισμός στο μέγεθος και στη συχνότητα των παραθεμάτων που μπορώ να εντάξω στην εργασία μου εντός εισαγωγικών. Κάθε μεγάλο παράθεμα (π.χ. σε πίνακα ή πλαίσιο, κλπ), προϋποθέτει ειδικές ρυθμίσεις, και όταν δημοσιεύεται προϋποθέτει την άδεια του συγγραφέα ή του εκδότη. Το ίδιο και οι πίνακες και τα σχέδια
4. Δέχομαι όλες τις συνέπειες σε περίπτωση λογοκλοπής ή αντιγραφής.

Ημερομηνία: .....

Ο δηλών

Κομπογιάννης Χρήστος

(1) «Όποιος εν γνώσει του δηλώνει ψευδή γεγονότα ή αρνείται ή αποκρύπτει τα αληθινά με έγγραφη υπεύθυνη δήλωση του άρθρου 8παρ. 4 Ν. 1599/1986 τιμωρείται με φυλάκιση τουλάχιστον τριών μηνών. Εάν ο υπαίτιος αυτών των πράξεων σκόπευε να προσπορίσει στον εαυτόν του ή σε άλλον περιουσιακό όφελος βλάπτοντας τρίτον ή σκόπευε να βλάψει άλλον, τιμωρείται με κάθειρξη μέχρι 10 ετών.

**«Ανάπτυξη νευρομορφικού κυκλώματος σε  
αναδιατασσομένη τεχνολογία για προσομοίωση συνάψεων»**

Κομπογιάννης Χρήστος

**Τριμελής Επιτροπή:**

Αθανάσιος Κακαρούντας, Αναπληρωτής Καθηγητής (επιβλέπων)

Σωτήριος Τασουλής, Επίκουρος Καθηγητής

Χαράλαμπος Καρανίκας, Λέκτορας



## Περίληψη

Τα νευρομορφικά κυκλώματα αποτελούν έναν τομέα ο οποίος έχει λάβει μεγάλη προσοχή τα τελευταία χρόνια σε ένα ευρύ φάσμα επιστημονικών κλάδων. Στην παρούσα πτυχιακή εργασία παρουσιάζονται αρχικά κάποιες βασικές έννοιες των βιολογικών νευρώνων και συνάψεων καθώς και τα θεμελιώδη χαρακτηριστικά των νευρομορφικών κυκλωμάτων. Στη συνέχεια αναλύεται μια πρωτότυπη υλοποίηση νευρομορφικής σύναψης σε πλακέτα ανάπτυξης υλικού FPGA, η οποία είναι συμβατή με κυκλώματα νευρώνων που βασίζονται στο μαθηματικό μοντέλο του Izhikevich. Η υλοποίηση αποτελείται από δύο πυρήνες υπολογισμού, ο πρώτος με πέντε στάδια διασωλήνωσης και ο δεύτερος με επτά. Σε σχέση με παρόμοιες νευρομορφικές συνάψεις, η συγκεκριμένη διατηρεί χαμηλή πολυπλοκότητα και υπολογίζει γρήγορα και αξιόπιστα τα συναπτικά ρεύματα τα οποία αποτελούν απαραίτητη είσοδο των συνδεδεμένων νευρώνων. Συγκεκριμένα όσον αφορά την ταχύτητα, η παράλληλη εκτέλεση των απαιτούμενων πράξεων σημαίνει ότι η σύναψη μπορεί να λειτουργήσει σε πραγματικό χρόνο.

**Λέξεις κλειδιά:** Νευρομορφικά κυκλώματα, τεχνητός νευρώνας, τεχνητή σύναψη, μοντέλο Izhikevich

## **Abstract**

Neuromorphic circuits comprise a topic that has received widespread attention across various scientific fields. In the introduction of this thesis some basic terminology concerning biological neurons and synapses, as well as the fundamental theory behind neuromorphic circuits, is presented. Following that, an original implementation of a neuromorphic synapse utilising FPGA technology is proposed. This circuit is designed with the goal of being compatible with neuromorphic neurons based on the mathematical Izhikevich neuron model. The implementation is comprised of two computation cores, the first with five pipeline stages and the second with seven. Compared to similar neuromorphic synapses, this one retains low complexity and can calculate the needed synaptic currents of the connected neurons quickly and reliably. The speed afforded by the parallel execution of instructions means that the system can function in real time.

**Key words:** Neuromorphic circuits, artificial neuron, artificial synapse, Izhikevich model



## Ευχαριστίες

Αρχικά θα ήθελα να ευχαριστήσω τον επιβλέποντα καθηγητή μου κ. Αθανάσιο Κακαρούντα. Η έμπνευση και καθοδήγησή του ήταν ανεκτίμητες καθ' όλη τη διάρκεια εκπόνησης της πτυχιακής μου.

Επίσης θα ήθελα να ευχαριστήσω από καρδιάς την υποψήφια διδάκτορα του τμήματος μας Μαρία Σαπουνάκη, η οποία ανέλαβε εκτός των άλλων και το ρόλο ελέγχου αυτού του κειμένου. Χωρίς την τεχνική της γνώση πάνω στο θέμα των νευρομορφικών κυκλωμάτων, την προθυμία της να με βοηθήσει σε όποιο πρόβλημα και αν προέκυπτε και τη συνεχή ύπαρξη της ως παράδειγμα προς μίμηση, δε θα ήταν δυνατή η ολοκλήρωση της παρούσας εργασίας.

Τέλος, θα ήθελα να ευχαριστήσω τους γονείς, την αδερφή και τους φίλους μου που στάθηκαν υπομονετικά δίπλα μου από την αρχή μέχρι το τέλος της διαδικασίας. Αποτελούν πηγή δύναμης για κάθε μου πράξη και δε θα βρισκόμουν εδώ που είμαι χωρίς αυτούς.

## Περιεχόμενα

Περίληψη .....	i
Abstract .....	ii
Ευχαριστίες .....	iii
1 Εισαγωγή.....	1
2 Νευρώνες, Συνάψεις και Δίκτυα.....	3
2.1 Εισαγωγή.....	3
2.2 Ο Βιολογικός Νευρώνας.....	3
2.2.1 Δομή.....	3
2.2.2 Κατηγοριοποιήσεις Βιολογικών Νευρώνων .....	4
2.3 Η Βιολογική Σύναψη .....	5
2.4 Ο Τεχνητός Νευρώνας.....	7
2.4.1 Εισαγωγή.....	7
2.4.2 McCulloch-Pitts .....	8
2.4.3 Hodgkin-Huxley .....	9
2.4.4 FitzHugh-Nagumo .....	11
2.4.5 Morris-Lecar .....	11
2.4.6 Hindmarsh-Rose .....	13
2.4.7 Integrate and Fire (I&F).....	13
2.4.8 Leaky Integrate and Fire (LIF) .....	14
2.4.9 Resonate and Fire.....	15
2.4.10 Εκθετικό Integrate and Fire με Προσαρμογή.....	16
2.4.11 Izhikevich .....	16
2.5 Η Τεχνητή Σύναψη .....	16
2.5.1 Εισαγωγή.....	16
2.5.2 Τεχνητές Συνάψεις σε Κλασικά Νευρωνικά Δίκτυα .....	17
2.5.3 Τεχνητές Συνάψεις σε Νευρομορφικά Κυκλώματα .....	20
2.6 Μοντέλα Νευρωνικών Δικτύων.....	20
2.6.1 Κεντρική Γεννήτρια Μοτίβων (Central Pattern Generator - CPG) .....	21
2.6.2 Νευρωνικά Δίκτυα Πρόσθιας Διάδοσης.....	21
2.6.3 Αναδρομικά Νευρωνικά Δίκτυα .....	22

2.6.4	Επαναλαμβανόμενα Νευρωνικά Δίκτυα.....	23
2.6.5	Συνελικτικά Νευρωνικά Δίκτυα .....	24
3	Νευρομορφικά Κυκλώματα σε Αναδιατασόμενη Τεχνολογία .....	26
3.1	Εισαγωγή.....	26
3.2	Νευρομορφικές Υλοποιήσεις σε Υλικό .....	27
3.2.1	Αναλογικά Κυκλώματα .....	28
3.2.2	Ψηφιακά Κυκλώματα.....	29
3.2.3	Υβριδικά Κυκλώματα .....	29
3.3	Σχετικά Projects .....	29
3.3.1	Human Brain Project.....	30
3.3.2	BrainScaleS.....	30
3.3.3	SpiNNaker.....	31
4	Μελέτη Περίπτωσης: Μοντέλο Izhikevich και Παράγωγα .....	32
4.1	Εισαγωγή.....	32
4.2	Το αρχικό μοντέλο .....	32
4.3	Άλλες Υλοποιήσεις .....	33
5	Η Προτεινόμενη Συναπτική Υλοποίηση.....	42
5.1	Εισαγωγή.....	42
5.2	Περιγραφή και Αρχιτεκτονική της Υλοποίησης.....	42
5.3	Το Κύκλωμα .....	43
6	Αποτελέσματα Υλοποίησης.....	46
6.1	Έλεγχος Αποτελεσμάτων.....	46
6.2	Πόροι Υλοποίησης.....	46
7	Συμπεράσματα .....	48
7.1	Εισαγωγή.....	48
7.2	Πιθανές Προεκτάσεις της Υλοποίησης.....	48
	ΠΑΡΑΡΤΗΜΑ Α.....	50
1.	Πλατφόρμα Υλικού.....	50
2.	Κώδικας .....	52
3.	GTKWave.....	53
	ΠΑΡΑΡΤΗΜΑ Β .....	54

1. Κώδικας 1 (Design) .....	54
2. Κώδικας 2 (Testbench) .....	77
Βιβλιογραφία .....	81

## Ευρετήριο Εικόνων

Εικόνα 1. Διάγραμμα ενός τυπικού νευρώνα και των συστατικών μερών του. ....	4
Εικόνα 2. Κατηγοριοποίηση των νευρώνων σε αισθητικούς, ενδιάμεσους και κινητήριους ανάλογα με τη λειτουργία τους.....	4
Εικόνα 3. Κατηγοριοποίηση των νευρώνων σε μονοπολικούς, διπολικούς, ψευδομονοπολικούς και πολυπολικούς ανάλογα με τη δομή τους.....	5
Εικόνα 4. Διάγραμμα χημικής σύναψης. ....	6
Εικόνα 5. Διάγραμμα ηλεκτρικής σύναψης.....	7
Εικόνα 6. Διάγραμμα τεχνητού νευρώνα.....	8
Εικόνα 7. Το ηλεκτρικό κύκλωμα του μοντέλου Hodgkin-Huxley.....	10
Εικόνα 8. Το αντίστοιχο κύκλωμα του μοντέλου FitzHugh-Nagumo.....	11
Εικόνα 9. Το αντίστοιχο κύκλωμα του μοντέλου Integrate and Fire. C είναι η μεμβρανική χωρητικότητα, R η μεμβρανική αντίσταση, V η μεμβρανική τάση, $V_{rest}$ η μεμβρανική τάση ηρεμίας, I η ένταση του ρεύματος εισόδου.....	14
Εικόνα 10. Σύγκριση της απόκρισης ενός Integrate and Fire νευρώνα και ενός Resonate and Fire νευρώνα.....	15
Εικόνα 11. Η συνάρτηση STDP μίας σύναψης. Αν η προσυναπτική ακίδα φτάσει πριν τη μετασυναπτική ( $\Delta t > 0$ ), παρατηρείται μακροπρόθεσμη ενδυνάμωση. Αλλιώς ( $\Delta t < 0$ ) παρατηρείται μακροπρόθεσμη ύφεση.....	19
Εικόνα 12. Η δομή ενός perceptron (αριστερά) και ενός MLP (δεξιά).....	22
Εικόνα 13. Εικόνα 14. Αναπαράσταση Αναδρομικού Νευρωνικού Δικτύου στον τομέα της επεξεργασίας φυσικής γλώσσας. Ξεκινώντας από κάτω προς τα πάνω, οι	

είσοδοι (εδώ ρήματα V και φράσεις ουσιαστικών NP) συνδυάζονται για να προκύψει στην έξοδο η πρόταση S. ....	22
Εικόνα 15. Ένα μονοεπίπεδο δίκτυο Hopfield. ....	24
Εικόνα 16. Μάθηση χαρακτηριστικών και ταξινόμηση χειρόγραφων αριθμών με τη χρήση συνελκτικού νευρωνικού δικτύου.....	25
Εικόνα 17. Χαρακτηριστικά των πλακέτων Xilinx FPGA από το 1988 μέχρι το 2015. Η χωρητικότητα είναι ο αριθμός λογικών μπλοκ ανά πλακέτα, η ταχύτητα αφορά την εκτέλεση της ίδιας συνάρτησης σε κάθε πλακέτα, το κόστος και η ηλεκτρική κατανάλωση δίνονται ανά λογικό μπλοκ πολ/μένες με το 10000. ....	27
Εικόνα 18. Τα στάδια διασωλήνωσης στους νευρώνες του δικτύου των Cassidy και Ανδρέου. ....	35
Εικόνα 19. Η αρχιτεκτονική του LIF δικτύου των Cassidy, Denham κ.α. στο οποίο βασίστηκε το Izhikevich δίκτυο των Cassidy και Ανδρέου. Με γκρι επισημαίνεται το συναπτικό κομμάτι της αρχιτεκτονικής. ....	36
Εικόνα 20. Τα 7 στάδια διασωλήνωσης υπολογισμού των u και v στο paper του Ambroise και συνεργατών. ....	37
Εικόνα 21. Τα δύο pipelines για τον υπολογισμό των διεγερτικών και ανασταλτικών ρευμάτων και της εκθετικής μείωσης τους στο δίκτυο του Ambroise και συνεργατών. ....	38
Εικόνα 22. Τα στάδια διασωλήνωσης στο νευρώνα του paper των Bandeira, Costa κ.α. ....	39
Εικόνα 23. Τα στάδια διασωλήνωσης στο νευρώνα του paper των Haghiri, Zahedi κ.α. ....	40
Εικόνα 24. Το συναπτικό κύκλωμα του paper των Haghiri, Zahedi κ.α. ....	41
Εικόνα 25. Η προτεινόμενη αρχιτεκτονική. ....	43
Εικόνα 26. Η προτεινόμενη σχεδίαση του πυρήνα υπολογισμού ρευμάτων.....	44
Εικόνα 27. Η προτεινόμενη σχεδίαση του πυρήνα υπολογισμού εκθετικής μείωσης ρευμάτων.....	45
Εικόνα 28. Τα αποτελέσματα της προσομοίωσης όσων αφορά τα συναπτικά ρεύματα. ....	46
Εικόνα 29. Αναφορά της κατανάλωσης ισχύος της υλοποίησης στο περιβάλλον Vivado.....	47

Εικόνα 30. Η αρχιτεκτονική του spiking νευρωνικού δικτύου του paper του Ambroise και συνεργατών.....	49
Εικόνα 31. Το μπλοκ διατασσόμενης λογικής (CLB) ενός Xilinx 7 FPGA.....	51
Εικόνα 32. Πλακέτα ανάπτυξης του FPGA Xilinx Artix-7.....	52

## 1 Εισαγωγή

Ο εγκέφαλος είναι ένα όργανο του κεντρικού νευρικού συστήματος (ΚΝΣ). Είναι υπεύθυνος για τον συντονισμό όλων των διεργασιών που λαμβάνουν χώρα στο σώμα. Ως εκ τούτου, βλάβες σε αυτόν ή σε άλλα όργανα του νευρικού συστήματος μπορεί να έχουν τεράστιες επιδράσεις στην ποιότητα ζωής των ασθενών. Ενδεικτικά αναφέρεται ότι στην Ευρωπαϊκή Ένωση, περίπου 165 εκατομμύρια άτομα πάσχουν από κάποια νευρολογική πάθηση [1].

Ο εγκέφαλος αποτελείται από δύο είδη κυττάρων: τα νευρογλοιακά κύτταρα και τους νευρώνες. Πάρα το γεγονός ότι η πλειοψηφία των εγκεφαλικών κυττάρων είναι τα νευρογλοιακά, ο ρόλος τους είναι επικουρικός. Οι νευρώνες αποτελούν τη βασική λειτουργική μονάδα. Ο ανθρώπινος εγκέφαλος απαρτίζεται από περίπου 100 δισεκατομμύρια νευρώνες, καθένας εκ των οποίων μπορεί να έχει μέχρι 10000 συνάψεις, δηλαδή συνδέσεις, με άλλους νευρώνες [2]. Οι νευρώνες είναι υπεύθυνοι για τη δημιουργία και το διαβιβασμό ηλεκτρικών τάσεων σε ολόκληρο το σώμα, ελέγχοντας και ρυθμίζοντας έτσι τις λειτουργίες του. Είναι λογικό επομένως ότι οι προσπάθειες της επιστήμης της πληροφορικής να επέμβει στο νευρικό σύστημα για την μερική ή ολική αποκατάσταση νευρολογικών παθήσεων ξεκινούν από τη δημιουργία τεχνητών νευρώνων. Ο όρος «νευρομορφικά συστήματα» που εφευρέθηκε το 1990 από τον C. Mead περιγράφει κυκλώματα που μιμούνται τη βιολογική λειτουργικότητα [3]. Αυτά τα συστήματα διακρίνονται σε αναλογικά, ψηφιακά ή υβριδικά και οι αρχιτεκτονικές που τα διέπουν διαφέρουν από την παραδοσιακή von Neumann αρχιτεκτονική καθώς δεν υπάρχει διαχωρισμός της κεντρικής μονάδας επεξεργασίας και της μνήμης [4].

Πέρα από την αποκατάσταση σωματικών βλαβών, η νευρομορφική υπολογιστική πλέον αναπτύσσεται ως ανεξάρτητος κλάδος της πληροφορικής προσφέροντας μία πιθανή διέξοδο από τους περιορισμούς της von Neumann αρχιτεκτονικής. Με τη σωστή υλοποίηση μπορούμε να έχουμε χαμηλότερη κατανάλωση ενέργειας, μικρότερη επιφάνεια και μεγαλύτερη ταχύτητα σε σχέση με τα τυπικά κυκλώματα. Αυτά τα χαρακτηριστικά κάνουν τον τομέα πολύ ελκυστικό για την ανάπτυξη συστημάτων που αφορούν τη βιοϊατρική μηχανική, την ασφάλεια συστημάτων κ.α.

Στην παρούσα εργασία παρουσιάζεται μια υλοποίηση νευρομορφικής σύναψης συμβατή με νευρώνες βασιζόμενους στο μαθηματικό νευρωνικό μοντέλο Izhikevich. Γίνεται χρήση της τεχνολογίας FPGA και της γλώσσας περιγραφής υλικού VHDL. Η υλοποίηση βασίζεται στην τεχνητή σύναψη που παρουσίασαν ο Ambroise και συνεργάτες [5] .

Η εργασία δομείται ως εξής: στο 2<sup>ο</sup> κεφάλαιο εισάγονται κάποιες βασικές έννοιες των βιολογικών και τεχνητών νευρώνων, συνάψεων και δικτύων. Στο 3<sup>ο</sup> κεφάλαιο παρουσιάζονται αρχές της αναδιατασσόμενης τεχνολογίας όπως αυτή χρησιμοποιείται για τη δημιουργία νευρομορφικών κυκλωμάτων, καθώς και άλλα σχετικά projects. Στο 4<sup>ο</sup> κεφάλαιο αναλύεται το μοντέλο Izhikevich και παράγωγες υλοποιήσεις τους, αφού η σύναψη απευθύνεται σε νευρώνες βασιζόμενος σε αυτό. Το προτεινόμενο κύκλωμα παρουσιάζεται εκτενώς στο 5<sup>ο</sup> κεφάλαιο. Οι επιδόσεις της υλοποίησης καταγράφονται στο 6<sup>ο</sup> κεφάλαιο. Τα συμπεράσματα της εργασίας βρίσκονται στο 7<sup>ο</sup> κεφάλαιο. Τέλος, στο παράρτημα Α παρουσιάζονται χαρακτηριστικά της πλατφόρμας ανάπτυξης υλικού FPGA και της γλώσσας περιγραφής υλικού VHDL και στο παράρτημα Β παρατίθεται ο κώδικας της υλοποίησης.



## **2 Νευρώνες, Συνάψεις και Δίκτυα**

### **2.1 Εισαγωγή**

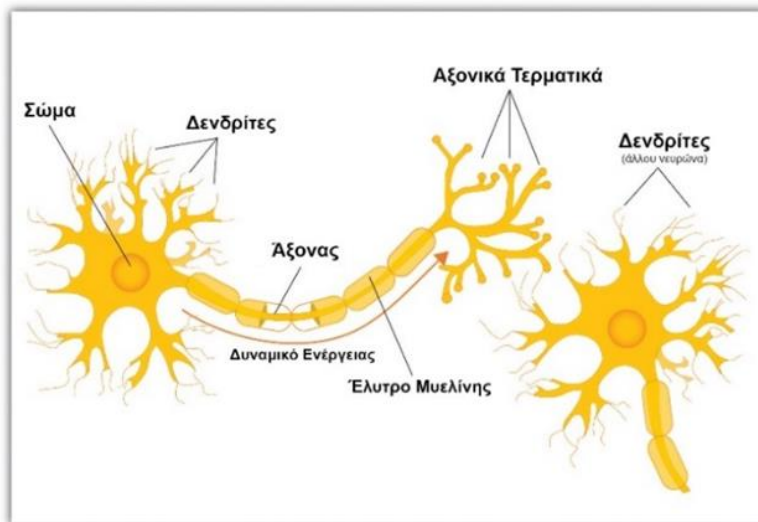
Όπως προαναφέρθηκε, ο εγκέφαλος αποτελείται από νευρώνες οι οποίοι συνδέονται μεταξύ τους μέσω των συνάψεων, μεταβιβάζοντας ηλεκτρικές τάσεις και σχηματίζοντας δίκτυα. Ο τεράστιος παραλληλισμός που προκύπτει και η χαμηλή συχνότητα λειτουργίας συμβάλλουν στις χαμηλές ενεργειακές απαιτήσεις που ενέπνευσαν τη δημιουργία των νευρομορφικών κυκλωμάτων.

Στο παρόν κεφάλαιο αναφέρονται αρχικά τα βιολογικά χαρακτηριστικά των νευρώνων και των συνάψεων. Στη συνέχεια γίνεται μια ιστορική αναδρομή στα διαφορετικά μαθηματικά μοντέλα που έχουν προταθεί για την προσομοίωση αυτών των βιολογικών χαρακτηριστικών. Τέλος, παρουσιάζονται οι τρόποι με τους οποίους τα μαθηματικά μοντέλα νευρώνων και συνάψεων μπορούν να οργανωθούν σε δίκτυα.

### **2.2 Ο Βιολογικός Νευρώνας**

#### **2.2.1 Δομή**

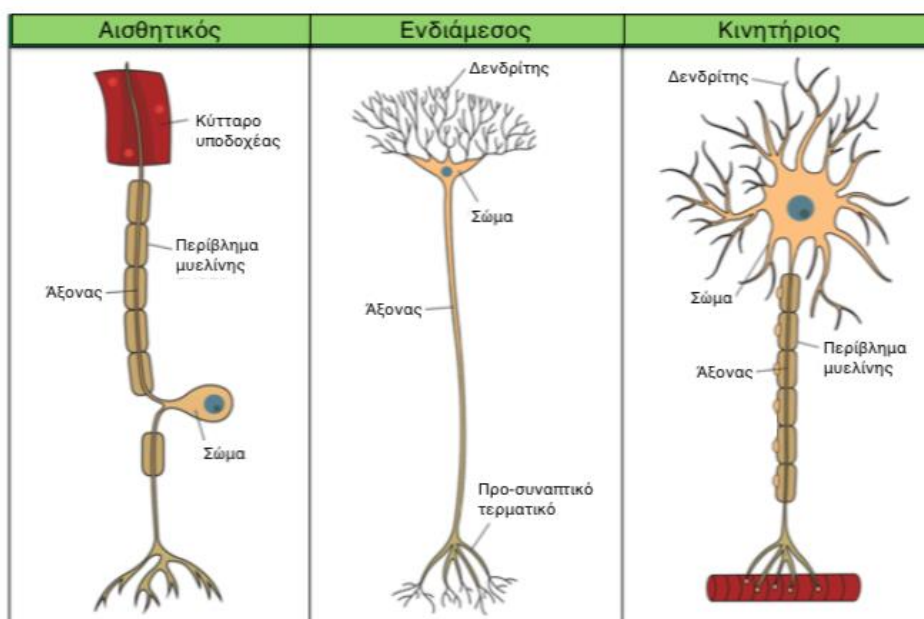
Όπως όλα τα κύτταρα του ανθρώπινου σώματος, οι δισεκατομμύρια νευρώνες παρουσιάζουν μεγάλη ποικιλομορφία μεταξύ τους. Παρόλα αυτά, κάθε νευρώνας αποτελείται από τρία μέρη, το κυτταρικό σώμα (ή απλά σώμα), τους δενδρίτες (dendrites) και τον άξονα (axon). Στο σώμα περιέχεται ο κυτταρικός πυρήνας με τη γενετική πληροφορία και τα υπόλοιπα οργανίδια που είναι κοινά με αυτά άλλων κυττάρων, όπως τα μιτοχόνδρια και τα ριβοσώματα. Εδώ δημιουργούνται οι ηλεκτρικοί παλμοί. Οι δενδρίτες αποτελούν την «είσοδο» του νευρώνα. Δέχονται παλμούς από τους άξονες άλλων νευρώνων και τους επεξεργάζονται. Για την αύξηση της επιφάνειας επαφής με τους άξονες, διακλαδίζονται στα λεγόμενα δενδριτικά δέντρα. Ο άξονας είναι η «έξοδος» του νευρώνα. Αυτή η μακριά απόληξη μεταφέρει τους παραγόμενους ηλεκτρικούς παλμούς του σώματος στο κύτταρο-στόχο. Περιβάλλεται από μία λιπαρή ουσία γνωστή ως μυελίνη η οποία επιταχύνει την ηλεκτρική διάδοση. Καταλήγει σε εξογκώματα γνωστά ως νευραξονικές απολήξεις [6].



Εικόνα 1. Διάγραμμα ενός τυπικού νευρώνα και των συστατικών μερών του.

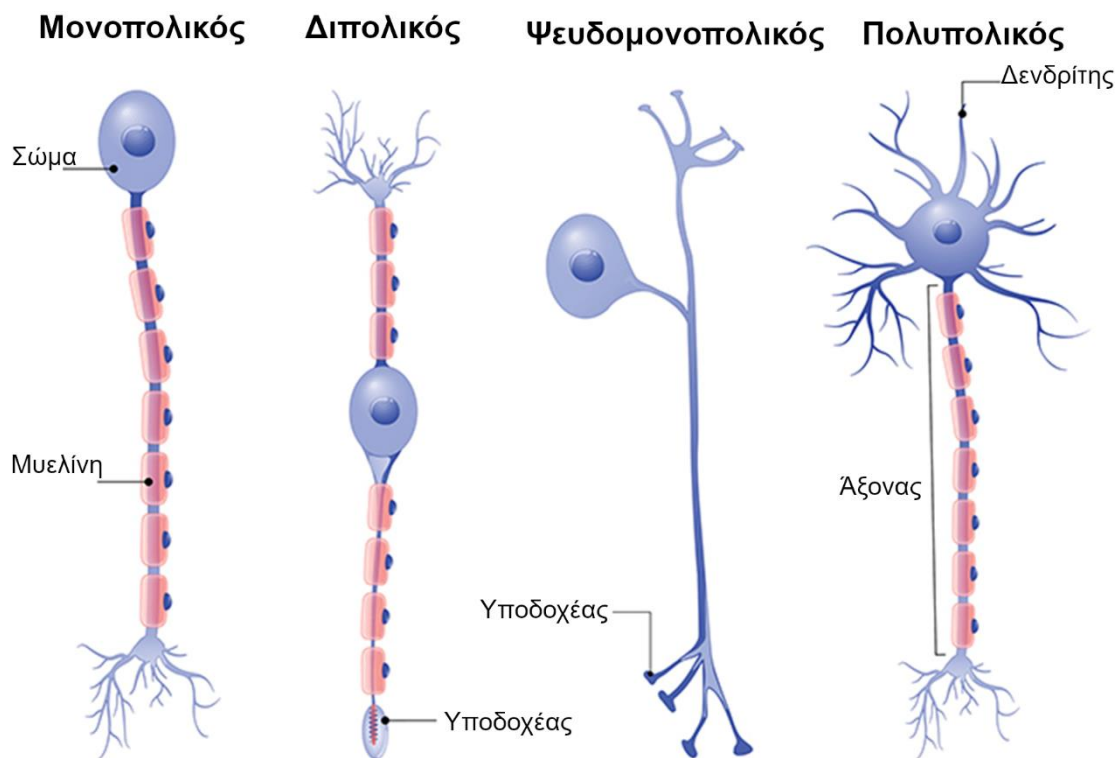
### 2.2.2 Κατηγοριοποιήσεις Βιολογικών Νευρώνων

Ένας τρόπος κατηγοριοποίησης των νευρώνων είναι με βάση τη λειτουργία τους. Έτσι χωρίζονται σε αισθητικούς, κινητήριους και ενδιάμεσους. Οι αισθητικοί νευρώνες είναι υπεύθυνοι για τη μεταφορά εξωτερικών ερεθισμάτων στο κεντρικό νευρικό σύστημα (ΚΝΣ). Δεν ενεργοποιούνται από άλλους νευρώνες. Οι κινητήριοι νευρώνες βρίσκονται στο εσωτερικό του ΚΝΣ και μέσω των αξόνων τους μεταβιβάζουν παλμούς έξω από αυτό, ελέγχοντας τους μυς. Οι ενδιάμεσοι νευρώνες λειτουργούν ως μεσάζοντες των άλλων δύο και βρίσκονται στο ΚΝΣ [7].



Εικόνα 2. Κατηγοριοποίηση των νευρώνων σε αισθητικούς, ενδιάμεσους και κινητήριους ανάλογα με τη λειτουργία τους.

Ένας άλλος τρόπος κατηγοριοποίησης των νευρώνων είναι με βάση τον αριθμό των κλάδων (διεργασίες) που εξέρχονται από το σώμα. Με αυτό το κριτήριο διαχωρίζονται σε μονοπολικούς, διπολικούς, ψευδομονοπολικούς και πολυπολικούς. Από το σώμα των μονοπολικών νευρώνων εκτείνεται μόνο ένας κλάδος το τέλος του οποίου διακλαδίζεται σε νευραξονικά και δενδριτικά τερματικά. Στους διπολικούς νευρώνες, δύο κλάδοι εξέρχονται αντιδιαμετρικά από το σώμα. Ο ένας αποτελεί τον άξονα και ο άλλος τερματίζεται στο δενδριτικό δέντρο. Ο κλάδος που ξεκινάει από το σώμα των ψευδομονοπολικών νευρώνων μετά από μικρή απόσταση χωρίζεται σε δύο μακριούς άξονες. Αυτοί οι άξονες είναι ιδιαίτεροι γιατί επιτελούν ταυτόχρονα την κανονική λειτουργία τους και τη λειτουργία των δενδριτών. Ονομάζονται ψευδομονοπολικοί γιατί στα πρώτα στάδια της ανάπτυξης είναι διπολικοί. Οι πολυπολικοί νευρώνες συμβαδίζουν περισσότερο από όλους με τον τυπικό νευρώνα της Εικόνας 1. Απαρτίζουν την πλειοψηφία των νευρώνων του ΚΝΣ, έχουν έναν μακρύ άξονα και τρεις ή περισσότερους δενδρίτες που δεν εκτείνονται πολύ μακριά από το σώμα [8].



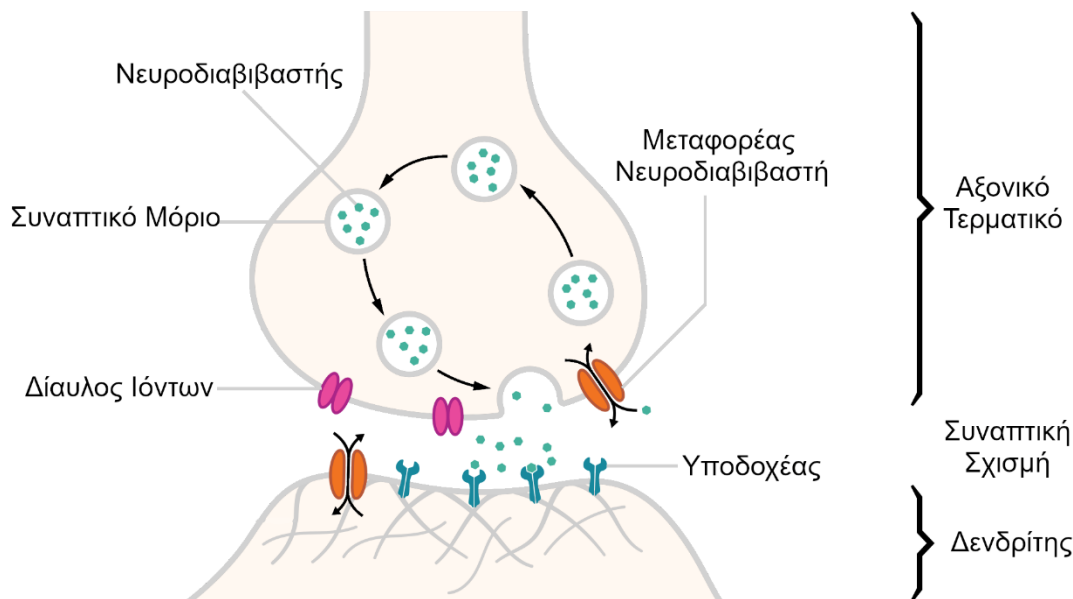
Εικόνα 3. Κατηγοριοποίηση των νευρώνων σε μονοπολικούς, διπολικούς, ψευδομονοπολικούς και πολυπολικούς ανάλογα με τη δομή τους.

### 2.3 Η Βιολογική Σύναψη

Κατά την επικοινωνία τους οι νευρώνες αποκαλούνται προσυναπτικοί και μετασυναπτικοί. Οι πρώτοι είναι αυτοί που δημιουργούν τον ηλεκτρικό παλμό και οι δεύτεροι είναι οι αποδέκτες. Ανάλογα με τη φύση της σύνδεσης τους, οι αποδέκτες

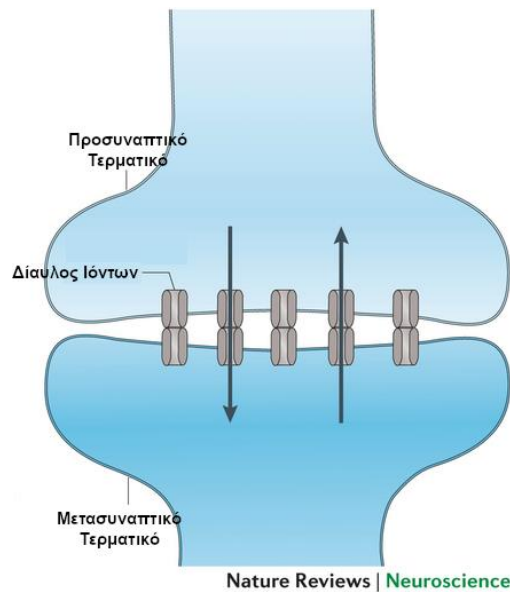
είτε μεταβιβάζουν τον παλμό που λαμβάνουν σε κάποιο άλλο κύτταρο (δηλαδή διεγείρονται) ή δεν το μεταβιβάζουν (δηλαδή αναστέλλονται). Οι συνδέσεις μεταξύ των νευρώνων ονομάζονται συνάψεις, όρος που χρησιμοποιήθηκε για πρώτη φορά από τον Sherrington το 1897. Χωρίζονται σε δύο μεγάλες κατηγορίες: τις χημικές και τις ηλεκτρικές.

Η πλειοψηφία των συνάψεων στο ανθρώπινο σώμα είναι χημικές. Σε αυτές η επικοινωνία είναι μονόδρομη και οι δύο νευρώνες δεν έρχονται σε επαφή. Αντιθέτως διαχωρίζονται από μία σχισμή μήκους 20-500 nm. Λόγω αυτού του κενού εμφανίζουν μια συναπτική καθυστέρηση που διαρκεί περίπου 0.3 ms [9]. Η επικοινωνία γίνεται εφικτή μέσω χημικών ουσιών γνωστών ως νευροδιαβιβαστές. Αφού εξέλθουν από το προσυναπτικό κύτταρο και διασχίσουν τη συναπτική σχισμή, δεσμεύονται από υποδοχείς του μετασυναπτικού κυττάρου. Αυτοί οι υποδοχείς μπορεί να είναι είτε ιοντικοί διάλυοι ή μεταβοτροπικοί υποδοχείς. Στην πρώτη περίπτωση η πρόσδεση του νευροδιαβιβαστή προκαλεί ιοντικό ρεύμα και μεταβάλλεται η μεμβρανική τάση του μετασυναπτικού κυττάρου. Στη δεύτερη περίπτωση η πρόσδεση του νευροδιαβιβαστή προκαλεί μια σειρά από βιοχημικές διεργασίες που έχει σαν αποτέλεσμα το άνοιγμα των ιοντικών διαύλων και τη μεταβολή της μεμβρανικής τάσης [10].



Εικόνα 4. Διάγραμμα χημικής σύναψης.

Οι ηλεκτρικές συνάψεις απαντώνται σπανιότερα. Διαφέρουν από τις χημικές καθώς τα δύο εμπλεκόμενα κύτταρα συνδέονται μέσω διαύλων επιτρέποντας την ελεύθερη και αμφίδρομη διέλευση ιόντων. Λόγω της άμεσης σύνδεσης των νευρώνων που παρουσιάζεται στις ηλεκτρικές συνάψεις, η επικοινωνία επιτυγχάνεται χωρίς σημαντική συναπτική καθυστέρηση. Οι ηλεκτρικές συνάψεις απαντώνται σε δίκτυα νευρώνων που πρέπει να δρουν συντονισμένα, όπως αυτά στον υποθάλαμο των θηλαστικών [11].



Εικόνα 5. Διάγραμμα ηλεκτρικής σύναψης

## 2.4 Ο Τεχνητός Νευρώνας

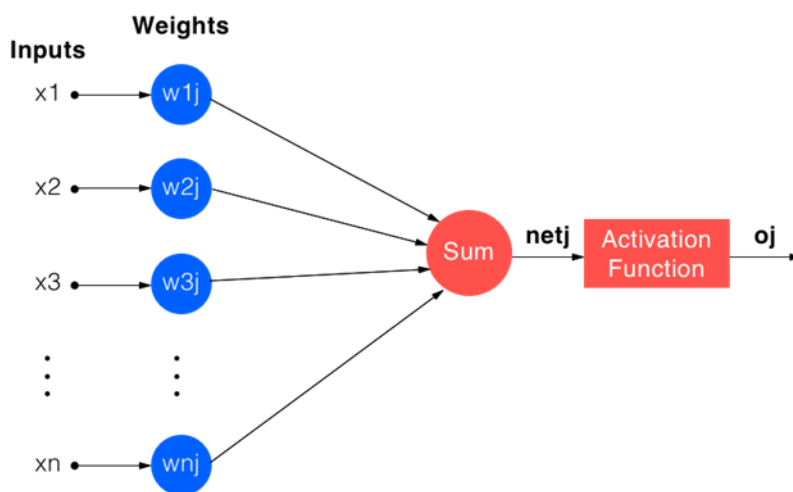
### 2.4.1 Εισαγωγή

Τα διαφορετικά μοντέλα που έχουν προταθεί για τη μίμηση των βιολογικών νευρώνων μπορούν να χωριστούν σε τρεις γενιές ανάλογα με την πολυπλοκότητα τους [12]:

1. Η πρώτη γενιά αφορά σε μοντέλα νευρώνων, τα οποία δεν περιλαμβάνουν τον χρόνο ως παράγοντα υπολογισμού της τιμής της εξόδου. Γίνεται η υπόθεση δηλαδή ότι οι είσοδοι του νευρώνα φτάνουν την ίδια ακριβώς στιγμή. Σε αυτή τη γενιά η έξοδος του νευρώνα είναι δυαδική, 1 εάν το άθροισμα των γινομένων ξεπερνά το κατώφλι και 0 διαφορετικά.
2. Στα μοντέλα της δεύτερης γενιάς εισάγεται η ιδέα της συνάρτησης ενεργοποίησης. Η συνάρτηση ενεργοποίησης είναι υπεύθυνη για τον υπολογισμό της εξόδου βάσει των εισόδων. Σε αντίθεση με την απλή πρόσθεση που συνέβαινε στα μοντέλα της πρώτης γενιάς, χρησιμοποιούνται συναρτήσεις όπως η σιγμοειδής και η συνάρτηση ακτινικής βάσης.
3. Η τρίτη γενιά είναι η πιο πρόσφατη και οι νευρώνες της (γνωστοί ως «spiking» νευρώνες) είναι πιο κοντά στους βιολογικούς από τους προηγούμενους. Αντί για διακριτές τιμές, αυτοί οι νευρώνες παίρνουν ως είσοδο και έχουν ως έξοδο σειρές αποκρίσεων ακίδας.

Τυπικά, τα μοντέλα τεχνητών νευρώνων της δεύτερης και τρίτης γενιάς έχουν την δομή που φαίνεται στην Εικόνα 6. Οι είσοδοι  $x_1, x_2 \dots x_n$  που προέρχονται από άλλους νευρώνες εισέρχονται στο σύστημα και πολλαπλασιάζονται με τις βαρύτητες

των συνδέσεων  $w_1, w_2, \dots, w_n$ . Τα γινόμενα αθροίζονται και εισάγονται στη συνάρτηση ενεργοποίησης η οποία παράγει την τελική έξοδο του νευρώνα. Αυτή η έξοδος αποτελεί στη συνέχεια την είσοδο για άλλους νευρώνες στο δίκτυο. Η αντιστοίχιση αυτής της δομής με αυτήν ενός βιολογικού νευρώνα μπορεί να γίνει ως εξής: ο πολλαπλασιασμός των εισόδων με τις βαρύτητες και η πρόσθεση των γινομένων λειτουργούν όπως οι δενδρίτες, η επεξεργασία των γινομένων με χρήση της συνάρτησης ενεργοποίησης προσομοιώνει το κυτταρικό σώμα και η τελική έξοδος συμπεριφέρεται όπως ο άξονας.



Εικόνα 6. Διάγραμμα τεχνητού νευρώνα.

Ακολουθεί μια παρουσίαση διαφορετικών μοντέλων τεχνητών νευρώνων που έχουν προταθεί από τους ερευνητές για τη χρήση σε δίκτυα.

### 2.4.2 McCulloch-Pitts

Ίσως το πρώτο υπολογιστικό μοντέλο ενός νευρώνα παρουσιάστηκε το 1943 από το νευροεπιστήμονα Warren McCulloch και τον μαθηματικό Walter Pitts. Αυτό και όλα τα μοντέλα που βασίστηκαν σε αυτό αποτελούν την πρώτη γενιά νευρωνικών μοντέλων.

Από μαθηματικής άποψης το μοντέλο είναι αρκετά απλό. Αρχικά παίρνει ως είσοδο τις δυαδικές τιμές  $x_1, x_2 \dots x_n$  και ανάλογα με το αν είναι διεγερτικές ή ανασταλτικές τις πολλαπλασιάζει με τις τιμές βαρύτητας  $w_1, w_2, \dots, w_n$  που ισούνται με +1 και -1 αντίστοιχα. Τα γινόμενα αθροίζονται και συγκρίνονται με μία προκαθορισμένη τιμή κατωφλίσωσης. Αν το άθροισμα των γινομένων είναι μικρότερο αυτής της τιμής η έξοδος του νευρώνα είναι 0, αλλιώς είναι 1 [13].

### 2.4.3 Hodgkin-Huxley

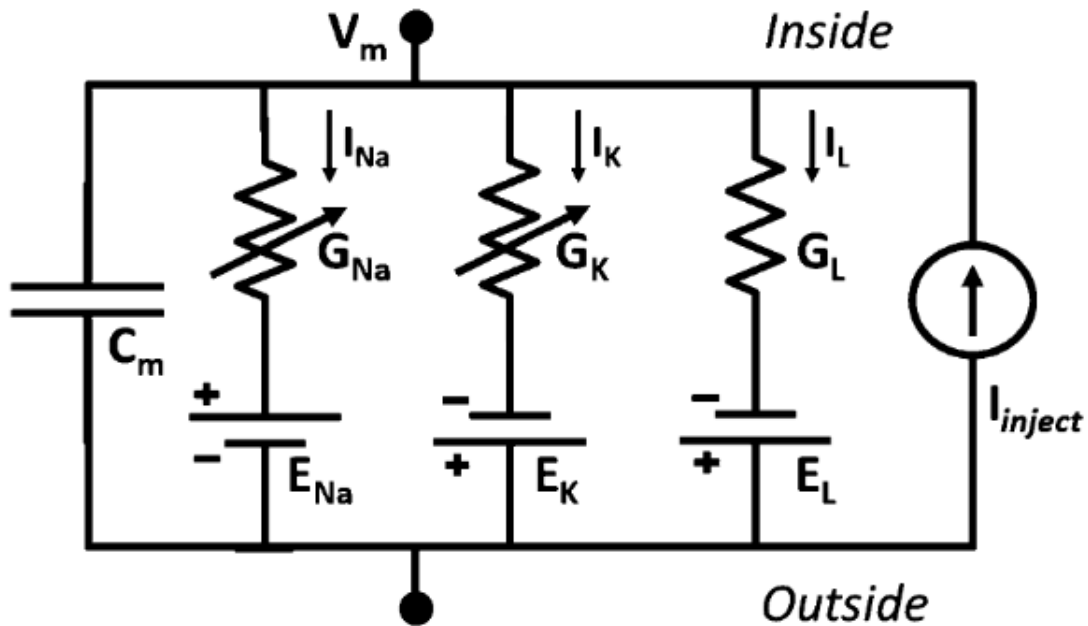
Αυτό το μαθηματικό μοντέλο προτάθηκε από τους φυσιολόγους και βιοφυσικούς Alan Hodgkin και Andrew Huxley το 1952 ακολουθώντας τα πειράματα που είχαν κάνει με τους τεράστιους νευρωνικούς άξονες των καλαμαριών. Σκοπός τους ήταν να αποτυπώσουν τη διάδοση ηλεκτρικού ρεύματος μέσα από κανάλια ιόντων στη νευρωνική μεμβράνη [14].

Αρχικά σχεδίασαν ένα ηλεκτρικό κύκλωμα που αντιπροσώπευε τα πειραματικά ευρήματα τους και στη συνέχεια το μοντελοποίησαν μαθηματικά με 4 διαφορετικές εξισώσεις πρώτου βαθμού.

Οι ρόλοι που επιτελούν τα εξαρτήματα του κυκλώματος της Εικόνας 7 είναι οι εξής:

1. Το  $V_m$  είναι το μεμβρανικό δυναμικό εντός του αξονικού κυττάρου σε σύγκριση με ένα σταθερό σημείο εκτός αυτού.
2. Το  $I_{inject}$  είναι η ένταση ενός εξωτερικού ρεύματος άσχετο με την ένταση των ιόντων
3. Ο πυκνωτής  $C_m$  αποτυπώνει τη μεμβρανική χωρητικότητα όπως προκύπτει από τη διαφορά αγωγιμότητας της σχετικά μη αγωγίμης κυτταρικής μεμβράνης και του αγωγίμου κυτοπλάσματος.
4. Οι διαγραμμένες αντιστάσεις  $G_{NA}$  και  $G_K$  αναπαριστούν την μεταβαλλόμενη αγωγιμότητα των ιόντων νατρίου και καλίου αντίστοιχα. Η σταθερή αγωγιμότητα  $G_L$  είναι η αγωγιμότητα διαρροής που οφείλεται σε παθητικά κανάλια ιόντων και δεν εξαρτάται από την τάση σε αντίθεση με τις  $G_{NA}$  και  $G_K$ .

Οι πηγές  $E_{NA}$ ,  $E_K$  και  $E_L$  αντιπροσωπεύουν το δυναμικό αντιστροφής των  $G_{NA}$ ,  $G_K$  και  $G_L$  αντίστοιχα.



Εικόνα 7. Το ηλεκτρικό κύκλωμα του μοντέλου Hodgkin-Huxley.

Το μαθηματικό μοντέλο περιγράφεται από τις εξής εξισώσεις:

$$C_m \frac{dV_m}{dt} = \bar{g}_{Na} m^3 h (E_{Na} - V_m) + \bar{g}_K n^4 (E_K - V_m) + \bar{g}_{leak} (E_L - V_m) + I_{leak} \quad (1)$$

$$\frac{dn}{dt} = \alpha_n(V)(1 - n) - \beta_n(V)n \quad (2)$$

$$\frac{dm}{dt} = \alpha_m(V)(1 - m) - \beta_m(V)m \quad (3)$$

$$\frac{dh}{dt} = \alpha_h(V)(1 - h) - \beta_h(V)h \quad (4)$$

Στις παραπάνω εξισώσεις, τα  $n$  και  $m$  είναι οι αδιάστατες σταθερές ενεργοποίησης του νατρίου και του καλίου αντίστοιχα και το  $h$  είναι η σταθερά απενεργοποίησης. Τα  $\alpha(V)$  και  $\beta(V)$  είναι τασο-εξαρτώμενες παράμετροι που απαιτούνται για τον



υπολογισμό των  $m$ ,  $h$  και  $n$ . Τα  $\bar{g}_{Na}$ ,  $\bar{g}_K$  και  $\bar{g}_{leak}$  είναι οι παράγοντες κλιμάκωσης των  $G_{Na}$ ,  $G_K$  και  $G_L$  αντίστοιχα.

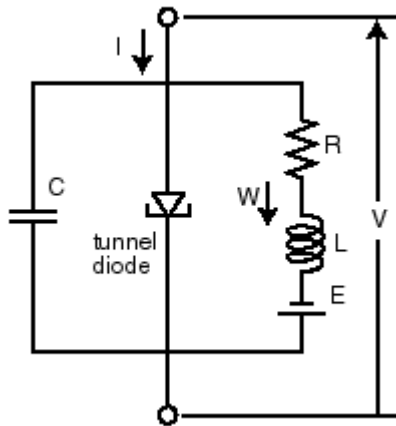
#### 2.4.4 FitzHugh-Nagumo

Το μαθηματικό μοντέλο FitzHugh-Nagumo προτάθηκε από τον βιοφυσικό Richard FitzHugh το 1961. Αποτελεί έναν ταλαντωτή τύπου «Bonhoeffer-van der Pol», είναι δηλαδή ένα σύστημα που αποτυπώνει το πώς αντιδρούν οι μεταβλητές του συναρτήσεως του χρόνου μετά από μια είσοδο ρεύματος. Το 1962, ο J. Nagumo κ.α. σχεδίασαν το αντίστοιχο ηλεκτρικό κύκλωμα (Εικόνα 8). Η δισδιάστατη αυτή απλοποίηση του μοντέλου Hodgkin-Huxley είχε ως στόχο να ξεχωρίσει τις μαθηματικές ιδιότητες της διέγερσης και διάδοσης δυναμικών ενέργειας στους νευρώνες από τις ηλεκτροχημικές ιδιότητες της ροής ιόντων [15]. Οι δύο εξισώσεις του μοντέλου είναι οι εξής:

$$\dot{V} = V - V^3/3 - W + I \quad (5)$$

$$\dot{W} = 0.08(V + 0.7 - 0.8W) \quad (6)$$

Όπου  $V$  είναι η μεμβρανική τάση,  $W$  είναι η μεταβλητή ηρεμίας και  $I$  είναι το ρεύμα εισόδου. Η εξίσωση (5) προσομοιώνει την αύξηση της μεμβρανικής τάσης και η εξίσωση (6) αναπαριστά τη σχετικά αργή απενεργοποίηση των καναλιών ιόντων.



Εικόνα 8. Το αντίστοιχο κύκλωμα του μοντέλου FitzHugh-Nagumo.

#### 2.4.5 Morris-Lecar

Το μοντέλο αυτό παρουσιάστηκε το 1981 από τη βιολόγο Catherine Morris και τον φυσικό Harold Lecar [16]. Αποτελεί μια απλοποιημένη και δισδιάστατη εκδοχή του Hodgkin-Huxley μοντέλου και έχει ως στόχο την προσομοίωση της δημιουργίας αποκρίσεων ακίδας στους νευρώνες. Οι δύο μη γραμμικές εξισώσεις που το περιγράφουν είναι οι εξής:

$$dV_t = \frac{1}{C} (-g_{Ca}m_\infty(V_t)(V_t - V_{Ca}) - g_K W_t (V_t - V_K) - g_L (V_t - V_L) + I) dt \quad (7)$$

$$dW_t = (\alpha(V_t)(1 - W_t) - \beta(W_t)V_t)dt \quad (8)$$

Όπου  $V_t$  είναι η μεμβρανική τάση,  $W_t$  είναι η αγωγιμότητα των ιόντων  $K^+$ ,  $g_{Ca}$ ,  $g_K$  και  $g_L$  είναι οι αγωγιμότητες των ρευμάτων ιόντων  $Ca^{2+}$ ,  $K^+$  και διαρροής αντίστοιχα,  $V_{Ca}$ ,  $V_K$  και  $V_L$  είναι τα δυναμικά των ρευμάτων ιόντων  $Ca^{2+}$ ,  $K^+$  και διαρροής αντίστοιχα,  $V_1$  έως  $V_4$  και  $\varphi$  είναι παράγοντες κλιμάκωσης και  $I$  είναι το ρεύμα εισόδου. Το  $W_t$  παίρνει τιμές από το 0 μέχρι το 1, εκφράζοντας τη πιθανότητα το κανάλι  $K^+$  να είναι ανοιχτό ή κλειστό μία δεδομένη χρονική στιγμή.

Αυτές οι εξισώσεις συμπληρώνονται από τις εξής τρεις βοηθητικές:

$$m_\infty(v) = \frac{1}{2} \left( 1 + \tanh \left( \frac{v - V_1}{V_2} \right) \right) \quad (9)$$

$$\alpha(v) = \frac{1}{2} \varphi \cosh \left( \frac{v - V_3}{2V_4} \right) \left( 1 + \tanh \left( \frac{v - V_3}{V_4} \right) \right) \quad (10)$$

$$\beta(v) = \frac{1}{2} \varphi \cosh \left( \frac{v - V_3}{2V_4} \right) \left( 1 - \tanh \left( \frac{v - V_3}{V_4} \right) \right) \quad (11)$$

Η εξίσωση (9) αναπαριστά την τιμή ισορροπίας της αγωγιμότητας των ιόντων  $Ca^{2+}$  για μία συγκεκριμένη τιμή μεμβρανικής τάσης. Οι εξισώσεις (10) και (11) μοντελοποιούν το άνοιγμα και το κλείσιμο αντίστοιχα των καναλιών ιόντων  $K^+$  συναρτήσει της τάσης.

Γίνεται προφανές ότι το μοντέλο για μία συγκεκριμένη είσοδο ρεύματος υπολογίζει πώς αντιδράν τα κανάλια ιόντων και η μεμβρανική τάση του νευρώνα.

### 2.4.6 Hindmarsh-Rose

Το μοντέλο παρουσιάστηκε το 1984 από τους J. L. Hindmarsh και R. M. Rose. Με τη χρήση τριών δυναμικών μη γραμμικών διαφορικών εξισώσεων πρώτου βαθμού, μπορεί να προσομοιώσει όλες τις δυναμικές λειτουργίες ενός νευρώνα διατηρώντας μια σχετικά χαμηλή υπολογιστική πολυπλοκότητα [17]. Το μοντέλο απέκτησε νέο ενδιαφέρον την τελευταία δεκαετία με τις προόδους που σημειώθηκαν στον χώρο των δυναμικών συστημάτων και του χάους.

Μπορεί να αντιμετωπιστεί ως μια απλοποίηση του μοντέλου Hodgkin-Huxley ή μια γενίκευση του FitzHugh-Nagumo. Το μοντέλο είναι το εξής:

$$\dot{x} = y + 3x^2 - x^3 - z + I \quad (12)$$

$$\dot{y} = 1 - 5x^2 - y \quad (13)$$

$$\dot{z} = r \left[ 4 \left( x + \frac{8}{5} \right) - z \right] \quad (14)$$

Η εξίσωση (12) περιγράφει τη δυναμική της μεμβρανικής τάσης ενώ οι εξισώσεις (13) και (14) περιγράφουν την ανταλλαγή ιόντων μέσω γρήγορων και αργών διαμεμβρανικών καναλιών αντίστοιχα. Όσον αφορά τις παραμέτρους του συστήματος,  $I$  είναι η ένταση του ρεύματος εισόδου και  $r$  είναι η αποτελεσματικότητα ανταλλαγής ιόντων του αργού καναλιού [18].

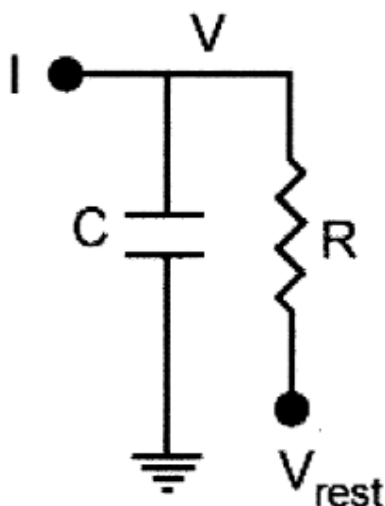
### 2.4.7 Integrate and Fire (I&F)

Παρά το γεγονός ότι το αρχικό Integrate and Fire μοντέλο προτάθηκε από τον Louis Lapicque το 1907, πολύ πριν δηλαδή ακόμα και από το McCulloch-Pitts μοντέλο, παρουσιάζεται τώρα για δύο λόγους. Πρώτον, παρά την ηλικία και την απλότητα του, ακόμα και σήμερα μπορεί να είναι αρκετό για συγκεκριμένες εφαρμογές. Δεύτερον, το απλό I&F ενέπνευσε τη δημιουργία πολλών άλλων μοντέλων, κάποια εκ των οποίων δημοσιεύτηκαν μέσα στην τελευταία εικοσαετία.

Η αρχική έμπνευση του Lapicque για το νευρώνα ήταν ένα κύκλωμα RC, δηλαδή ένα κύκλωμα αποτελούμενο από έναν πυκνωτή τοποθετημένο παράλληλα με μια αντίσταση. Ο πυκνωτής συμβολίζει την χωρητικότητα και η αντίσταση την αντίσταση διαρροής της κυτταρικής μεμβράνης [19]. Το ίδιο το μοντέλο δεν μπορεί να προσομοιώσει τη δημιουργία ακίδων, όμως εξηγεί πώς συμπεριφέρεται το φορτίο του νευρώνα όταν αυτός δέχεται κάποιο ρεύμα εισόδου. Συγκεκριμένα, το φορτίο σταδιακά αυξάνεται μέχρι ένα άνω όριο τάσης εισόδου. Τότε το σύστημα επαναφέρεται στην κατάσταση ηρεμίας [20]. Ακολουθεί το μαθηματικό μοντέλο και το αντίστοιχο ηλεκτρικό κύκλωμα.

$$I(t) = C_m \frac{dV(t)}{dt} \quad (15)$$

Όπου  $I$  το ρεύμα εισόδου,  $C_m$  η μεμβρανική χωρητικότητα και  $V$  η μεμβρανική τάση.



Εικόνα 9. Το αντίστοιχο κύκλωμα του μοντέλου Integrate and Fire.  $C$  είναι η μεμβρανική χωρητικότητα,  $R$  η μεμβρανική αντίσταση,  $V$  η μεμβρανική τάση,  $V_{rest}$  η μεμβρανική τάση ηρεμίας,  $I$  η ένταση του ρεύματος εισόδου.

#### 2.4.8 Leaky Integrate and Fire (LIF)

Μία από τις βελτιώσεις του απλού I&F που πρότεινε ο ίδιος ο Lapicque ονομάζεται Leaky Integrate and Fire (LIF). Η βασική προσθήκη που κάνει το LIF πιο βιολογικά αληθοφανές είναι ο όρος διαρροής,  $I_{leak}$ , που προσομοιώνει την έξοδο ιόντων μέσω της μεμβράνης. Επειδή το μοντέλο αυτό μπορεί να θεωρηθεί βιολογικά αληθοφανές διατηρώντας όμως την υπολογιστική απλότητα του προκατόχου του, είναι ιδιαίτερα δημοφιλές και αποτέλεσε έμπνευση για πιο μοντέρνα και πολύπλοκα μοντέλα όπως αυτά που παρουσιάζονται στις επόμενες ενότητες [20]. Το μαθηματικό μοντέλο είναι το εξής:

$$C_m \frac{dV}{dt} = I_{leak}(t) + I_s(t) + I_{inj}(t) \quad (16)$$

Όπου το  $I_{leak}(t)$  δίνεται από τον τύπο:

$$I_{leak}(t) = \frac{-C_m}{\tau_m} [V(t) - V_{rest}] \quad (17)$$

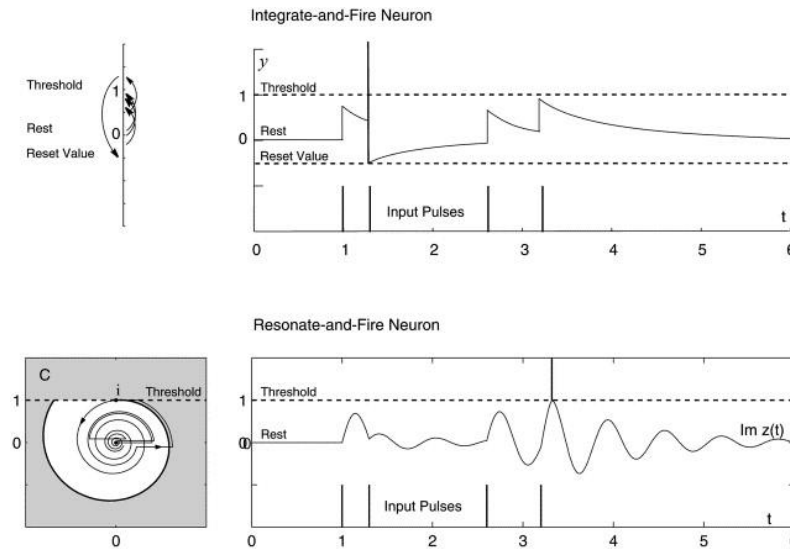
Σε αυτές τις εξισώσεις, το  $I_{leak}$  είναι το ρεύμα που οφείλεται στη διάχυση ιόντων μέσω της μεμβράνης, το  $I_s$  είναι το ρεύμα που οφείλεται σε είσοδο σύναψης στο νευρώνα, το  $I_{inj}$  είναι ένα εξωτερικό ρεύμα εισόδου (μη συναπτικό), το  $\tau_m$  είναι η

χρονική σταθερά της μεμβράνης υπό την υπόθεση ότι  $R_m$  και  $C_m$  είναι σταθερά, το  $V$  είναι η τρέχουσα μεμβρανική τάση και το  $V_{rest}$  είναι η μεμβρανική τάση όταν αυτή βρίσκεται σε κατάσταση ηρεμίας.

### 2.4.9 Resonate and Fire

Αυτός ο τύπος μοντέλων παρουσιάστηκε για πρώτη φορά από τον Eugene Izhikevich το 2001 [21]. Στα επόμενα χρόνια απασχόλησε πολλούς ερευνητές όπως οι Tetsuya Hishiki και Hiroyuki Torikai οι οποίοι το 2009 παρουσίασαν μια γενική εκδοχή της δουλειάς του Izhikevich [22]. Όπως τα Integrate and Fire μοντέλα, τα Resonate and Fire είναι μοντέλα για spiking νευρώνες με αυξημένη πολυπλοκότητα στη μεταβλητή κατάσταση. Επίσης παρουσιάζουν παρόμοια χαρακτηριστικά με αυτά του μοντέλου Hodgkin-Huxley.

Παρά τις γενικές ομοιότητες με τα IF, τα Resonate and Fire παρουσιάζουν συμπεριφορά ταλαντωτή. Αυτό σημαίνει ότι αντί να αυξάνεται ο ρυθμός πυροδότησης γραμμικά με τη συχνότητα των εισόδων, η πυροδότηση εξαρτάται από το χρόνο έλευσης των εισόδων σε σχέση με τη συχνότητα ταλάντωσης του νευρώνα. Για παράδειγμα αν η δεύτερη είσοδος συμπέσει ακριβώς με την συχνότητα ταλάντωσης, ο ρυθμός πυροδότησης μεγιστοποιείται. Η διαφορά στη συμπεριφορά των δύο μοντέλων φαίνεται σχηματικά στην Εικόνα 10.



Εικόνα 10. Σύγκριση της απόκρισης ενός Integrate and Fire νευρώνα και ενός Resonate and Fire νευρώνα.

Το μαθηματικό μοντέλο που πρότεινε ο Izhikevich είναι το εξής:

$$\dot{z} = (b_i + i\omega_i)z_i + \sum_{j=1}^n c_{ij}\delta(t - t_j^*) \quad (18)$$

Όπου  $z_i \in C$  είναι η κατάσταση του  $i$ -οστού νευρώνα,  $b_i + i\omega_i \in C$  είναι η εσωτερική παράμετρος του νευρώνα,  $c_{ij} \in C$  είναι ο συναπτικός συντελεστής,  $\delta$  είναι η συνάρτηση δέλτα και  $t_j^*$  είναι η πιο κοντινή στιγμή πυροδότησης του  $j$ -οστού νευρώνα.

#### 2.4.10 Εκθετικό Integrate and Fire με Προσαρμογή

Αυτή η δισδιάστατη εκδοχή του IF μοντέλου παρουσιάστηκε το 2005 από τους Romain Brette και Wulfram Gerstner. Συνδυάζει μια εκθετική μέθοδο πυροδότησης με μία συνάρτηση προσαρμογής [23].

Το μοντέλο μπορεί να περιγραφεί από τις εξής τρεις εξισώσεις:

$$C \frac{dV}{dt} = f(V) - w + I \quad (19)$$

$$f(V) = -g_L(V - E_L) + g_L \Delta_T \exp\left(\frac{V - V_T}{\Delta_T}\right) \quad (20)$$

$$\tau_w \frac{dw}{dt} = \alpha(V - E_L) - w \quad (21)$$

Η (19) είναι η βασική εξίσωση του μοντέλου, η (20) είναι η συνάρτηση που δίνει τα παθητικά χαρακτηριστικά και τον τρόπο πυροδότησης του νευρώνα και η (21) δίνει το ρεύμα προσαρμογής. Συγκεκριμένα,  $C$  είναι η μεμβρανική χωρητικότητα,  $w$  είναι η μεταβλητή προσαρμογής,  $I$  είναι το συναπτικό ρεύμα,  $g_L$  είναι η αγωγιμότητα διαρροής,  $E_L$  είναι το δυναμικό ηρεμίας,  $\Delta_T$  είναι ο παράγοντας κλίσης,  $V_T$  είναι το δυναμικό κατώφλιου,  $\tau_w$  είναι η χρονική σταθερά και το  $\alpha$  αναπαριστά το επίπεδο της προσαρμογής κάτω από το κατώφλι.

#### 2.4.11 Izhikevich

Το βιολογικά εμπνευσμένο μοντέλο Izhikevich αποτελεί ένα από τα πιο δημοφιλή στο χώρο των νευρομορφικών κυκλωμάτων. Πιο εκτενής περιγραφή θα παρουσιαστεί στο επόμενο κεφάλαιο όπου θα αναλυθούν και τα χαρακτηριστικά του.

## 2.5 Η Τεχνητή Σύναψη

### 2.5.1 Εισαγωγή

Όπως προαναφέρθηκε στην ενότητα 2.3, συνάψεις ονομάζονται οι συνδέσεις μεταξύ των νευρώνων. Η λειτουργία τους εξαρτάται από τα δυναμικά των μεμβρανών του σώματος, των δενδριτών και του άξονα όλων των νευρώνων που συμμετέχουν σε αυτές. Η σημαντικότητα τους στη διάδοση της πληροφορίας είναι αντίστοιχη με αυτή

των νευρώνων. Ως αποτέλεσμα, για να είναι ένα τεχνητό νευρωνικό μοντέλο πλήρες θα πρέπει να συνοδεύεται από ένα αντίστοιχο συναπτικό. Οι ερευνητές έχουν προτείνει ανά τα χρόνια πολλά διαφορετικά μοντέλα τεχνητών συνάψεων με λιγότερη ή περισσότερη βιολογική αληθοφάνεια.

Τα συναπτικά μοντέλα μπορούν να διαχωριστούν σε αυτά που προορίζονται για τη χρήση σε νευρομορφικά κυκλώματα και σε αυτά που εμφανίζονται στα κλασικά τεχνητά νευρωνικά δίκτυα.

### 2.5.2 Τεχνητές Συνάψεις σε Κλασικά Νευρωνικά Δίκτυα

Οι τεχνητές συνάψεις χρησιμοποιούνται ευρέως στα κλασικά νευρωνικά δίκτυα, όπως είναι τα δίκτυα πρόσθιας διάδοσης πολλαπλών επιπέδων (feed-forward multi-layer networks), τα δίκτυα “winner-take-all” και τα συνελκτικά δίκτυα (convolutional networks) (βλ. 2.6). Όπως στα βιολογικά, έτσι και στα τεχνητά νευρωνικά δίκτυα η συναπτική μάθηση ακολουθά κάποιους βασικούς κανόνες. Αυτοί περιγράφονται από τη μάθηση Hebbian και την εξέλιξη της, τη μάθηση STDP. Ανά τα χρόνια έχουν επίσης προταθεί εξειδικευμένοι κανόνες μάθησης συνήθως βασιζόμενοι στο STDP. Παραδείγματα αποτελούν οι SpikeProp και Tempotron.

Η μάθηση Hebbian είναι μια έκφραση της συναπτικής πλαστικότητας, της διαδικασίας δηλαδή κατά την οποία η αύξηση ή μείωση της δραστηριότητας μίας συγκεκριμένης σύναψης μπορεί να αυξήσει ή να μειώσει την πιθανότητα ενεργοποίησης της αντίστοιχα. Αυτό το είδος πλαστικότητας προτάθηκε από τον ψυχολόγο Donald Hebb το 1949 σε μια προσπάθεια του να εξηγήσει τον μηχανισμό της συσχετιστικής μάθησης [24]. Η κεντρική ιδέα της μάθησης Hebbian είναι η εξής: εάν ένας προσυναπτικός νευρώνας A ενεργοποιεί επαναλαμβανόμενα έναν μετασυναπτικό νευρώνα B ενόσω ο A είναι ενεργός, ο A θα γίνει πιο αποτελεσματικός στην ενεργοποίηση του B. Με άλλα λόγια, το βάρος της σύναψης των δύο νευρώνων θα αυξηθεί. Ένας απλός τρόπος να αποτυπωθεί αυτή η ιδέα με μαθηματικό συμβολισμό είναι ο εξής:

$$\frac{dw}{dt} = \eta xy \quad (22)$$

Όπου  $w$  είναι το συναπτικό βάρος,  $x$  και  $y$  είναι η δραστηριότητα του προσυναπτικού και μετασυναπτικού νευρώνα αντίστοιχα και  $\eta$  είναι ο (πάντα σταθερός και θετικός) ρυθμός εκμάθησης. Σύμφωνα με την (22), το συναπτικό βάρος μπορεί να αυξηθεί μονάχα όταν και οι δύο νευρώνες είναι ενεργοί. Στα τεχνητά νευρωνικά δίκτυα και ιδιαίτερα σε αυτά που χρησιμοποιούν τη μη επιβλεπόμενη μάθηση, τα συναπτικά βάρη ακολουθούν μία εκδοχή του Hebbian μοτίβου που συμπληρώνεται από τη διαδικασία STDP.

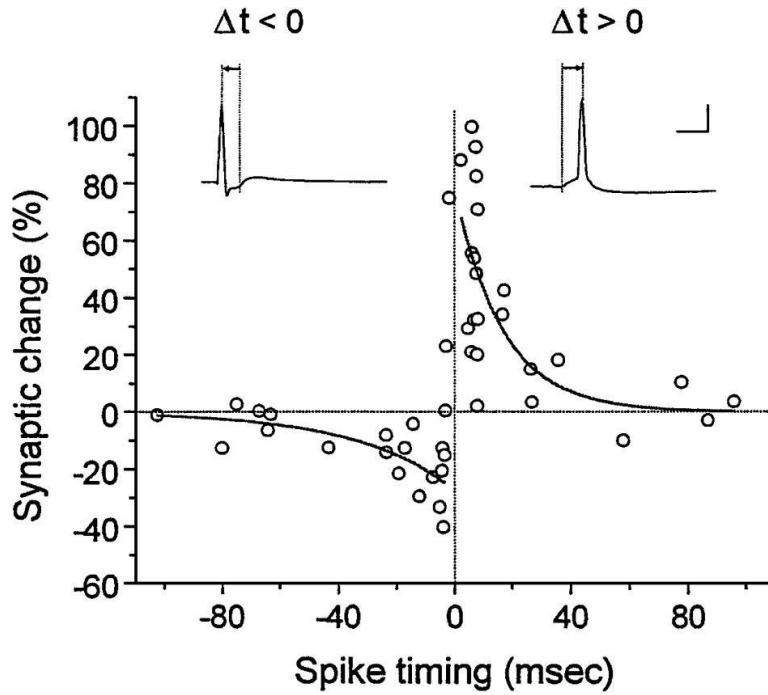
Ένας βασικός περιορισμός της θεωρίας του Hebbian είναι ότι δεν εξετάζει το χρονικό διάστημα ανάμεσα στην ενεργοποίηση του προσυναπτικού και του μετασυναπτικού

νευρώνα. Με άλλα λόγια, δεν υπάρχει τρόπος να είναι κανείς σίγουρος ότι η ενεργοποίηση του μετασυναπτικού νευρώνα όντως προκλήθηκε από την ενεργοποίηση του προσυναπτικού. Αντίθετα, η διαδικασία spike-timing dependent plasticity συνδέει άρρηκτα τη συναπτική αλληλεπίδραση με το χρονισμό παραγωγής ακίδων από τους δύο νευρώνες [25]. Αυτή η εξέλιξη της Hebbian μάθησης παρουσιάστηκε το 1997 από τους Henry Markram και Bert Sakmann ακολουθώντας πειράματα που είχαν διεξαγάγει τρία χρόνια πριν. Το μοντέλο έχει ως εξής:

$$\Delta w_j = \sum_{f=1}^N \sum_{n=1}^N W(t_i^n - t_j^f) \quad (23)$$

Σε αυτήν την εξίσωση  $\Delta w_j$  είναι η αλλαγή βάρους της σύναψης ενός προσυναπτικού νευρώνα  $j$  η οποία εξαρτάται από το σχετικό χρονισμό μεταξύ της άφιξης των προσυναπτικών ακίδων στη σύναψη και της παραγωγής των μετασυναπτικών ακίδων. Ο όρος  $t_j^f$  συμβολίζει το χρόνο άφιξης των προσυναπτικών ακίδων με το  $f = 1, 2, 3, \dots$  να είναι ο μετρητής αυτών των ακίδων. Αντίστοιχα ο όρος  $t_i^n$  με τον μετρητή  $n = 1, 2, 3, \dots$  συμβολίζει το χρόνο παραγωγής των μετασυναπτικών ακίδων. Η  $W(x)$  είναι η συνάρτηση εκμάθησης της STDP. Ανάλογα με τη συνάρτηση που επιλέγεται, το συναπτικό βάρος αλλάζει με διαφορετικό ρυθμό. Στη μάθηση STDP παρατηρούνται δύο διαφορετικά φαινόμενα: η μακροπρόθεσμη ενδυνάμωση (Long-Term Potentiation - LTP) και η μακροπρόθεσμη ύφεση (Long-Term Depression - LTD). Το πρώτο συμβαίνει όταν οι προσυναπτικές ακίδες φτάνουν στη σύναψη μερικά χιλιοστόλεπτα πριν την παραγωγή της μετασυναπτικής ακίδας. Το δεύτερο συμβαίνει όταν οι μετασυναπτικές ακίδες φτάνουν στη σύναψη μερικά χιλιοστόλεπτα πριν την παραγωγή της προσυναπτικής ακίδας. Τα δύο αυτά φαινόμενα αποτυπώνονται στην Εικόνα 11.





Εικόνα 11. Η συνάρτηση STDP μίας σύναψης. Αν η προσυναπτική ακίδα φτάσει πριν τη μετασυναπτική ( $\Delta t > 0$ ), παρατηρείται μακροπρόθεσμη ενδυνάμωση. Αλλιώς ( $\Delta t < 0$ ) παρατηρείται μακροπρόθεσμη ύφεση.

Το 2000 οι Sander Bohte, Joost Kok και Han La Poutre παρουσίασαν τον αλγόριθμο μάθησης επανατροφοδότησης για SNNs SpikeProp. Βασιζόμενοι σε μία σύγχρονη εκδοχή της μάθησης Hebbian, ανέπτυξαν τις εξής εξισώσεις για τον υπολογισμό των μετασυναπτικών εισόδων ενός spiking νευρώνα  $j$ :

$$y_i(t) = \sum_k^m y_i^k(t) \quad (24)$$

$$x_j(t) = \sum_i \sum_k w_{ij}^k y_i^k(t) \quad (25)$$

Η (24) περιγράφει την προσυναπτική είσοδο του νευρώνα  $i \in I$  στο νευρώνα  $j \in J$  ως άθροισμα συναπτικών συνεισφορών, όπου  $m$  είναι ο αριθμός των καθυστερήσεων και  $y_i^k(t)$  το καθυστερημένο προσυναπτικό δυναμικό κάθε τερματικού. Η (25) περιγράφει τις εισόδους  $x_j$  του μετασυναπτικού νευρώνα  $j$  ο οποίος λαμβάνει εισόδους από τους νευρώνες  $i$ . Ο παράγοντας  $w_{ij}^k$  είναι το βάρος του συναπτικού τερματικού  $k$  [26].

Ο κανόνας μάθησης Tempotron παρουσιάστηκε το 2006 από τους Robert Güttig και Haim Sompolinsky ως εξέλιξη του STDP. Αντί να βασίζεται στο μέσο ρυθμό πυροδότησης των προ- και μετα-συναπτικών νευρώνων, ο κανόνας είναι συνδεδεμένος με την χωροχρονική δομή των αποκρίσεων ακίδας. Περιγράφεται από τον τύπο υπολογισμού της μεμβρανικής τάσης υποκατωφλίωσης:

$$V(t) = \sum_i \omega_i \sum_k K(t - t_i) + V_{rest} \quad (26)$$

Όπου  $t_i$  είναι οι χρόνοι ακίδας της  $i$ -οστής άφιξης,  $K(t - t_i)$  είναι το μετασυναπτικό δυναμικό που προσφέρει κάθε εισερχόμενη ακίδα και  $\omega_i$  είναι οι συναπτικές αποτελεσματικότητες [27].

### 2.5.3 Τεχνητές Συνάψεις σε Νευρομορφικά Κυκλώματα

Οι συνάψεις των νευρομορφικών κυκλωμάτων κατά κανόνα προσομοιώνουν τις λειτουργίες στοιχείων των βιολογικών συνάψεων όπως οι διάυλοι ιόντων και οι χημικές αλληλεπιδράσεις των νευροδιαβιβαστών [28]. Αυτή η μοντελοποίηση πολλές φορές είναι πιο λεπτομερής από ό,τι θα έβρισκε κανείς, για παράδειγμα, σε υλοποιήσεις των καναλιών ιόντων του μοντέλου Hodgkin-Huxley. Πέρα από την προσομοίωση συγκεκριμένων στοιχείων, πολλές νευρομορφικές συνάψεις εμφανίζουν παρόμοια συμπεριφορά με τις βιολογικές. Για παράδειγμα μπορούν να εμφανίσουν πλαστικότητα και μάθηση [4]. Τα νευρομορφικά κυκλώματα στα οποία χρησιμοποιούνται αυτές οι συνάψεις συνήθως σχηματίζουν spiking νευρωνικά δίκτυα (Spiking Neural Networks - SNNs), τα οποία όμως ακολουθούν την τοπολογία των δικτύων που αναφέρθηκαν στην ενότητα 2.5.2. Ένα βιολογικά εμπνευσμένο νευρωνικό μοντέλο που συνήθως υλοποιείται με τη χρήση SNNs είναι η Κεντρική Γεννήτρια Μοτίβων (Central Pattern Generator - CPG) (βλ. 2.6).

## 2.6 Μοντέλα Νευρωνικών Δικτύων

Ο συνδυασμός πολλών νευρώνων και συνάψεων δημιουργεί δομές γνωστές ως νευρωνικά δίκτυα. Αυτά τα δίκτυα μπορούν να διαφοροποιούνται όσον αφορά τον αριθμό των νευρώνων που ενσωματώνουν, το νευρωνικό και συναπτικό μοντέλο που επιλέγεται και την «τοπολογία» που ακολουθείται. Η τοπολογία του δικτύου περιγράφει τη δομή του, δηλαδή τον τρόπο σύνδεσης των νευρώνων (γνωστοί επίσης ως «κόμβοι») μέσω των συνάψεων (γνωστές επίσης ως «ακμές»). Τα νευρωνικά δίκτυα ακολουθούν συγκεκριμένους μαθηματικούς τύπους για την παραγωγή της επιθυμητής εξόδου [4].

### 2.6.1 Κεντρική Γεννήτρια Μοτίβων (*Central Pattern Generator - CPG*)

Ένα μοντέλο δικτύου που έχει υλοποιηθεί με τη χρήση SNNs είναι η Κεντρική Γεννήτρια Μοτίβων (ΚΓΜ). Αυτό το βιολογικό δίκτυο είναι υπεύθυνο για την παραγωγή επαναλαμβανόμενων και κυκλικών κινήσεων στο σώμα με σκοπό την επίτευξη τόσο ενεργητικών όσο και παθητικών λειτουργιών. Στις ενεργητικές λειτουργίες ανήκουν το περπάτημα και το κολύμπι. Στις παθητικές λειτουργίες υπάγονται η αναπνοή και το χτύπημα της καρδιάς. Τα μοτίβα στα οποία βασίζονται αυτές οι κινήσεις παράγονται κατά κύριο λόγο από τις ΚΓΜ. Το βασικό τους χαρακτηριστικό είναι ότι δε βασίζονται σε εξωτερικά ερεθίσματα για τη λειτουργία τους [29].

Ο βιολογικός ρόλος των ΚΓΜ στην κίνηση ενέπνευσε πολλούς ερευνητές να προσομοιώσουν αυτά τα δίκτυα για να τα χρησιμοποιήσουν σε υλοποιήσεις όπως η ρομποτική. Ενδεικτικά αναφέρεται η δουλειά των Chen, Wang κ.ά. οι οποίοι υλοποίησαν ένα δίκτυο ΚΓΜ πραγματικού χρόνου σε αναδιατασσόμενη τεχνολογία [30]. Η υλοποίηση χρησιμοποίησε το νευρωνικό μοντέλο Komendantov–Kononenko το οποίο με τη σειρά του έχει βασιστεί στο μοντέλο Hodgkin-Huxley.

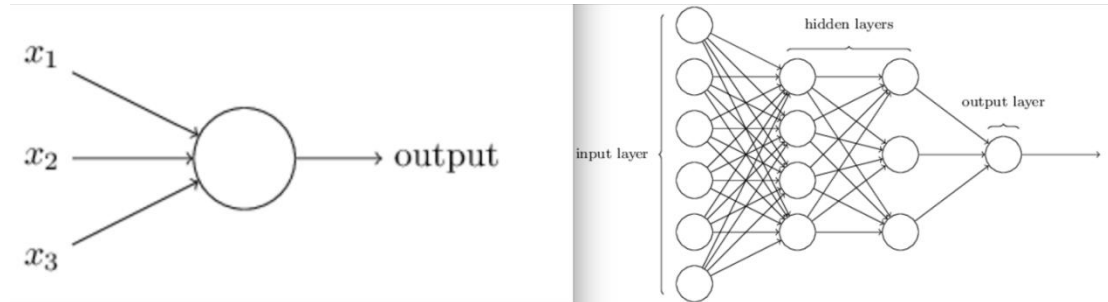
### 2.6.2 Νευρωνικά Δίκτυα Πρόσθιας Διάδοσης

Τα νευρωνικά δίκτυα πρόσθιας διάδοσης (feedforward neural networks) είναι ιδιαίτερα δημοφιλή στους επιστήμονες του κλάδου των νευρομορφικών κυκλωμάτων.

Τα δίκτυα αυτής της κατηγορίας αποτελούν τη βασικότερη έκφραση των μοντέλων βαθιάς μάθησης (deep learning) [31]. Παίρνουν το όνομα τους από το γεγονός ότι δεν υπάρχει ανατροφοδότηση (feedback), δηλαδή τα δεδομένα εντός του δικτύου ρέουν από την είσοδο, στη συνάρτηση υπολογισμού και τέλος στην έξοδο χωρίς να επιστρέφουν προς τα πίσω. Ο στόχος αυτών των δικτύων είναι να προσεγγίσουν μια συνάρτηση  $f$  αντιστοιχίζοντας την είσοδο  $x$  σε μια κατηγορία  $y$ :  $y = f(x; \theta)$  όπου  $\theta$  οι παράμετροι που μαθαίνονται από το δίκτυο και που οδηγούν πιο αποτελεσματικά στην προσέγγιση της  $f$ .

Μία κατηγορία των νευρωνικών δικτύων πρόσθιας διάδοσης είναι τα πολυεπίπεδα perceptrons (multilayer perceptrons - MLPs) [32]. Το perceptron είναι ένα από τα πρώτα και πιο απλά νευρωνικά δίκτυα. Αναπτύχθηκε τη δεκαετία του 1950 από τον Frank Rosenblatt και η λειτουργία του βασίζεται στο μοντέλο McCulloch-Pitts. Χρησιμοποιούνταν για την ταξινόμηση δεδομένων σε μία από δύο κλάσεις, όπως προκύπτει από το μαθηματικό μοντέλο που περιγράφηκε στην ενότητα 2.4.2. Η τοπολογία των MLP έχει ως εξής: πολλαπλοί νευρώνες οργανώνονται σε επίπεδα τριών ειδών· το επίπεδο εισόδου, το επίπεδο εξόδου και τα κρυφά επίπεδα ανάμεσα στα δύο. Κάθε νευρώνας έχει την ίδια συνάρτηση ενεργοποίησης και τα τελικά αποτελέσματα εξάγονται στο επίπεδο εξόδου.

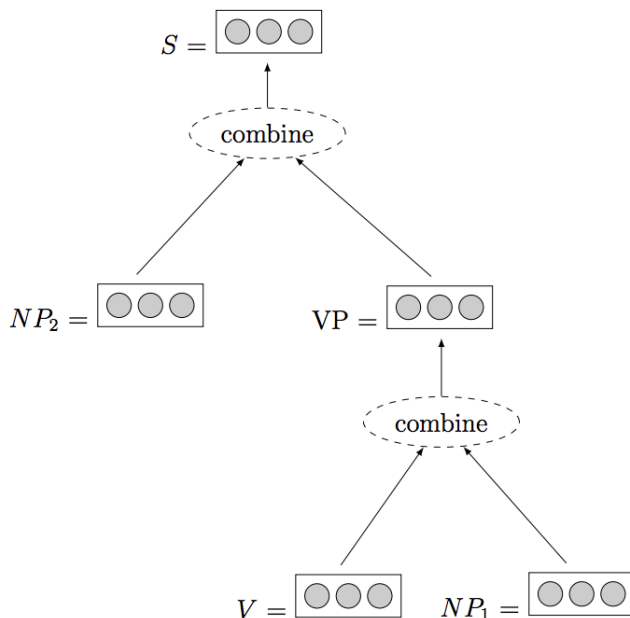
Πλέον αντί για απλή άθροιση και σύγκριση με την τιμή κατωφλίου όπως συνέβαινε στα perceptrons, τα MLP συνήθως έχουν ως συνάρτηση ενεργοποίησης κάποια μη γραμμική συνάρτηση όπως η σιγμοειδής. Στην Εικόνα 12 φαίνονται ένα perceptron και ένα MLP.



Εικόνα 12. Η δομή ενός perceptron (αριστερά) και ενός MLP (δεξιά).

### 2.6.3 Αναδρομικά Νευρωνικά Δίκτυα

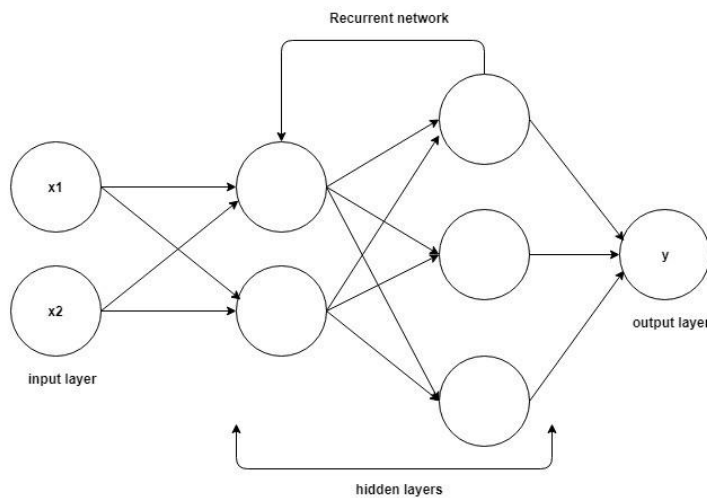
Τα αναδρομικά νευρωνικά δίκτυα (recursive neural networks) είναι μία δημοφιλής κατηγορία των νευρωνικών δικτύων βαθιάς μάθησης. Είναι μη γραμμικά και δέχονται δομημένες εισόδους στις οποίες εφαρμόζουν βάρη με αναδρομικό τρόπο. Έτσι σχηματίζουν μία ιεραρχική δομή δέντρου όπου το βάρος κάθε ακμής σχετίζεται με τις υπόλοιπες. Ο στόχος αυτών των δικτύων είναι να παράξουν μία δομημένη πρόβλεψη στην έξοδο διατρέχοντας την είσοδο. Για αυτό το λόγο έχουν χρησιμοποιηθεί επιτυχώς στην επεξεργασία φυσικής γλώσσας [33].



Εικόνα 13. Εικόνα 14. Αναπαράσταση Αναδρομικού Νευρωνικού Δικτύου στον τομέα της επεξεργασίας φυσικής γλώσσας. Ξεκινώντας από κάτω προς τα πάνω, οι εισόδους (εδώ ρήματα V και φράσεις ουσιαστικών NP) συνδυάζονται για να προκύψει στην έξοδο η πρόταση S.

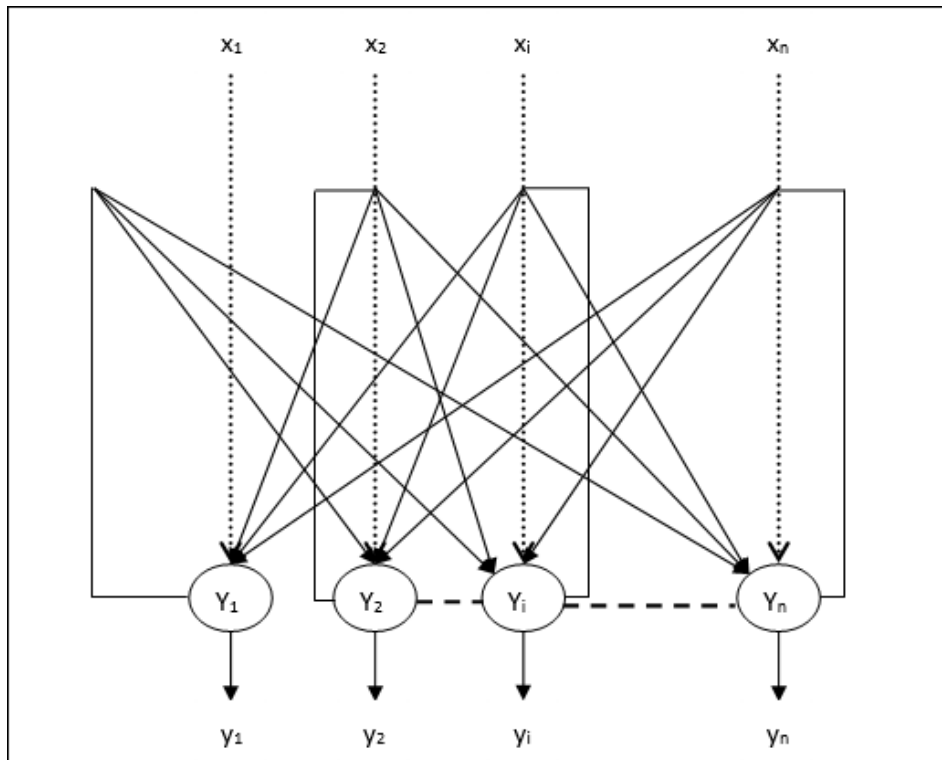
### 2.6.4 Επαναλαμβανόμενα Νευρωνικά Δίκτυα

Τα επαναλαμβανόμενα νευρωνικά δίκτυα (recurrent neural networks) είναι μια ειδική κατηγορία των αναδρομικών στα οποία οι ακμές μεταξύ των κόμβων δημιουργούν κατευθυνόμενους γράφους. Αυτή η ανατροφοδότηση σημαίνει ότι η έξοδος εμφανίζει χρονική εξάρτηση, εξαρτάται δηλαδή από την τωρινή είσοδο και την προηγούμενη έξοδο [34]. Ανάλογα με τον τρόπο σύνδεσης των κόμβων, αυτά τα δίκτυα μπορούν να χωριστούν σε διαφορετικές κατηγορίες. Ένα δημοφιλές παράδειγμα είναι τα πλήρως επαναλαμβανόμενα δίκτυα.



Εικόνα 14. Επαναλαμβανόμενο νευρωνικό δίκτυο.

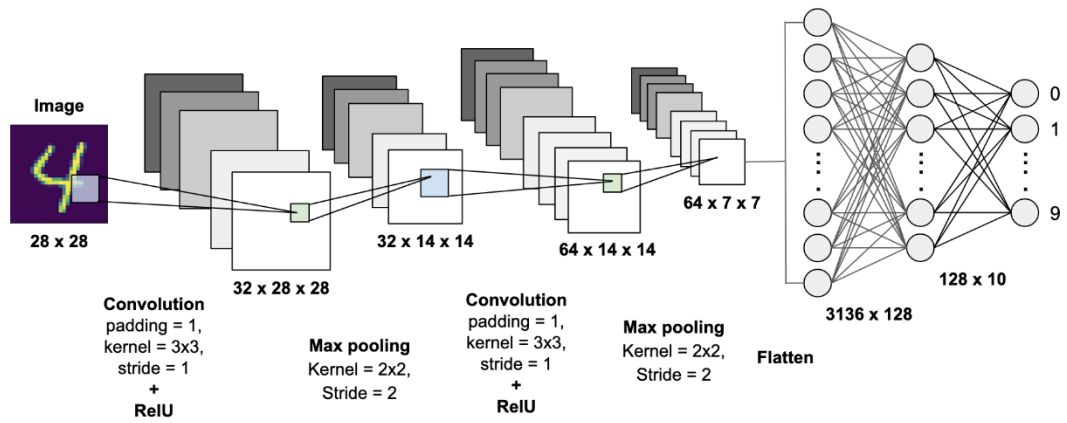
Μία ειδική κατηγορία νευρωνικών δικτύων που εμφανίζουν επανάληψη είναι τα δίκτυα Hopfield. Εφευρέθηκαν το 1982 από τον John Hopfield και αποτελούνται από ένα ή περισσότερα πλήρως συνδεδεμένα δομικά στοιχεία [35].



Εικόνα 15. Ένα μονοεπίπεδο δίκτυο Hopfield.

### 2.6.5 Συνελκτικά Νευρωνικά Δίκτυα

Τα συνελκτικά νευρωνικά δίκτυα (convolutional neural networks - CNNs) είναι μια κατηγορία εις βάθος νευρωνικών δικτύων πρόσθιας διάδοσης επηρεασμένα από το ανθρώπινο σύστημα όρασης. Χρησιμοποιούνται ευρέως στον κλάδο της υπολογιστικής όρασης για το διαχωρισμό αντικειμένων και την αναγνώριση χαρακτήρων. Ως αποτέλεσμα οι είσοδοι τους συνήθως είναι εικονοστοιχεία μιας εικόνας. Οι είσοδοι περνάν από πολλαπλά επίπεδα τεχνητών νευρώνων το καθένα εκ των οποίων εκτελεί την πράξη της συνέλιξης, το αποτέλεσμα της οποίας αποτελεί την είσοδο για το επόμενο επίπεδο. Αυτό έχει σαν αποτέλεσμα το δίκτυο να εξάγει και να μαθαίνει χαρακτηριστικά της εικόνας σε κάθε επίπεδο (π.χ. ύψος, βάθος κ.λπ.). Αφού τελειώσει η διαδικασία της μάθησης, το CNN είναι σε θέση να ταξινομήσει εικόνες σε κατηγορίες χρησιμοποιώντας χαρακτηριστικά τα οποία αντιστοιχεί σε αυτά που έχει μάθει (classification) [36].



Εικόνα 16. Μάθηση χαρακτηριστικών και ταξινόμηση χειρόγραφων αριθμών με τη χρήση συνελκτικού νευρωνικού δικτύου.

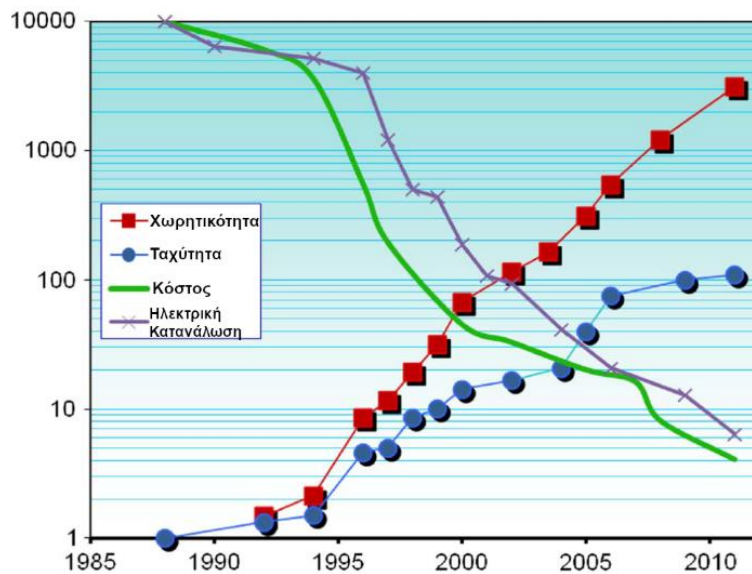
## **3 Νευρομορφικά Κυκλώματα σε Αναδιατασσόμενη Τεχνολογία**

### **3.1 Εισαγωγή**

Όπως αναφέρθηκε στην εισαγωγή της παρούσας πτυχιακής εργασίας, «νευρομορφικά» αποκαλούνται τα υπολογιστικά συστήματα που μιμούνται τη βιολογική λειτουργικότητα. Στις δεκαετίες που ακολούθησαν την πρόταση του όρου από τον Carver Mead το 1990, έχουν προταθεί πολλές υλοποιήσεις αυτών, οι οποίες κάνουν χρήση της αναδιατασσόμενης λογικής. Το βασικό όφελος αυτής της τεχνικής σχεδίασης είναι η ισορροπία μεταξύ υψηλών επιδόσεων και ευελιξίας. Για παράδειγμα, οι δημοφιλείς πλατφόρμες ανάπτυξης FPGA μπορούν να ανταγωνιστούν σε επιδόσεις τα κλασικά ολοκληρωμένα κυκλώματα, παρέχοντας επίσης τη δυνατότητα για επανακαθορισμό της λειτουργικότητας τους ακόμα και αφότου απομακρυνθούν από το εργοστάσιο παραγωγής. Παρά το γεγονός ότι τα ολοκληρωμένα κυκλώματα ειδικών εφαρμογών (Application-Specific Integrated Circuits - ASIC) μπορούν να ξεπεράσουν σε επιδόσεις τα FPGA, δεν παρέχουν τη δυνατότητα επανακαθορισμού του υλικού και κατά κανόνα είναι πιο ακριβά στην παραγωγή.

Πέρα από τα προαναφερθέντα πλεονεκτήματα, τα κυκλώματα αναδιατασσόμενης τεχνολογίας προσφέρουν μεγάλη παραλληλία στην επεξεργασία δεδομένων, είναι άμεσα διαθέσιμα και προσφέρουν ανεξαρτησία από τις εταιρείες-παραγωγούς μετά την αγορά αφού μπορούν να επαναπρογραμματιστούν για τις ανάγκες μίας νέας εφαρμογής [37]. Ως εκ τούτου, γίνεται φανερό ότι η δημοφιλία τους είναι δικαιολογημένη όχι μόνο στον τομέα της νευρομορφικής υπολογιστικής αλλά και σε οποιονδήποτε μπορεί να επωφεληθεί από μεγάλη σχεδιαστική ευελιξία. Στην Εικόνα 17 φαίνεται η εξέλιξη των χαρακτηριστικών των FPGA της εταιρίας Xilinx από το 1988 μέχρι το 2015.





Εικόνα 17. Χαρακτηριστικά των πλακέτων Xilinx FPGA από το 1988 μέχρι το 2015. Η χωρητικότητα είναι ο αριθμός λογικών μπλοκ ανά πλακέτα, η ταχύτητα αφορά την εκτέλεση της ίδιας συνάρτησης σε κάθε πλακέτα, το κόστος και η ηλεκτρική κατανάλωση δίνονται ανά λογικό μπλοκ πολ/μένες με το 10000.

### 3.2 Νευρομορφικές Υλοποιήσεις σε Υλικό

Οι νευρομορφικές υλοποιήσεις σε υλικό που έχουν παρουσιαστεί από τους ερευνητές ανάλογα με την πλατφόρμα που επιλέγεται μπορούν να χωριστούν σε τρεις μεγάλες κατηγορίες: στις αναλογικές, στις ψηφιακές και στις υβριδικές, όπου οι υβριδικές συνδυάζουν στοιχεία των άλλων δύο.

Οι διαφορές των αναλογικών και των ψηφιακών συστημάτων σχετίζονται με τα δεδομένα που διαχειρίζονται και τον τρόπο που συγχρονίζονται. Τα αναλογικά συστήματα χρησιμοποιούν συνεχείς τιμές και είναι ασύγχρονα, με τη λειτουργία τους να καθορίζεται από συμβάντα (event-driven) και όχι από ένα ρολόι. Από την άλλη πλευρά, επειδή τα ψηφιακά συστήματα αποτελούνται από κυκλώματα που λειτουργούν με βάση την δυϊκή άλγεβρα Boole, τα δεδομένα που διαχειρίζονται είναι εξ ορισμού διακριτά. Επίσης, τα ψηφιακά συστήματα είναι σύγχρονα καθώς η λειτουργία τους καθορίζεται από τους κύκλους ενός ρολογιού [38]. Όσον αφορά τις νευρομορφικές υλοποιήσεις συγκεκριμένα, τα πρώτα έχουν το πλεονέκτημα ότι είναι πιο βιολογικά αληθοφανή ακριβώς επειδή διαχειρίζονται συνεχείς τιμές όπως και ο εγκέφαλος. Παρόλα αυτά, η χρησιμότητα τους μπορεί να περιοριστεί από τον αναπόφευκτα αυξημένο θόρυβο σε σύγκριση με τα ψηφιακά. Τα υβριδικά συστήματα αναπτύχθηκαν ως μία μέση λύση μεταξύ της ακρίβειας των ψηφιακών και τα συνεχή δεδομένα των αναλογικών συστημάτων.

Οι υλοποιήσεις που εμπίπτουν σε μία από τις τρεις προαναφερθείσες κατηγορίες μπορούν επίσης να διαχωριστούν με βάση το υλικό που χρησιμοποιούν. Για παράδειγμα, κάποιες υλοποιήσεις βασίζονται σε ολοκληρωμένα κυκλώματα ειδικών εφαρμογών (ASIC) ενώ άλλες κάνουν χρήση των πλακετών ανάπτυξης τεχνολογίας FPGA οι οποίες θα αναλυθούν σε επόμενη ενότητα. Τα κυκλώματα ASIC είναι σχεδιασμένα για μία συγκεκριμένη εφαρμογή και ως αποτέλεσμα είναι ταχύτατα σε αυτή τη δουλειά. Πολλές φορές αποτελούνται από πολύ μεγάλης κλίμακας ολοκληρωμένα κυκλώματα (Very Large Scale Integration - VLSI). Όπως προαναφέρθηκε, το αυξημένο κόστος παραγωγής είναι το βασικό αρνητικό τους [39]. Οι πλακέτες FPGA (Field Programmable Gate Arrays) αποτελούν πλατφόρμες ανάπτυξης οι οποίες επιτρέπουν την επιτόπια αλλαγή του υλικού και ως εκ τούτου προσφέρουν πολύ μεγάλη ευελιξία.

### **3.2.1 Αναλογικά Κυκλώματα**

Οι αναλογικές υλοποιήσεις νευρομορφικών κυκλωμάτων συνήθως βασίζονται είτε σε αναλογικά VLSI ή σε επιτόπια προγραμματιζόμενες αναλογικές διατάξεις (Field Programmable Analog Arrays – FPAA).

Ο σκοπός των αναλογικών VLSI και των ASIC τα οποία απαρτίζουν, όπως όλων των νευρομορφικών κυκλωμάτων, είναι η μίμηση συγκεκριμένων λειτουργιών του εγκεφάλου σε βιολογικά εύλογο χρόνο. Για παράδειγμα, οι Koickal, Hamilton και συνεργάτες σχεδίασαν ένα νευρομορφικό αναλογικό VLSI κύκλωμα για την προσομοίωση της όσφρησης το οποίο επεξεργάζονταν τις εισόδους σε χρονικό πλαίσιο των 300 ms [40]. Σημειώνεται ότι έχει παρατηρηθεί πως η ανθρώπινη όσφρηση δρα στο πλαίσιο των 500 ms [41]. Παρόλα αυτά, τα αναλογικά VLSI μπορεί να μην λειτουργούν πάντα στο επιθυμητό χρονικό πλαίσιο ή να είναι πολύ ογκώδη.

Όπως φαίνεται από το όνομα τους, οι πλακέτες ανάπτυξης που βασίζονται στην τεχνολογία FPAA χαρακτηρίζονται και αυτά από αναδιατασσόμενη λογική. Αποτελούνται από αναλογικά λειτουργικά blocks τα οποία ενώνονται με ειδικούς δίαυλους. Για τις ανάγκες των νευρομορφικών υλοποιήσεων, οι πλακέτες FPAA αποτελούν τη βάση για πιο εξειδικευμένα κυκλώματα, όπως τα κυκλώματα τεχνολογίας FPNA και NeuroFPAA. Οι πλακέτες ανάπτυξης με τεχνολογία FPNA (Field Programmable Neural Array) είναι, κατ' ουσίαν, ένα αναλογικό κύκλωμα το οποίο μπορεί να χρησιμοποιηθεί για την προσομοίωση σύνθετων νευρωνικών κυττάρων ή πολλαπλών λιγότερο σύνθετων κυττάρων. Διατηρεί τη δυνατότητα επαναπροσδιορισμού των FPGA, αλλά αποτελείται από βιολογικά αληθοφανή αναλογικά εξαρτήματα όπως κανάλια ιόντων, δενδρίτες και συνάψεις [42]. Η NeuroFPAA είναι μια αρχιτεκτονική βασισμένη στα FPAA η οποία σχεδιάστηκε ως πλατφόρμα για την υλοποίηση σύνθετων νευρομορφικών κυκλωμάτων συνδυάζοντας μεγάλη πυκνότητα και χαμηλή κατανάλωση ενέργειας [43].

### **3.2.2 Ψηφιακά Κυκλώματα**

Στις ψηφιακές υλοποιήσεις νευρομορφικών κυκλωμάτων γίνεται πιο συχνή χρήση της τεχνολογίας FPGA. Τα κυκλώματα αυτά αποτελούνται από δυναμικές ή στατικές RAM, καταχωρητές, αθροιστές, πολλαπλασιαστές κ.λπ. Ενδεικτικά, ακολουθούν κάποιες νευρομορφικές υλοποιήσεις που κάνουν χρήση των FPGA.

Οι Ambroise, Levi, Joucla, Yvert και Saïghi υλοποίησαν μία Κεντρική Γεννήτρια Μοτίβων σε FPGA βασισμένη στο νευρωνικό μοντέλο Izhikevich. Για την υλοποίηση των νευρώνων σε αυτό το σύστημα, χρησιμοποιήθηκαν αθροιστές, πολλαπλασιαστές, συγκριτές και πολυπλέκτες. Για την υλοποίηση των σύναψεων, χρησιμοποιήθηκε μνήμη RAM στην οποία αποθηκεύτηκαν τα συναπτικά βάρη, ο παράγοντας ύφεσης και το ποσοστό διάλυσης [44].

Οι Hong, Kang και Chung παρουσίασαν ένα συνελκτικό νευρωνικό δίκτυο υλοποιημένο σε FPGA. Σε αυτό, οι συνάψεις αποτυπώνονται ως καταχωρητές στους οποίους αποθηκεύεται το συναπτικό βάρος. Αποτελούνται από θύρες AND, πλήρεις αθροιστές και flip-flops. Οι νευρώνες αποτυπώνονται ως σειριακοί αθροιστές. Αυτή η υλοποίηση δοκιμάστηκε στην ταξινόμηση του σετ δεδομένων χειρόγραφων αριθμών MNIST με ποσοστό ακρίβειας 99.37% και κατανάλωση ενέργειας 93μJ ανά εικόνα [45].

### **3.2.3 Υβριδικά Κυκλώματα**

Οι υβριδικές υλοποιήσεις νευρομορφικών κυκλωμάτων αναπτύχθηκαν για να συνδυάσουν τα πλεονεκτήματα των αναλογικών και ψηφιακών συστημάτων, μειώνοντας ταυτόχρονα τα μειονεκτήματά τους. Ακολουθεί μία υλοποίηση που κάνει χρήση αναλογικών και ψηφιακών εξαρτημάτων.

Οι Dalgaty, Payvand, Moro, Ly, Rebay-Peyroula και συνεργάτες παρουσίασαν μία σειρά νευρομορφικών κυκλωμάτων τα οποία συνδυάζουν ψηφιακή μνήμη RAM με αναλογικά εξαρτήματα για χρήση σε μαζικά παραλληλοποιημένες εφαρμογές, όπως τα επαναληπτικά νευρωνικά δίκτυα. Με αυτήν την τεχνική κατάφεραν να υλοποιήσουν ένα τέτοιο δίκτυο το οποίο παρουσίαζε πλαστικότητα πιο αποτελεσματικά από προηγούμενες εφαρμογές [46].

## **3.3 Σχετικά Projects**

Ένας από τους στόχους της επιστημονικής κοινότητας όσον αφορά τα νευρομορφικά κυκλώματα είναι η προσομοίωση του βιολογικού εγκεφάλου. Η προσπάθεια αυτή έχει οδηγήσει στην δημιουργία διαφόρων φορέων και υπερυπολογιστών, κάποιιοι από τους οποίους παρουσιάζονται παρακάτω.

### 3.3.1 *Human Brain Project*

Το Human Brain Project ξεκίνησε το 2013 με τη συνεργασία περισσότερων από 500 ερευνητών από 140 πανεπιστήμια, ερευνητικά κέντρα και νοσοκομεία και τη συγχρηματοδότηση της Ευρωπαϊκής Ένωσης. Ο αρχικός του στόχος να προσομοιώσει τον ανθρώπινο εγκέφαλο με τη χρήση υπερυπολογιστών [47] έχει διευρυνθεί για να συμπεριλάβει 11 υποπρογράμματα και δύο άλλα projects υπερυπολογιστών, ονόματι τα BrainScaleS και SpiNNaker [48].

Τα συμπεριλαμβανόμενα υποπρογράμματα είναι τα εξής [49]:

- Mouse Brain Organisation
- Human Brain Organisation
- Systems and Cognitive Neuroscience
- Theoretical Neuroscience
- Neuroinformatics Platform
- Brain Simulation Platform
- High-Performance Analytics & Computing Platform
- Medical Informatics Platform
- Neuromorphic Computing Platform
- Neurorobotics
- Ethics and Society

Πρόσβαση στους BrainScaleS και SpiNNaker για ερευνητική χρήση προσφέρεται εξ αποστάσεως μέσω της πλατφόρμας EBRAINS.

### 3.3.2 *BrainScaleS*

Το BrainScaleS (Brain-inspired multiscale computation in neuromorphic hybrid systems) είναι ένα ερευνητικό πρόγραμμα που ξεκίνησε στη Χαϊδελβέργη της Γερμανίας τον Ιανουάριο 2011 με τη σύμπραξη της Ευρωπαϊκής Ένωσης και 19 ερευνητικών κέντρων. Στόχος του είναι η κατανόηση και προσομοίωση λειτουργιών και αλληλεπιδράσεων του εγκεφάλου. Η αρχική ανάπτυξη ολοκληρώθηκε τον Μάρτιο 2015 και έκτοτε το project έχει ενταχθεί ως υποπρόγραμμα του Human Brain Project [50].

Ο παραγόμενος υπερυπολογιστής αποτελείται από 3.9 εκατομμύρια νευρώνες και 880 εκατομμύρια συνάψεις. Η τεχνολογία που ακολουθείται είναι αυτή του υβριδικού ASIC και ο συντελεστής εξέλιξης του μοντέλου που υλοποιείται μπορεί να είναι από  $10^3$  μέχρι  $10^5$  φορές πιο μεγάλος από τον βιολογικό χρόνο [51].

### 3.3.3 *SpiNNaker*

Η αρχιτεκτονική SpiNNaker (Spiking Neural Network Architecture) αναπτύχθηκε από ομάδα επιστημόνων στο Πανεπιστήμιο του Manchester βασισμένο στον επεξεργαστή ARM 968. Η βασική υλοποίηση της αρχιτεκτονικής είναι το SpiNNaker κύκλωμα το οποίο στη συνέχεια εντάχθηκε σε έναν υπερυπολογιστή με 50.000 τέτοια κυκλώματα και τελικά έκανε δυνατή τη προσομοίωση ενός δικτύου με ένα δισεκατομμύριο νευρώνες σε πραγματικό χρόνο.

Μέχρι σήμερα και υπό την αιγίδα του Human Brain Project η υλοποίηση έχει ξεπεράσει τους ένα εκατομμύριο πυρήνες και έχει ξεκινήσει η ανάπτυξη του SpiNNaker 2 στη Δρέσδη της Γερμανίας [52].

## 4 Μελέτη Περίπτωσης: Μοντέλο Izhikevich και Παράγωγα

### 4.1 Εισαγωγή

Όπως αναφέρθηκε στην ενότητα 2.4, το μοντέλο Izhikevich αποτελεί ένα από τα πλέον δημοφιλή βιολογικά εμπνευσμένα νευρωνικά μοντέλα. Παρουσιάστηκε για πρώτη φορά το 2003 από το μαθηματικό Eugene Izhikevich. Αυτή η δουλειά είναι επηρεασμένη από τα μοντέλα Hodgkin-Huxley και Integrate and Fire, συνδυάζοντας την αληθοφάνεια των δυναμικών του πρώτου με την αποδοτικότητα του δεύτερου [53]. Αξίζει να σημειωθεί ότι το μοντέλο Izhikevich είναι καθαρά νευρωνικό, δηλαδή ασχολείται μόνο με την παραγωγή και ενημέρωση των μεμβρανικών τάσεων ενός νευρώνα βάσει ενός προκαθορισμένου συναπτικού ρεύματος. Παράγωγες υλοποιήσεις επεκτείνουν τη λειτουργία του μοντέλου ώστε να συμπεριλάβουν το μηχανισμό με τον οποίο οι προ- και μετά- συναπτικοί νευρώνες επικοινωνούν, περιγράφοντας δηλαδή μία τεχνητή σύναψη.

### 4.2 Το αρχικό μοντέλο

Το μοντέλο που πρότεινε ο Izhikevich περιγράφεται από ένα διδιάστατο σύστημα συνήθων διαφορικών εξισώσεων:

$$v' = 0.04v^2 + 5v + 140 - u + I \quad (27)$$

$$u' = a(bv - u) \quad (28)$$

Επίσης ισχύει η εξής συνθήκη επαναφοράς μετά από την παραγωγή ακίδας:

$$\text{Εάν } v \geq 30mV, \text{ τότε } \begin{cases} v \leftarrow c \\ u \leftarrow u + d \end{cases} \quad (29)$$

Στα παραπάνω, τα  $u$  και  $v$  είναι αδιάστατες μεταβλητές και τα  $a$ ,  $b$ ,  $c$ ,  $d$  είναι αδιάστατες παράμετροι. Όσον αφορά τις μεταβλητές, το  $v$  είναι το μεμβρανικό δυναμικό, το  $u$  είναι μια μεταβλητή ανάκτησης της μεμβράνης και το  $I$  είναι το συναπτικό ρεύμα. Σχετικά με τις παραμέτρους, το  $a$  καθορίζει την χρονική κλίμακα του  $u$ , το  $b$  περιγράφει την ευαισθησία αλλαγής του  $u$  εξαιτίας ταλαντώσεων του μεμβρανικού δυναμικού  $v$  και τα  $c$  και  $d$  καθορίζουν την τιμή του  $v$  και του  $u$  μετά

την πυροδότηση αντίστοιχα. Διαφορετικά σεί παραμέτρων οδηγούν σε διαφορετικά μοτίβα πυροδότησης.

Στην εργασία που παρουσιάζει το μοντέλο του [53], ο Izhikevich απαριθμεί διεγερτικούς και ανασταλτικούς νευρώνες του εγκεφαλικού φλοιού και τους ταξινομεί με βάση το μοτίβο πυροδότησης τους. Οι διεγερτικοί νευρώνες χωρίζονται στις εξής τρεις κατηγορίες:

- Νευρώνες Τυπικής Απόκρισης Ακίδας (Regular Spiking - RS): Αυτοί είναι οι πιο συνηθισμένοι νευρώνες του φλοιού. Σε αυτούς, ένα παρατεταμένο ερέθισμα πρώτα προκαλεί κάποιες ακμές με μικρή περίοδο, η οποία στη συνέχεια μεγαλώνει. Ένα πιο δυνατό ερέθισμα μπορεί να αυξήσει τη συχνότητα των πυροδοτήσεων μέχρι ένα άνω όριο. Η προσομοίωση αυτού του νευρώνα επιτυγχάνεται με τις παραμετρικές τιμές  $c = -65 \text{ mV}$  και  $d = 8$ .
- Νευρώνες Εγγενούς Έξαρσης (Intrinsically Bursting - IB): Αυτοί οι νευρώνες παράγουν μία έξαρση ακίδων η οποία ακολουθείται από επαναλαμβανόμενες μονές ακίδες. Στην αρχική έξαρση η μεταβλητή  $u$  αυξάνεται προκαλώντας την αλλαγή από έξαρση σε πυροδότηση. Οι παραμετρικές τιμές είναι οι  $c = -55 \text{ mV}$  και  $d = 4$ .
- Chattering – CH Νευρώνες: Οι νευρώνες αυτοί παράγουν εξάρσεις κοντινών ακίδων με τη συχνότητα μεταξύ των εξάρσεων να φτάνει τα 40 Hz. Οι παραμετρικές τιμές είναι οι  $c = -50 \text{ mV}$  και  $d = 2$ .

Οι ανασταλτικοί νευρώνες χωρίζονται σε δύο κατηγορίες:

- Νευρώνες Γρήγορης Απόκρισης Ακίδας (Fast Spiking - FS): Αυτοί οι νευρώνες παράγουν σειρές από ακίδες με πολύ μεγάλη συχνότητα η οποία δε μειώνεται. Η παραμετρική τιμή  $a$  ισούται με 0.1.
- Νευρώνες Απόκρισης Ακίδας Χαμηλής Κατωφλίσωσης (Low-Threshold Spiking - LTS): Οι νευρώνες αυτοί παράγουν σειρές από ακίδες με υψηλή συχνότητα όπως οι FS, υπάρχει όμως αισθητή προσαρμογή και μείωση της συχνότητας. Η παραμετρική τιμή  $b$  ισούται με 0.25.

### 4.3 Άλλες Υλοποιήσεις

Το αρχικό μοντέλο Izhikevich αποτέλεσε τη βάση για πολλές άλλες υλοποιήσεις οι οποίες είτε τροποποιούν τις εξισώσεις για μεγαλύτερη αποδοτικότητα ή επεκτείνουν το μοντέλο για αυξημένη λειτουργικότητα. Σε αυτήν την ενότητα παρουσιάζονται σε χρονολογική σειρά πέντε από αυτές τις υλοποιήσεις οι οποίες με τη σειρά τους αποτέλεσαν πηγή έμπνευσης για τους ερευνητές του τομέα.

Το 2008 οι Andrew Cassidy και Ανδρέας Ανδρέου παρουσίασαν ένα νευρωνικό δίκτυο βασισμένο σε FPGA το οποίο αποτελείται από 32 φυσικούς νευρώνες, καθένας εκ των οποίων πολυπλέκει τη λειτουργία 8 εικονικών νευρώνων [54]. Οι νευρώνες υλοποιούν μια τροποποιημένη εκδοχή των εξισώσεων Izhikevich.

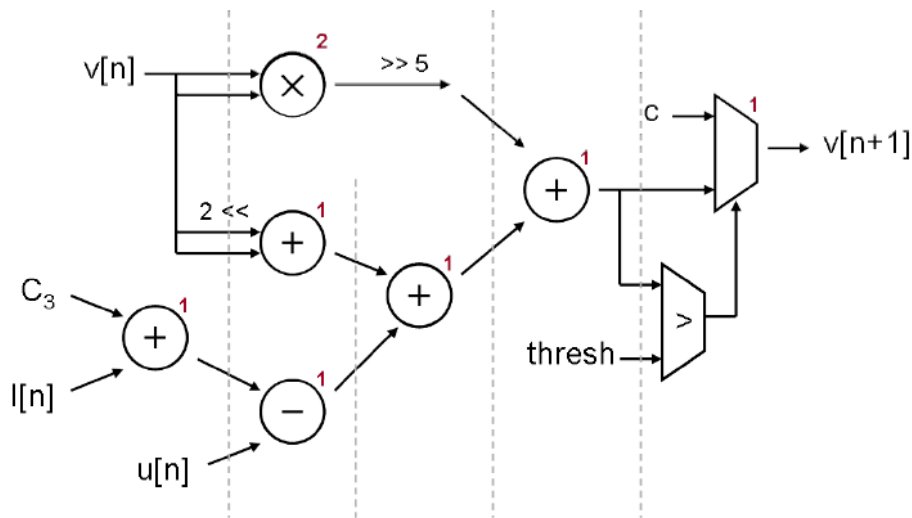
Αρχικά για λόγους συμβατότητας με το υλικό οι αριθμητικές τιμές αναπαρίστανται σε 18 bit, με τα 10 από αυτά να χρησιμοποιούνται για το ακέραιο κομμάτι και τα 8 για το δεκαδικό. Επίσης, για να επιτευχθεί αριθμητική με δυνάμεις του 2 στον πολλαπλασιασμό και τη διαίρεση, οι παράμετροι της (27) πολλαπλασιάστηκαν με το 0.78125:

$$v' = v + \frac{1}{32}v^2 + 4v + 109.375 - u + I \quad (30)$$

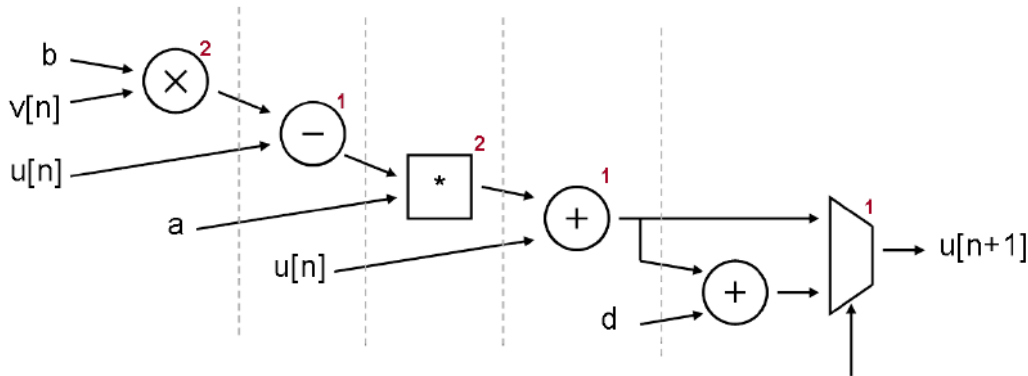
$$u' = u + a(bv - u) \quad (31)$$

Οι τιμές  $u$  και  $v$  υπολογίζονται μετά από 5 στάδια διασωλήνωσης τα οποία φαίνονται στην εικόνα 18.





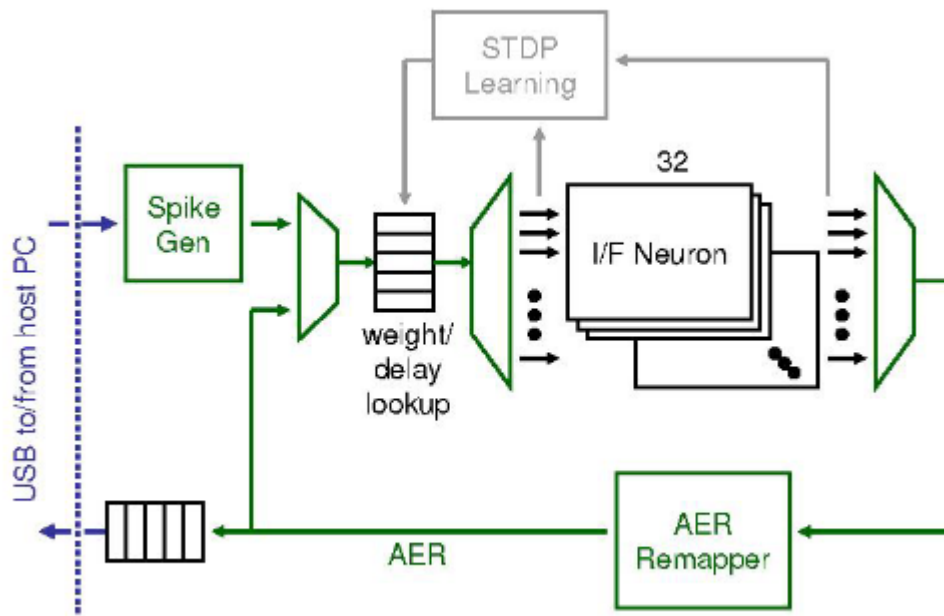
(a) 'v' pipeline



(b) 'u' pipeline

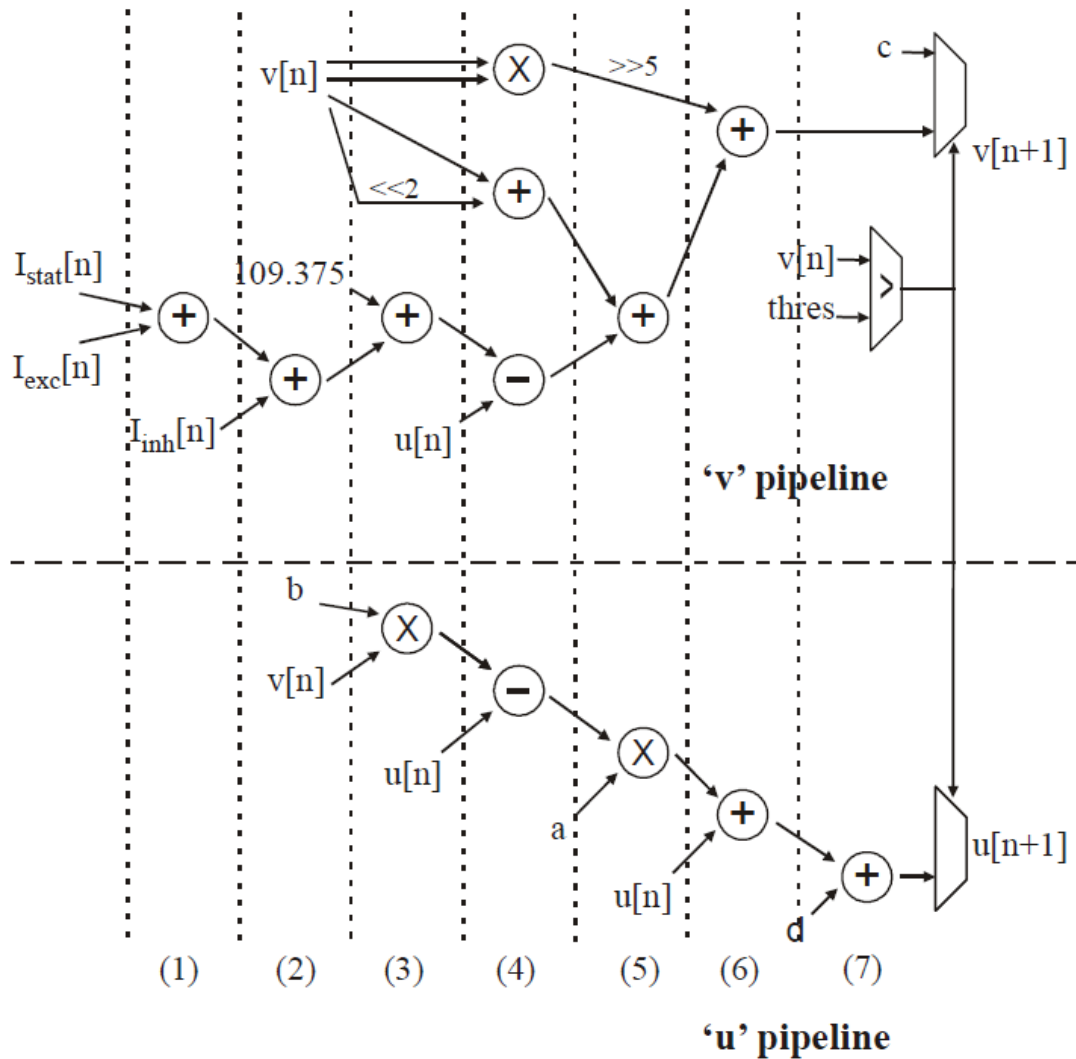
Εικόνα 18. Τα στάδια διασώληνωσης στους νευρώνες του δικτύου των Cassidy και Ανδρέου.

Όσον αφορά το συναπτικό κομμάτι του δικτύου, επαναχρησιμοποιείται η γενική αρχιτεκτονική που είχαν παρουσιάσει οι Cassidy, Denham και συνεργάτες για χρήση σε Leaky Integrate and Fire δίκτυο [55]. Συγκεκριμένα, οι συναπτικές εισοδοι των νευρώνων του δικτύου εξαρτώνται από τον ρυθμό παραγωγής αποκρίσεων ακίδας από αυτούς μέσω του μηχανισμού STDP. Το μόνο που αλλάζει στο Izhikevich δίκτυο είναι το νευρωνικό μοντέλο. Στην εικόνα 19 φαίνεται η αρχική αρχιτεκτονική.



Εικόνα 19. Η αρχιτεκτονική του LIF δικτύου των Cassidy, Denham κ.α. στο οποίο βασίστηκε το Izhikevich δίκτυο των Cassidy και Ανδρέου. Με γκρι επισημαίνεται το συναπτικό κομμάτι της αρχιτεκτονικής.

Το 2013 οι Ambroise, Levi, Bornat και Saïghi παρουσίασαν ένα spiking νευρωνικό δίκτυο αποτελούμενο από 117 νευρώνες Izhikevich σε αναδιατασσόμενη τεχνολογία [5]. Εκτός από το νευρωνικό μοντέλο, υλοποίησαν και ένα βιολογικά αληθοφάνες συναπτικό μοντέλο για τον ακριβή υπολογισμό των ρευμάτων που εισέρχονται σε κάθε νευρώνα. Για τον υπολογισμό των τιμών  $\mu$  και  $\nu$  κάθε νευρώνα χρησιμοποιήθηκαν αθροιστές, αφαιρέτες, ολισθητές, συγκριτές και πολυπλέκτες διατεταγμένοι όπως φαίνεται στην εικόνα 20 σε 7 στάδια διασωλήνωσης.



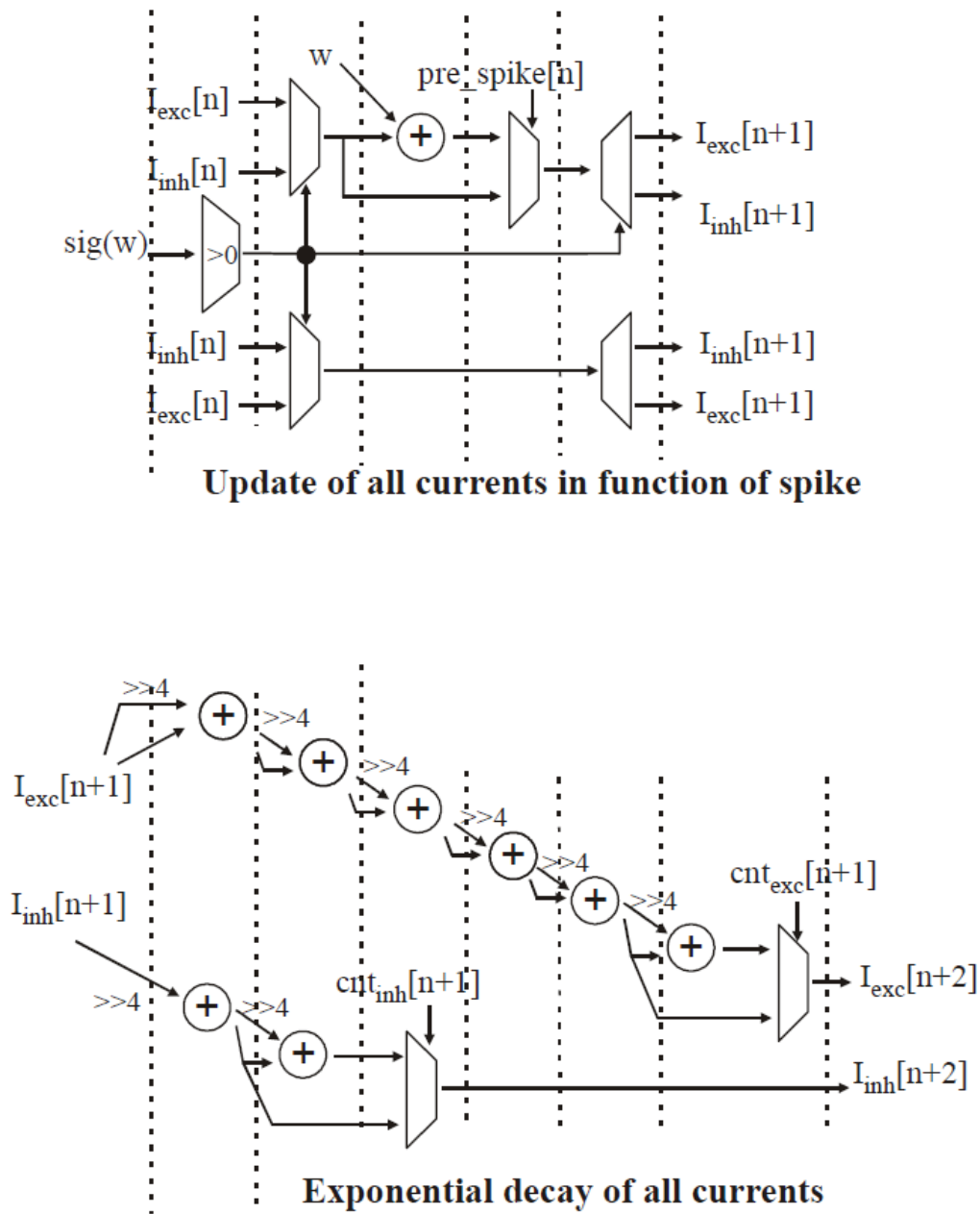
Εικόνα 20. Τα 7 στάδια διασωλήνωσης υπολογισμού των  $u$  και  $v$  στο paper του Ambroise και συνεργατών.

Σε αντίθεση με το μοντέλο Izhikevich, σε αυτήν την υλοποίηση το συναπτικό ρεύμα χωρίζεται σε τρία μέρη για μεγαλύτερη αληθοφάνεια:  $I_{stat}$ ,  $I_{exc}$  και  $I_{inh}$ , όπου το πρώτο είναι ένας στατικός παράγοντας πόλωσης, το δεύτερο είναι το ρεύμα που οφείλεται σε διεγερτικές συνάψεις και το τρίτο είναι το ρεύμα που οφείλεται σε ανασταλτικές συνάψεις. Μία άλλη προσθήκη που γίνεται στο αρχικό μοντέλο είναι ο υπολογισμός των τιμών των ρευμάτων  $I_{exc}$  και  $I_{inh}$  καθώς αυτές εκθετικά μειώνονται μετά την παραγωγή απόκρισης ακίδας στο νευρώνα. Για να επιτευχθεί αυτό, χρησιμοποιείται η μέθοδος της γραμμικής παλινδρόμησης που περιγράφεται από τον εξής τύπο:

$$I_{exp}(t + 1) = \left(1 - \frac{1}{\tau}\right) I_{exp}(t) \quad (32)$$

Όπου  $\tau$  είναι μια χρονική σταθερά με διαφορετική τιμή για το  $I_{exc}$  και το  $I_{inh}$ .

Συνολικά, τα συναπτικά ρεύματα υπολογίζονται σε δύο διαδοχικά βήματα. Το πρώτο από αυτά υπολογίζει και ενημερώνει τα ρεύματα που εισάγονται στους νευρώνες ανάλογα με το αν αυτοί παρήγαγαν απόκριση ακίδας. Το δεύτερο, μετά την παραγωγή απόκρισης ακίδας, υπολογίζει την εκθετική μείωση των ρευμάτων  $I_{exc}$  και  $I_{inh}$ , οι νέες τιμές των οποίων ανατροφοδοτούνται ως είσοδοι στο πρώτο κύκλωμα. Το πρώτο κύκλωμα αποτελείται από 5 στάδια διασωλήνωσης και το δεύτερο από 7, όπως φαίνεται στην εικόνα 21:



Εικόνα 21. Τα δύο pipelines για τον υπολογισμό των διεγερτικών και ανασταλτικών ρευμάτων και της εκθετικής μείωσής τους στο δίκτυο του Ambroise και συνεργατών.

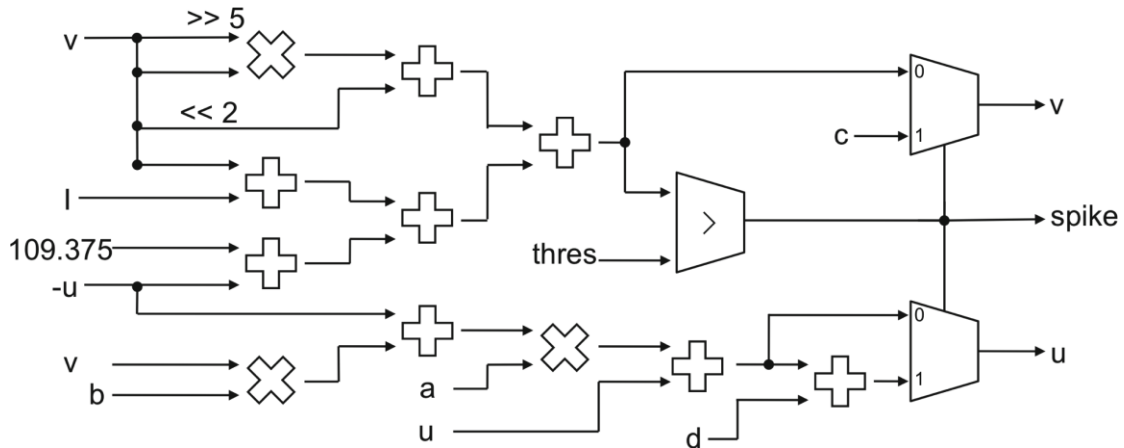
Το 2017 οι Vitor Bandeira, Vivianne L. Costa και συνεργάτες παρουσίασαν μία τροποποιημένη υλοποίηση του μοντέλου Izhikevich με στόχο την δημιουργία ενός εύκολα αναδιατασσόμενου δικτύου με χαμηλή καθυστέρηση [56]. Συγκεκριμένα, οι τιμές των  $u$  και  $I$  πολλαπλασιάζονται με το  $h = 0.78125$  και οι εξισώσεις (27), (28) και (29) παίρνουν την εξής μορφή:

$$v' = \frac{1}{32}v^2 + 3.90625v + 109.375 - uh + Ih \quad (33)$$

$$u' = ah(bhv - uh) \quad (34)$$

$$\text{Εάν } v \geq 30mV, \text{ τότε } \begin{cases} v \leftarrow c \\ uh \leftarrow uh + d \end{cases} \quad (35)$$

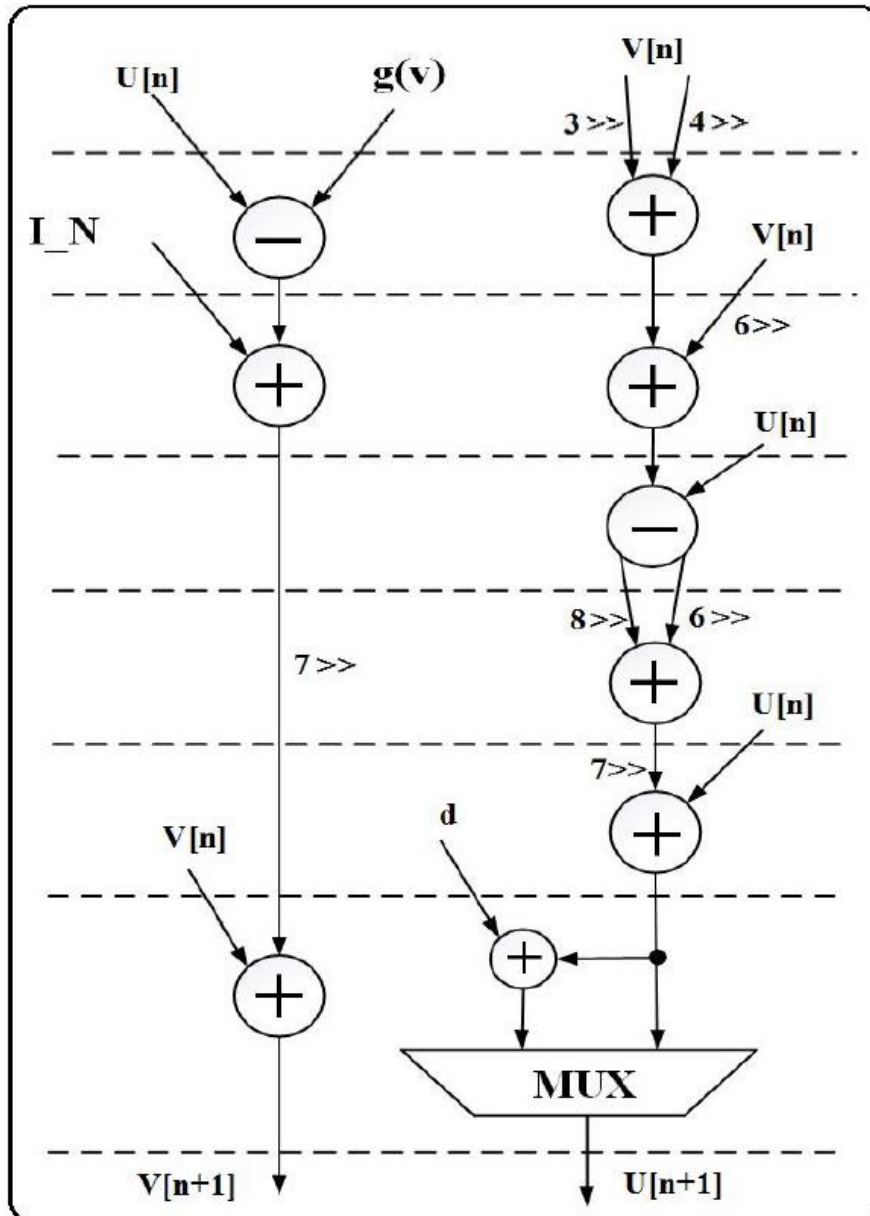
Ο υπολογισμός των  $u$  και  $v$  στο νευρώνα γίνεται στα εξής 6 στάδια διασώληνωσης:



Εικόνα 22. Τα στάδια διασώληνωσης στο νευρώνα του paper των Bandeira, Costa κ.α.

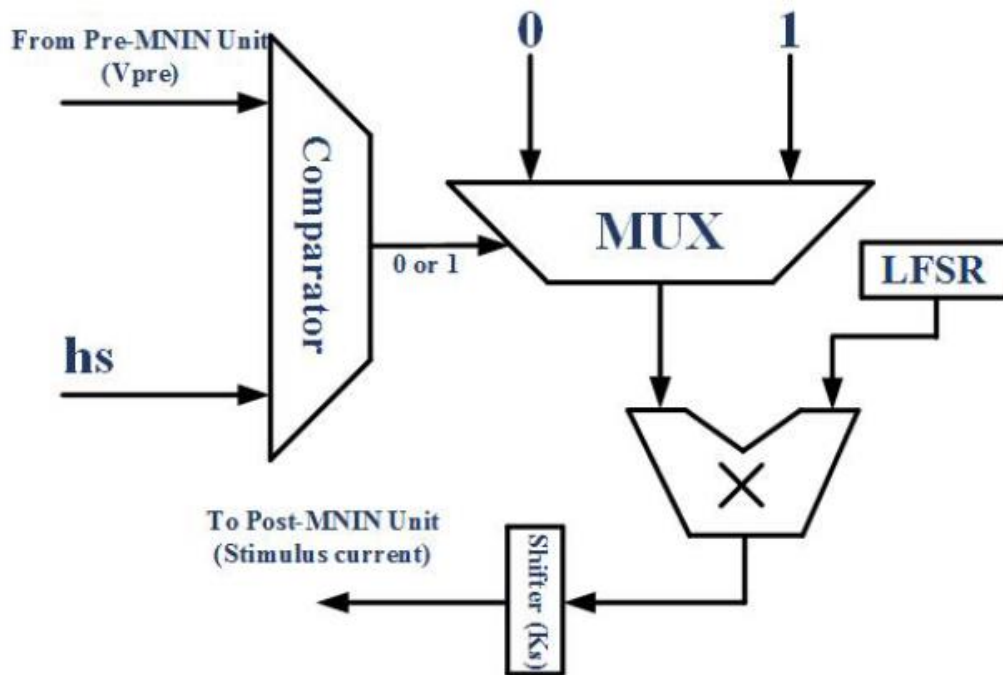
Οι νευρώνες οργανώνονται σε κυκλικό δίκτυο όπου κάθε νευρώνας συνδέεται μόνο με τους γείτονες του. Επίσης, τα ρεύματα θεωρούνται σταθερά και δεν ανανεώνονται, δεν υλοποιείται δηλαδή κάποιος συναπτικός μηχανισμός. Αυτό σημαίνει ότι πέρα από τους νευρώνες, το δίκτυο έχει μικρή βιολογική σημασία. Είναι παρόλα αυτά χρήσιμο ως εργαλείο για τον έλεγχο της επίδρασης των διαφορετικών παραμέτρων του μοντέλου Izhikevich στην παραγόμενη κυματομορφή.

Το 2018 οι Saeed Haghiri, Abdulhamid Zahedi και συνεργάτες παρουσίασαν μία παραλλαγή του μοντέλου Izhikevich η οποία υλοποιείται χωρίς τη χρήση πολλαπλασιαστών [57]. Συγκεκριμένα, οι πολλαπλασιασμοί έχουν αντικατασταθεί με ολισθήσεις και προσθέσεις και το μοντέλο έχει μετατραπεί σε θορυβώδες μοντέλο Izhikevich. Ο νευρώνας υλοποιείται σε 8 στάδια pipeline τα οποία φαίνονται στην εικόνα 23.



Εικόνα 23. Τα στάδια διασώληνωσης στο νευρόνα του paper των Haghiri, Zahedi κ.α.

Για την υλοποίηση της σύναψης χρησιμοποιείται ένας συγκριτής, ένας πολυπλέκτης, ένας πολλαπλασιαστής και ένας ολισθητής. Εκτός από το προσυναπτικό δυναμικό, στον υπολογισμό της μετασυναπτικής εξόδου συμμετέχει ένας γραμμικός ανατροφοδοτούμενος καταχωρητής ολίσθησης (Linear Feedback Shift Register) ο οποίος προσομοιώνει το θορυβώδες συναπτικό ρεύμα:



Εικόνα 24. Το συναπτικό κύκλωμα του paper των Haghiri, Zahedi κ.α.

Τέλος, το 2020 οι Alexander J. Leigh, Mitra Mirhassani και Roberto Muscedere παρουσίασαν άλλη μια τροποποιημένη μορφή του μοντέλου Izhikevich η οποία μπορεί να αναπαράγει όλες τις νευρωνικές συμπεριφορές της ενότητας 4.2 με μία μόνο παράμετρο-είσοδο [58]. Αυτή είναι άλλη μια υλοποίηση η οποία ασχολείται αποκλειστικά με το νευρωνικό κομμάτι και όχι το συναπτικό.

## 5 Η Προτεινόμενη Συναπτική Υλοποίηση

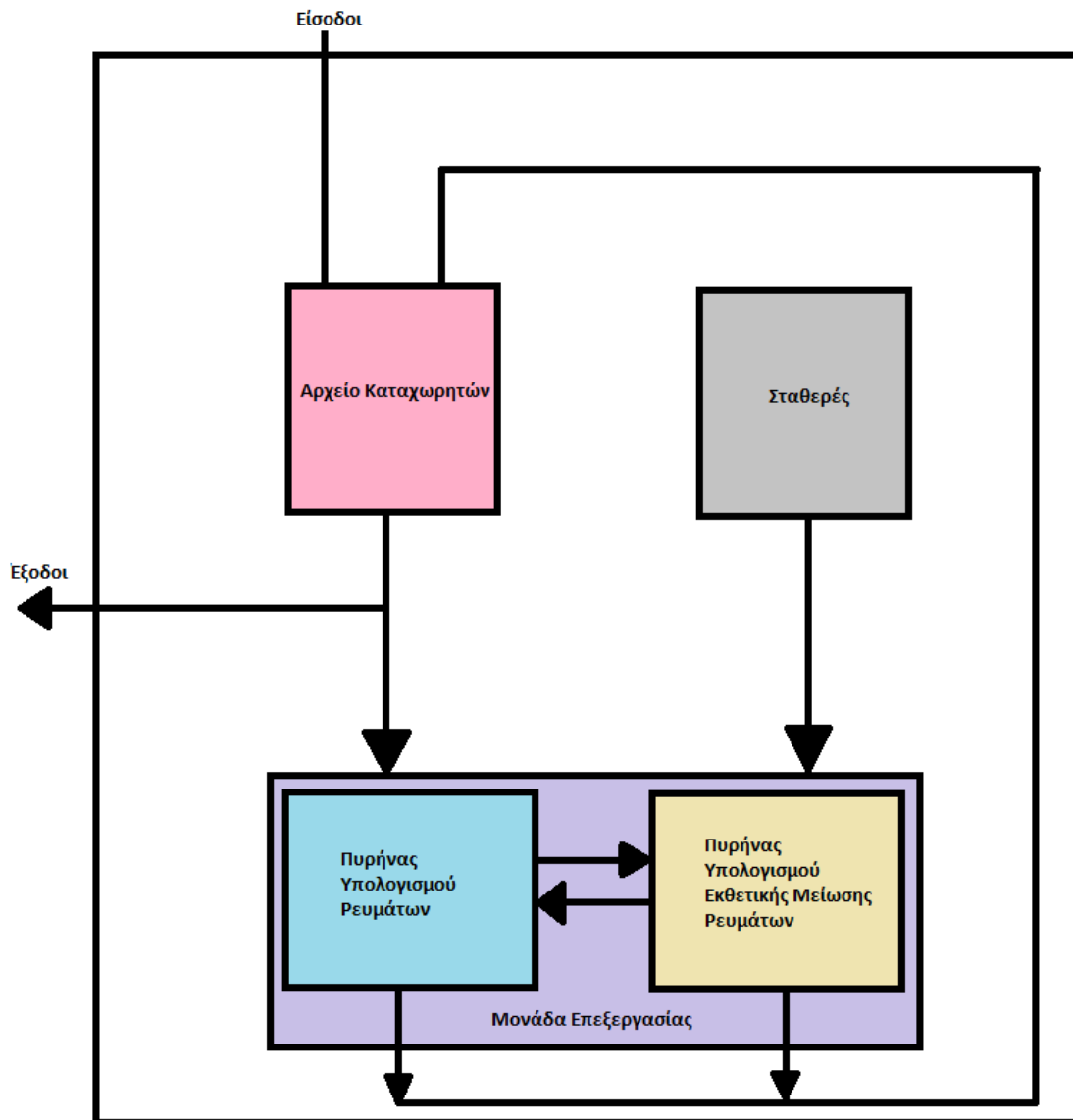
### 5.1 Εισαγωγή

Από όλα τα μοντέλα που παρουσιάστηκαν στην προηγούμενη ενότητα, αυτό με ίσως τις περισσότερες παραλλαγές και τροποποιήσεις είναι των Ambroise, Levi, Bornat και Saïghi. Μάλιστα, το νευρωνικό τμήμα του μοντέλου αποτέλεσε βάση για μία βελτιωμένη εκδοχή από την τότε προπτυχιακή φοιτήτρια του τμήματος Μαρία Σαπουνάκη [59]. Η παρούσα εργασία αποτελεί συνέχεια αυτής της υλοποίησης. Συγκεκριμένα προστίθεται ένας μηχανισμός για την επικοινωνία των νευρώνων, δηλαδή μία τεχνητή σύναψη.

### 5.2 Περιγραφή και Αρχιτεκτονική της Υλοποίησης

Η σύναψη βασίζεται σε μία απλοποιημένη εκδοχή της σύναψης του μοντέλου του Ambroise και συνεργατών. Αυτό σημαίνει ότι διατηρούνται τα δύο υπολογιστικά στάδια των current computation και exponential decay computation, αφαιρείται όμως η χρήση αραιών πινάκων για την αποθήκευση των συναπτικών βαρών. Αντιθέτως, αυτά αντλούνται ευθέως από ένα αρχείο καταχωρητών. Επίσης στο δεύτερο στάδιο αντί για μετρητές χρησιμοποιούνται προκαθορισμένες τιμές. Τέλος, για τεχνικούς λόγους, οι δύο αποπλέκτες του current computation αντικαθίστανται με πολυπλέκτες. Οι είσοδοι προέρχονται από τον προσυναπτικό νευρώνα και οι έξοδοι του πυρήνα υπολογισμού ρευμάτων εκτός από είσοδο στον πυρήνα υπολογισμού εκθετικής μείωσης αποτελούν και έξοδο της σύναψης και άρα είσοδο για τον μετασυναπτικό νευρώνα. Η γενική μορφή της αρχιτεκτονικής φαίνεται στην εικόνα 25:





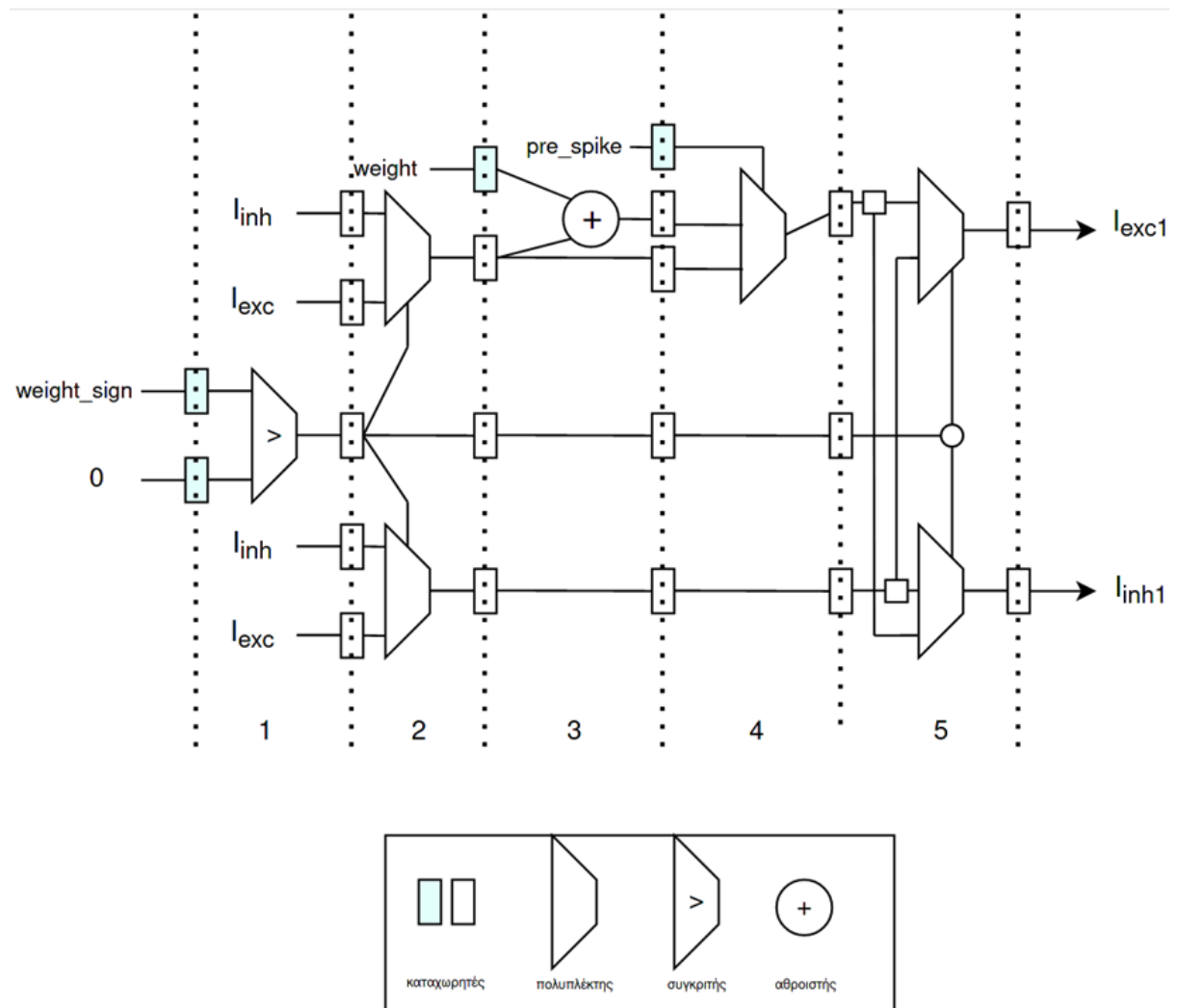
Εικόνα 25. Η προτεινόμενη αρχιτεκτονική.

### 5.3 Το Κύκλωμα

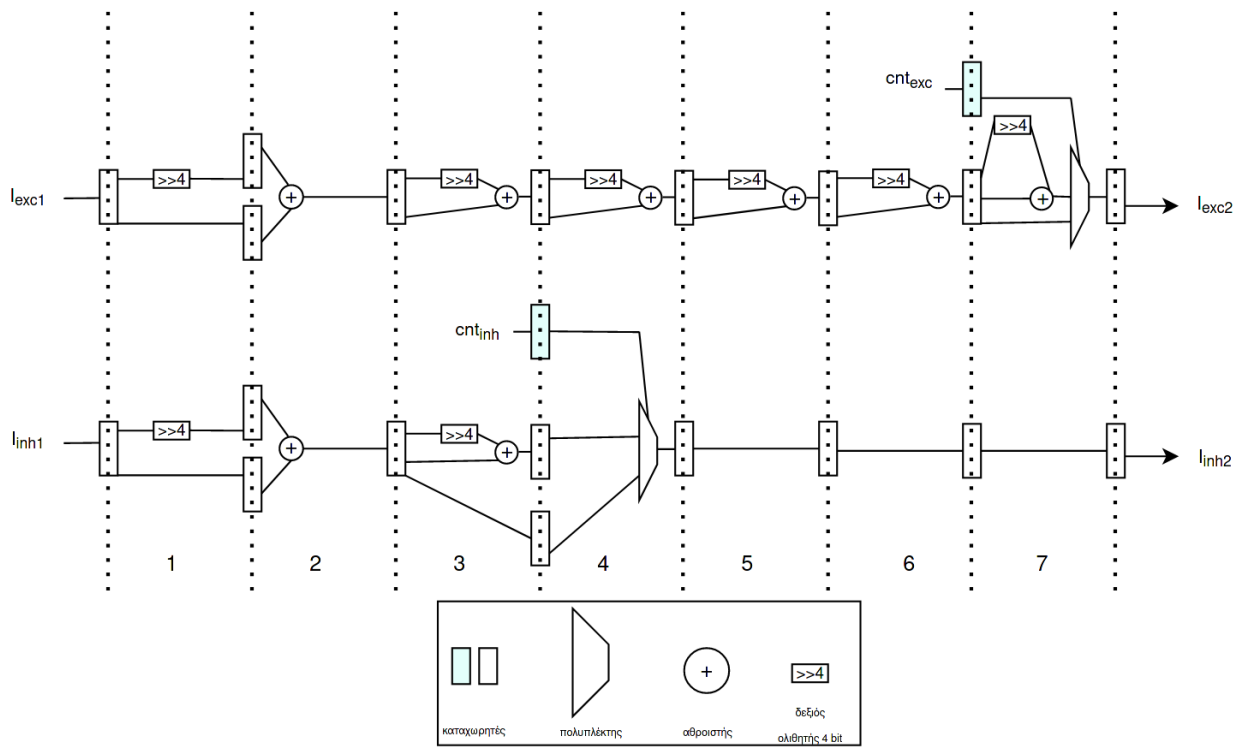
Επειδή ακριβώς η υλοποίηση ακολουθεί το κύκλωμα που παρουσίασαν ο Ambroise και συνεργάτες, αποτελείται από δύο υπολογιστικούς πυρήνες ο καθένας με τα δικά του στάδια διασωλήνωσης. Οι δύο αυτοί πυρήνες που ενσωματώνονται στη μονάδα επεξεργασίας λειτουργούν ο ένας διαδοχικά με τον άλλον ενώ και οι δύο παράγουν εξόδους απαραίτητες για τους νευρώνες. Ο πυρήνας υπολογισμού ρευμάτων παίρνει σαν είσοδο τα διεγερτικά και ανασταλτικά ρεύματα  $I_{exc}$  και  $I_{inh}$  (αρχικά ίσα με 0), το πρόσημο και την τιμή του βάρους της σύναψης, την πληροφορία του αν ο νευρώνας παρήγαγε απόκριση ακίδας ή όχι, ένα σήμα ρολογιού, ένα σήμα clear και ένα σήμα ready. Μετά από 5 στάδια διασωλήνωσης παράγεται η έξοδος των ανανεωμένων ρευμάτων  $I_{exc1}$  και  $I_{inh1}$ . Αυτά αποτελούν είσοδο για τον πυρήνα υπολογισμού εκθετικής μείωσης ρευμάτων αλλά και έξοδο της σύναψης, καθώς τροφοδοτούνται

στο μετασυναπτικό νευρώνα. Ο πυρήνας υπολογισμού εκθετικής μείωσης ρευμάτων παίρνει ως είσοδο τα προαναφερθέντα ρεύματα, τις προκαθορισμένες δυαδικές τιμές  $C_{nt_{exc}}$  και  $C_{nt_{inh}}$  και τα σήματα clock, clear και ready. Μετά από 7 στάδια διασωλήνωσης παράγονται οι έξοδοι  $I_{exc2}$  και  $I_{inh2}$ . Αυτές αποτελούν την επόμενη είσοδο του πυρήνα υπολογισμού ρευμάτων.

Συνολικά, το κύκλωμα κάνει χρήση 42 καταχωρητών: 21 στον πρώτο πυρήνα και 21 στο δεύτερο. Ο πρώτος πυρήνας περιλαμβάνει 1 συγκριτή, 5 πολυπλέκτες και 1 αθροιστή. Ο δεύτερος πυρήνας αποτελείται από 8 4 bit δεξιούς ολισθητές, 8 αθροιστές και 2 πολυπλέκτες. Οι ενδιάμεσες πράξεις είναι ίδιες με αυτές του μοντέλου του Ambroise και συνεργατών. Παρακάτω φαίνεται η προτεινόμενη σχεδίαση των δύο πυρήνων.



Εικόνα 26. Η προτεινόμενη σχεδίαση του πυρήνα υπολογισμού ρευμάτων.

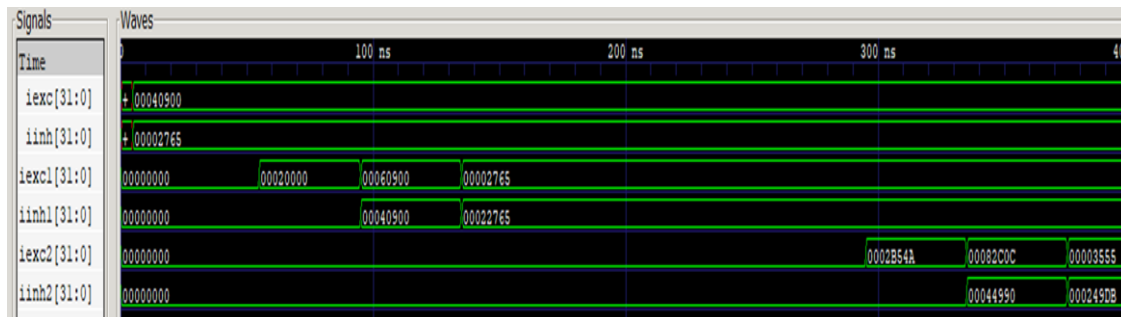


Εικόνα 27. Η προτεινόμενη σχεδίαση του πυρήνα υπολογισμού εκθετικής μείωσης ρευμάτων.

## 6 Αποτελέσματα Υλοποίησης

### 6.1 Έλεγχος Αποτελεσμάτων

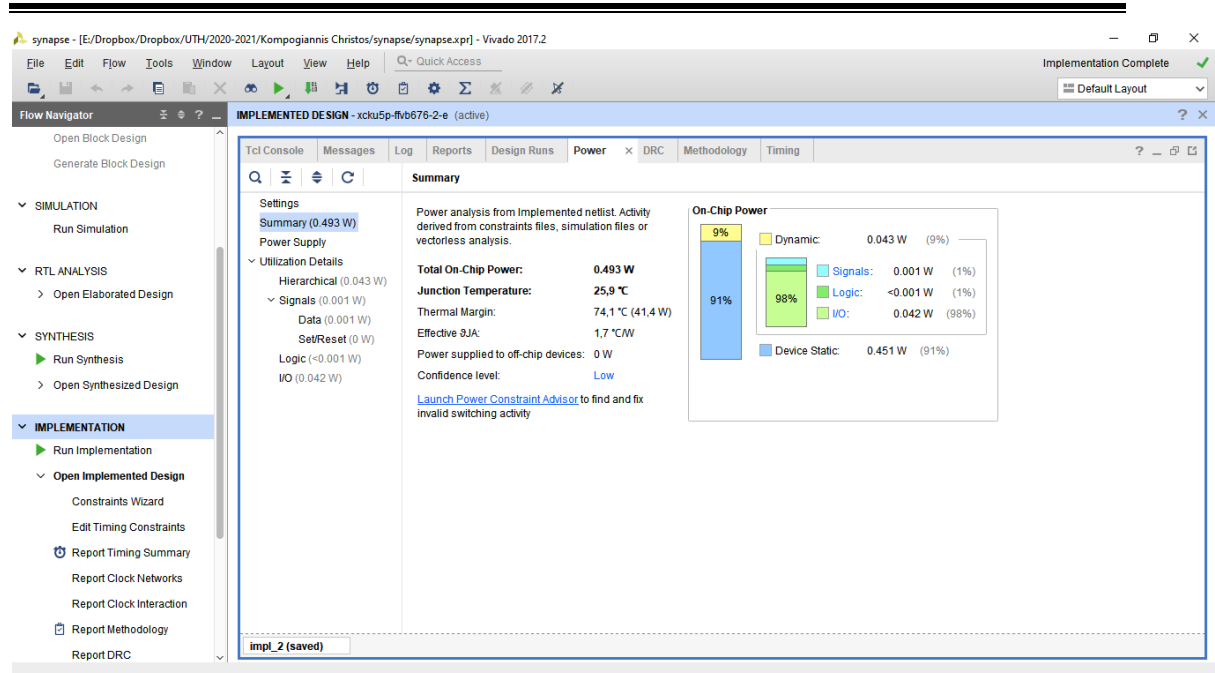
Για τον έλεγχο της ορθότητας των αποτελεσμάτων της υλοποίησης αρχικά έγινε συγγραφή ενός κώδικα επικύρωσης λειτουργίας (testbench). Μέσω αυτού δόθηκαν συγκεκριμένες αρχικές τιμές στις εισόδους της σχεδίασης. Στη συνέχεια χρησιμοποιήθηκε το εργαλείο GTKWave. Αυτό το πρόγραμμα επιτρέπει τον έλεγχο των παραγόμενων κυματομορφών της σχεδίασης για την επικύρωση των αποτελεσμάτων. Οι κυματομορφές ελέγχθηκαν σε διάρκεια 400 ns και επιβεβαιώθηκε ότι οι έξοδοι έχουν τις επιθυμητές τιμές.



Εικόνα 28. Τα αποτελέσματα της προσομοίωσης όσον αφορά τα συναπτικά ρεύματα.

### 6.2 Πόροι Υλοποίησης

Μετρήσεις για τις επιδόσεις της υλοποίησης έγιναν μέσω του εργαλείου Vivado. Έτσι διαπιστώθηκε ότι η σύνθεση στη συσκευή xc6slx16-2 χρησιμοποιεί μόνο τα 400 από τα 216960 Look-Up Tables των Configurable Logic Blocks και μόνο τους 644 από τους 433920 CLB καταχωρητές. Επίσης, όσον αφορά την ισχύ, χρησιμοποιήθηκαν μόλις 0.493 W. Ο τρόπος με τον οποίο κατανέμεται η ισχύς σε δυναμική και σταθερή φαίνεται στην παρακάτω εικόνα.



Εικόνα 29. Αναφορά της κατανάλωσης ισχύος της υλοποίησης στο περιβάλλον Vivado.

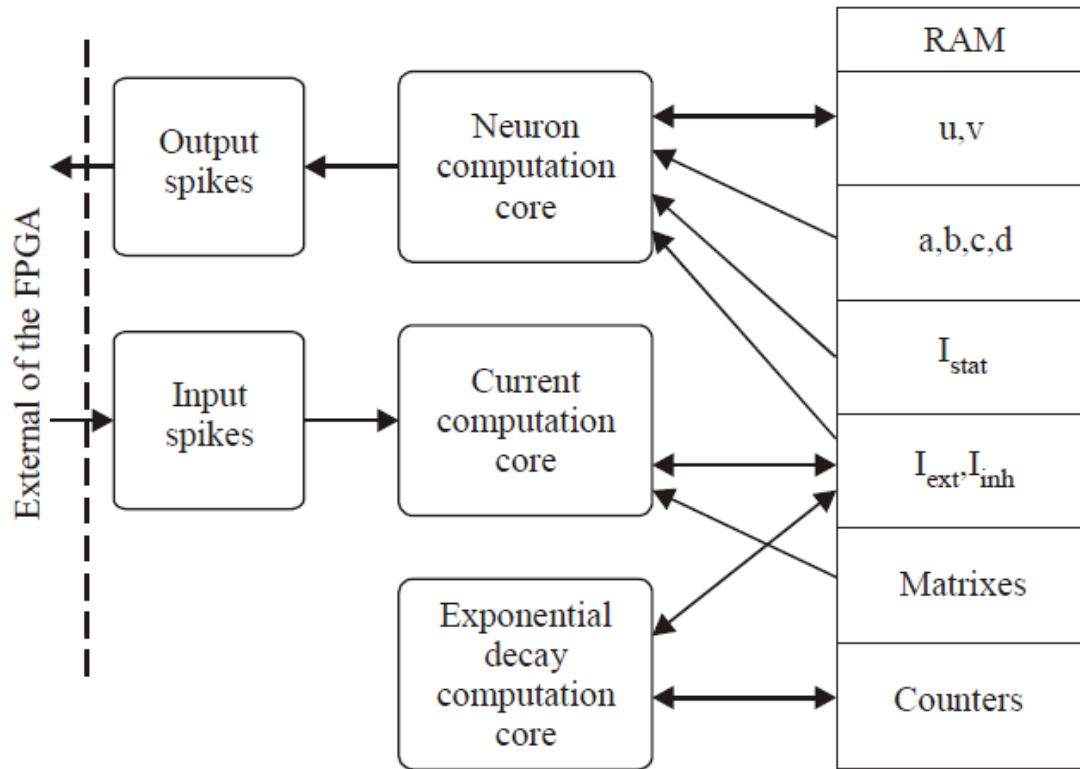
## 7 Συμπεράσματα

### 7.1 Εισαγωγή

Η εκπόνηση της παρούσας εργασίας ξεκίνησε με εκτενή μελέτη της διεθνούς βιβλιογραφίας των νευρομορφικών συστημάτων. Μελετήθηκαν το βιολογικό υπόβαθρο, η γενική δομή των νευρομορφικών κυκλωμάτων και οι πιθανές εφαρμογές τους. Διαπιστώθηκε ότι προσομοιώνοντας τους νευρωνικούς μηχανισμούς του εγκεφάλου σε επίπεδο υλικού, ερευνητές κατάφεραν να δημιουργήσουν βιολογικά αληθοφανή κυκλώματα με υψηλή αποδοτικότητα. Αυτά τα κυκλώματα βασίζονται σε μαθηματικά μοντέλα με διαφορετικούς βαθμούς αληθοφάνειας και πολυπλοκότητας. Επειδή ο στόχος της εργασίας ήταν η ανάπτυξη ενός συναπτικού μηχανισμού ο οποίος θα συμπλήρωνε μία συγκεκριμένη νευρωνική υλοποίηση, πέρα από την συνολική επίδοση, γνώμονας για την επιλογή μοντέλου ήταν και η συμβατότητα με αυτήν την υλοποίηση. Αυτοί οι παράγοντες οδήγησαν στην επιλογή του μοντέλου Izhikevich, το οποίο είχε χρησιμοποιηθεί προηγουμένως από τη Μαρία Σαπουνάκη στον τεχνητό νευρώνα της [59]. Το μοντέλο αυτό χαρακτηρίζεται από μεγάλη βιολογική αληθοφάνεια με μικρή πολυπλοκότητα. Μετά την επιλογή του μοντέλου μελετήθηκαν πολλές παράγωγες υλοποιήσεις οι οποίες περιλάμβαναν συναπτικούς μηχανισμούς. Τέλος αναπτύχθηκε η πρωτότυπη σχεδίαση τεχνητής σύναψης και γράφτηκε ο κώδικας-περιγραφή της για πλακέτα FPGA. Έτσι επιτεύχθηκε μία απλή και αποτελεσματική μέθοδος επικοινωνίας μεταξύ νευρώνων.

### 7.2 Πιθανές Προεκτάσεις της Υλοποίησης

Η παρούσα υλοποίηση έχει δύο πιθανές προεκτάσεις. Η πρώτη σχετίζεται με το ίδιο το κύκλωμα. Καθώς η σύναψη που παρουσιάζεται έχει μεγάλη ομοιότητα με αυτή του paper του Ambroise και συνεργατών [5], ο συναπτικός μηχανισμός επιδέχεται βελτιώσεις παρόμοιες με αυτές της αρχικής υλοποίησης. Συγκεκριμένα, ο πυρήνας υπολογισμού εκθετικής μείωσης ρευμάτων περιλαμβάνει πολλές επαναλαμβανόμενες πράξεις και τελεστές στα 7 στάδια διασωλήνωσης του. Μια πιθανή βελτίωση θα ήταν η συμπίκνωση αυτών των πράξεων με εκ των προτέρων υπολογισμό της μείωσης που υφίστανται οι είσοδοι με κάθε στάδιο διασωλήνωσης. Η δεύτερη προέκταση αφορά τον συνδυασμό αυτής της εργασίας με τη νευρωνική υλοποίηση που προηγήθηκε. Συγκεκριμένα, ο τελικός στόχος είναι οι δύο μηχανισμοί να συνδυαστούν ώστε να δημιουργηθεί μια πλήρης αρχιτεκτονική spiking νευρωνικού δικτύου. Αναφέρεται ότι η εργασία του Ambroise περιλαμβάνει μια τέτοια αρχιτεκτονική η οποία μπορεί να αποτελέσει πηγή έμπνευσης. Όπως και να έχει, η απλότητα της παρούσας υλοποίησης σημαίνει ότι εύκολα μπορεί να τροποποιηθεί και να επεκταθεί ανάλογα με τις ανάγκες κάποιας μελλοντικής εργασίας.



Εικόνα 30. Η αρχιτεκτονική του spiking νευρωνικού δικτύου του paper του Ambroise και συνεργατών.

## ΠΑΡΑΡΤΗΜΑ Α

### 1. Πλατφόρμα Υλικού

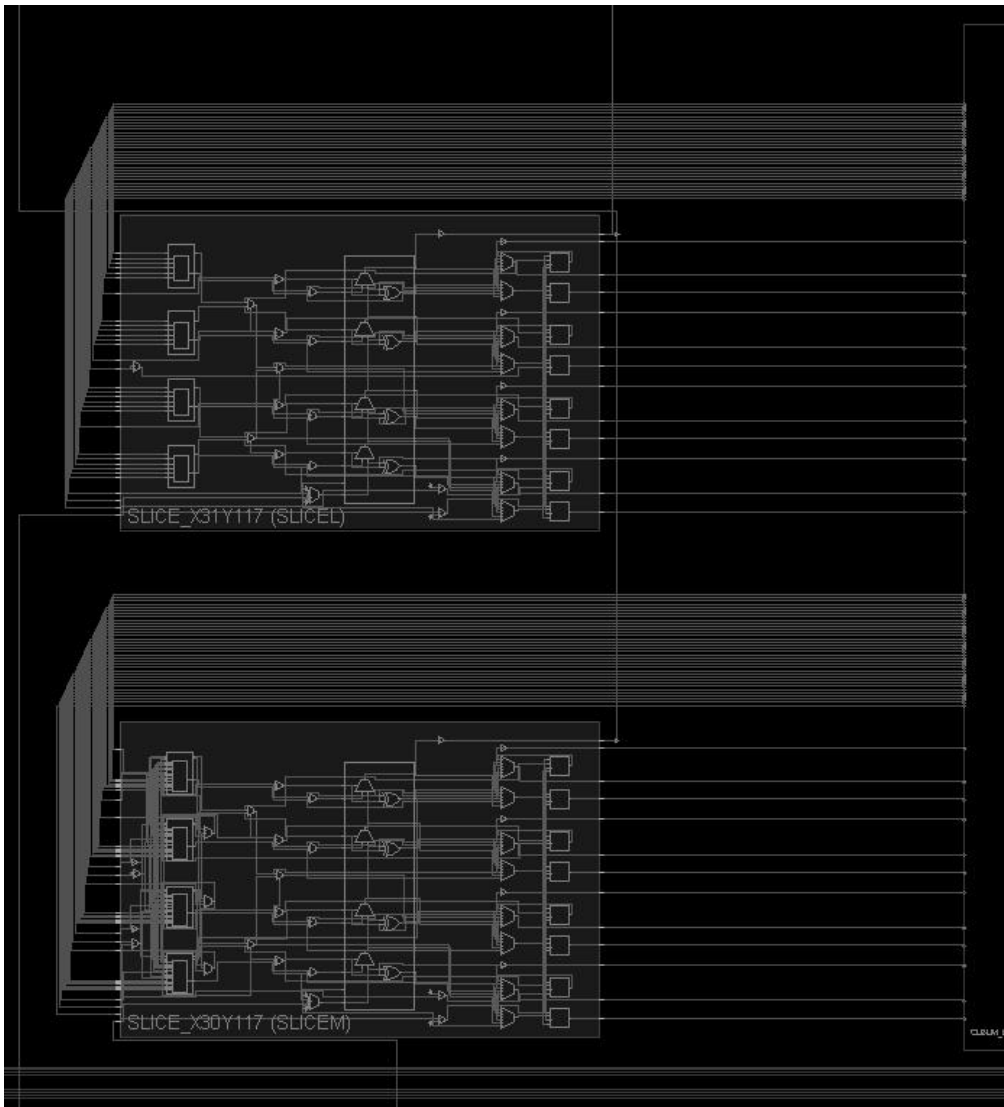
Ο κώδικας της υλοποίησης που παρουσιάστηκε στην παρούσα εργασία γράφτηκε με στόχο την εκτέλεσή του σε πλατφόρμα ανάπτυξης τεχνολογίας FPGA. Το ακρωνύμιο FPGA σημαίνει Field Programmable Gate Arrays, ή Συστοιχία Επιτόπια Προγραμματιζόμενων Πυλών. Όπως φαίνεται από το όνομα, αυτή η τεχνολογία επιτρέπει την τροποποίηση του υλικού ανάλογα με τις ανάγκες της κάθε υλοποίησης. Έτσι επιτυγχάνεται ισορροπία μεταξύ μεγάλων επιδόσεων και ευελιξίας. Η προαναφερθείσα τροποποίηση είναι δυνατή με τη συγγραφή κώδικα σε γλώσσα περιγραφής υλικού (Hardware Description Language - HDL). Για παράδειγμα, οι δομές που περιγράφονται στο ΠΑΡΑΡΤΗΜΑ Β υλοποιούνται με τη βοήθεια των διατάξεων πυλών του FPGA κυκλώματος.

Παρά το γεγονός ότι τα διαφορετικά FPGA μπορούν να διαφέρουν πολύ μεταξύ τους όσον αφορά τις επιδόσεις τους, τα εξής δομικά συστατικά υπάρχουν σε σχεδόν όλα τα κυκλώματα [60]:

1. Μπλοκ Διατασσόμενης Λογικής (Configurable Logic Block – CLB). Αυτός είναι ο πυρήνας του FPGA. Αυτό το εξάρτημα δίνει τη δυνατότητα αναδιάταξης της λειτουργικότητας του κυκλώματος. Αποτελείται από λογικές πύλες και Look-Up Tables (LUTs). Θεωρητικά μονάχα ο συνδυασμός τέτοιων μπλοκ μπορεί να θεωρηθεί ως FPGA.
2. Μπλοκ Επεξεργασίας Ψηφιακών Σημάτων (Digital Signal Processing Block). Ένα εξειδικευμένο FPGA εξάρτημα το οποίο είναι φτιαγμένο για να επεξεργάζεται ψηφιακά σήματα με λειτουργίες όπως το φιλτράρισμα και ο πολλαπλασιασμός.
3. Πομποδέκτες (Transceivers). Αυτά τα εξαρτήματα αναλαμβάνουν την είσοδο και έξοδο δεδομένων προς και από το FPGA. Δουλειά τους είναι η κωδικοποίηση/αποκωδικοποίηση των δεδομένων και ο έλεγχος για λάθη.
4. Μπλοκ Μνήμης (Block RAM). Η μνήμη του FPGA είναι διαταγμένη σε μπλοκ προκαθορισμένου μεγέθους τα οποία όμως μπορούν να διαιρεθούν ή να συνδυαστούν για την τροποποίηση της διαθέσιμης RAM.
5. Μπλοκ Εισόδου/Εξόδου (Input/Output Blocks). Αυτά τα μπλοκ επιτελούν παρόμοιο ρόλο με τους πομποδέκτες αλλά λειτουργούν σε μικρότερες ταχύτητες επιτρέποντας μεγαλύτερη ευελιξία.

Παρακάτω φαίνεται το μπλοκ διατασσόμενης λογικής ενός Xilinx 7 FPGA.





Εικόνα 31. Το μπλοκ διατασσόμενης λογικής (CLB) ενός Xilinx 7 FPGA.

Τα FPGA κυκλώματα συνήθως τοποθετούνται σε πλακέτες ανάπτυξης οι οποίες περιλαμβάνουν διάφορα ηλεκτρονικά στοιχεία ώστε να γίνεται πιο εύκολος ο προγραμματισμός και η είσοδος/έξοδος δεδομένων. Παρακάτω φαίνεται μια πλακέτα ανάπτυξης FPGA βασισμένη στο κύκλωμα Xilinx Artix-7.



Εικόνα 32. Πλακέτα ανάπτυξης του FPGA Xilinx Artix-7.

## 2. Κώδικας

Ο προγραμματισμός των κυκλωμάτων FPGA επιτυγχάνεται με τη χρήση γλωσσών περιγραφής υλικού (Hardware Description Language - HDL). Με αυτές τις γλώσσες καθορίζονται οι δομές, τα σήματα και η λειτουργικότητα του επιθυμητού κυκλώματος. Μετά τη συγγραφή του κώδικα ακολουθεί το στάδιο προσομοίωσης το οποίο χρησιμοποιείται για να επιβεβαιωθεί η σωστή λογική λειτουργία του συστήματος. Για την προσομοίωση απαιτείται ένας επιπλέον σχετικά σύντομος κώδικας με τον οποίο ελέγχεται η ορθότητα της σχεδίασης (testbench). Αυτός ο κώδικας περιλαμβάνει συγκεκριμένες τιμές σημάτων οι οποίες χρησιμοποιούνται ως είσοδοι στη σχεδίαση. Οι κυματομορφές που αποτελούν την έξοδο της σχεδίασης βάσει των εισόδων του testbench εξετάζονται για την ορθότητα τους. Ο προσομοιωτής που χρησιμοποιήθηκε είναι ο ανοιχτού κώδικα GHDL. Το τελικό στάδιο της διαδικασίας είναι η σύνθεση κατά την οποία οι προγραμματιστικές εντολές και δομές μεταφράζονται σε λογικά κυκλώματα.

Οι πιο συνηθισμένες γλώσσες περιγραφής υλικού είναι οι Verilog, VHDL και SystemC. Ο κώδικας της παρούσας υλοποίησης είναι γραμμένος σε VHDL ή αλλιώς VHSIC HDL (Very High Speed Integrated Circuit HDL). Όλοι οι κώδικες αυτής της γλώσσας περιλαμβάνουν τις βιβλιοθήκες, τις οντότητες και την αρχιτεκτονική. Οι βιβλιοθήκες περιέχουν έτοιμες και συχνά χρησιμοποιούμενες μονάδες σχεδίασης. Τέτοιες μονάδες είναι οι συναρτήσεις και οι τύποι δεδομένων. Οι οντότητες περιγράφουν επιμέρους τμήματα της σχεδίασης ως μαύρα κουτιά. Δηλαδή ορίζονται οι είσοδοι και οι έξοδοι τους και το πώς ενώνονται με άλλες οντότητες. Η λειτουργία

των οντοτήτων και άρα η λειτουργία ολόκληρου του συστήματος αναλύεται στην αρχιτεκτονική.

### **3. GTKWave**

Για την επιβεβαίωση των αποτελεσμάτων μετά την προσομοίωση χρησιμοποιήθηκε το εργαλείο ανοιχτού κώδικα GTKWave. Το περιβάλλον αυτό δίνει τη δυνατότητα στο χρήστη να οπτικοποιήσει την κυματομορφή των σημάτων μιας σχεδίασης και να ελέγξει εάν τα αποτελέσματα είναι τα επιθυμητά σε κάθε στιγμή της προσομοίωσης. Το GTKWave δέχεται ως είσοδο αρχεία τύπου .vcd που κατά την ροή εξομοίωσης ενός VHDL μοντέλου.

## ΠΑΡΑΡΤΗΜΑ Β

### 1. Κώδικας 1 (Design)

```
-----register32-----  
  
library IEEE;  
use IEEE.std_logic_1164.all;  
  
entity register32 is  
    Port(  
        D : in std_logic_vector(31 downto 0);  
        CLR : in std_logic;  
        CLK : in std_logic;  
        Q : out std_logic_vector(31 downto 0)  
    );  
end register32;  
  
architecture description of register32 is  
begin  
    process (CLK, CLR)  
        begin  
            if CLR = '1' then  
                Q <= x"00000000";  
            elsif rising_edge(CLK) then  
                q <= d;  
            end if;  
        end process;  
    end description;  
  
-----RegisterSTDLogic-----
```

```
library ieee;
```

---

```
use ieee.std_logic_1164.all;
```

```
entity register32logic is
```

```
    Port(  
        D   : in std_logic;  
        CLR : in std_logic;  
        CLK : in std_logic;  
        Q   : out std_logic  
    );
```

```
end register32logic;
```

```
architecture description of register32logic is
```

```
begin
```

```
    process(CLR, CLK)  
    begin  
        if CLR = '1' then  
            q <= '0';  
        elsif rising_edge(CLK) then  
            q <= d;  
        end if;
```

```
    end process;
```

```
end description;
```

```
-----adder32-----
```

```
library IEEE;
```

```
use IEEE.std_logic_1164.all;
```

```
use ieee.numeric_std.all;
```

```
entity adder32 is
```

```
    Port(  

```

## Ανάπτυξη Νευρομορφικού Κυκλώματος Σε Αναδιατασσόμενη Τεχνολογία Για Προσομοίωση Συνάψεων

---

```
A : in std_logic_vector(31 downto 0);
B : in std_logic_vector(31 downto 0);
SUM : out std_logic_vector(31 downto 0);
CARRY : out std_logic
);
end adder32;
```

architecture description of adder32 is

```
signal TEMP : signed(32 downto 0);

begin
  process (A, B)
  begin
    TEMP <= signed('0' & A) + signed('0' & B);
  end process;
  SUM <= std_logic_vector(TEMP(31 downto 0));
  CARRY <= std_logic(TEMP(32));
end description;
```

-----comparator-----

```
library IEEE;
use ieee.numeric_std.all;
use IEEE.std_logic_1164.all;
```

entity comparator is

```
Port(
  A : in std_logic_vector(31 downto 0);
  B : in std_logic_vector(31 downto 0);
  Q : out std_logic
);
```

end comparator;

architecture description of comparator is

signal ASigned, BSigned : signed(31 downto 0);

begin

ASigned <= signed(A);

BSigned <= signed(B);

Q <= '1' when (ASigned > BSigned) else '0';

end description;

-----mux2to1-----

library IEEE;

use IEEE.std\_logic\_1164.all;

use ieee.numeric\_std.all;

entity mux2to1 is

Port(

SEL : in std\_logic;

A : in std\_logic\_vector(31 downto 0);

B : in std\_logic\_vector(31 downto 0);

Q : out std\_logic\_vector(31 downto 0)

);

end mux2to1;

architecture description of mux2to1 is

begin

Q <= A when (SEL = '0') else B;

end description;

```
-----demux1to2-----  
  
library IEEE;  
use IEEE.std_logic_1164.all;  
  
entity demux1to2 is  
    Port(  
        SEL : in std_logic;  
        A  : in std_logic_vector(31 downto 0);  
        Q0 : out std_logic_vector(31 downto 0);  
        Q1 : out std_logic_vector(31 downto 0)  
    );  
end demux1to2;  
  
architecture description of demux1to2 is  
begin  
    process (SEL,A)  
    begin  
        if (SEL = '0') then  
            Q0 <= A;  
        elsif (SEL <= '1') then  
            Q1 <= A;  
        end if;  
    end process;  
end description;
```

```
-----rshift_4-----  
  
library ieee;  
use ieee.std_logic_1164.all;  
  
entity rshift_4 is  
    Port(  

```



```
A : in std_logic_vector(31 DOWNT0 0);  
Q : out std_logic_vector(31 DOWNT0 0)  
  
);  
end rshift_4;
```

architecture description of rshift\_4 is

```
begin  
    Q(31 downto 0) <= "0000" & A(31 downto 4);  
end description;
```

-----currentup-----

```
library ieee;  
use ieee.std_logic_1164.all;  
use ieee.numeric_std.all;
```

entity currentup is

```
Port(  
    IEXC      : in std_logic_vector(31 downto 0);  
    IINH      : in std_logic_vector(31 downto 0);  
    WEIGHT_SIG : in std_logic_vector(31 downto 0);  
    WEIGHT     : in std_logic_vector(31 downto 0);  
    PRE_SPIKE  : in std_logic;  
    CLK        : in std_logic;  
    CLR        : in std_logic;  
    IEXC1      : out std_logic_vector(31 downto 0);  
    IINH1      : out std_logic_vector(31 downto 0)  
  
);  
end currentup;
```

architecture description of currentup is

## Ανάπτυξη Νευρομορφικού Κυκλώματος Σε Αναδιατασσόμενη Τεχνολογία Για Προσομοίωση Συνάψεων

---

```
signal R1_PL1, R1_PL2, R3_PL3, R7_PL4, R10_PL5 : std_logic;
```

```
signal R2_PL2, R2_PL3, R4_PL2, R4_PL3, R5_PL3, R5_PL4, R6_PL4, R8_PL4, R9_PL4, R9_PL5, R11_PL5,  
MUX1_TEMP, MUX2_TEMP : std_logic_vector(31 downto 0);
```

```
signal ZERO : std_logic_vector(31 downto 0) := "00000000000000000000000000000000";
```

```
component register32 is
```

```
  Port(  
    D      : in std_logic_vector(31 downto 0);  
    CLR,CLK : in std_logic;  
    Q      : out std_logic_vector(31 downto 0)  
  );
```

```
end component;
```

```
component register32logic is
```

```
  Port(  
    D,CLR,CLK : in std_logic;  
    Q          : out std_logic  
  );
```

```
end component;
```

```
component adder32 is
```

```
  Port(  
    A,B : in std_logic_vector(31 downto 0);  
    SUM : out std_logic_vector(31 downto 0)  
  );
```

```
end component;
```

```
component mux2to1 is
```

```
  Port(  
    SEL : in std_logic;  
    A,B : in std_logic_vector(31 downto 0);  
    Q   : out std_logic_vector(31 downto 0)
```

```
);  
end component;  
  
component comparator is  
  Port(  
    A,B : in std_logic_vector(31 downto 0);  
    Q   : out std_logic_vector(31 downto 0)  
  );  
end component;
```

```
component demux1to2 is  
  Port(  
    SEL : in std_logic;  
    A   : in std_logic_vector(31 downto 0);  
    Q0,Q1 : out std_logic_vector(31 downto 0)  
  );  
end component;
```

```
begin  
-----1st pipeline stage-----  
  comp_pl1 : comparator port map(  
    A => WEIGHT_SIG,  
    B => ZERO,  
    Q => R1_PL1  
  );  
  
  reg_comp_pl1 : register32logic port map(  
    D => R1_PL1,  
    CLR => CLR,  
    CLK => CLK,  
    Q => R1_PL2  
  );
```

-----2nd pipeline stage-----

mux1\_pl2 : mux2to1 port map(

SEL => R1\_PL2,

A => IEXC,

B => IINH,

Q => R2\_PL2

);

reg\_mux1\_pl2 : register32 port map(

D => R2\_PL2,

CLR => CLR,

CLK => CLK,

Q => R2\_PL3

);

reg\_comp\_pl2 : register32logic port map(

D => R1\_PL2,

CLR => CLR,

CLK => CLK,

Q => R3\_PL3

);

mux2\_pl2 : mux2to1 port map(

SEL => R1\_PL2,

A => IINH,

B => IEXC,

Q => R4\_PL2

);

reg\_mux2\_pl2 : register32 port map(

D => R4\_PL2,

CLR => CLR,

```
CLK => CLK,  
Q => R4_PL3  
);
```

-----3rd pipeline stage-----

```
adder_pl3 : adder32 port map(  
  A => WEIGHT,  
  B => R2_PL3,  
  SUM => R5_PL3  
);
```

```
reg_adder_pl3 : register32 port map(  
  D => R5_PL3,  
  CLR => CLR,  
  CLK => CLK,  
  Q => R5_PL4  
);
```

```
reg_mux1_pl3 : register32 port map(  
  D => R2_PL3,  
  CLR => CLR,  
  CLK => CLK,  
  Q => R6_PL4  
);
```

```
reg_comp_pl3 : register32logic port map(  
  D => R3_PL3,  
  CLR => CLR,  
  CLK => CLK,  
  Q => R7_PL4  
);
```

```
reg_mux2_pl3 : register32 port map(  

```

```
D => R4_PL3,  
CLR => CLR,  
CLK => CLK,  
Q => R8_PL4  
);
```

-----4th pipeline stage-----

```
mux_pl4 : mux2to1 port map(  
  SEL => PRE_SPIKE,  
  A => R5_PL4,  
  B => R6_PL4,  
  Q => R9_PL4  
);
```

```
reg_mux1_pl4 : register32 port map(  
  D => R9_PL4,  
  CLR => CLR,  
  CLK => CLK,  
  Q => R9_PL5  
);
```

```
reg_comp_pl4 : register32logic port map(  
  D => R7_PL4,  
  CLR => CLR,  
  CLK => CLK,  
  Q => R10_PL5  
);
```

```
reg_mux2_pl4 : register32 port map(  
  D => R8_PL4,  
  CLR => CLR,  
  CLK => CLK,  
  Q => R11_PL5  
);
```

);

-----5th pipeline stage-----

mux1\_pl5 : mux2to1 port map(

SEL => R10\_PL5,

A => R9\_PL5,

B => R11\_PL5,

Q => MUX1\_TEMP

);

mux2\_pl5 : mux2to1 port map(

SEL => R10\_PL5,

A => R11\_PL5,

B => R9\_PL5,

Q => MUX2\_TEMP

);

IEXC1 <= MUX1\_TEMP;

IINH1 <= MUX2\_TEMP;

end description;

-----decay-----

library ieee;

use ieee.std\_logic\_1164.all;

use ieee.numeric\_std.all;

entity decay is

Port(

## Ανάπτυξη Νευρομορφικού Κυκλώματος Σε Αναδιατασσόμενη Τεχνολογία Για Προσομοίωση Συνάψεων

---

```
IEXC1      : in std_logic_vector(31 downto 0);
IINH1      : in std_logic_vector(31 downto 0);
CNTEXC     : in std_logic;
CNTINH     : in std_logic;
CLK        : in std_logic;
CLR        : in std_logic;
IEXC2      : out std_logic_vector(31 downto 0);
IINH2      : out std_logic_vector(31 downto 0)
);
end decay;
```

architecture description of decay is

```
signal R1_PL1, R1_PL2, R2_PL2, R3_PL1, R3_PL2, R4_PL2, R5_PL2, R5_PL3, R6_PL2, R6_PL3, R7_PL3,
R7_PL4, R8_PL3, R8_PL4, R9_PL4, R10_PL4, R10_PL5, R11_PL4, R11_PL5, R12_PL5, R12_PL6, R13_PL6,
R14_PL6, R14_PL7, R15_PL7, R16_PL7, S1_PL3, S2_PL3, S1_PL4, S1_PL5, S1_PL6, S1_PL7, IEXC2_TEMP
: std_logic_vector(31 downto 0);
```

component register32 is

```
Port(
  D      : in std_logic_vector(31 downto 0);
  CLR,CLK : in std_logic;
  Q      : out std_logic_vector(31 downto 0)
);
end component;
```

component adder32 is

```
Port(
  A,B : in std_logic_vector(31 downto 0);
  SUM : out std_logic_vector(31 downto 0)
);
end component;
```

component mux2to1 is



```
Port(  
    SEL : in std_logic;  
    A,B  : in std_logic_vector(31 downto 0);  
    Q    : out std_logic_vector(31 downto 0)  
);  
end component;
```

component rshift\_4 is

```
Port(  
    A  : in std_logic_vector(31 downto 0);  
    Q  : out std_logic_vector(31 downto 0)  
);  
end component;
```

begin

-----1st pipeline stage-----

```
shifter1_pl1 : rshift_4 port map(  
    A => IEXC1,  
    Q => R1_PL1  
);
```

```
reg_shifter1_pl1 : register32 port map(  
    D => R1_PL1,  
    CLR => CLR,  
    CLK => CLK,  
    Q => R1_PL2  
);
```

```
reg_exc_pl1 : register32 port map(  
    D => IEXC1,  
    CLR => CLR,  
    CLK => CLK,
```

```
    Q => R2_PL2
);

shifter2_pl1 : rshift_4 port map(
    A => IINH1,
    Q => R3_PL1
);

reg_shifter2_pl1 : register32 port map(
    D => R3_PL1,
    CLR => CLR,
    CLK => CLK,
    Q => R3_PL2
);

reg_inh_pl1 : register32 port map(
    D => IINH1,
    CLR => CLR,
    CLK => CLK,
    Q => R4_PL2
);

-----2nd pipeline stage-----

adder1_pl2 : adder32 port map(
    A => R1_PL2,
    B => R2_PL2,
    SUM => R5_PL2
);

reg_adder1_pl2 : register32 port map(
    D => R5_PL2,
    CLR => CLR,
    CLK => CLK,
```

```
    Q => R5_PL3  
);
```

```
adder2_pl2 : adder32 port map(  
    A => R3_PL2,  
    B => R4_PL2,  
    SUM => R6_PL2  
);
```

```
reg_adder2_pl2 : register32 port map(  
    D => R6_PL2,  
    CLR => CLR,  
    CLK => CLK,  
    Q => R6_PL3  
);
```

-----3rd pipeline stage-----

```
shifter1_pl3 : rshift_4 port map(  
    A => R5_PL3,  
    Q => S1_PL3  
);
```

```
adder1_pl3 : adder32 port map(  
    A => S1_PL3,  
    B => R5_PL3,  
    SUM => R7_PL3  
);
```

```
reg_adder1_pl3 : register32 port map(  
    D => R7_PL3,  
    CLR => CLR,  
    CLK => CLK,
```

```
Q => R7_PL4  
);
```

```
shifter2_pl3 : rshift_4 port map(  
  A => R6_PL3,  
  Q => S2_PL3  
);
```

```
adder2_pl3 : adder32 port map(  
  A => S2_PL3,  
  B => R6_PL3,  
  SUM => R8_PL3  
);
```

```
reg_adder2_pl3 : register32 port map(  
  D => R8_PL3,  
  CLR => CLR,  
  CLK => CLK,  
  Q => R8_PL4  
);
```

```
reg_adder3_pl3 : register32 port map(  
  D => R6_PL3,  
  CLR => CLR,  
  CLK => CLK,  
  Q => R9_PL4  
);
```

-----4th pipeline stage-----

```
shifter_pl4 : rshift_4 port map(  
  A => R7_PL4,  
  Q => S1_PL4
```

);

adder\_pl4 : adder32 port map(

  A => S1\_PL4,

  B => R7\_PL4,

  SUM => R10\_PL4

);

reg\_adder\_pl4 : register32 port map(

  D => R10\_PL4,

  CLR => CLR,

  CLK => CLK,

  Q => R10\_PL5

);

mux\_pl4 : mux2to1 port map(

  SEL => CNTINH,

  A => R8\_PL4,

  B => R9\_PL4,

  Q => R11\_PL4

);

reg\_mux\_pl4 : register32 port map(

  D => R11\_PL4,

  CLR => CLR,

  CLK => CLK,

  Q => R11\_PL5

);

-----5th pipeline stage-----

shifter\_pl5 : rshift\_4 port map(

  A => R10\_PL5,

```
    Q => S1_PL5
);

adder_pl5 : adder32 port map(
    A => S1_PL5,
    B => R10_PL5,
    SUM => R12_PL5
);

reg_adder_pl5 : register32 port map(
    D => R12_PL5,
    CLR => CLR,
    CLK => CLK,
    Q => R12_PL6
);

reg_mux_pl5 : register32 port map(
    D => R11_PL5,
    CLR => CLR,
    CLK => CLK,
    Q => R13_PL6
);
```

-----6th pipeline stage-----

```
shifter_pl6 : rshift_4 port map(
    A => R12_PL6,
    Q => S1_PL6
);

adder_pl6 : adder32 port map(
    A => S1_PL6,
    B => R12_PL6,
```

```
SUM => R14_PL6
);

reg_adder_pl6 : register32 port map(
    D => R14_PL6,
    CLR => CLR,
    CLK => CLK,
    Q => R14_PL7
);
```

```
reg_mux_pl6 : register32 port map(
    D => R13_PL6,
    CLR => CLR,
    CLK => CLK,
    Q => R15_PL7
);
```

-----7th pipeline stage-----

```
shifter_pl7 : rshift_4 port map(
    A => R14_PL7,
    Q => S1_PL7
);
```

```
adder_pl7 : adder32 port map(
    A => S1_PL7,
    B => R14_PL7,
    SUM => R16_PL7
);
```

```
mux_pl7 : mux2to1 port map(
    SEL => CNTEXC,
    A => R16_PL7,
    B => R14_PL7,
```

```
    Q => IEXC2_TEMP
);

IEXC2 <= IEXC2_TEMP;
IINH2 <= R15_PL7;

end description;

----currents----
library ieee;
use ieee.std_logic_1164.all;
use ieee.numeric_std.all;

entity currents is
    Port(
        IEXC      : in std_logic_vector(31 downto 0);
        IINH      : in std_logic_vector(31 downto 0);
        WEIGHT_SIG : in std_logic_vector(31 downto 0);
        WEIGHT     : in std_logic_vector(31 downto 0);
        PRE_SPIKE  : in std_logic;
        CNTEXC     : in std_logic;
        CNTINH     : in std_logic;
        CLK        : in std_logic;
        CLR        : in std_logic;
        IEXC1      : out std_logic_vector(31 downto 0);
        IINH1      : out std_logic_vector(31 downto 0);
        IEXC2      : out std_logic_vector(31 downto 0);
        IINH2      : out std_logic_vector(31 downto 0)
    );
end currents;

architecture behavior of currents is
```

---



```
signal IEXC1_S, IINH1_S, IEXC2_S, IINH2_S : std_logic_vector(31 downto 0);
```

```
component currentup is
```

```
Port(  
    IEXC      : in std_logic_vector(31 downto 0);  
    IINH      : in std_logic_vector(31 downto 0);  
    WEIGHT_SIG : in std_logic_vector(31 downto 0);  
    WEIGHT    : in std_logic_vector(31 downto 0);  
    PRE_SPIKE  : in std_logic;  
    CLK       : in std_logic;  
    CLR       : in std_logic;  
    IEXC1     : out std_logic_vector(31 downto 0);  
    IINH1     : out std_logic_vector(31 downto 0)  
);
```

```
end component;
```

```
component decay is
```

```
Port(  
    IEXC1     : in std_logic_vector(31 downto 0);  
    IINH1     : in std_logic_vector(31 downto 0);  
    CNTEXC    : in std_logic;  
    CNTINH    : in std_logic;  
    CLK       : in std_logic;  
    CLR       : in std_logic;  
    IEXC2     : out std_logic_vector(31 downto 0);  
    IINH2     : out std_logic_vector(31 downto 0)  
);
```

```
end component;
```

```
begin
```

```
    currentupdate : currentup port map(  
        IEXC => IEXC,
```

```
IINH => IINH,  
WEIGHT_SIG => WEIGHT_SIG,  
WEIGHT => WEIGHT,  
PRE_SPIKE => PRE_SPIKE,  
CLK => CLK,  
CLR => CLR,  
IEXC1 => IEXC1_S,  
IINH1 => IINH1_S  
);
```

```
decaycurrent : decay port map(
```

```
  IEXC1 => IEXC1_S,  
  IINH1 => IINH1_S,  
  CNTEXC => CNTEXC,  
  CNTINH => CNTINH,  
  CLK => CLK,  
  CLR => CLR,  
  IEXC2 => IEXC2_S,  
  IINH2 => IINH2_S  
);
```

```
IEXC1 <= IEXC1_S;
```

```
IINH1 <= IINH1_S;
```

```
IEXC2 <= IEXC2_S;
```

```
IINH2 <= IINH2_S;
```

```
end behavior;
```

## 2. Κώδικας 2 (Testbench)

```
library ieee;

use ieee.std_logic_1164.all;

use ieee.numeric_std.all;

entity testbench is

end testbench;

architecture behavior of testbench is

component currents is

    Port(
        IEXC      : in std_logic_vector(31 downto 0);
        IINH      : in std_logic_vector(31 downto 0);
        WEIGHT_SIG : in std_logic_vector(31 downto 0);
        WEIGHT     : in std_logic_vector(31 downto 0);
        PRE_SPIKE  : in std_logic;
        CNTEXC     : in std_logic;
        CNTINH     : in std_logic;
        CLK        : in std_logic;
        CLR        : in std_logic;
        IEXC1      : out std_logic_vector(31 downto 0);
        IINH1      : out std_logic_vector(31 downto 0);
        IEXC2      : out std_logic_vector(31 downto 0);
        IINH2      : out std_logic_vector(31 downto 0)
    );
end component;

signal IEXC_IN, IINH_IN, WEIGHT_SIG_IN, WEIGHT_IN, IEXC1_OUT, IINH1_OUT, IEXC2_OUT,
IINH2_OUT: std_logic_vector(31 downto 0);

signal PRE_SPIKE_IN, CNTEXC_IN, CNTINH_IN, CLK_IN, CLR_IN : std_logic;
```



```
clk_in<='1';  
wait for 20 ns;
```

```
clk_in<='0';  
wait for 20 ns;
```

```
clk_in<='1';  
wait for 20 ns;
```

```
clk_in<='0';  
wait for 20 ns;
```

```
clk_in<='1';  
wait for 20 ns;
```

```
clk_in<='0';  
wait for 20 ns;
```

```
clk_in<='1';  
wait for 20 ns;
```

```
clk_in<='0';  
wait for 20 ns;
```

```
clk_in<='1';  
wait for 20 ns;
```

```
clk_in<='0';  
wait for 20 ns;
```

```
clk_in<='1';  
wait for 20 ns;
```

wait;  
end process;  
end behavior;

## Βιβλιογραφία

- [1] European Commission, «EU support for research and innovation in the area of the brain,» 30 Ιούλιος 2020. [Ηλεκτρονικό]. Available: [https://ec.europa.eu/info/research-and-innovation/research-area/health-research-and-innovation/brain-research\\_en](https://ec.europa.eu/info/research-and-innovation/research-area/health-research-and-innovation/brain-research_en). [Πρόσβαση 8 Μάρτιος 2021].
- [2] D. Mishra, A. Yadav, S. Ray και P. Kalra, «Exploring biological neuron models,» *The Research Magazine of IIT Kanpur*, pp. 13-22, 1 Ιανουάριος 2006.
- [3] C. Mead, «Neuromorphic electronic systems,» *Proc IEEE. on Proceedings of the IEEE*, τόμ. 78, pp. 1629 - 1636, 1990.
- [4] C. D. Schuman, T. E. Potok, R. M. Patton, J. Douglas Birdwell, M. E. Dean, G. S. Rose και J. S. Plank, «A Survey of Neuromorphic Computing and Neural Networks in Hardware,» *ArXiv*, τόμ. abs/1705.06963, 2017.
- [5] M. Ambroise, T. Levi, Y. Bornat και S. Saïghi, «Biorealistic Spiking Neural Network on FPGA,» σε *47th Annual Conference on Information Sciences and Systems (CISS)*, Baltimore, 2013.
- [6] BCcampus, «BCcampus Open Publishing,» 17 10 2014. [Ηλεκτρονικό]. Available: <https://opentextbc.ca/introductiontopsychology/chapter/3-1-the-neuron-is-the-building-block-of-the-nervous-system/>. [Πρόσβαση 11 Μάρτιος 2021].
- [7] Lumen Learning, «Boundless Psychology,» 12 4 2019. [Ηλεκτρονικό]. Available: <https://courses.lumenlearning.com/boundless-psychology/chapter/neurons/>. [Πρόσβαση 12 Μάρτιος 2021].
- [8] J. Waymire, «Neuroscience Online,» [Ηλεκτρονικό]. Available: <https://nba.uth.tmc.edu/neuroscience/m/s1/chapter08.html>. [Πρόσβαση 12 Μάρτιος 2021].
- [9] Brittanica, «Action Potential,» [Ηλεκτρονικό]. Available: <https://www.britannica.com/science/nervous-system/Action-potential>. [Πρόσβαση 12 Μάρτιος 2021].
- [10] Κ. Παπαθεοδωρόπουλος, «Σύναψη και Συναπτική διαβίβαση [Κεφάλαιο Συγγράμματος],» σε *Έννοιες στην επιστήμη της μνήμης. [ηλεκτρ. βιβλ.]*, Αθήνα, Σύνδεσμος Ελληνικών Ακαδημαϊκών Βιβλιοθηκών, 2015.
- [11] D. Purves, G. J. Augustine, D. Fitzpatrick, L. C. Katz, A.-S. LaMantia, M. S. Williams και J. O. McNamara, «Chapter 5. Electric Transmission,» σε *Neuroscience, 3rd Edition*, Sunderland

- (MA), Sinauer Associates, 2004, pp. 93-95.
- [12] S. Ghosh-Dastidar και H. Adeli, «Third Generation Neural Networks: Spiking Neural Networks.,» *Advances in Computational Intelligence*, τόμ. 116, pp. 167-178, 2009.
- [13] U. Sinha, «AI Shack,» [Ηλεκτρονικό]. Available: <https://aishack.in/tutorials/artificial-neurons-mccullochpitts-model/>. [Πρόσβαση 18 Μάρτιος 2021].
- [14] D. Beeman, «Hodgkin-Huxley Model,» *Encyclopedia of Computational Neuroscience*, pp. 1-13, 14 Απρίλιος 2014.
- [15] E. M. Izhikevich και R. FitzHugh, «FitzHugh-Nagumo model,» *Scholarpedia*, τόμ. 1, αρ. 9, p. 1349, 2006.
- [16] S. Ditlevsen και P. Greenwood, «The Morris–Lecar neuron model embeds a leaky integrate-and-fire model,» *Mathematical Biology*, pp. 241-243, 24 Μάιος 2012.
- [17] M. Storace, L. Daniele και E. de Lange, «The Hindmarsh–Rose neuron model: Bifurcation analysis and piecewise-linear approximations,» *Chaos: An Interdisciplinary Journal of Nonlinear Science*, pp. 1-18, 5 Σεπτέμβριος 2008.
- [18] J. M. González, «Complex bifurcation structures in the Hindmarsh–Rose neuron model,» *International Journal of Bifurcation and Chaos*, pp. 3071-3083, Σεπτέμβριος 2007.
- [19] L. F. Abbott, «Lapicque's introduction of the integrate-and-fire model neuron (1907).,» *Brain Research Bulletin*, τόμ. 5, αρ. 6, pp. 303-304, 1999.
- [20] A. N. Burkitt, «A review of the integrate-and-fire neuron model: I. Homogeneous synaptic input,» *Biological Cybernetics*, pp. 1-19, 2006.
- [21] E. Izhikevich, «Resonate-and-fire neurons,» *Neural Networks*, τόμ. 14, p. 883, 2001.
- [22] T. Hishiki και H. Torikai, «A Novel Resonate-and-Fire-type Digital Spiking Neuron and its Bifurcation Analysis,» σε *2009 International Symposium on Nonlinear Theory and its Applications*, Sapporo, Japan, 2009.
- [23] W. Gerstner και R. Brette, «Adaptive exponential integrate-and-fire model,» *Scholarpedia*, τόμ. 4, αρ. 6, p. 8427, 2009.
- [24] Y. Choe, «Hebbian Learning,» σε *Encyclopedia of Computational Neuroscience*, New York, NY, Springer, 2014, pp. 3498-3502.
- [25] J. Sjöström και W. Gerstner, «Scholarpedia,» 10 Φεβρουάριος 2010. [Ηλεκτρονικό]. Available: [http://www.scholarpedia.org/article/Spike-timing\\_dependent\\_plasticity](http://www.scholarpedia.org/article/Spike-timing_dependent_plasticity).



- [Πρόσβαση 24 Μάρτιος 2021].
- [26] S. Bohte, J. Kok και H. L. Poutre, «SpikeProp: Backpropagation for Networks of Spiking Neurons,» σε *ESANN 2000, 8th European Symposium on Artificial Neural Networks*, Bruges, Belgium, 2000.
- [27] R. Gütig και H. Sompolinsky, «The tempotron: a neuron that learns spiketiming-based decisions,» *Nature Neuroscience*, pp. 420-428, 12 Φεβρουάριος 2006.
- [28] C. Gordon, E. Farquhar και P. Hasler, «A family of floating-gate adapting synapses based upon transistor channel models,» σε *IEEE International Symposium on Circuits and Systems*, Vancouver, 2004.
- [29] D. Bucher, «Central Pattern Generators,» σε *Encyclopedia of Neuroscience*, Amsterdam, Elsevier, 2009, p. 691–700.
- [30] Q. Chen, J. Wang, Y. Shuangming, Q. Yingmei, D. Bin και X. Wei, «A real-time FPGA implementation of a biologically inspired central pattern generator network,» *Neurocomputing*, pp. 63-80, 28 Μάρτιος 2017.
- [31] T. Gupta, «Towards Data Science,» 5 Ιανουάριος 2017. [Ηλεκτρονικό]. Available: <https://towardsdatascience.com/deep-learning-feedforward-neural-network-26a6705dbdc7>. [Πρόσβαση 25 Μάρτιος 2021].
- [32] J. Brownlee, «Crash Course On Multi-Layer Perceptron Neural Networks,» 17 Μάιος 2016. [Ηλεκτρονικό]. Available: <https://machinelearningmastery.com/neural-networks-crash-course/>. [Πρόσβαση 25 Μάρτιος 2021].
- [33] P. Vadapalli, «upGrad blog,» 11 Νοέμβριος 2020. [Ηλεκτρονικό]. Available: <https://www.upgrad.com/blog/introduction-to-recursive-neural-network/>. [Πρόσβαση 26 Μάρτιος 2021].
- [34] IBM Cloud Education , «IBM,» 14 Σεπτέμβριος 2020. [Ηλεκτρονικό]. Available: [https://www.ibm.com/cloud/learn/recurrent-neural-networks#toc-variant-rn-2xvhb\\_yi](https://www.ibm.com/cloud/learn/recurrent-neural-networks#toc-variant-rn-2xvhb_yi). [Πρόσβαση 26 Μάρτιος 2021].
- [35] Tutorials Point, «Artificial Neural Network - Hopfield Networks,» [Ηλεκτρονικό]. Available: [https://www.tutorialspoint.com/artificial\\_neural\\_network/artificial\\_neural\\_network\\_hopfield.htm](https://www.tutorialspoint.com/artificial_neural_network/artificial_neural_network_hopfield.htm). [Πρόσβαση 26 Μάρτιος 2021].
- [36] B. Dickson, «What are convolutional neural networks (CNN)?,» 6 Ιανουάριος 2020. [Ηλεκτρονικό]. Available: <https://bdtechtalks.com/2020/01/06/convolutional-neural-networks-cnn-convnets/>. [Πρόσβαση 26 Μάρτιος 2021].

- [37] af inventions - embedded electronics, «FPGA advantages,» 2017. [Ηλεκτρονικό]. Available: <https://af-inventions.de/en/services/fpga-development/fpga-advantages.html>. [Πρόσβαση Μάρτιος 29 2021].
- [38] TechDifferences, «Difference Between Analog and Digital Computer,» 13 Αύγουστος 2019. [Ηλεκτρονικό]. Available: <https://techdifferences.com/difference-between-analog-and-digital-computer.html>. [Πρόσβαση 30 Μάρτιος 2021].
- [39] Cadence, «FPGA vs. ASIC: Differences and Choosing Best for Your Business,» 17 Δεκέμβριος 2019. [Ηλεκτρονικό]. Available: <https://resources.pcb.cadence.com/blog/2019-fpga-vs-asic-differences-and-choosing-best-for-your-business>. [Πρόσβαση 31 Μάρτιος 2021].
- [40] T. J. Koickal, A. Hamilton, S. L. Tan, J. A. Covington και J. W. Gardner, «Analog VLSI Circuit Implementation of an Adaptive Neuromorphic Olfaction Chip,» *IEEE Transactions on Circuits and Systems I: Regular Papers*, pp. 60-73, Ιανουάριος 2007.
- [41] J. K. Olofsson, «Time to smell: a cascade model of human olfactory perception based on response-time (RT) measurement,» *Frontiers in Psychology*, pp. 33-41, 4 Φεβρουάριος 2014.
- [42] E. Farquhar, C. Gordon και P. Hasler, «A field programmable neural array,» σε *Circuits and Systems, 2006. ISCAS 2006. Proceedings. 2006 IEEE International Symposium on. IEEE*, Island of Kos, Greece, 2006.
- [43] M. Liu, H. Yu και W. Wang, «FPAA Based on Integration of CMOS and Nanojunction Devices for Neuromorphic Applications,» σε *Lecture Notes of the Institute for Computer Sciences, Social Informatics and Telecommunications Engineering*, Boston, MA, 2008.
- [44] M. Ambroise, T. Levi, S. Joucla, B. Yvert και S. Saïghi, «Real-time Biomimetic Central Pattern Generators in an FPGA for Hybrid Experiments,» *Frontiers in Neuroscience*, pp. 1-11, 21 Νοέμβριος 2013.
- [45] «InSight: An FPGA-Based Neuromorphic Computing System for Deep Neural Networks,» *Low Power Electronics and Applications*, pp. 10-18, 30 Οκτώβριος 2020.
- [46] T. Dalgaty, M. Payvand, F. Moro, D. R. B. Ly, F. Pebay-Peyroula, J. Casas, G. Indiveri και E. Vianello, «Hybrid neuromorphic circuits exploiting non-conventional properties of RRAM for massively parallel local plasticity mechanisms,» *APL Materials* 7, 29 Αύγουστος 2019.
- [47] Human Brain Project, «Overview: Human Brain Project,» [Ηλεκτρονικό]. Available: <https://www.humanbrainproject.eu/en/about/overview/>. [Πρόσβαση 2 Απρίλιος 2021].

- [48] Human Brain Project, «Silicon Brains,» [Ηλεκτρονικό]. Available: <https://www.humanbrainproject.eu/en/silicon-brains/>. [Πρόσβαση 2 Απρίλιος 2021].
- [49] Human Brain Project, «Human Brain Project Framework Partnership Agreement,» [Ηλεκτρονικό]. Available: <https://web.archive.org/web/20170202001252/https://www.humanbrainproject.eu/documents/10180/538356/FPA++Annex+1+Part+B/41c4da2e-0e69-4295-8e98-3484677d661f>. [Πρόσβαση 2 Απρίλιος 2021].
- [50] «BrainScaleS,» 10 Φεβρουάριος 2019. [Ηλεκτρονικό]. Available: <https://brainscales.kip.uni-heidelberg.de/>.
- [51] P. A. Davison, E. Müller, S. Schmitt, B. Vogginger, D. Lester και T. Pfeil, «HBP Neuromorphic Computing Platform Guidebook,» 26 Ιουνίου 2019. [Ηλεκτρονικό]. Available: [https://electronicvisions.github.io/hbp-sp9-guidebook/pm/pm\\_hardware\\_configuration.html](https://electronicvisions.github.io/hbp-sp9-guidebook/pm/pm_hardware_configuration.html). [Πρόσβαση Μάιος 2019].
- [52] Human Brain Project, «Second Generation SpiNNaker Neuromorphic Supercomputer to be Built at TU Dresden - News,» 24 Σεπτέμβριος 2019. [Ηλεκτρονικό]. Available: <https://www.humanbrainproject.eu/en/follow-hbp/news/second-generation-spinnaker-neuromorphic-supercomputer-to-be-built-at-tu-dresden/>. [Πρόσβαση 2 Απρίλιος 2021].
- [53] E. M. Izhikevich, «Simple Model of Spiking Neurons,» *IEEE Transactions on Neural Networks*, τόμ. 14, αρ. 6, p. 1569–1572, 2003.
- [54] A. Cassidy και A. G. Andreou, «Dynamical Digital Silicon Neurons,» *Biomedical Circuits and Systems Conference*, p. 289–292, 2008.
- [55] A. Cassidy, S. Denham, P. Kanold και A. Andreou, «FPGA Based Silicon Spiking Neural Array,» *Biomedical Circuits and Systems Conference*, p. 75–78, 2007.
- [56] V. Bandeira, V. L. Costa, G. Bontorin και R. A. L. Reis, «Low Latency FPGA Implementation of Izhikevich-Neuron Model,» 2017, pp. 210-217.
- [57] S. Haghiri, A. Zahedi, A. Naderi και A. Ahmadi, «Multiplierless Implementation of Noisy Izhikevich Neuron with Low Cost Digital Design,» *IEEE Transactions on Biomedical Circuits and Systems*, p. 11, 2018.
- [58] A. J. Leigh, M. Mirhassani και R. Muscedere, «An Efficient Spiking Neuron Hardware System Based on the Hardware-Oriented Modified Izhikevich Neuron (HOMIN) Model,» *IEEE Transactions on Circuits and Systems II: Express Briefs*, pp. 3377 - 3381, 1 Απρίλιος 2020.
- [59] M. Sapounaki και A. P. Kakarountas, «A High-Performance Neuron for Artificial Neural Network based on Izhikevich model,» σε *International Symposium on Power and Timing*

*Modeling, Optimization and Simulation*, Ρόδος, Ελλάδα, 2019.

- [60] N. Eastland, «Structure of an FPGA,» 13 Αύγουστος 2015. [Ηλεκτρονικό]. Available: <https://blog.digilentinc.com/structure-of-an-fpga/>. [Πρόσβαση 2 Ιούνιος 2021].

## Πηγές Εικόνων

Εικόνα 1. Διάγραμμα ενός τυπικού νευρώνα και των συστατικών μερών του.

<https://commons.wikimedia.org/w/index.php?curid=97847412>

Εικόνα 2. Κατηγοριοποίηση των νευρώνων σε αισθητικούς, ενδιάμεσους και κινητήριους ανάλογα με τη λειτουργία τους.

<https://www.tutor2u.net/psychology/reference/biopsychology-sensory-relay-and-motor-neurons>

Εικόνα 3. Κατηγοριοποίηση των νευρώνων σε μονοπολικούς, διπολικούς, ψευδομονοπολικούς και πολυπολικούς ανάλογα με τη δομή τους.

<https://qbi.uq.edu.au/brain/brain-anatomy/types-neurons>

Εικόνα 4. Διάγραμμα χημικής σύναψης.

[https://commons.wikimedia.org/wiki/File:SynapseSchematic\\_en.svg](https://commons.wikimedia.org/wiki/File:SynapseSchematic_en.svg)

Εικόνα 5. Διάγραμμα ηλεκτρικής σύναψης.

[https://www.researchgate.net/figure/Representation-of-an-electrical-synapse-91\\_fig2\\_289479733](https://www.researchgate.net/figure/Representation-of-an-electrical-synapse-91_fig2_289479733)

Εικόνα 6. Διάγραμμα τεχνητού νευρώνα.

<https://hackernoon.com/a-hands-on-introduction-to-neural-networks-6a03afb468b1>

Εικόνα 7. Το ηλεκτρικό κύκλωμα του μοντέλου Hodgkin-Huxley.

D. Beeman, «Hodgkin-Huxley Model,» *Encyclopedia of Computational Neuroscience*, pp. 3, 14 Απρίλιος 2014.

Εικόνα 8. Το αντίστοιχο κύκλωμα του μοντέλου FitzHugh-Nagumo.

[http://www.scholarpedia.org/article/Bonhoeffer-van\\_der\\_Pol\\_oscillator](http://www.scholarpedia.org/article/Bonhoeffer-van_der_Pol_oscillator)

Εικόνα 9. Το αντίστοιχο κύκλωμα του μοντέλου Integrate and Fire. C είναι η μεμβρανική χωρητικότητα, R η μεμβρανική αντίσταση, V η μεμβρανική τάση,  $V_{rest}$  η μεμβρανική τάση ηρεμίας, I η ένταση του ρεύματος εισόδου.

<https://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.330.5032&rep=rep1&type=pdf>

Εικόνα 10. Σύγκριση της απόκρισης ενός Integrate and Fire νευρώνα και ενός Resonate and Fire νευρώνα.

<https://www.sciencedirect.com/science/article/pii/S0893608001000788>

Εικόνα 11. Η συνάρτηση STDP μίας σύναψης. Αν η προσυναπτική ακίδα φτάσει πριν τη μετασυναπτική ( $\Delta t > 0$ ), παρατηρείται μακροπρόθεσμη ενδυνάμωση. Αλλιώς ( $\Delta t < 0$ ) παρατηρείται μακροπρόθεσμη ύφεση.

<https://www.sciencedirect.com/science/article/pii/S1434841116313292>

Εικόνα 12. Η δομή ενός perceptron (αριστερά) και ενός MLP (δεξιά).

<https://machinelearningmastery.com/neural-networks-crash-course/>

Εικόνα 13. Αναπαράσταση Αναδρομικού Νευρωνικού Δικτύου στον τομέα της επεξεργασίας φυσικής γλώσσας. Ξεκινώντας από κάτω προς τα πάνω, οι είσοδοι (εδώ ρήματα V και φράσεις ουσιαστικών NP) συνδυάζονται για να προκύψει στην έξοδο η πρόταση S.

<https://machinelearningmastery.com/primer-neural-network-models-natural-language-processing/>

Εικόνα 14. Επαναλαμβανόμενο νευρωνικό δίκτυο.

<https://laptrinhx.com/rnn-or-recurrent-neural-network-for-noobs-374348076/>

Εικόνα 15. Ένα μονοεπίπεδο δίκτυο Hopfield.

[https://www.tutorialspoint.com/artificial\\_neural\\_network/artificial\\_neural\\_network\\_hopfield.htm](https://www.tutorialspoint.com/artificial_neural_network/artificial_neural_network_hopfield.htm)

Εικόνα 16. Μάθηση χαρακτηριστικών και ταξινόμηση χειρόγραφων αριθμών με τη χρήση συνελκτικού νευρωνικού δικτύου.

<https://becominghuman.ai/building-a-convolutional-neural-network-cnn-model-for-image-classification-116f77a7a236>

Εικόνα 17. Χαρακτηριστικά των πλακέτων Xilinx FPGA από το 1988 μέχρι το 2015. Η χωρητικότητα είναι ο αριθμός λογικών μπλοκ ανά πλακέτα, η ταχύτητα αφορά την εκτέλεση της ίδιας συνάρτησης σε κάθε πλακέτα, το κόστος και η ηλεκτρική κατανάλωση δίνονται ανά λογικό μπλοκ πολ/μένες με το 10000.

<https://abdullahyildiz.github.io/files/07086413.pdf>

Εικόνα 18. Τα στάδια διασωλήνωσης στους νευρώνες του δικτύου των Cassidy και Ανδρέου.

A. Cassidy και A. G. Andreou, «Dynamical digital silicon neurons,» Biomedical Circuits and Systems Conference, p. 289–292, 2008.

Εικόνα 19. Η αρχιτεκτονική του LIF δικτύου των Cassidy, Denham κ.α. στο οποίο βασίστηκε το Izhikevich δίκτυο των Cassidy και Ανδρέου. Με γκρι επισημαίνεται το συναπτικό κομμάτι της αρχιτεκτονικής.

A. Cassidy, S. Denham, P. Kanold και A. Andreou, «FPGA Based Silicon Spiking Neural Array,» Biomedical Circuits and Systems Conference, p. 75–78, 2007.

Εικόνα 20. Τα 7 στάδια διασωλήνωσης υπολογισμού των  $u$  και  $v$  στο paper του Ambroise και συνεργατών.

M. Ambroise, T. Levi, Y. Bornat και S. Saïghi, «Biorealistic Spiking Neural Network on FPGA,» σε 47th Annual Conference on Information Sciences and Systems (CISS), Baltimore, 2013.

Εικόνα 21. Τα δύο pipelines για τον υπολογισμό των διεγερτικών και ανασταλτικών ρευμάτων και της εκθετικής μείωσης τους στο paper του Ambroise και συνεργατών.

M. Ambroise, T. Levi, Y. Bornat και S. Saïghi, «Biorealistic Spiking Neural Network on FPGA,» σε 47th Annual Conference on Information Sciences and Systems (CISS), Baltimore, 2013.

Εικόνα 22. Τα στάδια διασωλήνωσης στο νευρώνα του paper των Bandeira, Costa κ.α.

V. Bandeira, V. L. Costa, G. Bontorin και R. A. L. Reis, «Low Latency FPGA Implementation of Izhikevich-Neuron Model,» 2017, pp. 210-217.

Εικόνα 23. Τα στάδια διασωλήνωσης στο νευρώνα του paper των Haghiri, Zahedi κ.α.

S. Haghiri, A. Zahedi, A. Naderi και A. Ahmadi, «Multiplierless Implementation of Noisy Izhikevich Neuron with Low Cost Digital Design,» IEEE Transactions on Biomedical Circuits and Systems, p. 11, 2018.

Εικόνα 24. Το συναπτικό κύκλωμα του paper των Haghiri, Zahedi κ.α.

A. J. Leigh, M. Mirhassani και R. Muscedere, «An Efficient Spiking Neuron Hardware System Based on the Hardware-Oriented Modified Izhikevich Neuron (HOMIN) Model,» IEEE Transactions on Circuits and Systems II: Express Briefs, pp. 3377 - 3381, 1 Απρίλιος 2020.

Εικόνα 25. Η προτεινόμενη αρχιτεκτονική

Εικόνα 26. Η προτεινόμενη σχεδίαση του πυρήνα υπολογισμού ρευμάτων.

Εικόνα 27. Η προτεινόμενη σχεδίαση του πυρήνα υπολογισμού εκθετικής μείωσης ρευμάτων.

Εικόνα 28. Τα αποτελέσματα της προσομοίωσης όσων αφορά τα συναπτικά ρεύματα.

Εικόνα 29. Αναφορά της κατανάλωσης ισχύος της υλοποίησης στο περιβάλλον Vivado.

Εικόνα 30. Η αρχιτεκτονική του spiking νευρωνικού δικτύου του paper του Ambroise και συνεργατών.

M. Ambroise, T. Levi, Y. Bornat και S. Saïghi, «Biorealistic Spiking Neural Network on FPGA,» σε 47th Annual Conference on Information Sciences and Systems (CISS), Baltimore, 2013.

Εικόνα 31. Το μπλοκ διατασσόμενης λογικής (CLB) ενός Xilinx 7 FPGA.

<https://blog.digilentinc.com/structure-of-an-fpga/>

Εικόνα 32. Πλακέτα ανάπτυξης του FPGA Xilinx Artix-7.

<https://www.distrelec.de/en/fpga-board-uart-usb-digilent-410-183/p/30044247>

