



ΕΘΝΙΚΟ ΚΑΙ ΚΑΠΟΔΙΣΤΡΙΑΚΟ ΠΑΝΕΠΙΣΤΗΜΙΟ ΑΘΗΝΩΝ

**ΣΧΟΛΗ ΘΕΤΙΚΩΝ ΕΠΙΣΤΗΜΩΝ
ΤΜΗΜΑ ΠΛΗΡΟΦΟΡΙΚΗΣ ΚΑΙ ΤΗΛΕΠΙΚΟΙΝΩΝΙΩΝ**

**ΔΙΑΤΜΗΜΑΤΙΚΟ ΠΡΟΓΡΑΜΜΑ ΜΕΤΑΠΤΥΧΙΑΚΩΝ ΣΠΟΥΔΩΝ
"ΤΕΧΝΟΛΟΓΙΕΣ ΠΛΗΡΟΦΟΡΙΚΗΣ ΣΤΗΝ ΙΑΤΡΙΚΗ ΚΑΙ ΤΗ ΒΙΟΛΟΓΙΑ"**

ΔΙΠΛΩΜΑΤΙΚΗ ΕΡΓΑΣΙΑ

**Έλεγχος ρομποτικού μηχανισμού μέσω
εγκεφαλικών σημάτων με χρήση πλατφόρμας
ανοικτού υλικού**

Νικόλαος Αγγέλου Καραγιάννης

Επιβλέπων: Διονύσιος Κάβουρας, Καθηγητής Τ.Ε.Ι Αθήνας

ΑΘΗΝΑ

ΔΕΚΕΜΒΡΙΟΣ 2014

ΔΙΠΛΩΜΑΤΙΚΗ ΕΡΓΑΣΙΑ

Έλεγχος ρομποτικού μηχανισμού μέσω
εγκεφαλικών σημάτων με χρήση πλατφόρμας
ανοικτού υλικού.

Νικόλαος Α. Καραγιάννης

A.M.: ΠΙΒ 094

ΕΠΙΒΛΕΠΩΝ: Διονύσιος Κάβουρας, Καθηγητής Τ.Ε.Ι Αθήνας

ΕΞΕΤΑΣΤΙΚΗ ΕΠΙΤΡΟΠΗ: Διονύσιος Κάβουρας, Καθηγητής Τ.Ε.Ι Αθήνας
Εμμανουήλ Σαγκριώτης, Καθηγητής Ε.Κ.Π.Α
Παντελεήμων Ασβεστάς, Επίκουρος Καθηγητής, Τ.Ε.Ι
Αθήνας

Δεκέμβριος 2014

ΠΕΡΙΛΗΨΗ

Ο εγκέφαλος είναι για πολλούς το σημαντικότερο όργανο του ανθρώπου. Είναι υπεύθυνος για τον έλεγχο των συναισθημάτων και των κινήσεών μας, αποθηκεύει και επεξεργάζεται πληροφορίες και ερεθίσματα που λαμβάνει από το εξωτερικό περιβάλλον καθώς επίσης και μηνύματα από άλλα όργανα, μέσα από ένα τεράστιο και πολύπλοκο δίκτυο δισεκατομμυρίων νευρώνων. Είναι το κέντρο λήψης αποφάσεων σχετικών με τη συμπεριφορά του ατόμου και τη ζωτική λειτουργία των υπολοίπων οργάνων του σώματος.

Για πολλές και διάφορες αιτίες όμως, προκαλούνται δυσλειτουργίες με αποτέλεσμα να υπάρχει είτε ολική, είτε μερική ανικανότητα ελέγχου της λειτουργίας των υπόλοιπων δομών του σώματος και των κινήσεων του ατόμου μέσα από το νευρικό του σύστημα.

Η ολοένα και περισσότερο εισαγωγή νέων τεχνολογιών στον τομέα των νευροεπιστημών, έχει οδηγήσει τους επιστήμονες στην ανάπτυξη συστημάτων τα οποία μπορούν να βοηθήσουν άτομα με μειωμένη κινητικότητα να αντιμετωπίσουν διάφορα προβλήματα που προκύπτουν από ενδεχόμενη δυσλειτουργία του νευρικού συστήματος. Τα συστήματα αυτά είναι γνωστά με τον όρο **Διεπαφή Εγκεφάλου – Υπολογιστή (Brain Computer Interface, B.C.I)** και κάνουν χρήση των εγκεφαλικών σημάτων για τον έλεγχο διάφορων συσκευών (π.χ αναπηρικό αμαξίδιο), ρομπότ και υπολογιστικών συστημάτων, παρακάμπτοντας το υπόλοιπο νευρικό σύστημα.

Σκοπός της παρούσας διπλωματικής εργασίας είναι η ανάπτυξη συστήματος B.C.I πραγματικού χρόνου, μέσα από το οποίο γίνεται ο έλεγχος της κίνησης ρομποτικού οχήματος με χρήση εγκεφαλικών σημάτων, έχοντας ως βάση ρομποτική πλατφόρμα ανοικτού υλικού και συγκεκριμένα του Arduino.

Η λήψη του εγκεφαλικού σήματος του χρήστη γίνεται με χρήση της συσκευής Mindwave Mobile της εταιρείας Neurosky. Η μεταφορά των δεδομένων από το Mindwave Mobile στο Arduino γίνεται ασύρματα με χρήση πρωτοκόλλου Bluetooth. Εν συνεχεία γίνεται εξαγωγή όλων των χρήσιμων εγκεφαλικών χαρακτηριστικών μέσα από τη μονάδα επεξεργασίας. Συγκεκριμένα, εξάγονται οι τιμές των επιπέδων προσοχής και συγκέντρωσης, των εγκεφαλικών ρυθμών και των ακατέργαστων δεδομένων. Από τα τελευταία, μετά από φιλτράρισμα, γίνεται ανίχνευση του βλεφαρισμού των ματιών και αναγνώριση πότε αυτός είναι απλός και πότε διπλός. Αν ο βλεφαρισμός είναι απλός, γίνεται εμπρόσθια κίνηση ενός αμαξιδίου που ελέγχεται από το Arduino, ενώ αν είναι διπλός η κίνηση είναι προς τα πίσω. Ταυτόχρονα, μέσω ενός αισθητήρα υπερήχων, ελέγχεται η απόσταση του αμαξιδίου από πιθανά εμπόδια κατά την εμπρόσθια κίνηση και όταν η αυτή γίνει ίση ή μικρότερη των 20cm το αμαξίδιο σταματάει να κινείται.

ΘΕΜΑΤΙΚΗ ΠΕΡΙΟΧΗ: Διεπαφή Εγκεφάλου - Υπολογιστή

ΛΕΞΕΙΣ ΚΛΕΙΔΙΑ: Διεπαφή Εγκεφάλου - Υπολογιστή, Arduino, Mindwave Mobile, Ρομποτική, Ηλεκτροεγκεφαλογράφημα

ABSTRACT

Human brain is a multifunctional organ, important for the entire function of body. It is responsible for controlling the emotions and the movements, it stores and processes information and stimuli received from the external environment, as well as messages from other organs, through a vast and complex network of billions of neurons. It is the decision-making center concerning with the behavior of the individual and the vital function of other organs of the body.

For many different reasons, however, human brain disabilities can result either in total or partial inability to control the operation of the remaining structures of the body and the movement of the individual through his/hers nervous system.

The increased use of new technologies in the field of neuroscience, has led scientists to develop systems that can help people with reduced mobility, to deal with various problems arising from any possible disability of the nervous system. These systems are known as Brain Computer Interface (B.C.I) and make use of brain signals to control various devices (e.g wheelchair), robots and computer systems, bypassing the disabled part of nervous system.

The aim of this thesis is to develop a real-time B.C.I system, which controls a robotic vehicle using brain signals, based on an open hardware robotic platform and specifically the Arduino.

The reception of user's brain signal is made by using Neurosky's Mindwave Mobile. The data transfer from Mindwave Mobile to Arduino is being made wirelessly using Bluetooth protocol. Thereafter, we export all the values of the useful features of brain through the procession unit. Specifically, there is exportation of the values of the levels of attention and concentration, of brain rhythms and raw data. By filtering raw data, there is the detection of eye blinking and the recognition whether it is a simple or a double eye blink. If the eye blinking is simple, a chassis, which is controlled by Arduino, moves forward and if it is double, the chassis moves backward. Simultaneously, an ultrasound sensor, checks the distance of the chassis from a possible obstacle during the forward motion and when it becomes equal to or less than 20cm chassis stops moving.

SUBJECT AREA: Brain Computer Interface

KEYWORDS: BCI, Arduino, Mindwave Mobile, Robotics, EEG

Στους γονείς μου.

ΕΥΧΑΡΙΣΤΙΕΣ

Για την πραγματοποίηση της παρούσας διπλωματικής εργασίας χρειάστηκαν αμέτρητες ώρες μελέτης, έρευνας και επικοινωνίας με οποιονδήποτε ήταν σε θέση να δώσει την παραμικρή βοήθεια σε ένα άγνωστο προς εμένα αντικείμενο. Βρισκόμενος μπροστά από την οθόνη ενός ηλεκτρονικού υπολογιστή και βλέποντας να περνάνε από τα μάτια μου χιλιάδες γραμμών, αναρωτιόμουν πολλές φορές, τι θα προκύψει από όλη αυτή την προσπάθεια και αν θα μπορέσω να τα καταφέρω. Ένωθα βαθιά μέσα μου την ανάγκη να βγει κάτι αξιοπρεπές αλλά πάνω απ' όλα χρήσιμο τόσο για την επιστήμη όσο και για τους ανθρώπους. Τον αρχικό προβληματισμό μου, λόγω της άγνοιας που είχα πάνω στο συγκεκριμένο τομέα, διαδέχθηκε η περιέργεια για το τι ακριβώς είναι το Brain Computer Interface και που μπορεί να έχει εφαρμογές. Καθώς έβρισκα όλο και περισσότερες πληροφορίες για το αντικείμενο, έρχονταν στο μυαλό μου καινούριες ιδέες προς υλοποίηση.

Θα ήθελα να ευχαριστήσω μέσα από την καρδιά μου τον υπεύθυνο του εργαστηρίου Επεξεργασίας Ιατρικής Εικόνας του Τ.Ε.Ι Αθήνας και επιβλέποντα της διπλωματικής, Δρ. Διονύσιο Κάβουρα, ο οποίος από την πρώτη κιόλας μέρα, με καλωσόρισε και με έκανε να νιώσω μέλος μιας ομάδας, στην οποία είναι εκείνος ηγέτης, και η οποία έχει ως στόχο την παραγωγή ουσιώδους και αποτελεσματικής έρευνας σε πρακτικό κυρίως επίπεδο.

Σε αυτό το δύσκολο εγχείρημα, βρήκα ακλόνητο στήριγμα στο πρόσωπο του Παντελή Ασβεστά, Επίκουρου Καθηγητή στο Τ.Ε.Ι Αθήνας, ο οποίος για 10 συνεχόμενους μήνες δεν έπαψε ποτέ να χαμογελάει και να είναι θετικά προδιατεθειμένος να με βοηθήσει, ακόμα και νυχτερινές ώρες. Ένα μεγάλο «ευχαριστώ» είμαι σίγουρος ότι δε φτάνει για να δείξει την ευγνωμοσύνη μου για όλη τη βοήθεια που μου προσέφερε, όχι μόνο σε επίπεδο γνωστικού αντικείμενου αλλά κυρίως σε επίπεδο ψυχολογικής υποστήριξης και συμπαράστασης.

Δεν μπορώ επίσης να μην ευχαριστήσω τους πολύ καλούς φίλους και συναδέλφους Ευριπίδη Μήκο, Ειρήνη – Κωνσταντίνα Νικολάτου και Γιώργο Ξενογιαννόπουλο, ερευνητές στο παραπάνω εργαστήριο, για την αμέριστη συμπαράστασή τους ιδιαίτερα σε στιγμές που τίποτα δεν έδειχνε να πηγαίνει σωστά.

Τέλος, οφείλω να ευχαριστήσω φίλες και φίλους που με τη μικρή του συμβουλή, πρόταση ή διόρθωση ο καθένας, συμμετείχαν ώστε να βγει το καλύτερο δυνατό αποτέλεσμα.

ΠΕΡΙΕΧΟΜΕΝΑ

ΠΡΟΛΟΓΟΣ	XV
1. ΕΙΣΑΓΩΓΗ	1
1.1 ΙΣΤΟΡΙΚΗ ΑΝΑΔΡΟΜΗ	1
1.1.1 Ερευνητές - Σκαπανείς	1
1.2 ΤΟ ΝΕΥΡΙΚΟ ΣΥΣΤΗΜΑ	4
1.2.1 Εισαγωγή.....	4
1.2.2 Δομή.....	4
1.2.3 Λειτουργία	4
1.2.4 Εγκέφαλος.....	6
1.3 ΗΛΕΚΤΡΟΕΓΚΕΦΑΛΟΓΡΑΦΙΑ (ELECTROENCEPHALOGRAPHY)	8
1.3.1 Γενικές πληροφορίες	8
1.3.2 Οι Εγκεφαλικοί Ρυθμοί (Brain Rhythms)	9
1.4 BRAIN COMPUTER INTERFACE (B.C.I)	11
1.4.1 Γενικά	11
1.4.2 Ιστορική εξέλιξη B.C.I.....	12
1.4.3 Κατηγορίες Brain – Computer Interfaces.....	13
1.4.4 Λειτουργία	14
1.4.5 Πλεονεκτήματα – Μειονεκτήματα των B.C.I	16
1.4.6 Εφαρμογές	17
1.4.7 Εμπορικά Συστήματα	17
1.4.8 Ηθικά ζητήματα	20
2. ΣΧΕΔΙΑΣΗ ΣΥΣΤΗΜΑΤΟΣ	21
2.1 Κατανόηση του σκοπού της εργασίας	21
2.2 Συγκέντρωση υλικών και εξαρτημάτων	22
2.3 Περιγραφή τρόπου λειτουργίας συσκευών	23
2.3.1 Mindwave Mobile	23
2.3.2 BlueSmirf RN- 42	33
2.3.3 Αισθητήρας Υπερήχων HC – SR04.....	34
2.3.4 Ολοκληρωμένο Κύκλωμα L293D	36
2.3.5 Arduino	36

2.4	Διάγραμμα ροής συστήματος	39
3.	ΥΛΟΠΟΙΗΣΗ ΣΥΣΤΗΜΑΤΟΣ	41
3.1	Συνδεσμολογία κυκλώματος ελέγχου	41
3.1.1	Σύνδεση Ολοκληρωμένου Κυκλώματος L293D	41
3.1.2	Σύνδεση Arduino με L293D.....	42
3.1.3	Σύνδεση αισθητήρα Υπερήχων HC – SR04 με Arduino	43
3.1.4	Σύνδεση Bluetooth module Bluesmirf RN - 42	43
3.1.5	Σύνδεση κινητήρων αμαξιδίου	44
3.2	Ρυθμίσεις επικοινωνίας και σύνδεσης Mindwave Mobile και Arduino.....	45
3.3	Υλοποίηση αλγορίθμου για τον έλεγχο του αμαξιδίου	47
3.3.1	Αρχικές ρυθμίσεις και δηλώσεις μεταβλητών	48
3.3.2	Ανάγνωση δεδομένων	48
3.3.3	Έλεγχος αρχικών bytes.....	49
3.3.4	Έλεγχος δεδομένων DataPayload	51
3.3.5	Απεικόνιση – Οπτικοποίηση εγκεφαλικών ρυθμών	55
3.3.6	Συνάρτηση ανίχνευσης βλεφαρισμού ματιών	58
3.3.7	Αλγόριθμος αναγνώρισης βλεφαρισμών. Κίνηση κινητήρων - αμαξιδίου.....	59
3.3.8	Αλγόριθμοι κίνησης κινητήρων	62
3.3.9	Αλγόριθμος αποφυγής εμποδίων με χρήση αισθητήρα υπερήχων	63
4.	ΣΥΜΠΕΡΑΣΜΑΤΑ	67
5.	ΜΕΛΛΟΝΤΙΚΕΣ ΕΠΕΚΤΑΣΕΙΣ	70
ΠΑΡΑΡΤΗΜΑ Α.....	73
ΠΑΡΑΡΤΗΜΑ Β.....	82
Γενικά περί κινητήρων	82
Ηλεκτρομαγνητική Επαγωγή	82
Δομή DC Κινητήρα	84
Βασική αρχή λειτουργίας DC κινητήρα	85
Γέφυρα H (H - Bridge).....	87

Είδη DC κινητήρων.....	89
Πλεονεκτήματα – Μειονεκτήματα DC Κινητήρων.....	89
ΑΝΑΦΟΡΕΣ.....	90

ΚΑΤΑΛΟΓΟΣ ΣΧΗΜΑΤΩΝ

Σχήμα 1: Διάγραμμα ροής επεξεργασίας πακέτου δεδομένων	32
Σχήμα 2: Σχηματικό διάγραμμα σύνδεσης BlueSmirf με Arduino	34
Σχήμα 3: Σχηματικό διάγραμμα σύνδεσης HC-SR04 με Arduino.....	35
Σχήμα 4: Σχηματικό διάγραμμα σύνδεσης IC L293D με Arduino.....	36
Σχήμα 5: Διάγραμμα ροής συστήματος	40
Σχήμα 6: Ανάγνωση λήψης δεδομένων από σειριακή θύρα	49
Σχήμα 7: Έλεγχος bytes κεφαλίδας πακέτου	50
Σχήμα 8: Έλεγχος bytes Payload Data	52
Σχήμα 9: Αλγόριθμος κατηγοριοποίησης βλεφαρισμών	60

ΚΑΤΑΛΟΓΟΣ ΕΙΚΟΝΩΝ

Εικόνα 1: Richard Caton	1
Εικόνα 2: Hans Berger.....	2
Εικόνα 3: George Bishop.....	2
Εικόνα 4: Herbert H. Jasper	2
Εικόνα 5: Edgar Douglas Adrian.....	2
Εικόνα 6: Franklin Offner	3
Εικόνα 7: Eli Goldensohn	3
Εικόνα 8: Ανθρώπινο Νευρικό Σύστημα	4
Εικόνα 9: Νευρώνες – Νευρογλοιακά Κύτταρα	4
Εικόνα 10: Τυπικός Εγκεφαλικός Νευρώνας	5
Εικόνα 11: Σύναψη	5
Εικόνα 12: Κύριες υποκατηγορίες εγκεφάλου	6
Εικόνα 13: Βασικές δομές ανθρώπινου εγκεφάλου	6
Εικόνα 14: Εγκεφαλικοί Λοβοί	7
Εικόνα 15: Εγκεφαλικό Στέλεχος (Brainstem).....	8
Εικόνα 16: Δέλτα ρυθμός.....	9
Εικόνα 17: Θήτα ρυθμός.....	9
Εικόνα 18: Άλφα ρυθμός	10
Εικόνα 19: Βήτα ρυθμός	10
Εικόνα 20: Γάμμα ρυθμός.....	10
Εικόνα 21: Εγκεφαλικοί ρυθμοί.....	11
Εικόνα 22: Jacques Vidal	12
Εικόνα 23: William Dobbelle	12
Εικόνα 24: Yang Dan.....	13
Εικόνα 25: Ray Lee	13
Εικόνα 26: Βασική Λειτουργία B.C.I.....	14

Εικόνα 27: Τυπική διάταξη ενός συστήματος B.C.I.....	15
Εικόνα 28: Διεθνές σύστημα τοποθέτησης ηλεκτροδίων 10-20	16
Εικόνα 29: Neural Pulse Actuator	18
Εικόνα 30: Emotiv Eroc.....	18
Εικόνα 31: Mindwave Mobile	19
Εικόνα 32: Muse.....	19
Εικόνα 33: OpenBCI	20
Εικόνα 34: Βασικό διάγραμμα project.....	21
Εικόνα 35: Mindwave Mobile	24
Εικόνα 36 Mindwave Mobile	25
Εικόνα 37 Mindwave Mobile	25
Εικόνα 38: ThinkGear ASIC Module (TGAM)	26
Εικόνα 39: Δομή ThinkGear Packet.....	28
Εικόνα 40: Δομή Data Row.....	28
Εικόνα 41: Διάγραμμα λειτουργίας αισθητήρα υπερήχων HC-SR04	35
Εικόνα 42: Arduino Uno Rev. 3.....	37
Εικόνα 43: Περιβάλλον προγραμματισμού Arduino	39
Εικόνα 44: Βασική συνδεσμολογία κυκλώματος ελέγχου	41
Εικόνα 45: Προετοιμασία συνδέσεων ολοκληρωμένου κυκλώματος	42
Εικόνα 46: Σύνδεση Arduino με L293D	42
Εικόνα 47: Σύνδεση αισθητήρα Υπερήχων HC – SR04 με Arduino.....	43
Εικόνα 48: Σύνδεση Bluetooth module Bluesmirf RN - 42	44
Εικόνα 49: Σύνδεση κινητήρων αμαξιδίου	44
Εικόνα 50: Τελική μορφή σχεδίου	45
Εικόνα 51: Διεύθυνση MAC Mindwave Mobile	46
Εικόνα 52: Ρυθμίσεις BlueSmirf.....	47
Εικόνα 53: Σειριακή Έξοδος Arduino	57

Εικόνα 54: Γραφική απεικόνιση εγκεφαλικών ρυθμών μέσω Processing	57
Εικόνα 55: Intel® Galileo Gen 2 Development Board	71
Εικόνα 56: Beagle Bone Black	71
Εικόνα 57: Raspberry Pi	72
Εικόνα 58: Rabbit	72
Εικόνα 59: Michael Faraday	82
Εικόνα 60: Αρνητική φορά ηλεκτρικού ρεύματος	83
Εικόνα 61: Θετική φορά ηλεκτρικού ρεύματος	83
Εικόνα 62: Άξονας	84
Εικόνα 63: Ρότορας	84
Εικόνα 64: Συλλέκτης	84
Εικόνα 65: Στάτης	84
Εικόνα 66: Ψήκτρες	84
Εικόνα 67: Διάγραμμα DC κινητήρα	85
Εικόνα 68: Συλλέκτης – Ψήκτρες	85
Εικόνα 69 Φόρτιση δαχτυλιδιών	86
Εικόνα 70 Εναλλαγή φόρτισης δαχτυλιδιών	86
Εικόνα 71: Κίνηση κινητήρα με έλξη ηλεκτρομαγνήτη	86
Εικόνα 72: Κίνηση κινητήρα με απώθηση ηλεκτρομαγνήτη	87
Εικόνα 73: Παύση κινητήρα	87
Εικόνα 74: Διάγραμμα Γέφυρας H	87
Εικόνα 75: Δεξιόστροφη λειτουργία κινητήρα	88
Εικόνα 76: Block diagram L293D.....	88

ΚΑΤΑΛΟΓΟΣ ΠΙΝΑΚΩΝ

Πίνακας 1: Λίστα κόστους υλικών και εξαρτημάτων	22
Πίνακας 2: Πίνακας επεξήγησης πεδίων κάθε Data Row	29
Πίνακας 3: Ανάλυση ακατέργαστων δεδομένων (raw data)	53
Πίνακας 4: Γράφημα ακατέργαστων δεδομένων.....	53
Πίνακας 5: Ανάλυση σήματος ASIC EEG Power	54
Πίνακας 6: Βλεφαρισμοί.....	58

ΠΡΟΛΟΓΟΣ

Η παρούσα διπλωματική εργασία πραγματοποιήθηκε εξ' ολοκλήρου στον χώρο του εργαστηρίου Επεξεργασίας Ιατρικού Σήματος και Εικόνας του Τ.Ε.Ι Αθήνας, το διάστημα από τα μέσα Ιανουαρίου 2014 έως και τα τέλη Νοεμβρίου 2014. Η αρχική ιδέα για την εργασία ανήκει στον επιβλέποντα, ο οποίος μετά από συζήτηση που είχαμε στο γραφείο του, όπου του επεσήμανα πως θα ήθελα να κάνω κάτι σχετικά με ρομποτικούς μηχανισμούς και αυτοματισμούς, μου πρότεινε να ρίξω μια ματιά σε διάφορους ιστότοπους αναζητώντας πληροφορίες σχετικές με το Brain Computer Interface.

Η εργασία αποτελείται από 4 κεφάλαια και 2 παραρτήματα.

Στο πρώτο κεφάλαιο γίνεται εισαγωγή στον τομέα του Brain Computer Interface και στους τομείς που αποτελούν τη βάση για την καλύτερη κατανόησή του. Ξεκινώντας από μια σύντομη παρουσίαση της ιστορική εξέλιξης που προηγήθηκε, αναλύονται βασικές έννοιες του ανθρώπινου νευρικού συστήματος και του εγκεφάλου και παρουσιάζεται η τεχνική της ηλεκτροεγκεφαλογραφίας που αποτελεί τον βασικό κορμό για την ανάπτυξη του B.C.I.

Στο δεύτερο κεφάλαιο αναλύεται ο σχεδιασμός του συστήματος. Αρχικά παρουσιάζονται τα απαραίτητα υλικά που χρησιμοποιήθηκαν και γίνεται ανάλυση του τρόπου λειτουργίας τους με αρκετές λεπτομέρειες. Εν συνεχεία, μέσω διαγραμμάτων ροής γίνεται επεξήγηση της λογικής η οποία ακολουθήθηκε για να μπορέσει να υλοποιηθεί ο στόχος μας.

Το τρίτο κεφάλαιο περιέχει οδηγίες καθώς επίσης εικόνες και σχέδια, για την υλοποίηση του συστήματος. Γίνεται επίσης τμηματική επεξήγηση του κώδικα που χρησιμοποιήθηκε ανάλογα με το βήμα που εκτελείται κάθε φορά. Ο ενδιαφερόμενος μπορεί να ακολουθήσει πιστά τα συγκεκριμένα βήματα με τη σειρά που παρουσιάζονται για να το πραγματοποιήσει και ο ίδιος.

Στο τέταρτο και τελευταίο κεφάλαιο, παρουσιάζονται τα συμπεράσματα από την ολοκλήρωση της εργασίας και γίνεται αναφορά σε μελλοντικές επεκτάσεις και σε περιορισμούς που υπήρχαν εξ' αρχής.

Στα δύο παραρτήματα που κλείνουν την εργασία παρατίθενται ο πλήρης κώδικας που υλοποιήθηκε καθώς επίσης και πληροφορίες σχετικές με τον τρόπο λειτουργίας των κινητήρων D.C

1. ΕΙΣΑΓΩΓΗ

1.1 ΙΣΤΟΡΙΚΗ ΑΝΑΔΡΟΜΗ

Ο εγκέφαλος αποτελεί για πολλούς το σημαντικότερο όργανο του ανθρώπου και η λειτουργία του ένα από τα μεγαλύτερα μυστήρια της φύσης. Πρόκειται για εκείνο το όργανο που ελέγχει τη σκέψη, τα συναισθήματα και τις κινήσεις μας, συλλέγοντας πληροφορίες από τα αισθητήρια όργανα και τα νευρικά κύτταρα του σώματος. Αποτελείται από περίπου 100 δισεκατομμύρια νευρώνες, οι οποίοι συνεργάζονται για να δημιουργήσουν ένα όργανο βάρους περίπου 1,5kg.

Οι δυνατότητες του ανθρώπινου εγκέφαλου είναι τεράστιες και ανεξερεύνητες. Μέσα στο απέραντο πλήθος των δυνατοτήτων του είναι και αυτή του ελέγχου, μέσω της παραγωγής ηλεκτρικών δυναμικών, συστημάτων και συσκευών όπως συστημάτων ήχου, φωτισμό και ηλεκτρικών συσκευών, αλλά και περισσότερο πολύπλοκων συστημάτων όπως αναπηρικά αμαξίδια και ρομποτικούς μηχανισμούς, αρκεί να μπορέσουν οι ιδέες και οι σκέψεις να μετατραπούν σε ηλεκτρικά σήματα κατάλληλης μορφής. Οι ερευνητές έχουν τα τελευταία χρόνια αναπτύξει μία περίπλοκη τεχνική για να επιτύχουν την επικοινωνία ανάμεσα στον ανθρώπινο εγκέφαλο και στους υπολογιστές. Έτσι, είναι δυνατό να ελεγχθούν διάφορα πολύπλοκα συστήματα κάνοντας χρήση μόνο των σκέψεων και των συναισθημάτων ενός ανθρώπου ακόμα και στην περίπτωση που τα περιφερειακά νεύρα δυσλειτουργούν.

Η τεχνική αυτή που επιτυγχάνει τη σύνδεση μεταξύ του εγκεφάλου και του υπολογιστή ονομάζεται **Διεπαφή Εγκεφάλου – Υπολογιστή** ή, όπως είναι ευρέως γνωστή, **Brain Computer Interface (B.C.I)**. Στηρίζεται στο γεγονός ότι κάθε εντολή που εκτελείται από τον εγκέφαλο μπορεί να αναπαρασταθεί με ηλεκτρικό τρόπο, αφού στην ουσία η επικοινωνία μεταξύ των νευρώνων γίνεται με ηλεκτρικά δυναμικά. Έτσι είναι δυνατό να καταγραφεί σχεδόν κάθε σκέψη, ιδέα και συναίσθημα του ανθρώπου και με γέφυρα τα ηλεκτρικά δυναμικά που παράγονται μπορεί να επιτευχθεί με κατάλληλο τρόπο, ο έλεγχος διαφόρων συστημάτων.

Ο τομέας εκείνος της ιατρικής επιστήμης που ασχολείται με τη μελέτη του εγκεφάλου και κατ' επέκταση με τη μελέτη και καταγραφή των ηλεκτρικών φαινομένων που λαμβάνουν χώρα, ονομάζεται **Νευροεπιστήμη** και η ανάπτυξη του ξεκίνησε στα τέλη του 18^{ου} αιώνα.

1.1.1 Ερευνητές - Σκαπανείς

Σημείο κλειδί για τη μελέτη και εξέλιξη τέτοιων φαινομένων υπήρξε η ανάπτυξη του **γαλβανομέτρου** από τον **Leopoldo Nobili** (1784 - 1835) στη Φλωρεντία το 1825 και η βελτίωσή του από τον **William Thompson** στην Αγγλία το 1858. Τα γαλβανόμετρα αυτά ήταν σε θέση να μετρήσουν συνεχή ηλεκτρικά ρεύματα, αλλά όχι στιγμιαία ηλεκτρικά φαινόμενα [1].

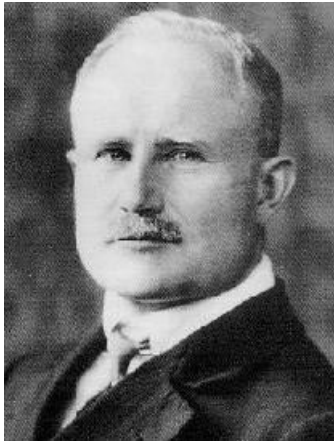
Ο Γερμανός φυσιολόγος **Emil Du Bois-Reymond** (1818 - 1896) επινόησε τον όρο «αρνητική διαφορά» για το φαινόμενο που συμβαίνει κατά τη διάρκεια μυϊκών συσπάσεων όταν το γαλβανόμετρο ανίχνευσε απροσδόκητη πτώση στην ένταση του ρεύματος [1].

Το 1875, ο Βρετανός επιστήμονας **Richard Caton** (1842 - 1926) (Εικόνα 1) ανέφερε ενώπιον της Βρετανικής Ιατρικής Ακαδημίας, ότι χρησιμοποίησε γαλβανόμετρο για να παρατηρήσει τις ηλεκτρικές δονήσεις στην επιφάνεια των εγκεφάλων ζωντανών οργανισμών [3].



Εικόνα 1: Richard Caton [2]

Το 1912, ο Ρώσος φυσιολόγος **Vladimir Vladimirovich Pravdich-Neminsky** (1879 - 1952), δημοσίευσε το πρώτο ηλεκτροεγκεφαλογράφημα ζώου και το προκλητό δυναμικό από ένα θηλαστικό [4].



Εικόνα 2: Hans Berger [5]

Μετά το θάνατο του Richard Caton, ο Γερμανός νευρολόγος **Hans Berger** (1873 - 1941) (Εικόνα 2) το 1924 χρησιμοποιώντας αρχικά ένα γαλβανόμετρο χορδής και μετέπειτα ένα γαλβανόμετρο διπλού πηνίου, κατάφερε να καταγράψει αυθόρμητες διακυμάνσεις τάσης του ανθρώπινου εγκεφάλου και να παρατηρήσει εξασθένιση κατά την αισθησιακή διέγερση. Επανάλαβε το πείραμα του Pravdich-Neminsky στον άνθρωπο παρατηρώντας τη δραστηριότητα των ζωνών συχνότητας **Alpha** και **Beta** και επινόησε τον όρο **Ηλεκτροεγκεφαλογράφημα** (Electroencephalogram, EEG) [6]. Στην ουσία πρόκειται για την πρώτη εμπειριστατωμένη αναφορά για την μέτρηση διαφορών δυναμικού από την εξωτερική επιφάνεια του ανθρώπινου κεφαλιού, γεγονός το οποίο οριοθετεί την έναρξη της μελέτης των λειτουργιών του εγκεφάλου μέσω του Ηλεκτροεγκεφαλογραφήματος (ΗΕΓ).

Δέκα χρόνια αργότερα, το 1934, οι Βρετανοί **Edgar Douglas Adrian** (1889 - 1977) (Εικόνα 3) και **Brian Matthews** επιβεβαίωσαν τα αποτελέσματα του Berger, χρησιμοποιώντας καινοτόμο διαφορικό ενισχυτή εισόδου που είχε εφεύρει ο Matthews [8].

Η έρευνα γύρω από το ηλεκτροεγκεφαλογράφημα επεκτάθηκε και στις Ηνωμένες Πολιτείες Αμερικής, με τους **George Bishop** (Εικόνα 4) και **S. Howard Bartley** να επιτυγχάνουν στις αρχές της δεκαετίας του 1930 την καταγραφή ιχνών ΗΕΓ από σκύλους στο Πανεπιστήμιο Washington του St. Louis [8].



Εικόνα 3:
George Bishop [9]

Την ίδια περίοδο, ο **Herbert H. Jasper** (Εικόνα 5) ήταν ο πρώτος στη Βόρειο Αμερική που επιβεβαίωσε τις παρατηρήσεις του Berger στο ανθρώπινο ΗΕΓ. Το 1934 σε συνεργασία με τον **Leonard Carmichael**, πέτυχαν την πρώτη καταγραφή ΗΕΓ στη δυτική πλευρά του Ατλαντικού και οι αναφορές τους περιγράφουν την παρεμπόδιση του Άλφα ρυθμού με διέγερση φωτός. Εν συνεχεία, ο Jasper συνεργάστηκε με τον **Wilder Penfield** με τον οποίο εγχείρησαν επιτυχώς δύο ασθενείς με επιληπτικές κρίσεις. Οι εγχειρήσεις αυτές ήταν η αφορμή για τον Jasper να μετακομίσει στο Montreal έτσι ώστε να

συνεχίσει το έργο του ιδρύοντας το Εργαστήριο Ηλεκτροεγκεφαλογραφήματος στο οποίο κατά το έτος 1939 είχαν γίνει περίπου 1000 ηλεκτροεγκεφαλογραφήματα σε πάνω από 500 επιληπτικούς ασθενείς [8].



Εικόνα 5:
Edgar Douglas Adrian [7]



Εικόνα 4:
Herbert H. Jasper [10]

Στα μέσα της δεκαετίας του 1930 τα συστήματα ηλεκτροεγκεφαλογραφήματος είχαν γίνει εμπορικά, ενώ στα μέσα της δεκαετίας του 1950 ο Αμερικανός **Franklin Offner** (Εικόνα 6) κατασκεύασε τον πρώτο ηλεκτροεγκεφαλογράφο που χρησιμοποιούσε ενισχυτές με τρανζίστορ, τα οποία είχαν εφευρεθεί το 1947 στα εργαστήρια Bell. Ο Offner κατασκεύασε επίσης τον **ηλεκτροεγκεφαλογράφο Τύπου - Τ** που αποτελούσε μία φορητή έκδοση ενός ηλεκτροεγκεφαλογράφου και ο οποίος απαρτιζόταν από δύο κουτιά σε μέγεθος βαλίτσας ταξιδιού [8].



Εικόνα 6:
Franklin Offner [11]

Το 1958 ο Jasper ηγείται μιας επιτροπής για την τυποποίηση του συστήματος 10/20 το οποίο σχετίζεται με τον τρόπο τοποθέτησης καλωδίων πάνω στο κρανίο.

Η εξέλιξη της τεχνολογίας που ακολούθησε μετά την ανακάλυψη των τρανζίστορ, είχε τεράστιες επιδράσεις και στον τομέα του σχεδιασμού και κατασκευής ηλεκτροεγκεφαλογράφων. Η μετάβαση από την αναλογική στην ψηφιακή εποχή κατά τις δεκαετίες του 1960 και 1970, συνέβαλλε στην προσθήκη σημαντικών δυνατοτήτων όπως η αποθήκευση και ανάκληση του σήματος, η ψηφιακή επεξεργασία και η σταθερή απεικόνιση σε πραγματικό χρόνο.



Εικόνα 7:
Eli Goldensohn [12]

Το 1966, ο **Eli Goldensohn** (Εικόνα 7) κατασκεύασε το πρώτο κλειστό κύκλωμα καταγραφής ηλεκτροεγκεφαλογραφήματος ενώ από τη δεκαετία του 1980 οι ψηφιακοί ηλεκτροεγκεφαλογράφοι έγιναν κυρίαρχοι σε διάφορα πεδία κλινικών ερευνών κυρίως μέσω της δυνατότητας τοπογραφικής αποτύπωσης που είχαν.

Αργότερα, κατά τη δεκαετία του 1990, τυποποιήθηκε η απεικόνιση σε πραγματικό χρόνο μέσω ψηφιακού ηλεκτροεγκεφαλογράφου και βελτιώθηκαν οι αλγόριθμοι επεξεργασίας σήματος παρέχοντας τη δυνατότητα στους ηλεκτροεγκεφαλογράφους να αποτελέσουν ξεχωριστό αντικείμενο έρευνας στο πεδίο των γνωστικών νευροεπιστημών [13].

1.2 ΤΟ ΝΕΥΡΙΚΟ ΣΥΣΤΗΜΑ

1.2.1 Εισαγωγή

Το νευρικό σύστημα είναι μοναδικό όσον αφορά στην πολυπλοκότητα που σχετίζεται με τις διαδικασίες σκέψης και τις ενέργειες ελέγχου τις οποίες πραγματοποιεί. Κάθε λεπτό λαμβάνει εκατομμύρια bits πληροφορίας από τα διάφορα αισθητηριακά νεύρα και όργανα και στη συνέχεια ενοποιεί όλες αυτές τις πληροφορίες για να προσδιορίσει τις ανάλογες αποκρίσεις που θα πραγματοποιηθούν από το σώμα [14].

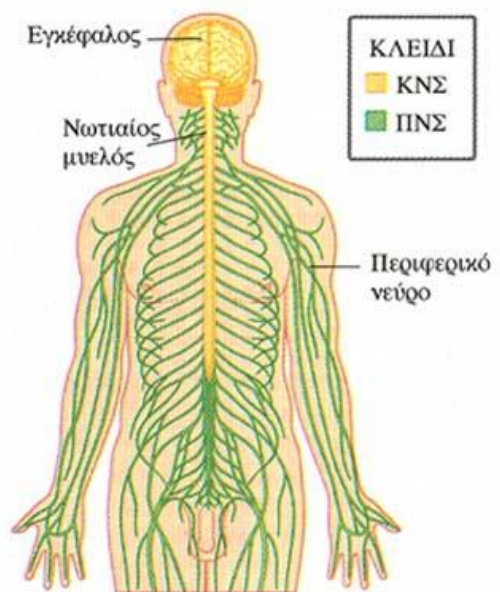
1.2.2 Δομή

Το ανθρώπινο νευρικό σύστημα (Εικόνα 8) χωρίζεται σε δύο μεγάλες κατηγορίες:

1. Το **Κεντρικό Νευρικό Σύστημα** (Central Nervous System, C.N.S)
2. Το **Περιφερειακό Νευρικό Σύστημα** (Peripheral Nervous System, P.N.S)

Το Κεντρικό Νευρικό Σύστημα περιλαμβάνει τον **εγκέφαλο**, τα **κρανιακά νεύρα** και τον **νωτιαίο μυελό**, ενώ το Περιφερειακό Νευρικό Σύστημα περιλαμβάνει το σύνολο των νεύρων που ξεκινούν από τον νωτιαίο μυελό σε διάφορα επίπεδα της σπονδυλικής στήλης καθώς επίσης και τα παρακλάδια τους. Ο ανθρώπινος εγκέφαλος απαρτίζεται από δύο τύπους κυττάρων [16]:

- Τους **νευρώνες** (Neurons) που σχετίζονται με όλες τις εγκεφαλικές λειτουργίες



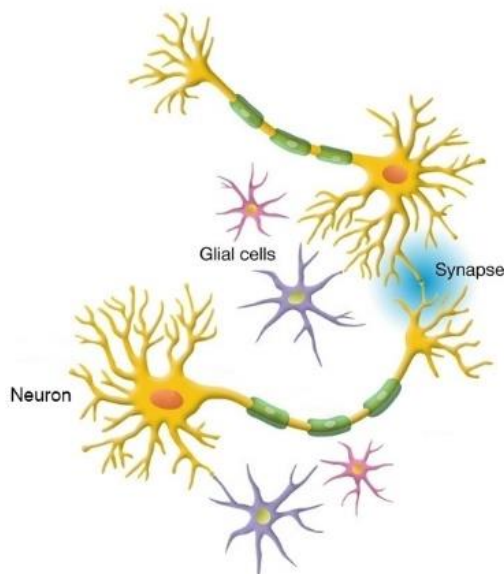
Εικόνα 8: Ανθρώπινο Νευρικό Σύστημα [15]

- Τα **νευρογλοιακά κύτταρα** (Glial Cells) (Εικόνα 9) τα οποία είναι μη – νευρωνικά κύτταρα και υποστηρίζουν τη λειτουργία των νευρώνων.

Στο περιφερειακό νευρικό σύστημα κατατάσσεται και το αυτόνομο νευρικό σύστημα που ελέγχει τις βασικές λειτουργίες όπως η αναπνοή και το ανοιγοκλείσιμο των ματιών [16].

1.2.3 Λειτουργία

Το Κεντρικό Νευρικό Σύστημα περιλαμβάνει περισσότερα από 100 δισεκατομμύρια κύτταρα που ονομάζονται **νευρώνες** (Εικόνα 10).



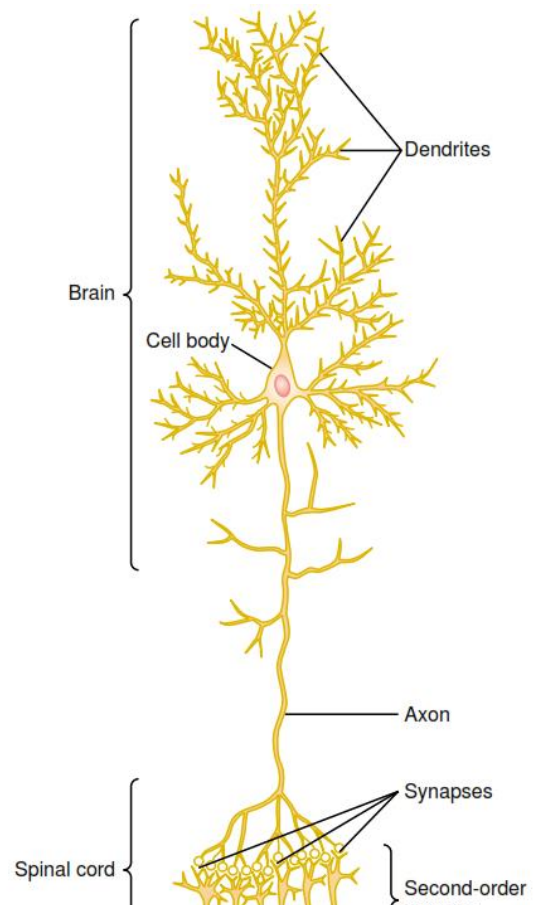
Εικόνα 9: Νευρώνες – Νευρογλοιακά Κύτταρα [16]

Τα σήματα εισέρχονται στο νευρώνα μέσω των συνάψεων που υπάρχουν συνήθως στους **δενδρίτες** (dendrites) αλλά και στο **σώμα του κυττάρου** (cell body). Αντίστροφα, τα εξερχόμενα σήματα ταξιδεύουν κατά μήκος του **άξονα** (αχον) που φεύγει από το νευρώνα. Ο άξονας αυτός εν συνεχεία συνδέεται μέσω διακλαδώσεων με άλλα σημεία του νευρικού συστήματος ή με το περιφερειακό σώμα [14].

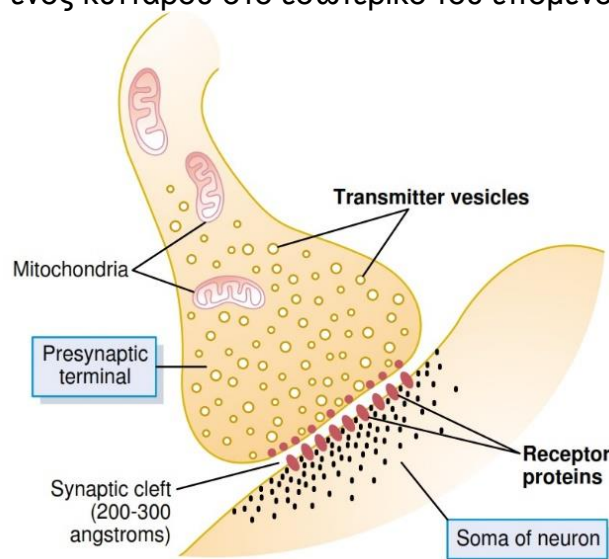
Με τον όρο **σύναψη** (synapse) (Εικόνα 11) ορίζεται η μεταφορά πληροφορίας στο κεντρικό νευρικό σύστημα υπό τη μορφή δυναμικών δράσης μέσω διαδοχικών νευρώνων. Υπάρχουν δύο είδη συνάψεων [14]:

- Η **χημική σύναψη** (chemical synapse). Όλες σχεδόν οι συνάψεις για τη μεταφορά σημάτων στο κεντρικό νευρικό σύστημα του ανθρώπου είναι χημικές συνάψεις. Σε αυτές ο αρχικός νευρώνας, εκκρίνει στη νευρική απόληξη συνάψεων μία ουσία που ονομάζεται νευροδιαβιβαστής (neurotransmitter) και αυτός με τη σειρά του δρα στις πρωτεΐνες υποδοχέων στη μεμβράνη του επόμενου νευρώνα για να τον διεγείρει, να αναστείλει τη λειτουργία του ή για να τροποποιήσει την ευαισθησία του.

- Η **ηλεκτρική σύναψη** (electrical synapse). Οι ηλεκτρικές συνάψεις, εν αντιθέσει με τις χημικές, χαρακτηρίζονται από άμεσα ανοικτά κανάλια τα οποία δημιουργούν ηλεκτρισμό από το ένα κύτταρο στο επόμενο. Οι περισσότερες από αυτές, αποτελούνται από πρωτεϊνικές σωληνοειδείς δομές, που επιτρέπουν την ελεύθερη μετακίνηση ιόντων από το εσωτερικό του ενός κυττάρου στο εσωτερικό του επόμενου.



Εικόνα 10: Τυπικός Εγκεφαλικός Νευρώνας [14]



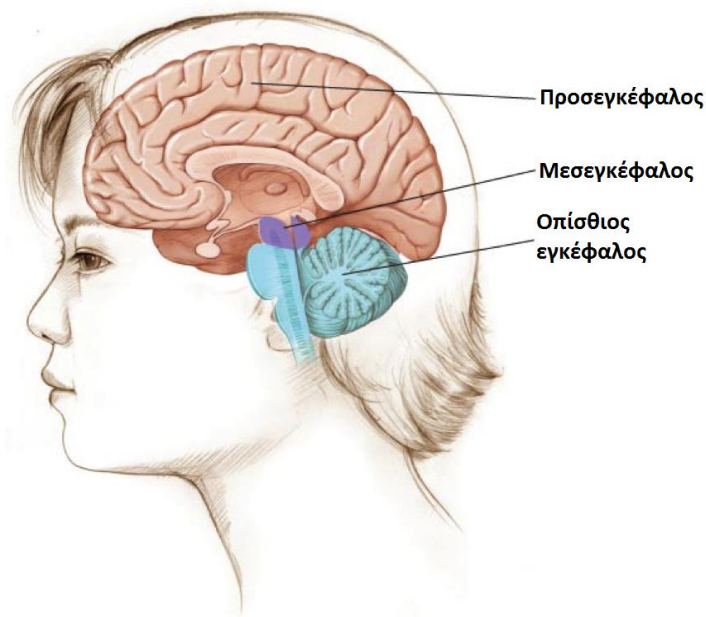
Εικόνα 11: Σύναψη [14]

1.2.4 Εγκέφαλος

Ο εγκέφαλος αποτελεί το πιο πολύπλοκο μέρος του ανθρώπινου σώματος. Είναι το όργανο εκείνο που ελέγχει τις αισθήσεις, τις σκέψεις, τις κινήσεις και είναι η πηγή των εντολών προς τα υπόλοιπα όργανα του σώματος. Ζυγίζει περίπου 1,5 kg και αποτελείται από 100 δισεκατομμύρια νευρώνες.

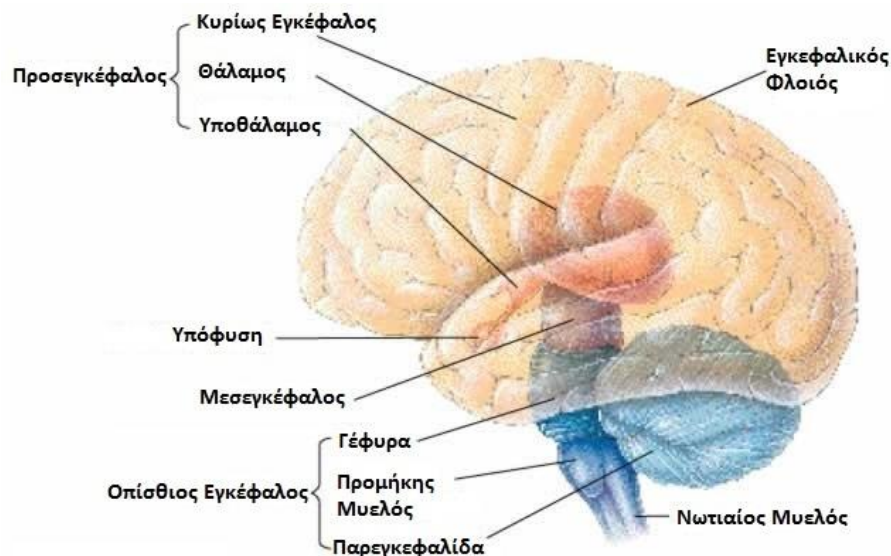
Ο εγκέφαλος διαιρείται σε 3 κύριες υποκατηγορίες (Εικόνα 12):

- Τον **Προσεγκέφαλο** (Forebrain)
- Το **Μεσεγκέφαλο** (Midbrain)
- Τον **Οπίσθιο Εγκέφαλο** (Hindbrain)



Εικόνα 12: Κύριες υποκατηγορίες εγκεφάλου [17]

Κάθε μία από τις παραπάνω υποκατηγορίες, αποτελείται από επιμέρους κομμάτια (Εικόνα 13).



Εικόνα 13: Βασικές δομές ανθρώπινου εγκεφάλου [18]

Πιο αναλυτικά:

Ο προσεγγέφαλος περιλαμβάνει :

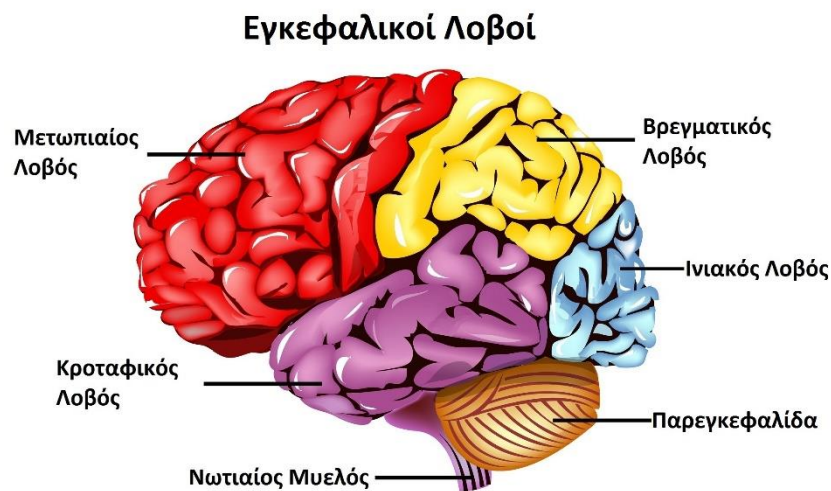
1. Τον **Κύριο Εγκέφαλο** (Cerebrum) που βρίσκεται στο ανώτερο μέρος του εγκεφάλου και είναι η πηγή των διανοητικών δραστηριοτήτων του ανθρώπου. Στο τμήμα αυτό υπάρχουν οι αναμνήσεις, η δυνατότητα του σχεδιασμού και της φαντασίας, της ανάγνωσης βιβλίων και της συμμετοχής σε παιχνίδια. Αποτελείται από ημισφαίρια που διαχωρίζονται με μία βαθιά σχισμή, η οποία δεν αποτρέπει την μεταξύ τους επικοινωνία, αφού αυτή μπορεί και γίνεται μέσω ενός πυκνού συστήματος νευρικών ινών.

Το αριστερό ημισφαίριο ελέγχει το δεξί ήμισυ του σώματος αισθητικά και κινητικά και σε αυτό γίνονται κυρίως οι λεκτικές, αναλυτικές και λογικές διεργασίες. Είναι επίσης υπεύθυνο για την αντίληψη του χρόνου, την ομιλία, τη γραφή, την αντίληψη του λόγου, τον συμβολισμό, τη λεκτική μνήμη και την αναλυτική σκέψη.

Το δεξί ημισφαίριο του εγκεφάλου ελέγχει το αριστερό ήμισυ του σώματος αισθητικά και κινητικά και είναι υπεύθυνο για την οπτική αντίληψη του χώρου, τη συσχέτιση, τη συναισθηματική φόρτιση, την επικοινωνία, τόσο με τον τόνο της φωνής όσο και με εκφράσεις του προσώπου.

Κάθε ημισφαίριο αποτελείται από τους παρακάτω 4 λοβούς (Εικόνα 14):

- Μετωπιαίο Λοβό (Frontal Lobe)
- Κροταφικό Λοβό (Temporal Lobe)
- Βρεγματικό Λοβό (Parietal Lobe)
- Ινιακό Λοβό (Occipital Lobe)



Εικόνα 14: Εγκεφαλικοί Λοβοί [19]

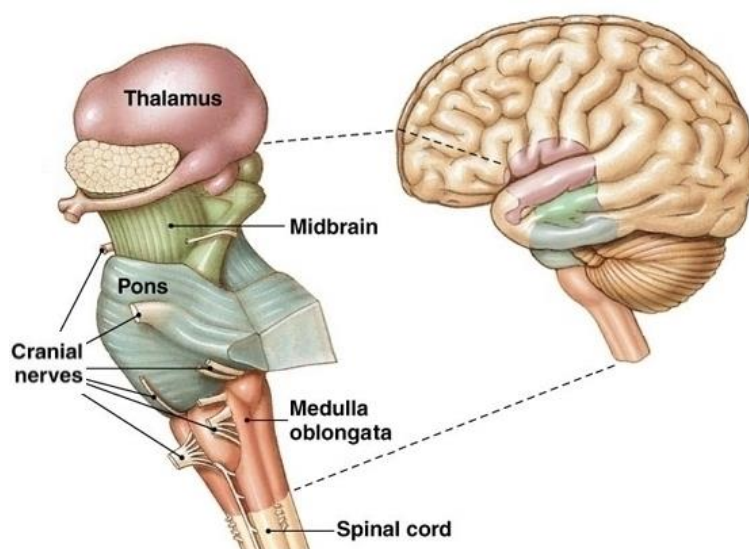
2. Τον **Θάλαμο** (Thalamus) που λειτουργεί ως χώρος αναμονής για τα μηνύματα που ταξιδεύουν από τον εγκεφαλικό φλοιό προς το υπόλοιπο σώμα και αντίστροφα.

3. Τον **Υποθάλαμο** (Hypothalamus), ο οποίος με σχήμα μαργαριταριού, κατευθύνει ένα πλήθος σημαντικών λειτουργιών. Πρόκειται για μία δομή που επικοινωνεί με την υπόφυση, προκειμένου να διαχειριστεί εκκρίσεις ορμονών καθώς και τον έλεγχο των λειτουργιών, όπως το φαγητό, το ποτό, τη σεξουαλική συμπεριφορά, τον ύπνο, τη θερμοκρασία του σώματος και τα συναισθήματα.

Ο μεσεγκέφαλος βρίσκεται κάτω ακριβώς από τον Θάλαμο και αποτελεί το βορειότερο τμήμα του **Εγκεφαλικού Στελέχους** (Brainstem) έχοντας τον έλεγχο κάποιων αντανακλαστικών κινήσεων και συμμετέχοντας επίσης στον έλεγχο της κίνησης των ματιών.

Κάτω ακριβώς από τον μεσεγκέφαλο και σε συνέχεια του εγκεφαλικού στελέχους (Εικόνα 15), βρίσκεται ο οπίσθιος εγκέφαλος που αποτελείται από:

1. Τη **Γέφυρα** (Pons), που έχει τον συντονισμό των κινήσεων των ματιών και του προσώπου καθώς επίσης και την αίσθηση του προσώπου, της ακοής και της ισορροπίας.
2. Τον **Προμήκη Μυελό** (Medulla Oblongata), ο οποίος ελέγχει τις βασικές λειτουργίες, όπως η αναπνοή, η αρτηριακή πίεση, ο καρδιακός ρυθμός καθώς επίσης και την κατάποση.
3. Δεξιά από τον προμήκη μυελό και αποτελώντας μέρος του οπίσθιου εγκεφάλου, βρίσκεται η **Παρεγκεφαλίδα** (Cerebellum), η οποία όμως δεν αποτελεί μέρος του εγκεφαλικού στελέχους. Είναι υπεύθυνη για τη διατήρηση της ισορροπίας και του συντονισμού των κινήσεων.



Εικόνα 15: Εγκεφαλικό Στέλεχος (Brainstem) [16]

1.3 ΗΛΕΚΤΡΟΕΓΚΕΦΑΛΟΓΡΑΦΙΑ (ELECTROENCEPHALOGRAPHY)

1.3.1 Γενικές πληροφορίες

Η σύγχρονη Ιατρική εφαρμόζει διάφορες τεχνικές απεικόνισης του ανθρώπινου σώματος. Το σύνολο των ιατροβιολογικών μετρήσεων περιλαμβάνει τεχνικές όπως το **Ηλεκτροκαρδιογράφημα** (ElectroCardioGraphy, ECG), το **Ηλεκτρομυογράφημα** (ElectroMyoGraphy, EMG), το **Ηλεκτροεγκεφαλογράφημα** (ElectroEncephaloGraphy, EEG), το **Μαγνητοεγκεφαλογράφημα** (MagnetoEncephaloGraphy, MEG) και το **Ηλεκτρογαστρογράφημα** (ElectroGastroGraphy, EGG).

Το ηλεκτροεγκεφαλογράφημα αποτελεί μία τεχνική ιατρικής απεικόνισης η οποία καταγράφει την κρανιακή ηλεκτρική δραστηριότητα που παράγεται από τις εγκεφαλικές δομές. Ορίζεται ως «η ηλεκτρική δραστηριότητα ενός εναλλασσόμενου τύπου ο οποίος καταγράφεται στην επιφάνεια του κρανίου αφού πρώτα έχει συλληχθεί μέσω μεταλλικών ηλεκτροδίων και αγώγιμων υλικών» [1]

Το ηλεκτροεγκεφαλογράφημα μετράει κυρίως το ρεύμα που ρέει κατά τις συναπτικές διεγέρσεις των δενδριτών πολλών πυραμιδωτών νευρώνων του εγκεφαλικού φλοιού [20]. Οι διαφορές ηλεκτρικών δυναμικών προκαλούνται από τα συνολικά μετασυναπτικά δυναμικά των πυραμιδωτών κυττάρων, τα οποία δημιουργούν ηλεκτρικά δίπολα ανάμεσα στο σώμα του νευρώνα και στις νευρικές διακλαδώσεις.

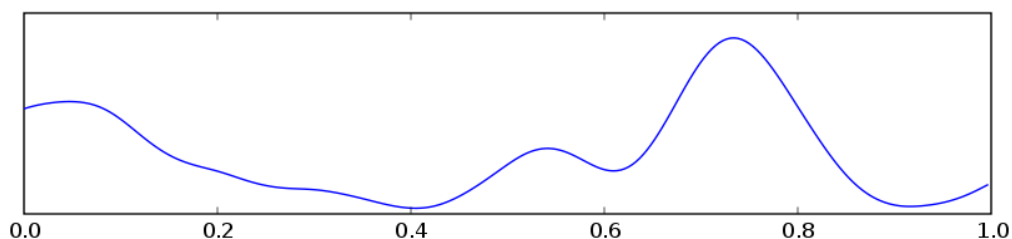
Το ηλεκτρικό ρεύμα στον εγκέφαλο αποτελείται κυρίως από ιόντα Na^+ , K^+ , Ca^{++} και Cl^- τα οποία διαχέονται μέσω καναλιών στις νευρωνικές μεμβράνες κατά την κατεύθυνση του δυναμικού των μεμβρανών [21]. Μόνο μεγάλοι πληθυσμοί νευρώνων μπορούν να παράγουν ηλεκτρική δραστηριότητα ικανή να καταγραφεί από την επιφάνεια του κεφαλιού. Ανάμεσα στο ηλεκτρόδιο καταγραφής και στο επίπεδο των νευρώνων, το ηλεκτρικό ρεύμα διεισδύει μέσω του δέρματος, του κρανίου και πολλών άλλων επιπέδων. Τα ασθενή ηλεκτρικά σήματα που ανιχνεύονται, ενισχύονται αρκετά και εν συνεχεία αποθηκεύονται στη μνήμη του υπολογιστή ή εκτυπώνονται [22].

1.3.2 Οι Εγκεφαλικοί Ρυθμοί (Brain Rhythms)

Οι ηλεκτρικές καταγραφές από την επιφάνεια του εγκεφάλου δείχνουν ότι υπάρχει ηλεκτρική δραστηριότητα. Τόσο η ένταση όσο και οι σχηματισμοί της ηλεκτρικής δραστηριότητας, καθορίζονται από το επίπεδο διέγερσης διαφορετικών σημείων του εγκεφάλου που προκαλούνται από την κατάσταση του ύπνου ή της εγρήγορσης ή από διάφορες παθήσεις όπως η επιληψία και οι ψυχώσεις. Οι κυματισμοί που υπάρχουν στο καταγεγραμμένο ηλεκτρικό σήμα ονομάζονται **Εγκεφαλικοί Ρυθμοί** (Brain Rhythms) ή **Εγκεφαλικά Κύματα** (Brain Waves).

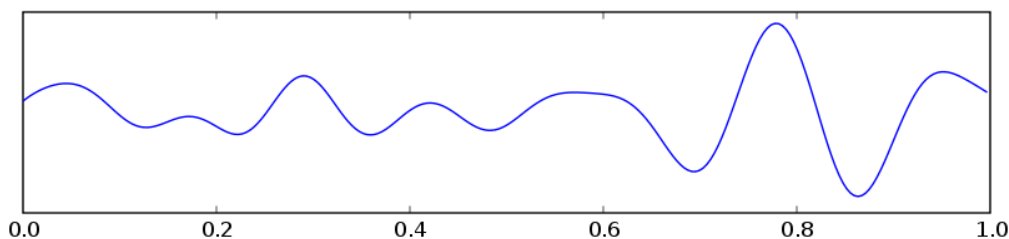
Οι κυριότεροι εγκεφαλικοί ρυθμοί που είναι παγκοσμίως αποδεκτοί και χρησιμοποιούνται από τους ερευνητές είναι οι ακόλουθοι:

- **Δέλτα** (Delta): Ο ρυθμός αυτός εντοπίζεται σε συχνότητες μικρότερες των 4 Hz και σχετίζεται με την απόλυτη χαλαρότητα και ηρεμία. Χαρακτηρίζεται από βαθύ ύπνο χωρίς όνειρα.



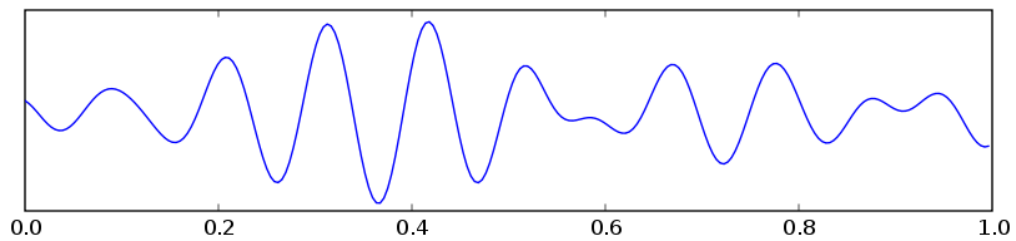
Εικόνα 16: Δέλτα ρυθμός [23]

- **Θήτα** (Theta): Ο ρυθμός αυτός εντοπίζεται σε συχνότητες από 4 Hz έως και 7 Hz και σχετίζεται με σχετική ηρεμία και μικρή εγρήγορση. Χαρακτηρίζεται από ταχεία εκμάθηση κυρίως στα παιδιά, θέληση για επίτευξη στόχων, υιοθέτηση νέων συμπεριφορών και έντονη δημιουργικότητα.



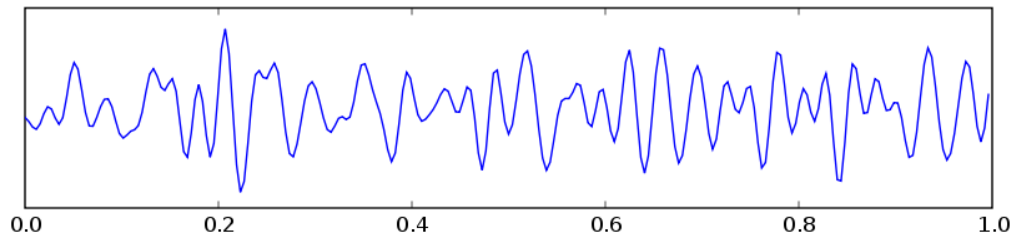
Εικόνα 17: Θήτα ρυθμός [23]

- **Άλφα** (Alpha): Ο ρυθμός αυτός εντοπίζεται σε συχνότητες από 8 Hz έως και 15 Hz και σχετίζεται με την ηρεμία και ψυχραιμία ενός ατόμου και παράλληλα το άτομο αυτό να είναι πνευματικά ενεργό. Χαρακτηρίζεται από διαλογισμό και όνειρα κατά τη διάρκεια της ημέρας.



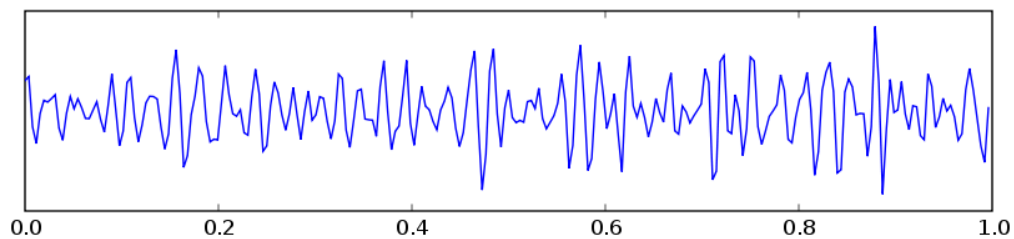
Εικόνα 18: Άλφα ρυθμός [23]

- **Βήτα** (Beta): Ο ρυθμός αυτός εντοπίζεται σε συχνότητες από 16 Hz έως και 31 Hz και χαρακτηρίζεται από έντονη, λογική και αναλυτική σκέψη, προφορική επικοινωνία και αγχώδη συμπεριφορά.



Εικόνα 19: Βήτα ρυθμός [23]

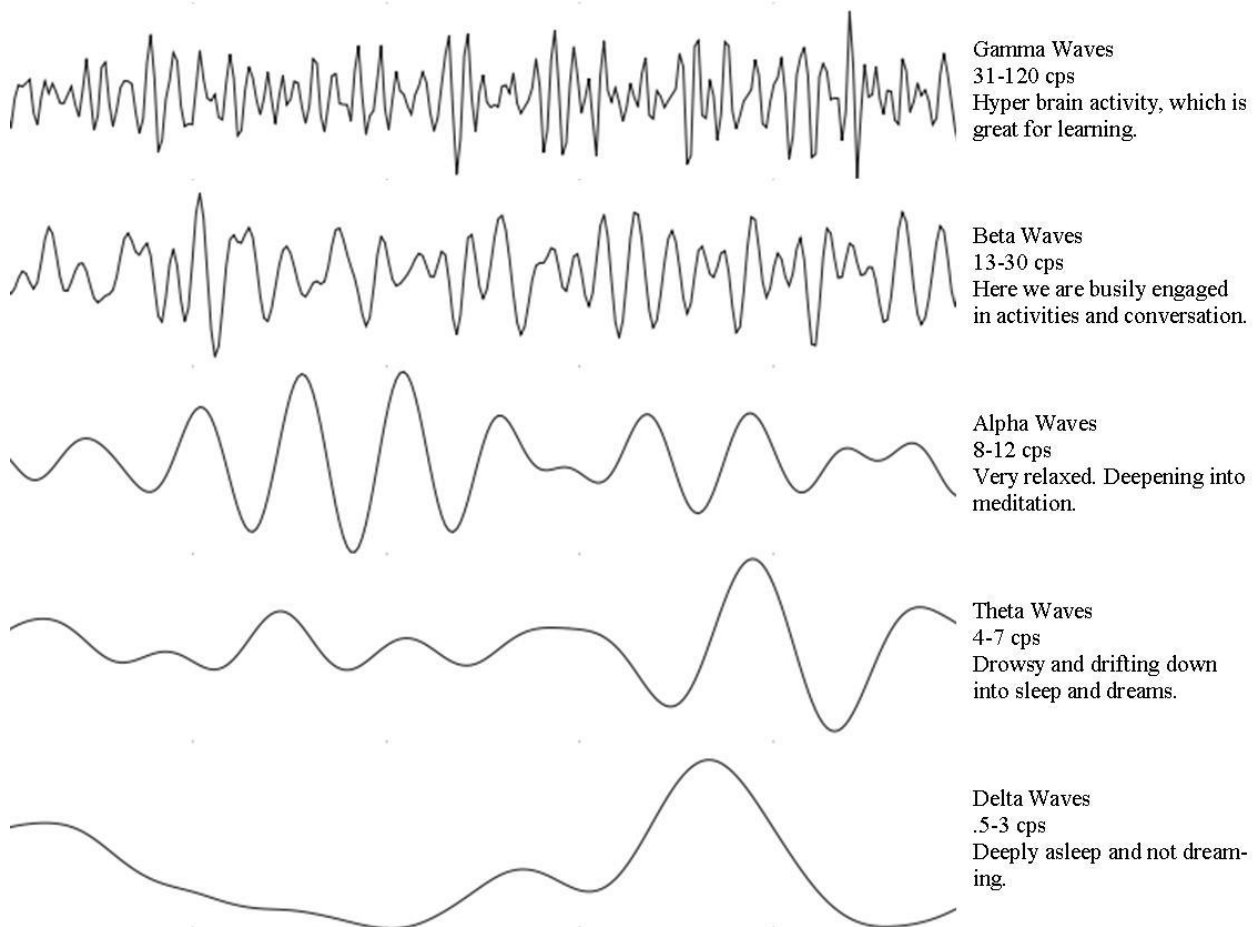
- **Γάμμα** (Gamma): Ο ρυθμός αυτός εντοπίζεται σε συχνότητες μεγαλύτερες από 31 Hz και χαρακτηρίζεται από την εκμάθηση ή επεξεργασία νέων πληροφοριών. Επίσης ο ρυθμός Γάμμα σχετίζεται με αισθητικές και μηχανικές λειτουργίες.



Εικόνα 20: Γάμμα ρυθμός [23]

Στην ακόλουθη εικόνα (Εικόνα 21) φαίνονται όλοι οι εγκεφαλικοί ρυθμοί συγκεντρωτικά.

Brain Waves Graph



Εικόνα 21: Εγκεφαλικοί ρυθμοί [24]

1.4 BRAIN COMPUTER INTERFACE (B.C.I)

1.4.1 Γενικά

Σύμφωνα με τον Jonathan R. Wolpaw [25], το B.C.I είναι ένα σύστημα επικοινωνίας μέσω του οποίου μηνύματα ή εντολές που ένας άνθρωπος στέλνει προς τον έξω κόσμο δεν περνάνε μέσα από τις κανονικές εγκεφαλικές διόδους εξόδου, όπως τα περιφερειακά νεύρα και οι μύες. Ένα σύστημα B.C.I παρέχει στον χρήστη μία εναλλακτική μέθοδο για να δράσει και να επικοινωνήσει με τον έξω κόσμο. Γενικότερα, ο όρος **Brain Computer Interface** (B.C.I) αναφέρεται σε ένα εύρος μεθόδων και τεχνολογιών με τον συνδυασμό των οποίων επιτυγχάνεται μία απευθείας διεπαφή προς το νευρικό σύστημα μέσω υπολογιστικών συστημάτων [26]. Στην ουσία πρόκειται για περιφερειακή συσκευή η οποία δίνει τη δυνατότητα στον χρήστη να επικοινωνήσει με έναν υπολογιστή μέσω εγκεφαλικών σημάτων.

Η τεχνολογία που χρησιμοποιείται στο B.C.I είναι συνδυασμός πολλών επιστημών όπως η νευροβιολογία, η ψυχολογία, η μηχανολογία, η πληροφορική, τα μαθηματικά, οι επικοινωνίες, η κλινική αποκατάσταση και η γλωσσολογία. Τα κριτήρια που είναι απαραίτητα να πληροί ένα σύστημα B.C.I είναι τα εξής [27]:

1. Η συσκευή πρέπει να χρησιμοποιεί σήματα που προέρχονται απευθείας από εγκεφαλική δραστηριότητα.

2. Είναι απαραίτητο να υπάρχει ένα τουλάχιστον εγκεφαλικό σήμα που είναι δυνατό να καταγραφεί και το οποίο μπορεί σκοπίμως να χειριστεί και να τροποποιήσει ο χρήστης για να επιτύχει την επιθυμητή συμπεριφορά.
3. Η επεξεργασία των δεδομένων να γίνεται σε πραγματικό χρόνο.
4. Ο χρήστης πρέπει να λαμβάνει ανατροφοδότηση από το σύστημα.

1.4.2 Ιστορική εξέλιξη B.C.I

Το 1970, ο Jonathan Wolpaw μαζί με τους συνεργάτες του, βασιζόμενοι στη δουλειά του Hans Berger το 1929 για την καταγραφή εγκεφαλικών σημάτων μέσω του ηλεκτροεγκεφαλογραφήματος, προχώρησαν στον σχεδιασμό και ανάπτυξη πρωτόγονων συστημάτων ελέγχου βασισμένων στην ηλεκτρική δραστηριότητα του εγκεφάλου.

Η συστηματική έρευνα πάνω στα συστήματα B.C.I έχει τις αρχές της σ' ένα ερευνητικό έργο που έγινε στο Πανεπιστήμιο της California στο Los Angeles, η υλοποίηση του οποίου ξεκίνησε στα μέσα του 1971 από τον **Jacques Vidal** (Εικόνα 22) και την ομάδα του. Σύμφωνα με το ερευνητικό έργο αυτό, ο ασθενής παρακολουθείται μέσα από ένα ειδικό θωρακισμένο περίβλημα το οποίο περιέχει διάφορες συσκευές εισόδου και εξόδου, ενώ ο ερευνητής βρίσκεται σε παρακείμενο δωμάτιο στο οποίο υπάρχουν τοποθετημένα τα τερματικά ελέγχου των υπολογιστών καθώς επίσης και ο εξοπλισμός καταγραφής του ηλεκτροεγκεφαλογραφήματος (EEG) και άλλων βιοσημάτων [29]. Το project αυτό χρηματοδοτήθηκε από την υπηρεσία D.A.R.P.A (Defense Advanced Research Projects Agency) του Αμερικανικού Υπουργείου Άμυνας, για την ανάπτυξη βιονικών συσκευών που θα βοηθούσαν τους στρατιώτες.



Εικόνα 22:
Jacques Vidal [28]

Το 1972 ο **Robin Michelson**, M.D. έπεισε την επιστημονική κοινότητα ότι κάθε αξιοσημείωτος ήχος μπορεί να μεταφερθεί στον εγκέφαλο με κατάλληλο ηλεκτρικό ερεθισμό του ακουστικού νεύρου. Η παρατήρηση αυτή οδήγησε στην ανάπτυξη του «βιονικού αυτιού (Cochlear implant)».



Εικόνα 23:
William Dobbelle
[31]

Το 1976, πάλι ο **Jacques Vidal**, επέκτεινε το ενδιαφέρον της D.A.R.P.A για τα συστήματα B.C.I στο γενικότερο τομέα της βιοκυβερνητικής (biocybernetics) με στόχο τον έλεγχο συσκευών μέσω επεξεργασίας οποιουδήποτε βιολογικού σήματος σε πραγματικό χρόνο [30]. Έδωσε στοιχεία ότι μπορεί να γίνει χρήση των οπτικών προκλητών δυναμικών ως κανάλι επικοινωνίας για τον έλεγχο κέρσορα μέσω δισδιάστατου λαβυρίνθου.

Ο Δρ. **William Dobbelle** (Εικόνα 23), ανέπτυξε το 1978 ένα σύστημα τεχνητής όρασης το οποίο έπαιρνε εικόνα από μία μικρή κάμερα καθώς και πληροφορίες απόστασης μέσω αισθητήρων τα οποία ήταν τοποθετημένα σε ένα ζευγάρι γυαλιά. Τα λαμβανόμενα σήματα επεξεργάζονταν από έναν μικρό φορητό υπολογιστή βάρους 5Kg και στη συνέχεια ένα νέο σήμα στέλνονταν σε 68 πλατινένια ηλεκτρόδια που είχαν εμφυτευτεί στον εγκέφαλο. Με τον τρόπο αυτό κατάφερε και ενεργοποιούσε τα οπτικά νεύρα μέσω των οποίων ο ασθενής ήταν σε θέση να δει κάποια στίγματα φωτός τα οποία έδειχναν τις άκρες των διάφορων αντικειμένων.

Το 1999 η Δρ. **Yang Dan** (Εικόνα 24), επικεφαλής ερευνητικής ομάδας από το Πανεπιστήμιο της California, προχώρησε σε καταγραφή της ηλεκτρικής δραστηριότητας των νευρικών κυττάρων που βρίσκονται στον θάλαμο, την περιοχή του εγκεφάλου η οποία λαμβάνει τα σήματα από τα μάτια. Το ιδιαίτερο με αυτό το πείραμα είναι ότι διεξήχθη σε μία γάτα και οι επιστήμονες κατέγραψαν την έξοδο 177 εγκεφαλικών κυττάρων που ανταποκρίνονταν στο φως και το σκοτάδι στο οπτικό πεδίο της [33].



Εικόνα 24: Yang Dan [32]

Το 2003, η εταιρεία **Cyberkinetics** σε συνεργασία με το τμήμα Νευροεπιστημών του Πανεπιστημίου Brown, παρουσίασε το **Braingate**, ένα σύστημα εγκεφαλικού εμφυτεύματος που κατέγραφε την εγκεφαλική δραστηριότητα των ασθενών και μετέτρεπε τις προθέσεις τους σε εντολές.



Το Φεβρουάριο του 2011, ο **Ray Lee** (Εικόνα 25), τεχνικός διευθυντής του Ινστιτούτου Νευροεπιστημών στο Πανεπιστήμιο Princeton, ανέπτυξε ένα σύστημα MRI με το οποίο γίνεται ταυτόχρονη σάρωση δύο εγκεφάλων. Η εξέλιξη αυτή δίνει τη δυνατότητα να εξεταστεί κατά τον τρόπο που αντιδρούν δύο διαφορετικοί εγκέφαλοι στα ίδια ερεθίσματα αλλά και μεταξύ τους [35].

Εικόνα 25:

Ray Lee [34]

1.4.3 Κατηγορίες Brain – Computer Interfaces.

Τα B.C.I διακρίνονται κυρίως σε δύο μεγάλες κατηγορίες ανάλογα με τον τρόπο που μεταφέρεται στο περιβάλλον η εντολή που δίνει ο εγκέφαλος. Συγκεκριμένα οι κατηγορίες αυτές είναι [36]:

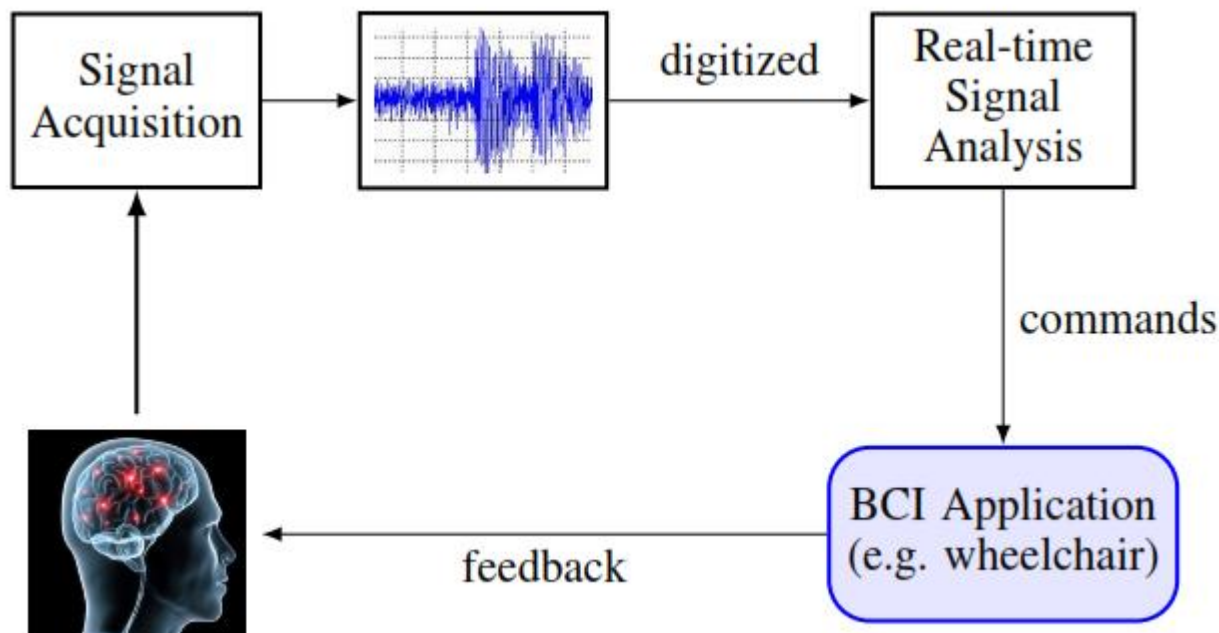
1. **Τα εξαρτημένα B.C.I** τα οποία δεν χρησιμοποιούν τις φυσιολογικές εγκεφαλικές διόδους για να μεταφέρουν τις εντολές που παράγονται, αλλά η δραστηριότητα σε αυτές τις διόδους είναι απαραίτητη για να δημιουργήσει την εγκεφαλική δραστηριότητα που θα μεταφέρει τις εντολές αυτές.
2. **Τα μη εξαρτημένα B.C.I** τα οποία δεν χρησιμοποιούν με κανέναν τρόπο τις φυσιολογικές εγκεφαλικές διόδους. Τα μηνύματα δεν μεταφέρονται από τα περιφερειακά νεύρα και τους μύες και κατ' επέκταση δε χρειάζεται δραστηριότητα στις διόδους αυτές που θα δημιουργήσει την αντίστοιχη εγκεφαλική δραστηριότητα (π.χ. EEG) που θα μεταφέρει τις εντολές αυτές.

Ένας άλλος διαχωρισμός των συστημάτων B.C.I βασίζεται στον τρόπο με τον οποίο γίνεται η λήψη των βιοσημάτων από τον εγκέφαλο και διακρίνονται στις ακόλουθες δύο κατηγορίες [36]:

1. Τα **επεμβατικά**, με τα οποία γίνεται εμφύτευση ηλεκτροδίων στη φαιά ουσία του ανθρώπινου εγκεφάλου για τη λήψη υψηλής ποιότητας σημάτων αλλά ενέχει ο κίνδυνος να προκληθεί ζημιά στον ιστό ή μόλυνση.
2. Τα **μη - επεμβατικά**, με τα οποία τα ηλεκτρόδια τοποθετούνται πάνω στην επιφάνεια του εγκεφάλου για να μετρήσουν τα σήματα του εγκεφάλου, αλλά ο σηματοθορυβικός λόγος (Signal to Noise Ratio, SNR) είναι ιδιαίτερα χαμηλός.

1.4.4 Λειτουργία

Μία σύντομη περιγραφή της λειτουργίας ενός B.C.I είναι ότι μετράει την εγκεφαλική δραστηριότητα, την επεξεργάζεται και παράγει σήματα ελέγχου που αντανακλούν την πρόθεση του χρήστη (Εικόνα 26).

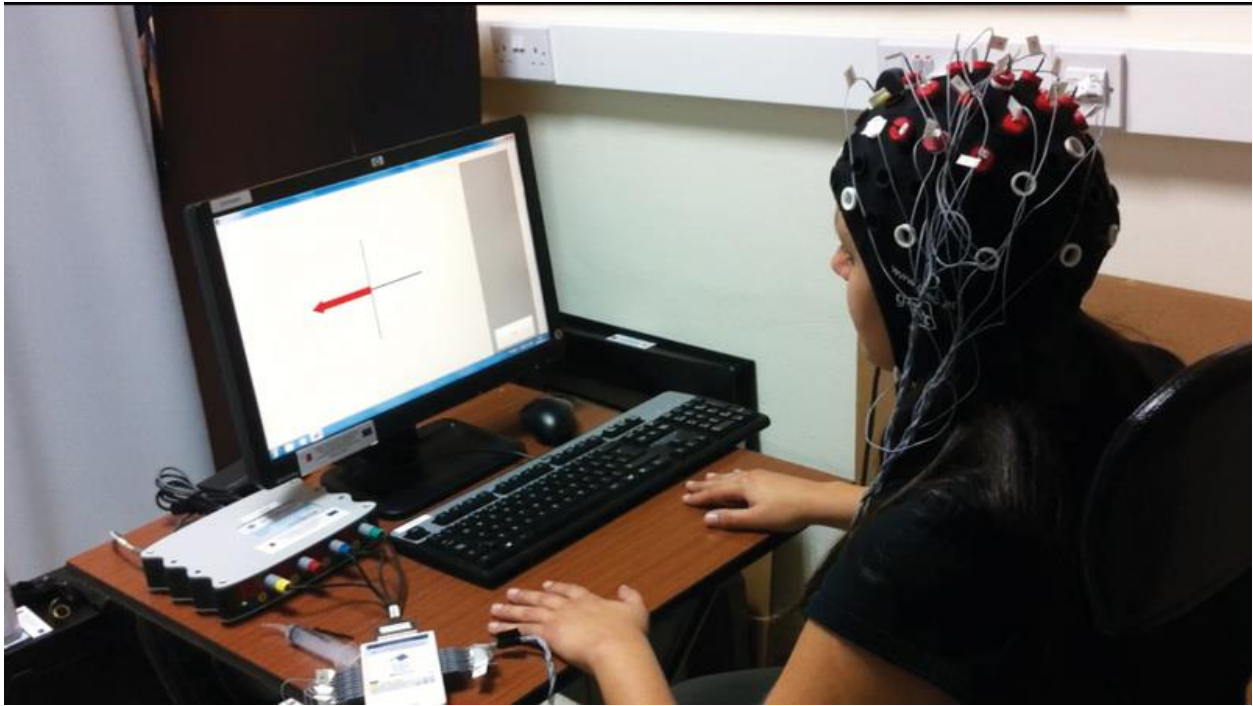


Εικόνα 26: Βασική Λειτουργία B.C.I [36]

Για να κατανοήσουμε όμως καλύτερα τον τρόπο λειτουργίας ενός B.C.I θα πρέπει πρώτα να κατανοήσουμε τον τρόπο με τον οποίο μπορεί να μετρηθεί η εγκεφαλική δραστηριότητα και ποια εγκεφαλικά σήματα μπορούν να χρησιμοποιηθούν.

Η εγκεφαλική δραστηριότητα παράγει ηλεκτρική και μαγνητική δραστηριότητα, με συνέπεια ειδικοί αισθητήρες να μπορούν να ανιχνεύουν τις αλλαγές που συμβαίνουν σε διαφορετικές χρονικές στιγμές και σε διαφορετικές περιοχές του εγκεφάλου. Τα περισσότερα συστήματα B.C.I στηρίζονται σε αυτές τις ηλεκτρικές μετρήσεις της εγκεφαλικής δραστηριότητας και σε αισθητήρες που τοποθετούνται πάνω από το κεφάλι για να μετρήσουν τη δραστηριότητα αυτή. Με τη μέθοδο του ηλεκτροεγκεφαλογραφήματος, ηλεκτρόδια που είναι συνδεδεμένα σε μία κάσκα καταγράφουν τις ηλεκτρικές διαφορές που συμβαίνουν στον εγκέφαλο και οι οποίες ονομάζονται **δυναμικά ενέργειας**. Εν συνεχεία, καλώδια μεταφέρουν τα καταγεγραμμένα σήματα από τα ηλεκτρόδια σε έναν ενισχυτή που μετατρέπει τα αναλογικά αυτά σήματα σε ψηφιακά. Τέλος, ένας ηλεκτρονικός υπολογιστής είναι υπεύθυνος για την επεξεργασία των δεδομένων και για την εκτέλεση της αντίστοιχης εφαρμογής.

Μια τυπική διάταξη B.C.I φαίνεται στην παρακάτω εικόνα:



Εικόνα 27: Τυπική διάταξη ενός συστήματος B.C.I [37]

Μεγάλη σημασία στην μέτρηση των εγκεφαλικών σημάτων έχει ο τρόπος τοποθέτησης των ηλεκτροδίων πάνω στο κεφάλι του χρήστη. Η τοποθέτηση των ηλεκτροδίων γίνεται με βάση το διεθνές πρότυπο **10-20** (Εικόνα 26), το οποίο βασίζεται σε μια επαναληπτική υποδιαίρεση των τόξων του κρανίου, αρχίζοντας από κρανιομετρικά σημεία αναφοράς [38]:

- Nasion (Ns),
- Inion (In),
- Αριστερά (PAL) και Δεξιά (PAR) προ-ωτικά σημεία.

Η τομή της διαμήκουσ των Nasion και Inion με τη διαμήκη των αριστερών και δεξιών προωτικών σημείων, ονομάζεται κορυφή του κρανίου (Vertex).

3. Ηθικά ζητήματα μπορεί να εμποδίσουν την περαιτέρω ανάπτυξη τέτοιων συστημάτων.
4. Τα ηλεκτρόδια έξω από το κρανίο μπορούν να ανιχνεύσουν πολύ λίγα εγκεφαλικά σήματα.
5. Τα ηλεκτρόδια μέσα στο κρανίο προκαλούν πληγές στον ιστό

1.4.6 Εφαρμογές

Τα συστήματα B.C.I έχουν πληθώρα εφαρμογών τόσο σε επιστημονικό όσο και σε εμπορικό επίπεδο. Μερικές από αυτές τις εφαρμογές είναι:

- Να δοθεί η δυνατότητα της επικοινωνίας, περιβαλλοντικού ελέγχου και κινητικής αποκατάστασης σε ανθρώπους που αντιμετωπίζουν ανάλογα προβλήματα.
- Να δοθεί η δυνατότητα ελέγχου συσκευών όπως αναπηρικών αμαξιδίων, οχημάτων ή βοηθητικών robot σε όσους έχουν περιορισμένη κινητικότητα.
- Η επέκταση ελέγχου ηλεκτρονικών παιχνιδιών μέσω πρόσθετου καναλιού.
- Η ανάπτυξη «έξυπνων» συσκευών χαλάρωσης.
- Ο έλεγχος robot σε επικίνδυνο ή αφιλόξενο για τους ανθρώπους περιβάλλον (π.χ. υποβρυχίως, σε υπερβολική ζέστη ή κρύο) .
- Ο σχεδιασμός και ανάπτυξη για την παρακολούθηση διαφόρων καταστάσεων του οργανισμού όπως παρακολούθηση μακροχρόνιας λήψης φαρμάκων και αξιολόγηση ψυχολογικής κατάστασης.
- Βελτίωση της γνώσης. Μία απλή μη ιατρική εφαρμογή που περιλαμβάνει σύστημα BCI αποτελεί ένα είδος εκπαίδευσης με ανατροφοδότηση στην οποία ο εκπαιδευόμενος τροποποιεί εγκεφαλική συμπεριφορά με σκοπό τη βελτίωση της προσοχής, της μνήμης και άλλων βασικών λειτουργιών.

1.4.7 Εμπορικά Συστήματα

Όπως ήταν αναμενόμενο, η ανάπτυξη συστημάτων B.C.I με σκοπό την εμπορική χρήση έγινε αντικείμενο έρευνας αρκετών εταιρειών οι κυριότερες από τις οποίες είναι οι ακόλουθες:

1. OCZ Technology (<http://www.ocz.com/>),
2. Emotiv Systems (<http://www.emotiv.com/>),
3. Neurosky (<http://www.neurosky.com/>),
4. InteraXon (<http://www.interaxon.ca/>),
5. OpenBCI (<http://www.openbci.com/>).

Το Μάιο του 2008, η εταιρεία OCZ Technology, παρουσίασε το Neural Pulse Actuator (Εικόνα 29), ένα σύστημα το οποίο κατέγραφε έναν συνδυασμό μυϊκής, δερματικής και νευρικής δραστηριότητας μεταξύ του συμπαθητικού και παρασυμπαθητικού συστήματος και το οποίο ήταν κυρίως βιοδυναμικά και λιγότερα νευρικά σήματα. Τα βιοδυναμικά αυτά αναλύονταν περαιτέρω σε διαφορετικά φάσματα συχνοτήτων, κάνοντας έτσι δυνατό το διαχωρισμό των φασμάτων αυτών σε ομάδες ηλεκτρικών σημάτων. Ορισμένα από αυτά τα ηλεκτρικά σήματα περιλάμβαναν τους ρυθμούς Άλφα και Βήτα.

Τρία χρόνια αργότερα, το Μάιο του 2011, η εταιρεία με ανακοίνωσή της, σταμάτησε την παραγωγή του Neural Pulse Actuator.



Εικόνα 29: Neural Pulse Actuator [40]

Το Δεκέμβριο του 2009 η εταιρεία **Emotiv Systems**, παρουσίασε το **Emotiv EPOC** (Εικόνα 30), ένα υψηλής ανάλυσης σύστημα για την καταγραφή ηλεκτροεγκεφαλογραφήματος, το οποίο διαθέτει 14 ηλεκτρόδια (5 λιγότερα από το διεθνές πρότυπο 10-20) καθώς επίσης και ένα γυροσκόπιο δύο αξόνων για την ανίχνευση κινήσεων του κεφαλιού. Τα 14 αυτά ηλεκτρόδια τοποθετούνται στα σημεία AF3, F7, F3, FC5, T7, P7, O1, O2, P8, T8, FC6, F4, F8 και AF4 που ορίζονται από το διεθνές πρότυπο 10-20 ενώ τα ηλεκτρόδια αναφοράς **Common Mode Sense** (CMS, ενεργό ηλεκτρόδιο) και **Driven Right Leg** (DLR, παθητικό ηλεκτρόδιο) τοποθετούνται στα σημεία P3 και P4 αντίστοιχα. Συγκεκριμένα, είναι σε θέση να ανιχνεύσει 6 κινήσεις του κεφαλιού (πάνω, κάτω, αριστερά, δεξιά, μπροστά και τράβηγμα προς τα πίσω), 6 περιστροφές (κατά τη φορά και αντίθετα των δεικτών του ρολογιού, στροφή αριστερά ή δεξιά και ταλάντωση μπροστά ή πίσω), τα συναισθήματα χαράς (excitement), ανίας (boredom), πνευματικής χαλάρωσης (meditation) και ενόχλησης (frustration) καθώς επίσης και τις εκφράσεις του προσώπου που σχετίζονται με το χαμόγελο και το γέλιο.



Εικόνα 30: Emotiv EPOC [41]

Το Μάρτιο του 2011 η εταιρεία **Neurosky** ανέπτυξε το **Mindwave** (Εικόνα 31), μία συσκευή καταγραφής του ηλεκτροεγκεφαλογραφήματος κάνοντας χρήση ενός μόνο ηλεκτροδίου, παρέχοντας στον χρήστη τη δυνατότητα απεικόνισης και χρήσης δύο «πνευματικών» καταστάσεων, της προσοχής και της πνευματικής ηρεμίας, που βασίζονται σε 5 ρυθμούς (Άλφα, Βήτα, Γάμμα, Δέλτα και Θήτα) και στον βλεφαρισμό των ματιών. Περισσότερες πληροφορίες για το Mindwave δίνονται στην παρούσα εργασία, στην παράγραφο 2.3.1.

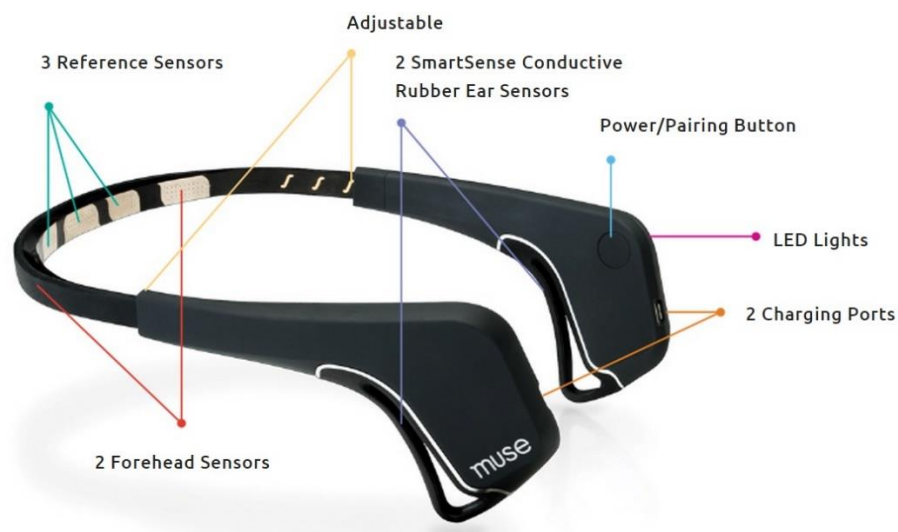


Εικόνα 31: Mindwave Mobile [42]

Τον Απρίλιο του 2014 η εταιρεία **Interaxon** παρουσίασε το **Muse** (Εικόνα 32), ένα εργαλείο εκγύμνασης του εγκεφάλου το οποίο μετράει τη δραστηριότητα σε αυτόν όπως ένας καρδιογράφος μετράει τον ρυθμό της καρδιάς. Αποτελείται από 7 βαθμονομημένους αισθητήρες εκ των οποίων δύο βρίσκονται στο μέτωπο, δύο πίσω από τα αυτιά και τρεις αποτελούν τους αισθητήρες αναφοράς.

Μέσω της καταγραφής του ηλεκτροεγκεφαλογραφήματος, εξάγονται οι εγκεφαλικοί ρυθμοί Άλφα, Βήτα, Γάμμα, Δέλτα και Θήτα. Αν οι ρυθμοί αυτοί βρίσκονται σε ισορροπία μεταξύ τους τότε αυτό είναι ένδειξη ότι ο χρήστης βρίσκεται σε ηρεμία και συγκεντρωμένος. Σε αντίθετη περίπτωση, η προσοχή του χρήστη αποσπάται από διάφορα ερεθίσματα με αποτέλεσμα να σκέφτεται πράγματα που δεν σχετίζονται μεταξύ τους. Το Muse βοηθάει τον χρήστη, μέσω κατάλληλα σχεδιασμένων ασκήσεων, να αναγνωρίσει τις αιτίες που του αποσπών την προσοχή και να επανακτήσει τον έλεγχό της.

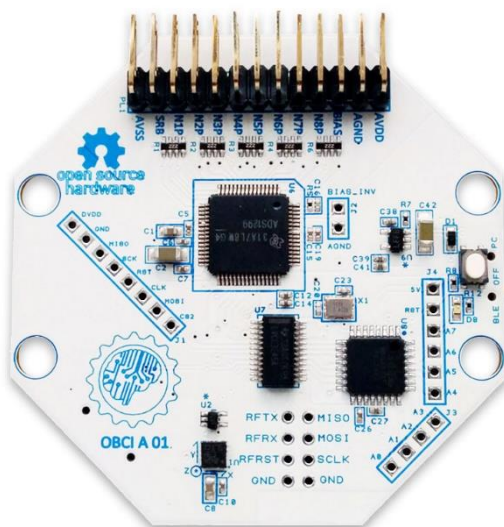
Είναι πολύ εύκολο στη χρήση και το μεγαλύτερο πλεονέκτημά του είναι η δυνατότητα σύνδεσής του με συσκευές που υποστηρίζουν το πρωτόκολλο Bluetooth, κάτι που του δίνει τη δυνατότητα να συνδέεται με smartphones μέσω των οποίων ο χρήστης μπορεί να χρησιμοποιεί τις εφαρμογές που το συνοδεύουν ακόμα και όταν βρίσκεται εν κινήσει.



Εικόνα 32: Muse [43]

Τέλος στα μέσα του 2014 ξεκίνησε μία προσπάθεια από μηχανικούς, επιστήμονες, καλλιτέχνες και ερασιτέχνες για τη δημιουργία μιας ανοικτής κοινότητας που ως σκοπό έχει την εξερεύνηση των δυνατοτήτων του ανθρώπινου εγκεφάλου. Το αποτέλεσμα ήταν να δημιουργηθεί το σχέδιο **OpenBCI** το οποίο υποστηρίζεται από διάφορες εταιρείες όπως η Fab Labs UK (<http://www.fablabsuk.co.uk/>) και η Mensia Technologies (<http://www.mensiatech.com/>).

Βασικό εργαλείο που υποστηρίζει η κοινότητα είναι το OpenBCI Board (Εικόνα 33) το οποίο αποτελεί στην ουσία μία παραλλαγή του Arduino με τη διαφορά ότι έχει τη δυνατότητα να συνδέεται μέσω θύρας USB σε μία κάσκα που φοράει ο χρήστης στο κεφάλι του και στην οποία βρίσκονται αισθητήρες τοποθετημένοι κατά το διεθνές πρότυπο 10-20. Μέσω των αισθητήρων αυτών καταγράφεται το ηλεκτροεγκεφαλογράφημα του χρήστη και με κατάλληλους αλγόριθμους μπορεί να ελεγχθεί η λειτουργία άλλων συστημάτων που συνδέονται με αυτό.



Εικόνα 33: OpenBCI [44]

1.4.8 Ηθικά ζητήματα

Όπως σε καθετί καινοτόμο και πρωτοποριακό, έτσι και με τα συστήματα B.C.I εγείρονται αρκετά ηθικά, κοινωνικά και νομικά ζητήματα τα κυριότερα εκ των οποίων είναι:

- Εννοιολογικά ζητήματα. Η επιστημονική κοινότητα έχει διαφωνίες σχετικά με το τι είναι και τι δεν είναι ένα σύστημα B.C.I.
- Οι συνέπειες από τη χρήση τέτοιων τεχνολογιών τόσο για τους ασθενείς όσο και για τις οικογένειές τους, είναι ακόμα άγνωστες.
- Πιθανές παρενέργειες.
- Πιθανή αλλαγή της προσωπικότητας του ατόμου.
- Ανάγνωση εγκεφαλικών σκέψεων και ιδιωτικότητας.
- Έλεγχος εγκεφάλου από τρίτους.
- Χρήση της τεχνολογίας σε προχωρημένες τεχνικές ανάκρισης από αστυνομικές αρχές.
- Ηθικά ερωτήματα για την εφαρμογή πειραμάτων σε ανθρώπους.
- Επιλεκτική ενίσχυση και κοινωνική διαστρωμάτωση.

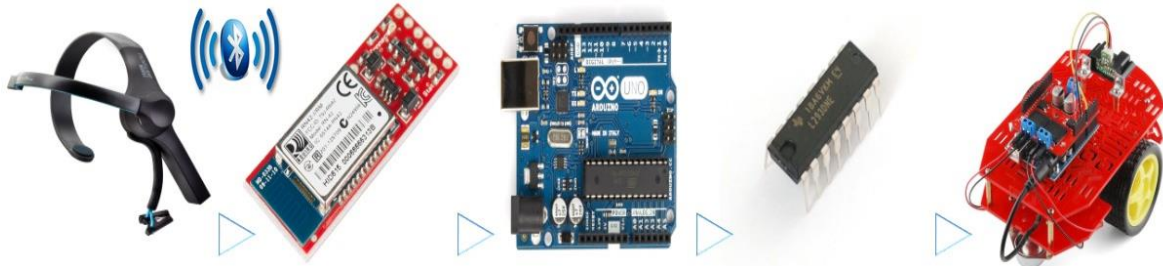
2. ΣΧΕΔΙΑΣΗ ΣΥΣΤΗΜΑΤΟΣ

Κατά τον σχεδιασμό του συστήματος ήταν απαραίτητο να γίνει μία καταγραφή των βημάτων τα οποία έπρεπε να πραγματοποιηθούν με συγκεκριμένη σειρά. Τα βήματα αυτά είναι τα ακόλουθα:

1. Κατανόηση του σκοπού της εργασίας.
2. Συγκέντρωση υλικών και εξαρτημάτων.
3. Περιγραφή τρόπου λειτουργίας συσκευών.
4. Επεξεργασία των δεδομένων.
5. Δημιουργία διαγράμματος ροής.

2.1 Κατανόηση του σκοπού της εργασίας

Σκοπός της εργασίας είναι η κατασκευή ενός ρομπωτικού μηχανισμού, ο έλεγχος του οποίου θα γίνεται μέσω των βλεφαρισμών των ματιών με χρήση πλατφόρμας ανοικτού υλικού και ειδικότερα του Arduino.



Εικόνα 34: Βασικό διάγραμμα project

Συγκεκριμένα, η πηγή των δεδομένων είναι η συσκευή καταγραφής ηλεκτροεγκεφαλογραφήματος Mindwave Mobile της εταιρείας Neurosky. Η συσκευή αυτή μετατρέπει την ηλεκτρική δραστηριότητα από την επιφάνεια του εγκεφάλου, και ειδικότερα από τον μετωπιαίο λοβό, σε ψηφιακές τιμές μέσω ενός εσωτερικού chip τύπου ASIC και τις στέλνει μέσω πρωτοκόλλου Bluetooth με ρυθμό μετάδοσης συμβόλων (baud rate) 57.600 σε ένα ειδικό Bluetooth άρθρωμα που συνδέεται στο Arduino. Οι τιμές που υπάρχουν στο απεσταλμένο σήμα είναι οι τιμές της Προσοχής και της Συγκέντρωσης, καθώς επίσης οι πέντε βασικοί εγκεφαλικοί ρυθμοί και ο βλεφαρισμός των ματιών. Εν συνεχεία γίνεται η επεξεργασία του ηλεκτροεγκεφαλογραφήματος στο Arduino για την εξαγωγή των χρήσιμων δεδομένων βάσει των οποίων θα γίνει ο έλεγχος του μηχανισμού. Ο μηχανισμός αυτός είναι ένα μικρό αυτοκινητάκι που αποτελείται από τη βάση και δύο κινητήρες DC που κινούν τους τροχούς, για τη λειτουργία των οποίων χρησιμοποιείται τροφοδοσία 5 V.

Η εργασία βασίζεται στο γενικότερο επιστημονικό ενδιαφέρον που υπάρχει παγκοσμίως και γνωρίζει ιδιαίτερη ανάπτυξη τα τελευταία χρόνια, σχετικά με τη δημιουργία εφαρμογών που θα βασίζονται στην επικοινωνία μεταξύ υπολογιστή και εγκεφάλου (B.C.I) και απευθύνεται κυρίως σε ανθρώπους που αντιμετωπίζουν κάποιου είδους κινητικής δυσκολίας.

2.2 Συγκέντρωση υλικών και εξαρτημάτων





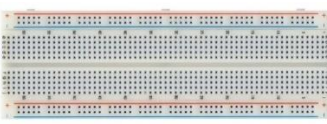

Τα υλικά και τα εξαρτήματα που χρησιμοποιήθηκαν στην εργασία ήταν τα ακόλουθα:

1. 1 Mindwave Mobile
2. 1 Arduino Uno Rev. 3
3. 1 Magician Chassis
4. 1 Αισθητήρας υπερήχων HC-SR04
5. 1 Bluesmirf RN-42
6. 4 Μπαταρίες AA
7. 1 μπαταρία AAA
8. 1 Ολοκληρωμένο κύκλωμα γέφυρας Η L293D
9. 1 Καλώδιο 15' USB
10. 1 breadboard
11. 1 πακέτο jumper breadboard wires των 65 τεμαχίων

Το συνολικό κόστος όλων των εξαρτημάτων που χρησιμοποιήθηκαν στην εργασία φαίνεται στον παρακάτω πίνακα:

Πίνακας 1: Λίστα κόστους υλικών και εξαρτημάτων

Είδος	Εικόνα	Τιμή
Mindwave Mobile		100,74 €
Arduino Uno Rev. 3		22,00 €
Magician Chassis		18,50 €
Αισθητήρας υπερήχων HC-SR04		4,00 €
Bluesmirf RN-42		36,00 €

4 Μπαταρίες AA		1,60 €
1 Μπαταρία AAA		0,80 €
1 Ολοκληρωμένο κύκλωμα γέφυρας Η L293D		2,00 €
1 Καλώδιο 15' USB		0,50 €
1 Breadboard		4,99 €
1 πακέτο jumper breadboard wires των 65 τεμαχίων		4,80 €
Συνολικό Κόστος		195,93 €

2.3 Περιγραφή τρόπου λειτουργίας συσκευών

2.3.1 Mindwave Mobile

Ένα από τα δυσκολότερα σημεία της εργασίας αυτής ήταν η κατανόηση του τρόπου λειτουργίας του Mindwave Mobile και ειδικότερα του τρόπου μετάδοσης μέσω Bluetooth των καταγεγραμμένων ψηφιακών τιμών.

2.3.1.1 Γενικές πληροφορίες

Οι συσκευές που χρησιμοποιούνται σήμερα σε κλινικές εφαρμογές για την καταγραφή ηλεκτροεγκεφαλογραφημάτων αποτελούνται από διάφορα ηλεκτρόδια για την καταγραφή των βασικότερων κυμάτων και κοστίζουν χιλιάδες δολάρια. Τα τελευταία χρόνια όμως,

έχουν αναπτυχθεί φθηνές φορητές συσκευές, που καταγράφουν το ηλεκτροεγκεφαλογράφημα, από τις ακόλουθες εταιρείες:

1. Avatar EEG Solutions (<http://www.avatarsolutions.com/>),
2. Neurosky (<http://www.neurosky.com/>),
3. OCZ Technology (<http://www.ocz.com/>),
4. InteraXon (<http://www.interaxon.ca/>),
5. PLX Devices (<http://www.plxdevices.com/>),
6. Emotiv Systems (<http://www.emotiv.com/>).

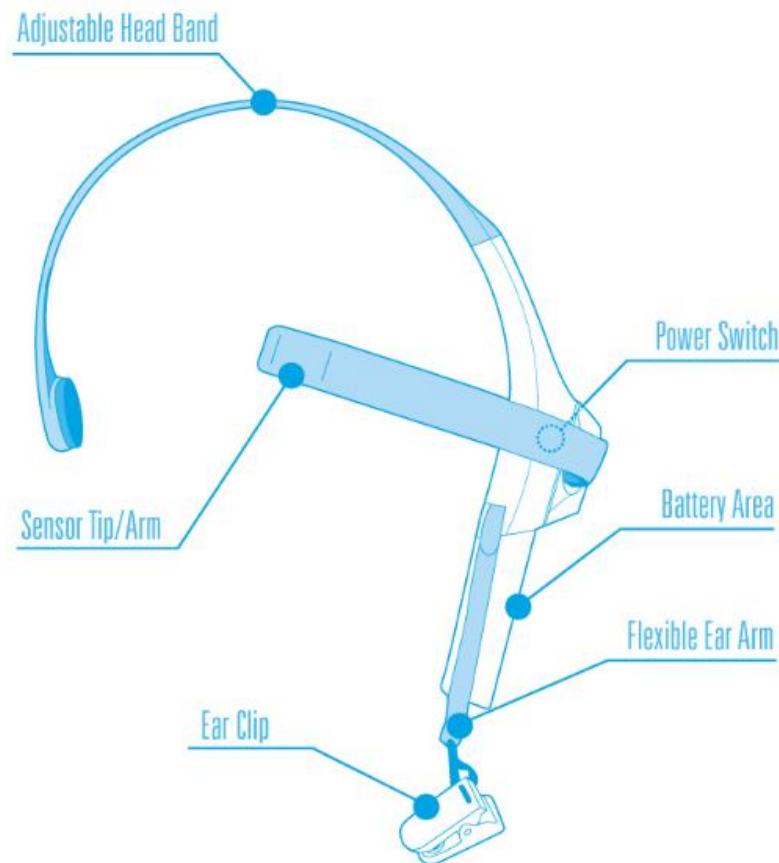
Η εταιρεία Neurosky© με έδρα την πόλη San Jose στην Καλιφόρνια των Η.Π.Α, έχει αναπτύξει τη συσκευή **Mindwave Mobile** (Εικόνα 35) για την καταγραφή του ηλεκτροεγκεφαλογραφήματος.



Εικόνα 35: Mindwave Mobile [45]

Η συσκευή αυτή, με κόστος περίπου \$100 (<https://www.sparkfun.com/products/12805>) έχει μόνο ένα κανάλι για την καταγραφή των δεδομένων και αποτελείται από τα παρακάτω μέρη, όπως φαίνεται στην εικόνα 36:

1. Βραχίονα αισθητήρων (Sensor Tip / Arm)
2. Ρυθμιζόμενο κεφαλόδεσμο (Adjustable Head Band)
3. Διακόπτη λειτουργίας (Power Switch)
4. Περιοχή Μπαταρίας (Battery Area)
5. Ευέλικτο Βραχίονα Αυτιού (Flexible Ear Arm)
6. Κλιπ αυτιού (Ear Clip)



Εικόνα 36 Mindwave Mobile [46]



Εικόνα 37 Mindwave Mobile [45]

Στο εσωτερικό του Mindwave Mobile (Εικόνα 37) συναντούμε ένα μονό ηλεκτρόδιο που ανιχνεύει τα ηλεκτρικά εγκεφαλικά σήματα που υπάρχουν στο μέτωπο, αλλά καταγράφει επίσης και τον εξωτερικό θόρυβο που δημιουργείται από μυσικές κινήσεις, υπολογιστές, λάμπες, ηλεκτρικές πρίζες και άλλες συσκευές που πιθανόν να βρίσκονται στον περιβάλλοντα χώρο, καθώς επίσης και ένα διπλό ηλεκτρόδιο που χρησιμοποιείται ως γείωση και σημείο αναφοράς επιτρέποντας στο thinkgear chip να φιλτράρει τον ηλεκτρικό θόρυβο [47].

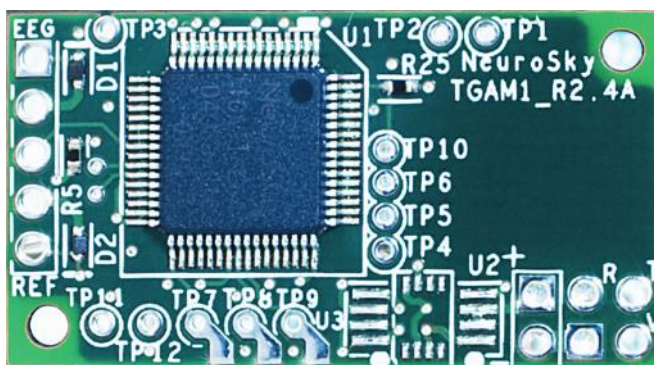
Τα ηλεκτρόδια αυτά συνδέονται στην κύρια μονάδα που υπάρχει σε όλα τα προϊόντα της εταιρείας και ονομάζεται **TGAM** (ThinkGear ASIC Module) (Εικόνα 38). Η μονάδα αυτή αποτελεί τον κύριο αισθητήρα

ASIC (Application Specific Integrated Circuit) και είναι ειδικά σχεδιασμένη για εμπορικές εφαρμογές, αποτελώντας τον κυριότερο αισθητήρα για ηλεκτροεγκεφαλογραφήματα (EEG) με εκατομμύρια πωλήσεις παγκοσμίως. Τα κυριότερα πλεονεκτήματα του TGAM είναι τα ακόλουθα [48]:

1. Συνδέεται απευθείας σε ξηρό ηλεκτρόδιο, εν αντιθέσει με τους συμβατικούς υγρούς αισθητήρες.
2. Έχει ένα κανάλι EEG με τρεις επαφές:
 - a. EEG
 - b. REF

c. GND

3. Ανιχνεύει μη σωστή τοποθέτηση της συσκευής και προειδοποιεί για «*Ασθενή Ποιότητα Σήματος (Poor Signal Quality)*» μέσω του ASIC.
4. Διαθέτει εξελιγμένη τεχνολογία φιλτραρίσματος με υψηλή ανοχή θορύβου.
5. Η κατανάλωση ενέργειας είναι αρκετά χαμηλή, με αποτέλεσμα να το καθιστά κατάλληλο για φορητές εφαρμογές.
6. Η μέγιστη κατανάλωση ενέργειας ανέρχεται στα 15mA@3.3V
7. Έχει έξοδο ακατέργαστων (RAW) δεδομένων με ρυθμό μετάδοσης 512 bits ανά δευτερόλεπτο.



Εικόνα 38: ThinkGear ASIC Module (TGAM) [48]

2.3.1.2 ThinkGear Data Values

Σύμφωνα με το πρωτόκολλο επικοινωνίας του Mindwave Mobile, το TGAM1 μεταδίδει τις ακόλουθες τιμές που ονομάζονται **ThinkGear Data Values**:

- **Poor Signal Quality:**

Η μέτρηση αυτή είναι **ένας μη-προσημασμένος ακέραιος ενός byte** και περιγράφει πόσο αδύναμη είναι η ποιότητα του σήματος που μετριέται από το ThinkGear. Παίρνει τιμές από 0 έως και 200 και οποιαδήποτε μη μηδενική τιμή είναι ένδειξη ότι υπάρχει κάποιος θόρυβος από διάφορες αιτίες όπως ελλιπής/περιορισμένη επαφή του αισθητηρίου με το μέτωπο του χρήστη, υπερβολική κίνηση του χρήστη και αρκετός περιβαλλοντικός ηλεκτροστατικός θόρυβος. Όσο μεγαλύτερη είναι η τιμή, τόσο μεγαλύτερος είναι ο θόρυβος που ανιχνεύεται. Στην περίπτωση που η τιμή του Poor Signal Quality φτάσει στο 200, αυτό σημαίνει ότι δεν υπάρχει επαφή του ThinkGear με το δέρμα του χρήστη.

- **eSense Meters:**

Η λειτουργία eSense δίνει τη δυνατότητα να εξαχθούν δύο τιμές για την παρατήρηση των λειτουργιών **Προσοχής (Attention)** και **Συγκέντρωσης (Meditation)** του χρήστη. Κάθε μία από τις δύο αυτές μεταβλητές, Attention και Meditation, παίρνει τιμές από 0 έως και 100. Όταν οι τιμές αυτές είναι ανάμεσα σε 1 και 20 θεωρούνται αρκετά αδύναμες ή όταν είναι ανάμεσα στο 20 και το 40 θεωρούνται ελαφρώς αδύναμες και είναι ένδειξη για απόσπαση προσοχής και ταραχής. Αντίθετα, όταν είναι ανάμεσα σε 40 και 60 θεωρούνται ως «ουδέτερες», ενώ οι τιμές από 60 έως και 80 είναι ελαφρώς πάνω από το φυσιολογικό. Οι τιμές που υπερβαίνουν το 80 και φτάνουν στο 100 θεωρούνται ως εξαιρετικά ανεβασμένες.

Attention eSense

Η μέτρηση αυτή είναι **ένας μη-προσημασμένος ακέραιος ενός byte** και περιγράφει την ένταση του επιπέδου προσοχής/εστίασης του χρήστη. Οποιαδήποτε πηγή που μπορεί

να προκαλέσει απώλεια της εστίασης του χρήστη, θα έχει σαν αποτέλεσμα τη μείωση του μετρητή Attention eSense.

Meditation eSense

Η μέτρηση αυτή είναι ένας **μη-προσημασμένος ακέραιος ενός byte** και περιγράφει το επίπεδο της πνευματικής ηρεμίας ή χαλάρωσης του χρήστη. Ο μετρητής αυτός σχετίζεται με το πνευματικό επίπεδο του χρήστη και όχι με το σωματικό, κάτι που σημαίνει πως ακόμα κι αν ο χρήστης χαλαρώσει σωματικά δεν είναι σίγουρο ότι θα επέλθει μείωση του συγκεκριμένου μετρητή. Παρ' όλα αυτά, στους περισσότερους χρήστες η σωματική χαλάρωση συνήθως βοηθάει το μυαλό να ηρεμήσει. Απόσπαση της προσοχής, διαταραχές και αισθητηριακά ερεθίσματα προκαλούν μείωση του επιπέδου πνευματικής χαλάρωσης.

- **RAW Wave Value (16 - bit)**

Η μέτρηση αυτή αποτελείται από δύο bytes και αντιπροσωπεύει τα ακατέργαστα δεδομένα (RAW data). Είναι ένας **προσημασμένος ακέραιος 16-bit**, οι τιμές του οποίου εκτείνονται από -32768 έως και 32767. Το πρώτο byte δείχνει τα bits υψηλής προτεραιότητας του συμπληρώματος ως προς 2, ενώ το δεύτερο byte δείχνει τα bits χαμηλής προτεραιότητας. Για την ανακατασκευή των ακατέργαστων δεδομένων, γίνεται αριστερή ολίσθηση του πρώτου byte κατά 8 bits και λογικό OR σε επίπεδο bit με το δεύτερο byte.

`short raw = (Value[0] << 8) | Value[1];`

Αν η γλώσσα προγραμματισμού που χρησιμοποιείται δεν υποστηρίζει πράξεις σε επίπεδο bit, τότε ο αλγόριθμος για την ανακατασκευή των RAW data, είναι ο ακόλουθος:

`raw = Value[0] * 256 + Value[1];`

`if (raw >= 32768) raw = raw - 65536;`

- **ASIC EEG POWER**

Η τιμή αυτή αντιπροσωπεύει το τρέχον μέγεθος των 8 συνηθέστερων τύπων του ηλεκτροεγκεφαλογράφηματος EEG. Κάθε ένας από αυτούς του τύπους αποτελεί έναν **μη – προσημασμένο ακέραιο τριών bytes**. Οι τύποι αυτοί, γνωστοί ως «**Εγκεφαλικοί Ρυθμοί** (Brain Rhythms)», είναι οι ακόλουθοι:

- Δέλτα (0.5 - 2.75Hz),
- Θήτα (3.5 - 6.75Hz),
- Χαμηλός Άλφα (7.5 - 9.25Hz),
- Υψηλός Άλφα (10 - 11.75Hz),
- Χαμηλός Βήτα (13 - 16.75Hz),
- Υψηλός Βήτα (18 - 29.75Hz),
- Χαμηλός Γάμμα (31 - 39.75Hz) και
- Μεσαίος Γάμμα (41 - 49.75Hz).

- **Blink Strength**

Είναι ένας **μη-προσημασμένος ακέραιος ενός byte** και περιγράφει την ένταση του πιο πρόσφατου βλεφαρισμού ματιού του χρήστη. Παίρνει τιμές από 1 έως και 255 και ανιχνεύεται όταν ο χρήστης ανοιγοκλείνει τα μάτια του.

Το ThinkGear chip που υπάρχει στο Mindwave Mobile μεταδίδει τις τιμές ThinkGear Data Values, κωδικοποιημένες μέσα σε πακέτα που ονομάζονται ThinkGear Packets σαν μία σειριακή ακολουθία από bytes μέσω του πρωτόκολλου Bluetooth με Baud Rate στα 57.600.

2.3.1.3 Δομή Δεδομένων Mindwave Mobile

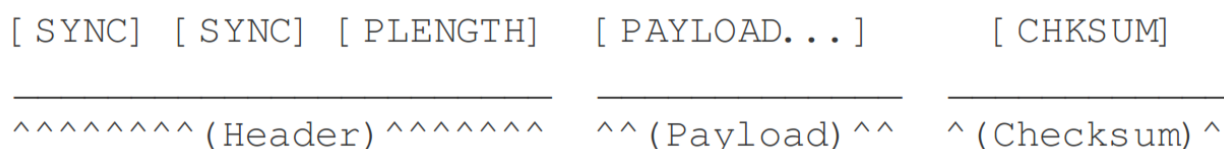
Η μετάδοση των παραπάνω ψηφιακών δεδομένων πραγματοποιείται μέσω ασύγχρονης σειριακής ροής bytes. Ο σειριακός αυτός συρμός αναλύεται και μεταφράζεται σε πακέτα, τα οποία ονομάζονται **ThinkGear Packets**, και από τα οποία γίνεται εξαγωγή όλων των προηγούμενων τιμών (ThinkGear Data Values).

Κάθε πακέτο ThinkGear Packet, αποτελείται από τα ακόλουθα 3 μέρη:

1. Την **Κεφαλίδα** (Packet Header)
2. Το **Ωφέλιμο Φορτίο** (Packet Payload)
3. Το **Αθροισμα των ψηφίων** (Packet Checksum)

Το μέσο που μεταδίδει το σειριακό ρεύμα μπορεί να είναι **Universal Asynchronous Receiver/Transmitter** (U.A.R.T), **Σειριακή Θύρα**, **Universal Serial Bus** (U.S.B), **Bluetooth** ή οποιοσδήποτε άλλος μηχανισμός ο οποίος μπορεί να μεταφέρει δεδομένα.

Κάθε πακέτο ξεκινάει με τα bytes που αποτελούν την Κεφαλίδα, τα οποία ακολουθούνται από τα bytes του Ωφέλιμου Φορτίου και τελειώνει με το byte του σωστού αθροίσματος ψηφίων.



Εικόνα 39: Δομή ThinkGear Packet [49]

- **Κεφαλίδα Πακέτου (Packet Header)**

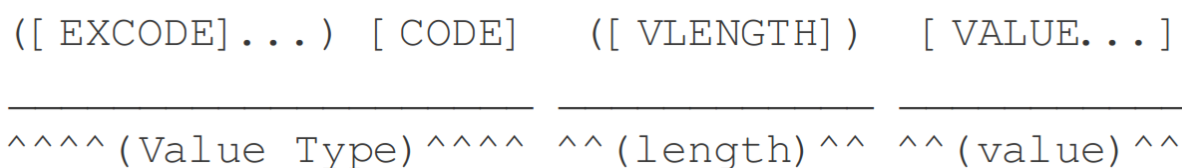
Η κεφαλίδα του πακέτου αποτελείται από δύο bytes συγχρονισμού [SYNC], των οποίων η δεκαεξαδική τιμή είναι 0xAA (η δεκαδική είναι 170). Χρησιμοποιούνται δύο bytes αντί του ενός για να μειωθεί η πιθανότητα να μπερδευτεί κάποιο από τα bytes συγχρονισμού ως αρχή του πακέτου. Το τρίτο byte που αποτελεί την κεφαλίδα [PLENGTH], μας δίνει το μήκος του πακέτου που αποτελούν το [PAYLOAD] και μπορεί να πάρει τιμές από 0 (άδειο PAYLOAD) μέχρι 169 (γεμάτο PAYLOAD).

- **Ωφέλιμο Φορτίο Πακέτου (Packet Payload)**

Το ωφέλιμο φορτίο πακέτου δεδομένων (Data Packet PayLoad) είναι απλά μια σειρά από bytes στην οποία περιέχονται διαδοχικά οι τιμές των ThinkGear Data Values και η κάθε μία από αυτές περιέχεται σε μία σειρά από bytes που ονομάζεται **Data Row**. Κάθε DataRow περιέχει τις ακόλουθες πληροφορίες για τα δεδομένα:

- Τι αντιπροσωπεύει το κάθε Data Value
- Το μήκος του Data Value
- Τα bytes του Data Value

Η δομή που έχει κάθε Data Row είναι η ακόλουθη:



Εικόνα 40: Δομή Data Row [51]

Κάθε σειρά Data Row ξεκινάει με κανένα (μηδέν) ή περισσότερα bytes **Extended Code** (EX.CODE) τα οποία έχουν δεκαεξαδική τιμή 0x55. Τα bytes [EXCODE] και [CODE] χρησιμοποιούνται για την περιγραφή του τύπου του Data Value που περιέχεται σε αυτή τη σειρά. Όταν η τιμή του byte του πεδίου [CODE] κυμαίνεται από 0x00 έως και 0x7F τότε το πεδίο [VALUE] έχει μήκος ενός byte και δεν υπάρχει καθόλου πεδίο [VLENGTH] και χαρακτηρίζεται ως **Single Byte Value**. Αν όμως η τιμή του byte του πεδίου [CODE] είναι μεγαλύτερη από 0x7F τότε εμφανίζεται το πεδίο [VLENGTH] που δίνει τον αριθμό των bytes που περιέχονται στο πεδίο [VALUE]. Παρακάτω φαίνεται ο πίνακας με τις επεξηγήσεις των πεδίων για κάθε σειρά Data Row.

Πίνακας 2: Πίνακας επεξήγησης πεδίων κάθε Data Row [51]

Single-Byte CODEs			
Extended Code Level	[CODE]	(Byte) [LENGTH]	Data Value Meaning
0	0x02	-	POOR_SIGNAL Quality (0-255)
0	0x04	-	ATTENTION eSense (0 to 100)
0	0x05	-	MEDITATION eSense (0 to 100)
0	0x16	-	Blink Strength. (0-255) Sent only when Blink event occurs.
Multi-Byte CODEs			
Extended Code Level	[CODE]	(Byte) [LENGTH]	Data Value Meaning
0	0x80	2	RAW Wave Value: a single big-endian 16-bit two's complement signed value (high order byte followed by low-order byte) (-32768 to 32767)
0	0x83	24	ASIC_EEG_POWER: eight big-endian 3-byte unsigned integer values representing delta, theta, low-alpha, high-alpha, low-beta, high-beta, low-gamma and mid-gamma EEG band power values
Any	0x55	-	NEVER (reserved for [EXCODE])
Any	0xAA	-	NEVER(reserved for [SYNC])

2.3.1.4 Επεξεργασία των δεδομένων ενός πακέτου (frame)

Ο τρόπος με τον οποίο αναλύεται κάθε πακέτο δεδομένων, είναι ο ακόλουθος:

1. Μέσω της σειριακής θύρας γίνεται συνεχής λήψη και ανάγνωση δεδομένων που προέρχονται από το Mindwave Mobile, μέχρι να εμφανιστεί το πρώτο byte συγχρονισμού [SYNC] το οποίο θα έχει την τιμή $0xAA_{(HEX)} = 170_{(DEC)}$
2. Εάν και το αμέσως επόμενο byte είναι byte συγχρονισμού [SYNC] (δηλαδή έχει και αυτό την τιμή $0xAA_{(HEX)}$), τότε ακολουθεί το επόμενο βήμα, αλλιώς η διαδικασία επιστρέφει στο βήμα 1 για εκ νέου ανάγνωση δεδομένων.
3. Αφού έχουν ήδη ληφθεί δύο bytes συγχρονισμού, τότε ελέγχεται αν το byte που ακολουθεί είναι εκείνο που περιέχει τον αριθμό του μήκους του ωφέλιμου φορτίου [PAYLOAD] του πακέτου. Το byte αυτό ονομάζεται Payload Length και η τιμή του μπορεί να κυμαίνεται μεταξύ 0 και 169. Αν ισχύει κάτι τέτοιο ο αλγόριθμος πηγαίνει στο επόμενο βήμα. Σε περίπτωση που η τιμή του byte αυτού είναι ίση με 170 τότε επαναλαμβάνεται το βήμα 3, ενώ αν είναι μεγαλύτερη του 170 γίνεται επιστροφή στο βήμα 1 για εκ νέου ανάγνωση δεδομένων.
4. Τα προηγούμενα τρία βήματα μας δίνουν τις πληροφορίες που υπάρχουν στην κεφαλίδα (Header) του πακέτου. Με την ολοκλήρωση και του τρίτου βήματος και με την προϋπόθεση ότι το μήκος του Payload Data είναι μεταξύ 0 και 169, γίνεται αποθήκευση των bytes του αριθμού αυτού σε έναν πίνακα Payload Data ο οποίος στην εργασία αυτή ορίστηκε να έχει 64 θέσεις, επειδή σε κάθε δευτερόλεπτο γίνεται λήψη 512 bits ακατέργαστων δεδομένων από το Mindwave Mobile, κάτι που ισούται με 64 bytes ανά δευτερόλεπτο. Επίσης, καθώς γίνεται η ανάγνωση κάθε byte από τη σειριακή θύρα και αποθηκεύεται στον πίνακα, ταυτόχρονα υπολογίζεται και το άθροισμα των ληφθέντων bytes (Payload CheckSum).
5. Εν συνεχεία διαβάζεται και το τελευταίο byte που υπάρχει στο πακέτο και το οποίο ονομάζεται CHECKSUM για να επιβεβαιωθεί η λήψη όλων των δεδομένων του πακέτου. Εάν η τιμή του byte αυτού είναι ίση με το συμπλήρωμα ως προς 1 του Payload CheckSum που υπολογίστηκε στο προηγούμενο βήμα, τότε η διαδικασία μπορεί να προχωρήσει στην ανάλυση των bytes που έχουν αποθηκευτεί στον πίνακα για να εξαχθούν τα χρήσιμα δεδομένα. Σε αντίθετη περίπτωση, η διαδικασία επαναλαμβάνεται ξεκινώντας από το βήμα 1.
6. Αφού επιβεβαιωθεί η λήψη όλων των δεδομένων του πακέτου στο προηγούμενο βήμα, για κάθε σειρά δεδομένων (Data Row) που υπάρχει στον πίνακα Payload Data γίνεται καταμέτρηση του αριθμού των bytes των οποίων η δεκαεξαδική τιμή είναι $0x55$. Τα bytes αυτά ονομάζονται Extended Code [EXCODE] bytes και μπορεί να υπάρχουν στην αρχή της κάθε σειράς δεδομένων. Ακόμα και αν δεν υπάρχουν, τότε θεωρείται πως η τιμή τους είναι 0 (μηδέν) και ακολουθεί το επόμενο βήμα.
7. Το επόμενο byte ονομάζεται [CODE] και σε συνεργασία με το [EXCODE] byte δείχνει ποια τιμή από τις Data Values (βλ. παρ 2.3.1.2 και πίνακα 2) περιέχεται στη συγκεκριμένη σειρά δεδομένων. Εάν η τιμή του [CODE] είναι ανάμεσα σε $0x00$ και $0x7F$ (0 και 127 στο δεκαδικό σύστημα), τότε η τιμή του [VALUE] είναι 1 και το ακολουθούμενο byte [VLENGTH] δεν παίρνει καμία τιμή.
8. Στην περίπτωση όμως που η τιμή του [CODE] είναι μεγαλύτερη από 127, τότε το byte [VLENGTH] μπορεί να αποτελείται από 2 ή περισσότερα bytes που αντιστοιχούν στον αριθμό των bytes που περιέχονται στην τιμή του [VALUE]. Αναλύοντας περαιτέρω το τρέχον βήμα, ισχύουν τα ακόλουθα:
 - Όταν η τιμή του [CODE] είναι ίση με $0x02_{(HEX)} = 2_{(DEC)}$ τότε αντιπροσωπεύει την τιμή της ποιότητας του σήματος, η οποία πρέπει να κυμαίνεται από 0 έως και 255. Όταν είναι ίση με 0 τότε η ποιότητα του σήματος είναι άριστη, ενώ όταν είναι 255 δεν υπάρχει καθόλου σήμα.

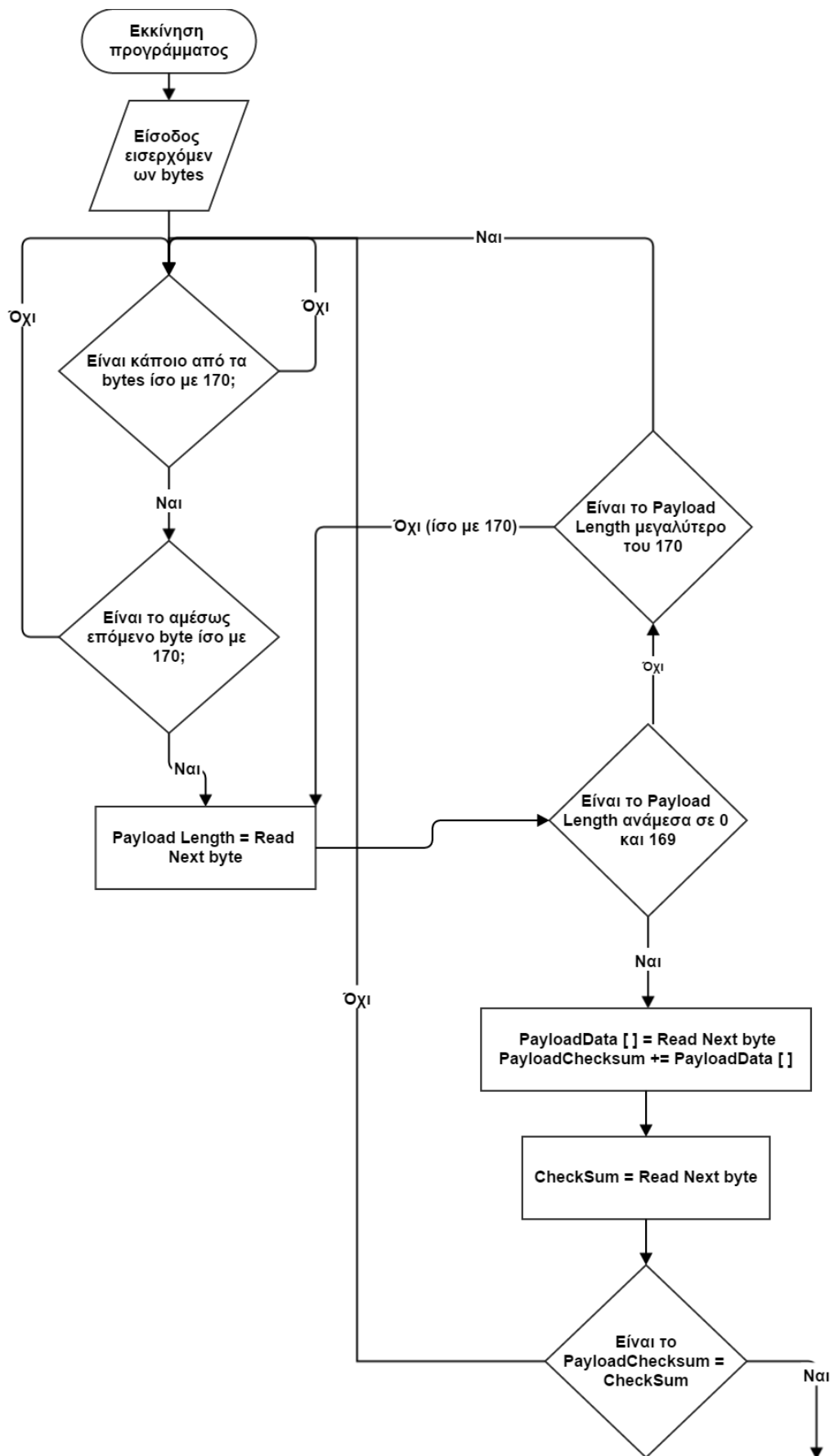
- Όταν η τιμή του [CODE] είναι ίση με $0x04_{(HEX)} = 4_{(DEC)}$ τότε αντιπροσωπεύει την τιμή του επιπέδου προσοχής του χρήστη και κυμαίνεται από 0 έως 100.
- Όταν η τιμή του [CODE] είναι ίση με $0x05_{(HEX)} = 5_{(DEC)}$ τότε αντιπροσωπεύει την τιμή του επιπέδου συγκέντρωσης του χρήστη και κυμαίνεται από 0 έως 100.
- Όταν η τιμή του [CODE] είναι ίση με $0x80_{(HEX)} = 128_{(DEC)}$ τότε έχουμε την τιμή των ακατέργαστων δεδομένων που αποτελείται από 2 bytes (16 bits).
- Όταν η τιμή του [CODE] είναι ίση με $0x83_{(HEX)} = 131_{(DEC)}$ έχουμε την τιμή του ASIC EEG POWER, που αποτελείται από 24 bytes. Στα bytes αυτά περιέχονται τα bytes για τους εγκεφαλικούς ρυθμούς (Delta, Theta, Low Alpha, High Alpha, Low Beta, High Beta, Low Gamma, Mid Gamma). Κάθε ρυθμός καταλαμβάνει τρία από αυτά τα 24 bytes.

9. Εν συνεχεία γίνεται η γραφική απεικόνιση μέσω του προγραμματιστικού περιβάλλοντος Processing (<https://processing.org/>) όλων των τιμών που υπάρχουν στο ASIC EEG Power.

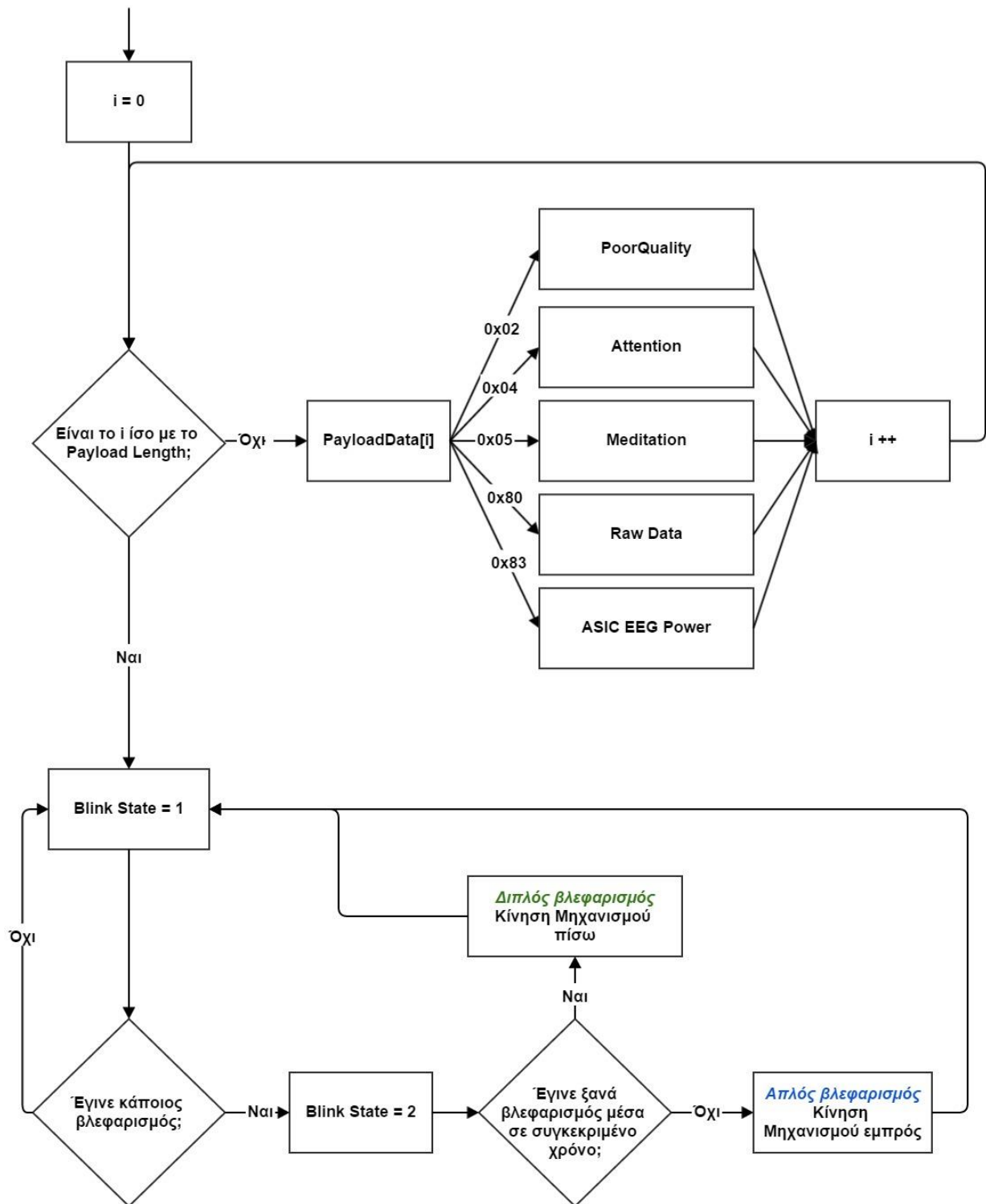
10. Την ίδια στιγμή, γίνεται έλεγχος βάσει των τιμών των ακατέργαστων δεδομένων για το αν συμβαίνει βλεφαρισμός των ματιών ή όχι. Αν συμβαίνει κάτι τέτοιο τότε ανάλογα με τον χρόνο που μεσολαβεί μέχρι τον επόμενο βλεφαρισμό, γίνεται χαρακτηρισμός ως απλού (Single) ή διπλού (Double) βλεφαρισμού και το αμαξίδιο κινείται μπροστά ή πίσω αντίστοιχα, σε ευθεία γραμμή.

11. Επίσης, πάλι σε πραγματικό χρόνο, ελέγχεται μέσω του αισθητήρα υπερήχων, η απόσταση από πιθανό εμπόδιο που μπορεί να υπάρχει κατά την εμπρόσθια κίνηση του αμαξιδίου. Αν η απόσταση αυτή είναι μικρότερη των 20 cm ή περίπου 1200 μs, τότε το αμαξίδιο σταματάει και περιμένει τον επόμενο βλεφαρισμό για να ξεκινήσει ξανά.

Η παραπάνω αλγοριθμική λογική φαίνεται στο ακόλουθο διάγραμμα ροής.



Σχήμα 1: Διάγραμμα ροής επεξεργασίας πακέτου δεδομένων (1/2)



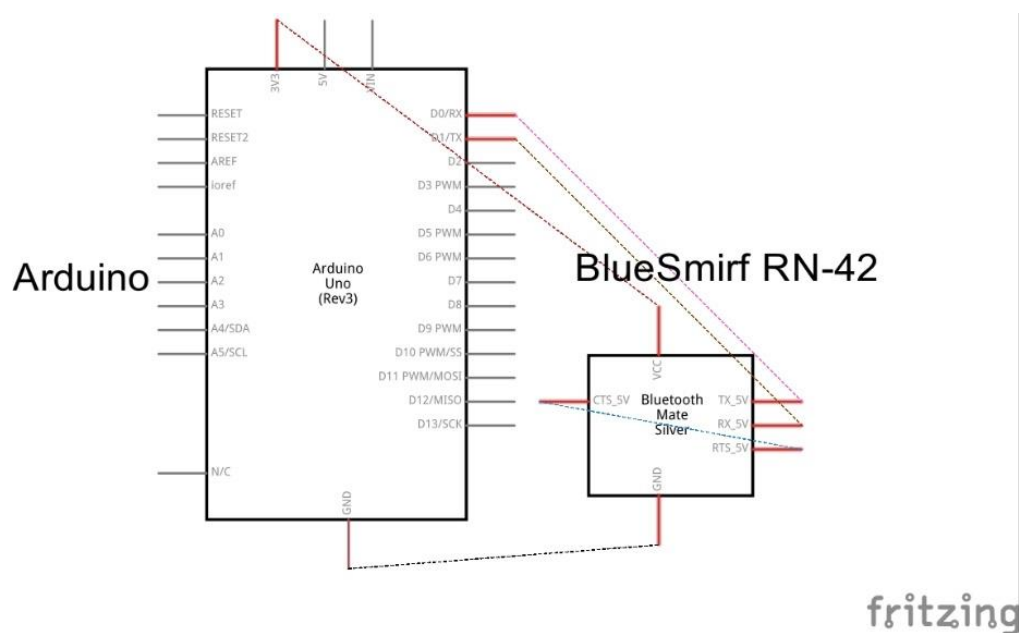
Σχήμα 1: Διάγραμμα ροής επεξεργασίας πακέτου δεδομένων (2/2)

2.3.2 BlueSmirf RN- 42

Η συσκευή που χρησιμοποιήθηκε στην εργασία αυτή ως δέκτης δεδομένων μέσω Bluetooth είναι το Bluesmirf RN-42. Το module αυτό είναι συμβατό με τις εκδόσεις 2.1/2.0/1.2/1.1 του πρωτοκόλλου Bluetooth [50] που παρέχει τη δυνατότητα να επιτυγχάνονται ταχύτητες επικοινωνίας έως και 3 Mbit/s, που όμως στην πράξη δεν υπερβαίνουν τα 2,1 Mbit/s. Επίσης, η μέγιστη απόσταση στην οποία η μεταφορά των

δεδομένων από τον πομπό στο δέκτη είναι άριστη ορίζεται στα 18 μέτρα, κάτι που το καθιστά ιδανικό για χρήση σε οικιακό περιβάλλον και μικρές εφαρμογές.

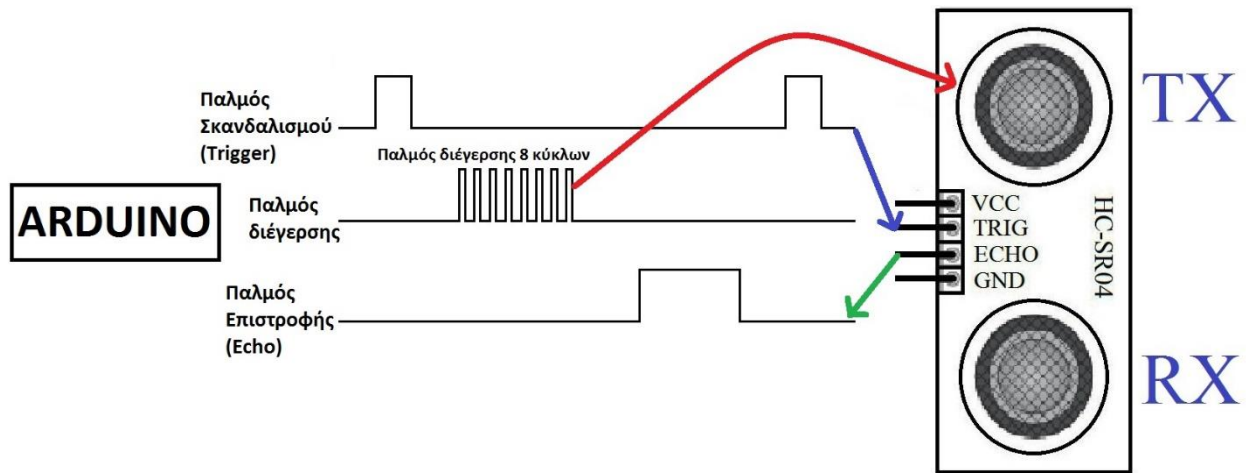
Η σύνδεση του module αυτού στον Arduino γίνεται χρησιμοποιώντας τους ακροδέκτες TX - 0 και RX - 1 τα οποία συνδέονται στους ακροδέκτες RX (ακροδέκτης 0) και TX (ακροδέκτης 1) του Arduino αντίστοιχα. Στο παρόν project ήταν αναγκαία η χρήση του Bluesmirf ως δέκτη και όχι ως πομπού και για το λόγο αυτό χρησιμοποιήθηκε μόνο το ζεύγος TX – 0 του Bluesmirf και RX (ακροδέκτης 0) του Arduino μέσω του οποίου γινόταν η λήψη των δεδομένων. Το TX (ακροδέκτης 1) του Arduino χρησιμοποιήθηκε για την αποστολή των λαμβανόμενων δεδομένων μέσω της θύρας USB στον υπολογιστή και την απεικόνιση αυτών στο ενσωματωμένο serial monitor που διαθέτει το interface του Arduino. Με τον τρόπο αυτό, κάθε πακέτο ακατέργαστων δεδομένων που έστειλε το TGAM του Mindwave Mobile, μεταδιδόταν ασύρματα μέσω του Bluesmirf με χρήση πρωτοκόλλου Bluetooth και από εκεί μέσω της θύρας USB του Arduino εμφανιζόταν στο serial monitor του.



Σχήμα 2: Σχηματικό διάγραμμα σύνδεσης BlueSmirf με Arduino

2.3.3 Αισθητήρας Υπερήχων HC – SR04

Για την αποφυγή των εμποδίων που πιθανόν να συναντήσει το αμαξίδιο κατά την εμπρόσθια κίνησή του χρησιμοποιήθηκε ένας **Ανιχνευτής Υπερήχων** και συγκεκριμένα ο **HC-SR04**. Πρόκειται για έναν αισθητήρα που χρησιμοποιείται ευρέως σε πολλές εφαρμογές κυρίως λόγω του χαμηλού κόστους και είναι δυνατόν η μέτρηση της απόστασης από ένα οποιοδήποτε αντικείμενο που βρίσκεται εντός της εμβέλειας του, η οποία είναι από 2 cm έως 400 cm [51]. Η λειτουργία του (Εικόνα 41) βασίζεται στην αποστολή ενός παλμού σκανδαλισμού διάρκειας 10 μs με τον οποίο προκαλείται ένας δεύτερος παλμός διέγερσης αποτελούμενος από 8 μικρότερους παλμούς. Ο παλμός διέγερσης είναι αυτός ο οποίος ταξιδεύει και όταν επιστρέφει καταγράφεται από το ακροδέκτη Echo του αισθητήρα σαν παλμός Ηχώ (Echo Pulse).

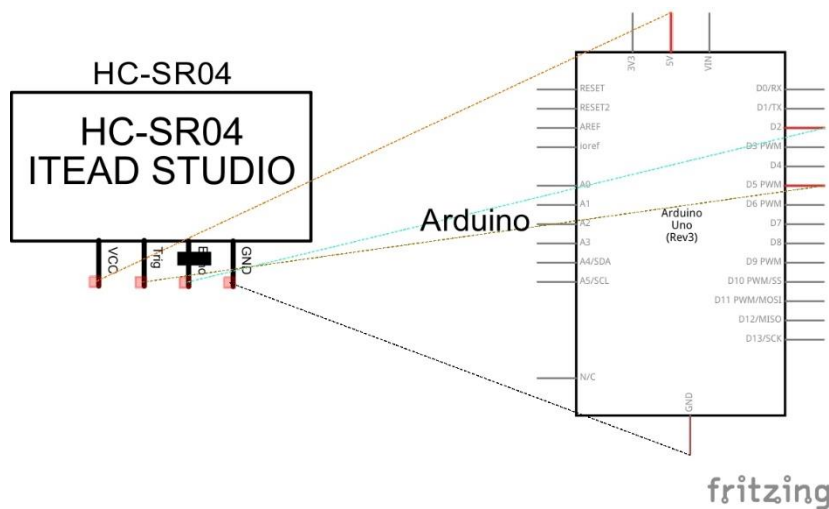


Εικόνα 41: Διάγραμμα λειτουργίας αισθητήρα υπερήχων HC-SR04

Η χρονική διαφορά από τη στιγμή αποστολής του παλμού σκανδαλισμού μέχρι τη στιγμή που επιστρέφει ο παλμός ηχώ χρησιμοποιείται για τον υπολογισμό της απόστασης από το εμπόδιο. Ως γνωστόν, η ταχύτητα διάδοσης του ήχου είναι 340 m/s ή ισοδύναμα χρειάζονται 29 μs για να διανυθεί 1 cm. Λαμβάνοντας επίσης υπόψη ότι ο παλμός διανύει την ίδια απόσταση δύο φορές, μία για να πάει μέχρι το αντικείμενο και μία για να επιστρέψει από αυτό, τότε γνωρίζοντας τη διάρκεια του παλμού ο τύπος που μας δίνει την απόσταση από το εμπόδιο είναι ο εξής:

$$\text{απόσταση (cm)} = \frac{\text{διάρκεια παλμού (}\mu\text{s)}}{\frac{29}{2}}$$

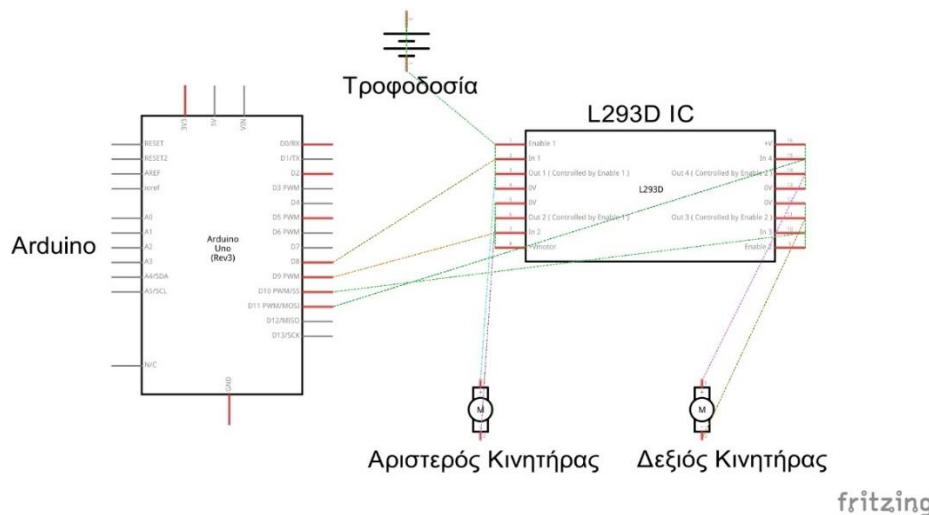
Η σύνδεση του αισθητήρα υπερήχων στον Arduino (Σχήμα 3) είναι αρκετά απλή. Συνδέουμε τον ακροδέκτη V_{cc} στον ακροδέκτη 5V του Arduino και τον ακροδέκτη GND σε έναν από τους ακροδέκτες του Arduino που αντιστοιχούν σε γείωση. Ο ακροδέκτης TRIG συνδέεται σε κάποιον από τους ψηφιακούς ακροδέκτες 4-13 ενώ ο ακροδέκτης ECHO συνδέεται σε έναν από τους δύο ακροδέκτες 2 ή 3 του Arduino που αντιστοιχούν στους ακροδέκτες για τα εξωτερικά interrupts που παρέχει. Συγκεκριμένα, οι ακροδέκτες 2 και 3 αντιστοιχούν στα interrupts 0 και 1 του Arduino αντίστοιχα.



Σχήμα 3: Σχηματικό διάγραμμα σύνδεσης HC-SR04 με Arduino

2.3.4 Ολοκληρωμένο Κύκλωμα L293D

Το ολοκληρωμένο κύκλωμα L293D έχει σκοπό να οδηγήσει τους δύο DC κινητήρες (περισσότερα για τους DC κινητήρες στο [Παράρτημα Β](http://dl.nmh9ip6v2uc.cloudfront.net/datasheets/Robotics/DG01D.pdf)) DG01D - A130 (<http://dl.nmh9ip6v2uc.cloudfront.net/datasheets/Robotics/DG01D.pdf>). Έχει τη δυνατότητα να αντέξει ρεύμα μέγιστης εντάσεως 1A, πολύ μεγαλύτερο από το μέγιστο ρεύμα των 250 mA που μπορεί να τραβήξει κάθε ένας από τους κινητήρες.



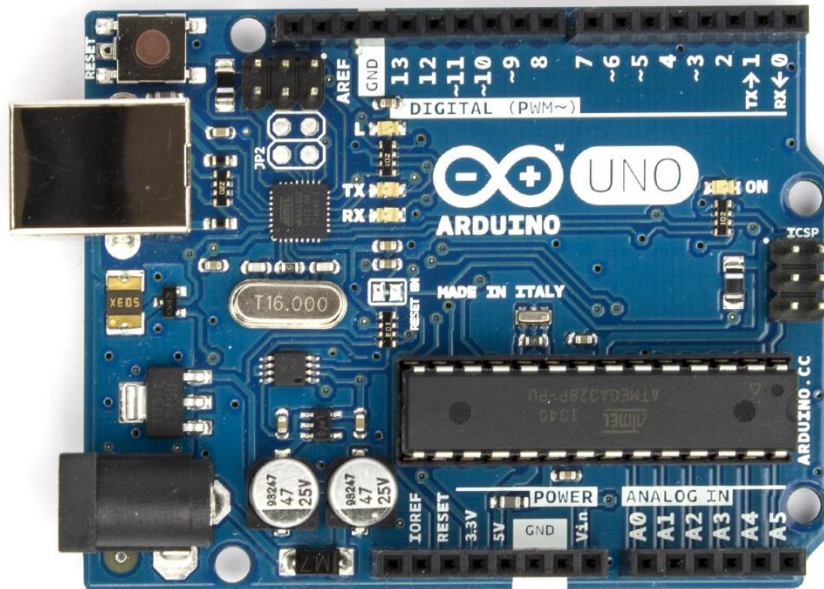
Σχήμα 4: Σχηματικό διάγραμμα σύνδεσης IC L293D με Arduino

2.3.5 Arduino

Στο σημείο αυτό θα ήταν χρήσιμο να δοθούν μερικές πληροφορίες σχετικά με την καρδιά όλων των εξαρτημάτων και των συστημάτων που χρησιμοποιήθηκαν, τον μικροελεγκτή Arduino.

2.3.5.1 Τι είναι το Arduino

Το Arduino (Εικόνα 42) είναι μία υπολογιστική πλατφόρμα ανοικτού υλικού βασισμένη σε έναν απλό μικροελεγκτή και ένα φιλικό περιβάλλον ανάπτυξης λογισμικού για την πλατφόρμα. Μπορεί να χρησιμοποιηθεί για την ανάπτυξη διαδραστικών αντικειμένων, λαμβάνοντας δεδομένα από μια ποικιλία από διακόπτες ή αισθητήρες, ελέγχοντας μια ποικιλία περιφερειακών εξαρτημάτων, όπως LEDs, κινητήρες και άλλες φυσικές εξόδους. Τα αντικείμενα αυτά μπορούν να χρησιμοποιηθούν είτε ως αυτόνομες εφαρμογές είτε συνδυαστικά με άλλα προγράμματα στον υπολογιστή (π.χ. Flash, Processing). Το υλικό είναι δυνατόν να συναρμολογηθεί από τον χρήστη ή να έρθει προσυναρμολογημένο. Το προγραμματιστικό περιβάλλον είναι μία εφαρμογή σε Wiring, μία παρόμοια φυσική υπολογιστική πλατφόρμα που έχει βασιστεί στο προγραμματιστικό περιβάλλον Processing [52].



Εικόνα 42: Arduino Uno Rev. 3 [53]

2.3.5.2 Μικροελεγκτής ATmega328

Το Arduino βασίζεται στο μικροελεγκτή ATmega328 της Atmel, έναν 8-bit **επεξεργαστή περιορισμένου συνόλου εντολών** (Reduced Instruction Set Computer, R.I.S.C), με συχνότητα λειτουργίας στα 16MHz. Ο ATmega328 διαθέτει ενσωματωμένη μνήμη τριών τύπων:

- 2KB μνήμης **SRAM** που είναι η ωφέλιμη μνήμη που μπορούν να χρησιμοποιήσουν οι χρήστες στα προγράμματά τους για να αποθηκεύουν μεταβλητές, πίνακες κ.λπ. Όπως και σε έναν υπολογιστή, αυτή η μνήμη χάνει τα δεδομένα της όταν η παροχή ρεύματος στο Arduino σταματήσει ή αν γίνει reset.
- 1KB μνήμης **EEPROM** η οποία μπορεί να χρησιμοποιηθεί για εγγραφή/ανάγνωση δεδομένων ανά byte από τα προγράμματα των χρηστών. Μπορούμε να παραλληλίσουμε τη μνήμη EEPROM με ένα σκληρό δίσκο μιας και δεν χάνει τα περιεχόμενά της με απώλεια τροφοδοσίας ή reset όπως η SRAM.
- 32KB μνήμης **Flash**, από τα οποία τα 2Kb χρησιμοποιούνται από το υλικολογισμικό (firmware) του Arduino που έχει εγκαταστήσει ήδη ο κατασκευαστής του. Το firmware αυτό που ονομάζεται **bootloader** είναι αναγκαίο για την εγκατάσταση των προγραμμάτων των χρηστών στον μικροελεγκτή μέσω της θύρας USB. Τα υπόλοιπα 30Kb της μνήμης Flash χρησιμοποιούνται για την αποθήκευση αυτών ακριβώς των προγραμμάτων, αφού πρώτα γίνει η μεταγλώττιση στον υπολογιστή. Η μνήμη Flash, όπως και η EEPROM δεν χάνει τα περιεχόμενά της με απώλεια τροφοδοσίας ή reset.

2.3.5.3 Είσοδοι - Έξοδοι

Ο μικροελεγκτής ATmega υποστηρίζει σειριακή επικοινωνία, την οποία το Arduino προωθεί μέσα από έναν ελεγκτή Serial-over-USB ώστε να συνδέεται με τον υπολογιστή μέσω USB. Η σύνδεση αυτή χρησιμοποιείται για την μεταφορά των προγραμμάτων από τον υπολογιστή στο Arduino αλλά και για αμφίδρομη επικοινωνία του Arduino με τον υπολογιστή την ώρα που εκτελείται ένα πρόγραμμα.

Στη μία πλευρά του Arduino βρίσκονται 14 ψηφιακοί ακροδέκτες με την ένδειξη DIGITAL (PWM~), αριθμημένα από 0 ως 13, που μπορούν να λειτουργήσουν ως είσοδοι και έξοδοι χρησιμοποιώντας τις συναρτήσεις `pinMode()`, `digitalWrite()` και `digitalRead()`.

Λειτουργούν στα 5V και καθένα μπορεί να παρέχει ή να δεχτεί το πολύ 40mA. Μερικά από αυτούς τους 14 ακροδέκτες, εκτός από ψηφιακές εισοδοί/έξοδοι έχουν και δεύτερη λειτουργία. Συγκεκριμένα:

- Οι ακροδέκτες 0 και 1 λειτουργούν ως RX και TX της σειριακής όταν το πρόγραμμα ενεργοποιεί τη σειριακή θύρα. Έτσι, όταν στέλνονται δεδομένα μέσω της σειριακής, αυτά προωθούνται και στην θύρα USB μέσω του ελεγκτή Serial-Over-USB αλλά και στον ακροδέκτη 0 για να τα διαβάσει ενδεχομένως μια άλλη συσκευή. Αυτό έχει σαν συνέπεια να χάνεται η δυνατότητα χρήσης των συγκεκριμένων ακροδεκτών όταν ενεργοποιείται το σειριακό interface.
- Οι ακροδέκτες 2 και 3 λειτουργούν και ως εξωτερικές διακοπές (interrupts) (interrupts 0 και 1 αντίστοιχα). Αυτό είναι μια πολύ σημαντική ιδιότητα αφού όταν συμβαίνουν συγκεκριμένες αλλαγές, η κανονική ροή του προγράμματος σταματάει άμεσα και εκτελείται μια συγκεκριμένη συνάρτηση. Τα εξωτερικά interrupts είναι ιδιαίτερα χρήσιμα σε εφαρμογές που απαιτούν συγχρονισμό μεγάλης ακρίβειας.
- Οι ακροδέκτες 3, 5, 6, 9, 10 και 11 μπορούν να λειτουργήσουν και ως ψευδοαναλογικές έξοδοι με την τεχνική **Διαμόρφωσης Πλάτους Παλμού** (Pulse Width Modulation, PWM).

Στην άλλη πλευρά του Arduino, υπάρχουν 6 ακροδέκτες με τη σήμανση ANALOG IN, αριθμημένοι από το 0 ως το 5, το καθένα από τα οποία λειτουργεί ως αναλογική είσοδος κάνοντας χρήση του ADC (Analog to Digital Converter) που είναι ενσωματωμένο στον μικροελεγκτή. Παρέχεται έτσι η δυνατότητα στον χρήστη να δέχεται δεδομένα από αναλογική πηγή (π.χ. ποτενσιόμετρο) και να βλέπει την ψηφιακή απεικόνιση των διαφόρων τιμών μέσα από ακέραιο αριθμό ανάλυσης 10-bit.

Δίπλα από τους ακροδέκτες αναλογικής εισόδου, υπάρχει συστοιχία από 6 ακροδέκτες με την σήμανση POWER. Η λειτουργία του καθενός έχει ως εξής [54]:

- Ο πρώτος, με την ένδειξη RESET, όταν γειωθεί (σε οποιοδήποτε από τους 3 ακροδέκτες με την ένδειξη GND που υπάρχουν στο Arduino) έχει ως αποτέλεσμα την επανεκκίνηση του Arduino.
- Ο δεύτερος, με την ένδειξη 3.3V, μπορεί να τροφοδοτήσει τα εξαρτήματα με τάση 3.3V.
- Ο τρίτος, με την ένδειξη 5V, μπορεί να τροφοδοτήσει τα εξαρτήματα με τάση 5V.
- Ο τέταρτος και ο πέμπτος ακροδέκτες, με την ένδειξη GND, είναι γειώσεις.
- Ο έκτος και τελευταίος ακροδέκτης, με την ένδειξη Vin έχει διπλό ρόλο. Σε συνδυασμό με τον ακροδέκτη γείωσης δίπλα του, μπορεί να λειτουργήσει ως μέθοδος εξωτερικής τροφοδοσίας του Arduino, στην περίπτωση που δεν μπορεί να χρησιμοποιηθεί η υποδοχή του φισ των 2,1mm. Αν όμως υπάρχει συνδεδεμένη εξωτερική τροφοδοσία μέσω του φισ, τότε μέσω του ακροδέκτη V_{in} μπορούν να τροφοδοτηθούν εξαρτήματα με την πλήρη τάση της εξωτερικής τροφοδοσίας (7~12V), πριν αυτή περάσει από τον ρυθμιστή τάσης όπως γίνεται με τον ακροδέκτη των 5V.

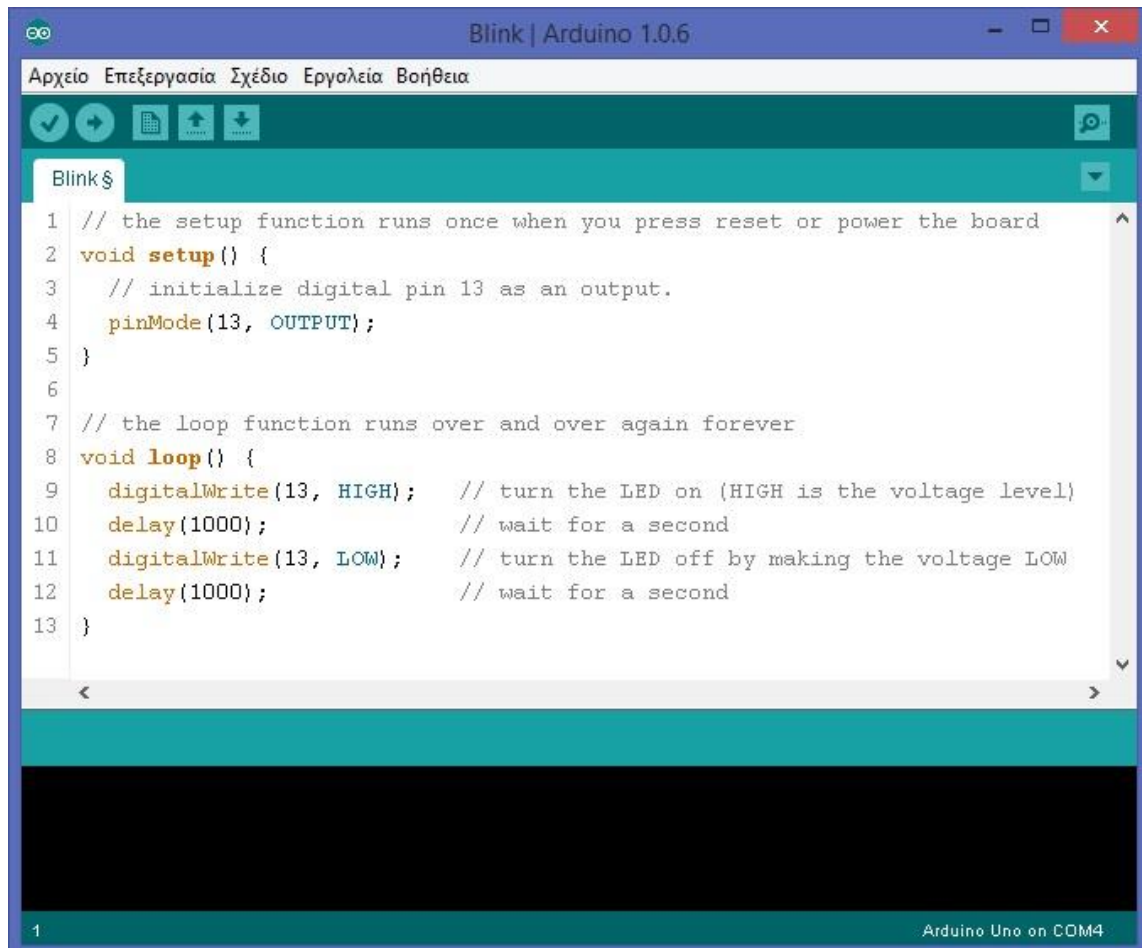
2.3.5.4 Τροφοδοσία

Το Arduino μπορεί να τροφοδοτηθεί με ρεύμα είτε από τον υπολογιστή μέσω της σύνδεσης USB, είτε από εξωτερική τροφοδοσία που παρέχεται μέσω μιας υποδοχής φισ των 2,1mm. Για να μην υπάρχουν προβλήματα, η εξωτερική τροφοδοσία πρέπει να είναι από 7V ως 12V και μπορεί να προέρχεται από έναν κοινό μετασχηματιστή του εμπορίου, από μπαταρίες ή οποιαδήποτε άλλη πηγή DC.

2.3.5.5 Γραφικό Περιβάλλον Προγραμματισμού (Programming Interface)

Ο προγραμματισμός του Arduino γίνεται μέσω ειδικού περιβάλλοντος (Εικόνα 43) το οποίο είναι βασισμένο στη γλώσσα προγραμματισμού Wiring, η οποία παρουσιάζει αρκετές ομοιότητες με τη γλώσσα C.

Ο βασικός κορμός ενός προγράμματος (sketch) του Arduino αποτελείται από δύο κύριες συναρτήσεις. Η πρώτη ονομάζεται **setup()** η οποία εκτελείται μόνο μία φορά, στην αρχή του προγράμματος, και σε αυτήν αρχικοποιούνται οι μεταβλητές του. Η δεύτερη ονομάζεται **loop()** και εκτελείται συνεχώς από τον μικροελεγκτή.

The image shows the Arduino IDE interface. The title bar reads "Blink | Arduino 1.0.6". The menu bar includes "Αρχείο", "Επεξεργασία", "Σχέδιο", "Εργαλεία", and "Βοήθεια". The toolbar contains icons for file operations and execution. The main text area shows the following code:

```
1 // the setup function runs once when you press reset or power the board
2 void setup() {
3   // initialize digital pin 13 as an output.
4   pinMode(13, OUTPUT);
5 }
6
7 // the loop function runs over and over again forever
8 void loop() {
9   digitalWrite(13, HIGH); // turn the LED on (HIGH is the voltage level)
10  delay(1000); // wait for a second
11  digitalWrite(13, LOW); // turn the LED off by making the voltage LOW
12  delay(1000); // wait for a second
13 }
```

The status bar at the bottom indicates "1" on the left and "Arduino Uno on COM4" on the right.

Εικόνα 43: Περιβάλλον προγραμματισμού Arduino

Περισσότερες πληροφορίες σχετικά με τον προγραμματισμό του Arduino, μπορεί κανείς να βρει στη διεύθυνση <http://arduino.cc/en/Guide/HomePage>.

2.4 Διάγραμμα ροής συστήματος

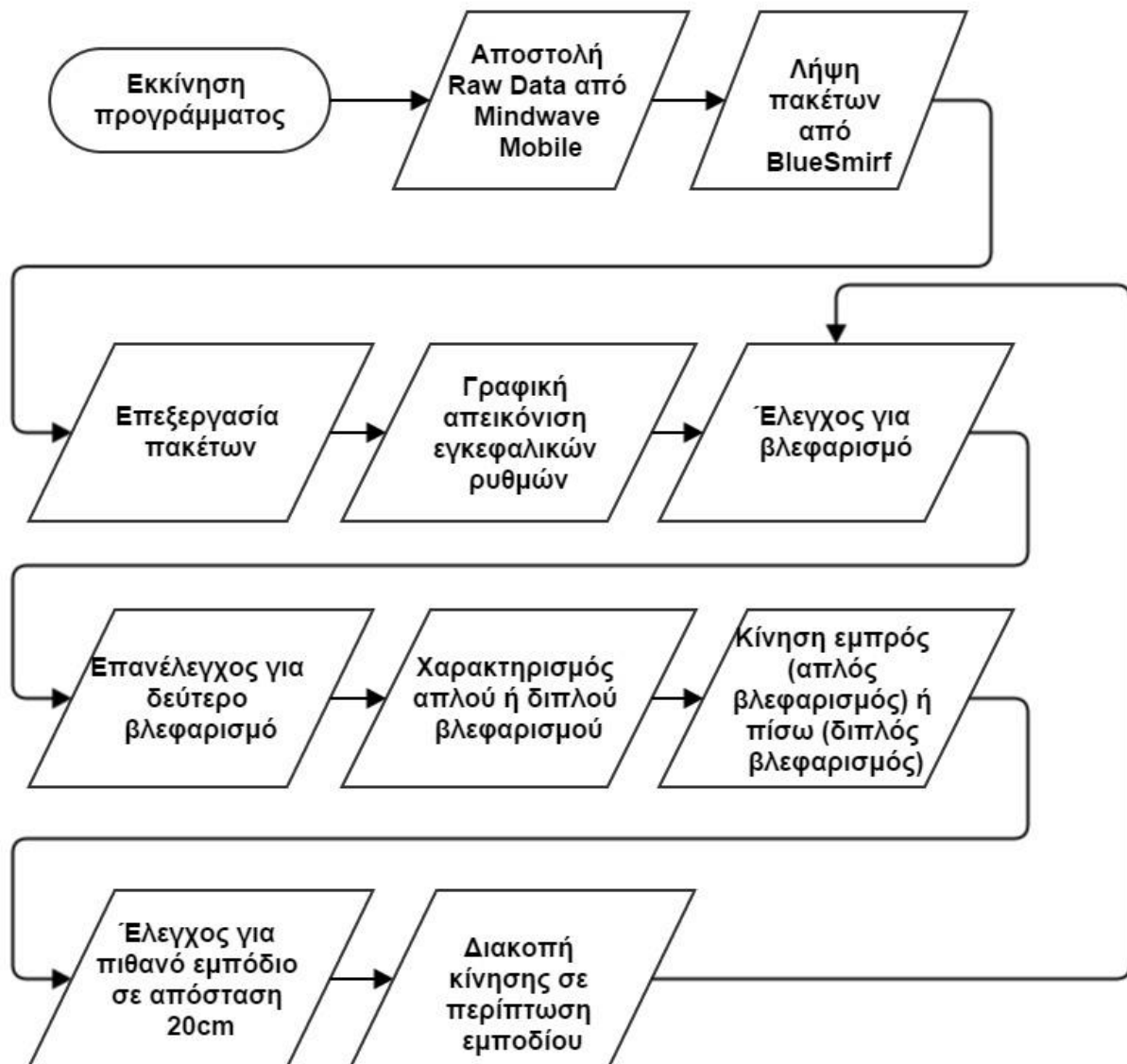
Το σύστημα εκτελεί έναν συγκεκριμένο κύκλο λειτουργίας, ο οποίος επαναλαμβάνεται συνεχώς. Όταν το σύστημα τεθεί σε λειτουργία, το Mindwave Mobile αποστέλλει συνεχώς δεδομένα στο Arduino μέσω πρωτοκόλλου Bluetooth. Κάθε πακέτο που λαμβάνεται οδηγείται προς επεξεργασία για την εξαγωγή των τιμών των επιπέδων προσοχής και συγκέντρωσης, των εγκεφαλικών ρυθμών και των ακατέργαστων δεδομένων. Σε περίπτωση που το σύστημα έχει συνδεθεί σε κάποιον υπολογιστή μέσω θύρας USB μπορεί να γίνει γραφική απεικόνιση των τιμών αυτών.

Στη συνέχεια, γίνεται έλεγχος των τιμών των ακατέργαστων δεδομένων για την εύρεση βλεφαρισμών. Ανάλογα με τη χρονική διαφορά μεταξύ δύο συνεχόμενων βλεφαρισμών,

το αμαξίδιο κινείται είτε εμπρός στη περίπτωση απλού βλεφαρισμού, είτε προς τα πίσω για 1 sec σε περίπτωση διπλού βλεφαρισμού.

Καθώς το σύστημα βρίσκεται σε εμπρός κίνηση, γίνεται έλεγχος της απόστασης από πιθανά εμπόδια, με χρήση ενός αισθητήρα υπερήχων. Σε περίπτωση που ανιχνευθεί εμπόδιο σε απόσταση μικρότερη των 20 cm, το αμαξίδιο σταματάει να κινείται και το σύστημα τίθεται σε αναμονή μέχρι τον επόμενο απλό ή διπλό βλεφαρισμό.

Ο παραπάνω κύκλος λειτουργίας φαίνεται στο ακόλουθο διάγραμμα ροής.



Σχήμα 5: Διάγραμμα ροής συστήματος

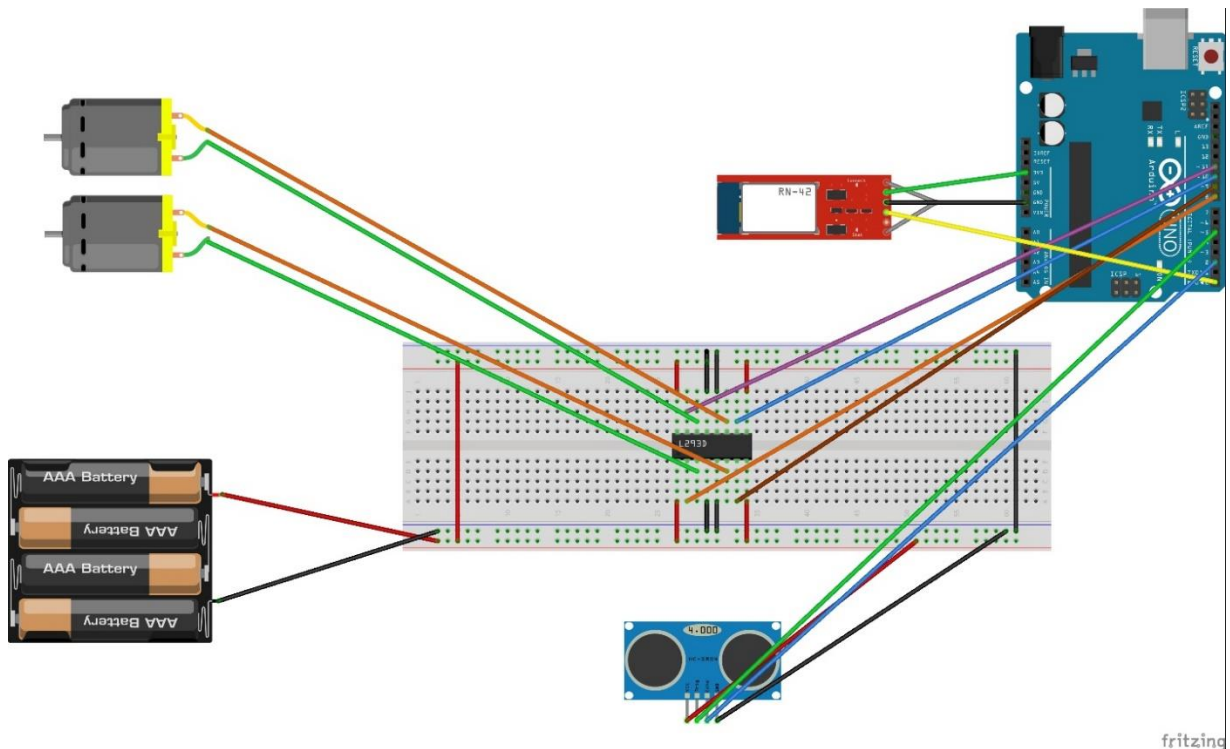
3. ΥΛΟΠΟΙΗΣΗ ΣΥΣΤΗΜΑΤΟΣ

Η υλοποίηση της παρούσας εργασίας αποτελείται από δύο βασικά μέρη:

1. Τις συνδεσμολογίες του Υλικού (Hardware)
2. Τη δημιουργία του κατάλληλου αλγορίθμου (Software)

3.1 Συνδεσμολογία κυκλώματος ελέγχου

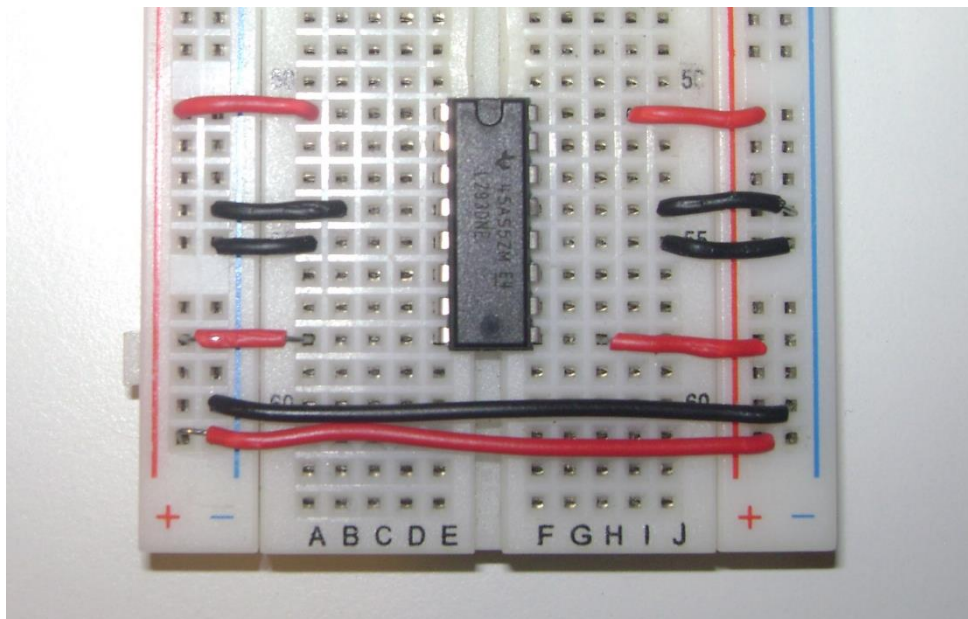
Αφού έγινε η συγκέντρωση των απαραίτητων υλικών (βλ. παρ 2.2), έπρεπε γίνει η κατάλληλη συνδεσμολογία όλων αυτών, όπως στην εικόνα 44.



Εικόνα 44: Βασική συνδεσμολογία κυκλώματος ελέγχου

3.1.1 Σύνδεση Ολοκληρωμένου Κυκλώματος L293D

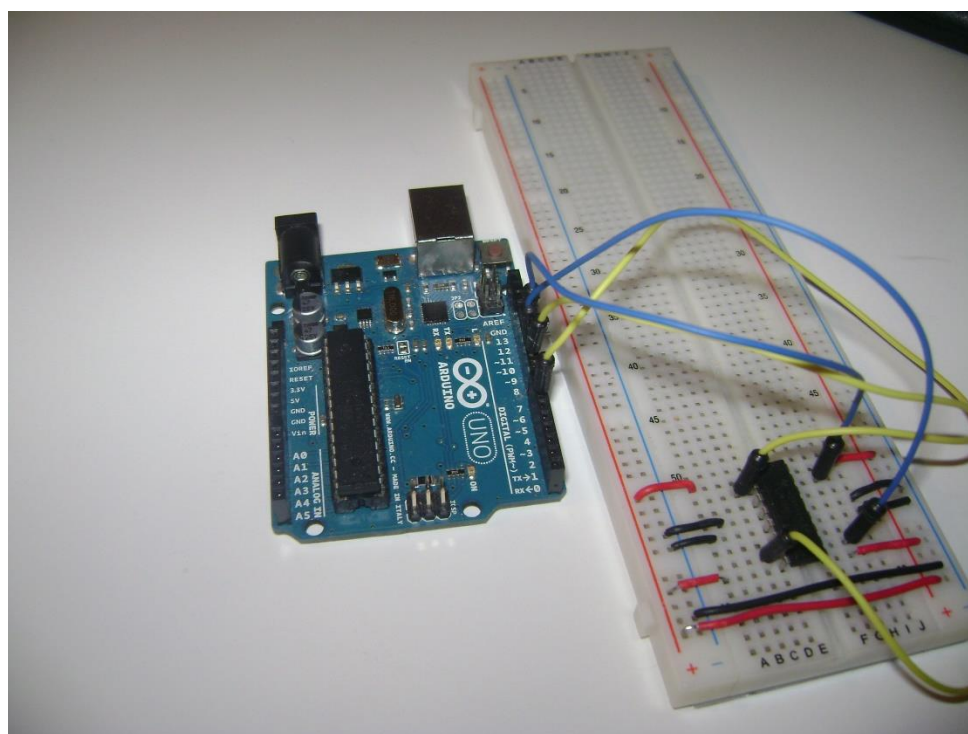
Αρχικά τοποθετήθηκε το ολοκληρωμένο κύκλωμα L293D που υλοποιεί τη γέφυρα H, στο αριστερό μέρος του breadboard και στη συνέχεια έγιναν οι απαραίτητες ενώσεις για την τροφοδοσία του ολοκληρωμένου. Σύμφωνα με το φυλλάδιο τεχνικών προδιαγραφών που δίνει η κατασκευάστρια εταιρεία (<http://www.ti.com/lit/ds/symlink/l293d.pdf>), στους ακροδέκτες 1 και 9 είναι αναγκαίο να δοθούν τάσεις ενεργοποίησης για τον πρώτο και δεύτερο κινητήρα αντίστοιχα. Στην περίπτωση μας οι τάσεις αυτές είναι ίσες με 5V που προέρχονται από την πηγή της τροφοδοσίας (μπαταρίες). Επιπρόσθετα, στους ακροδέκτες 8 και 16 αντιστοιχούν δύο τάσεις τροφοδοσίας του ολοκληρωμένου V_{cc2} και V_{cc1} οι οποίες είναι και αυτές 5V. Σύμφωνα επίσης με το φυλλάδιο του κατασκευαστή, οι ακροδέκτες 4, 5, 12 και 13 οδηγούνται στη γείωση. Μετά από αυτά τα βήματα, το ολοκληρωμένο είναι έτοιμο όσο αφορά την τροφοδοσία του και την προετοιμασία του να δεχτεί τις υπόλοιπες συνδέσεις (Εικόνα 45).



Εικόνα 45: Προετοιμασία συνδέσεων ολοκληρωμένου κυκλώματος

3.1.2 Σύνδεση Arduino με L293D

Ο έλεγχος των κινητήρων γίνεται με κατάλληλα σήματα που παράγει το Arduino μέσα από τον αλγόριθμο που έχουμε υλοποιήσει. Τα σήματα στέλνονται από τις ψηφιακές εξόδους 6 και 9 του Arduino προς τους ακροδέκτες 2 (1A) και 7 (2A) του L293D για τον πρώτο (δεξιό) κινητήρα και από τις ψηφιακές εξόδους 10 και 11 του Arduino προς τους ακροδέκτες 10 (3A) και 15 (4A) του L293D για το δεύτερο (αριστερό) κινητήρα. Στην εικόνα 46 φαίνεται αυτή ακριβώς η συνδεσμολογία.

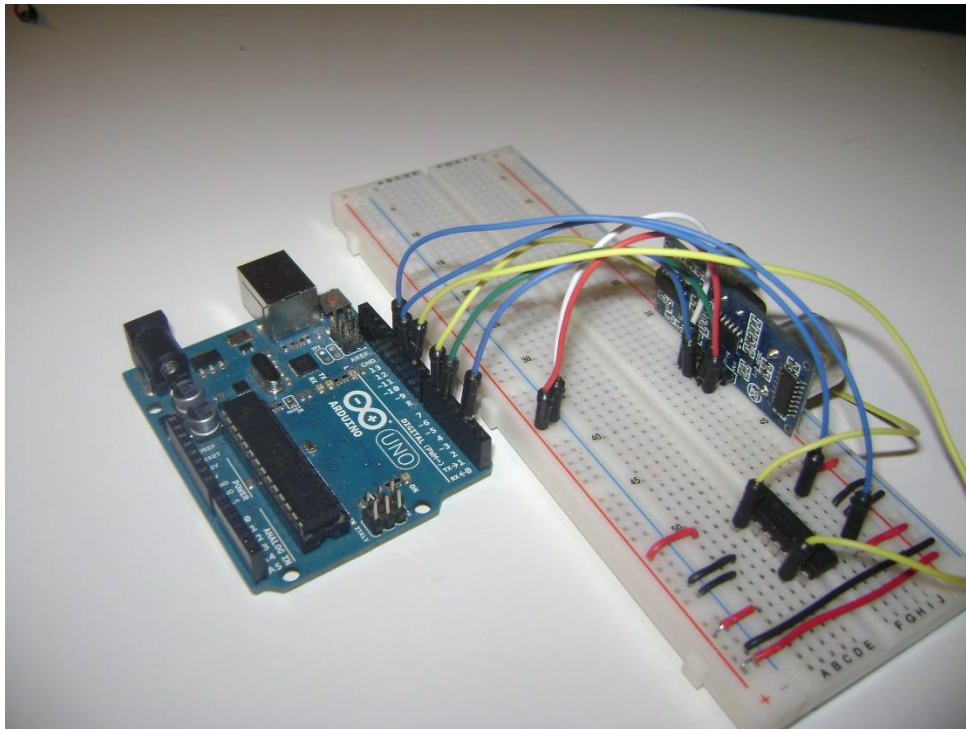


Εικόνα 46: Σύνδεση Arduino με L293D

3.1.3 Σύνδεση αισθητήρα Υπερήχων HC – SR04 με Arduino

Επόμενο βήμα ήταν η σύνδεση του αισθητήρα υπερήχων HC - SR04 με το Arduino για να γίνεται η ανίχνευση πιθανών εμποδίων που εντοπίζονται κατά την κίνηση του αμαξιδίου εμπρός. Σκοπός του αισθητήρα αυτού είναι να δίνει εντολή στο αμαξίδιο να σταματάει την κίνηση σε περίπτωση που ανιχνευτεί εμπόδιο σε απόσταση 20 cm.

Ο αισθητήρα αυτός έχει τέσσερις ακροδέκτες: V_{cc} για την τροφοδοσία του στα 5V, GND για γείωση, TRIG για την αποστολή του παλμού διέγερσης και ECHO για τη λήψη του παλμού επιστροφής. Ο ακροδέκτης V_{cc} συνδέεται στην τροφοδοσία 5V που υπάρχει στο breadboard και ο ακροδέκτης GND στην γείωση που υπάρχει επίσης στο breadboard. Οι ακροδέκτες όμως TRIG και ECHO συνδέονται στον Arduino και συγκεκριμένα στους ακροδέκτες 5 και 2 αντίστοιχα. Στην εικόνα 47 φαίνεται η συνδεσμολογία αυτή.

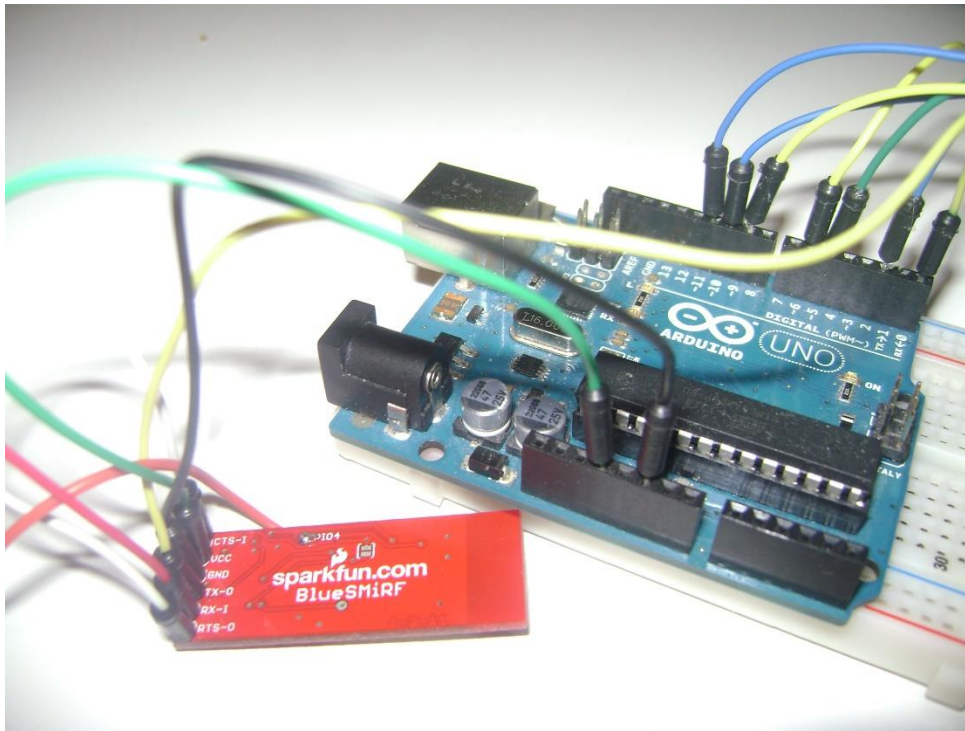


Εικόνα 47: Σύνδεση αισθητήρα Υπερήχων HC – SR04 με Arduino

3.1.4 Σύνδεση Bluetooth module Bluesmirf RN - 42

Στο επόμενο βήμα ήταν απαραίτητο να γίνει η σύνδεση του Bluesmirf, του Bluetooth module με το οποίο θα γινόταν η ασύρματη ζεύξη του Mindwave Mobile προς το Arduino. Το Bluesmirf RN – 42 διαθέτει 6 ακροδέκτες εκ των οποίων οι δύο είναι απενεργοποιημένοι (έχουν ενωθεί μεταξύ τους σύμφωνα με οδηγίες της Neurosky). Οι τέσσερις εναπομείναντες ακροδέκτες είναι ο ακροδέκτης τροφοδοσίας V_{cc} , ο ακροδέκτης της γείωσης GND, ο ακροδέκτης TX – 0 για την μετάδοση δεδομένων και ο ακροδέκτης RX – 1 για τη λήψη δεδομένων. Επειδή στην παρούσα εργασία μας ενδιέφερε μόνο η αποστολή δεδομένων από το Bluesmirf προς το Arduino συνδέσαμε μόνο τον ακροδέκτη TX – 0 αφήνοντας τον RX – 1 ελεύθερο.

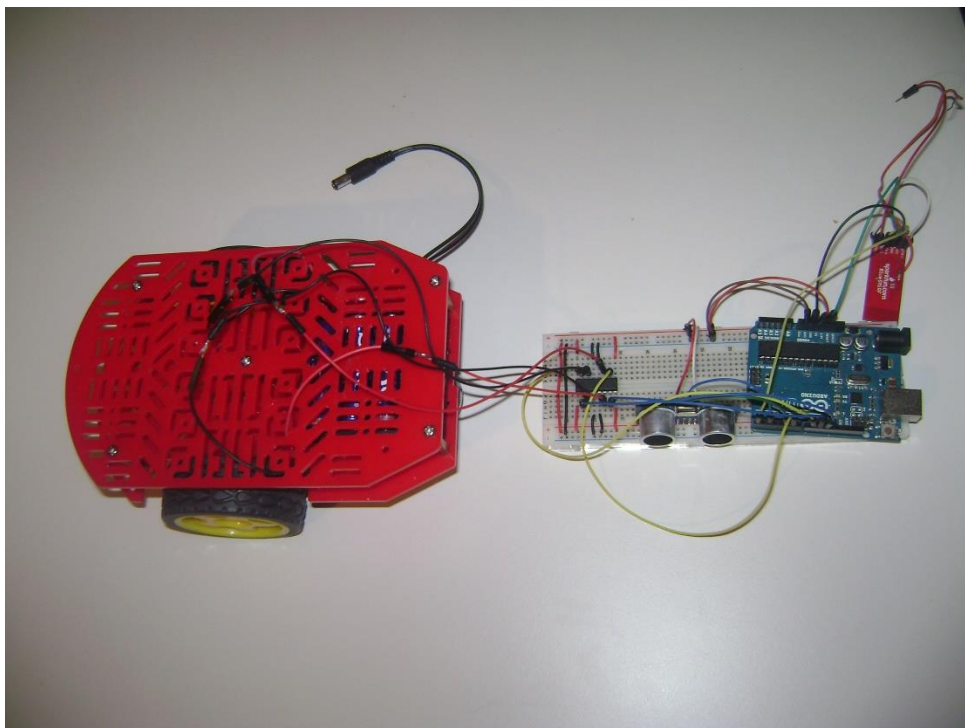
Συγκεκριμένα συνδέσαμε (Εικόνα 48) το V_{cc} στον ακροδέκτη 3,3 V του Arduino, τη γείωση GND σε αντίστοιχο ακροδέκτη του Arduino και τον TX – 0 στο ακροδέκτη 0 του Arduino που ονομάζεται RX.



Εικόνα 48: Σύνδεση Bluetooth module Bluesmifrn - 42

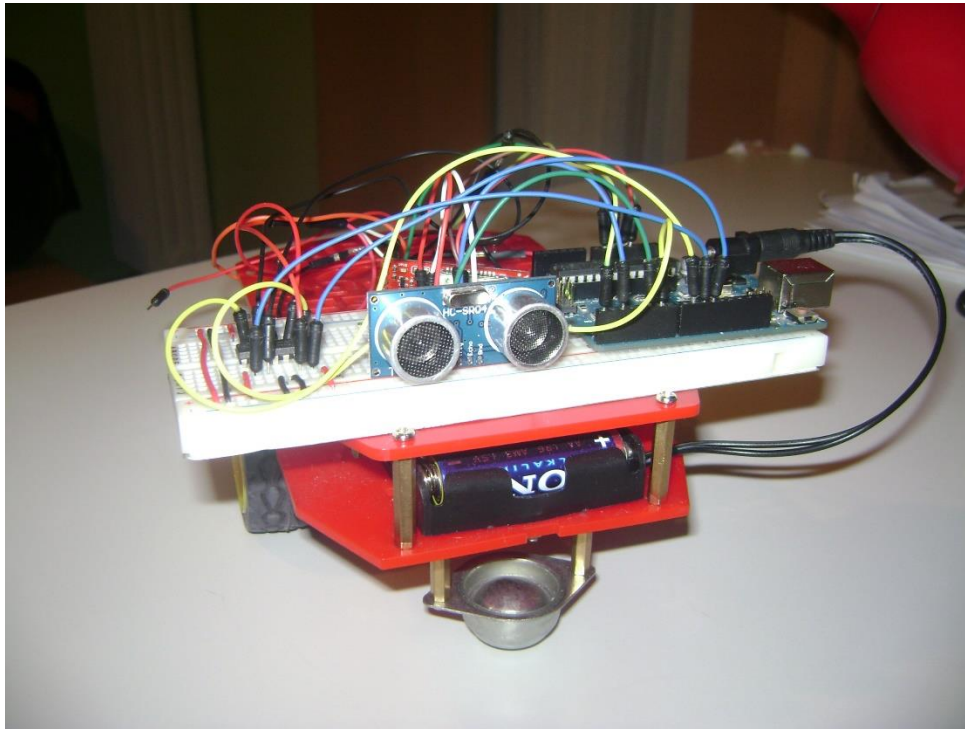
3.1.5 Σύνδεση κινητήρων αμαξιδίου

Η σύνδεση των κινητήρων είναι αρκετά εύκολη (Εικόνα 49). Ο ένας εκ των δύο κινητήρων συνδέεται στους ακροδέκτες 3 (1Y) και 6 (2Y) του chip ενώ ο δεύτερος κινητήρας στους ακροδέκτες 11 (3Y) και 14 (4Y). Η πολικότητα σύνδεσης είναι αδιάφορη αλλά σε περίπτωση αντίστροφης περιστροφής του κινητήρα απ' ό,τι το αναμενόμενο απλά αλλάζουμε τις συνδέσεις στους ακροδέκτες.



Εικόνα 49: Σύνδεση κινητήρων αμαξιδίου

Το τελικό μας σχέδιο έχει την ακόλουθη μορφή:

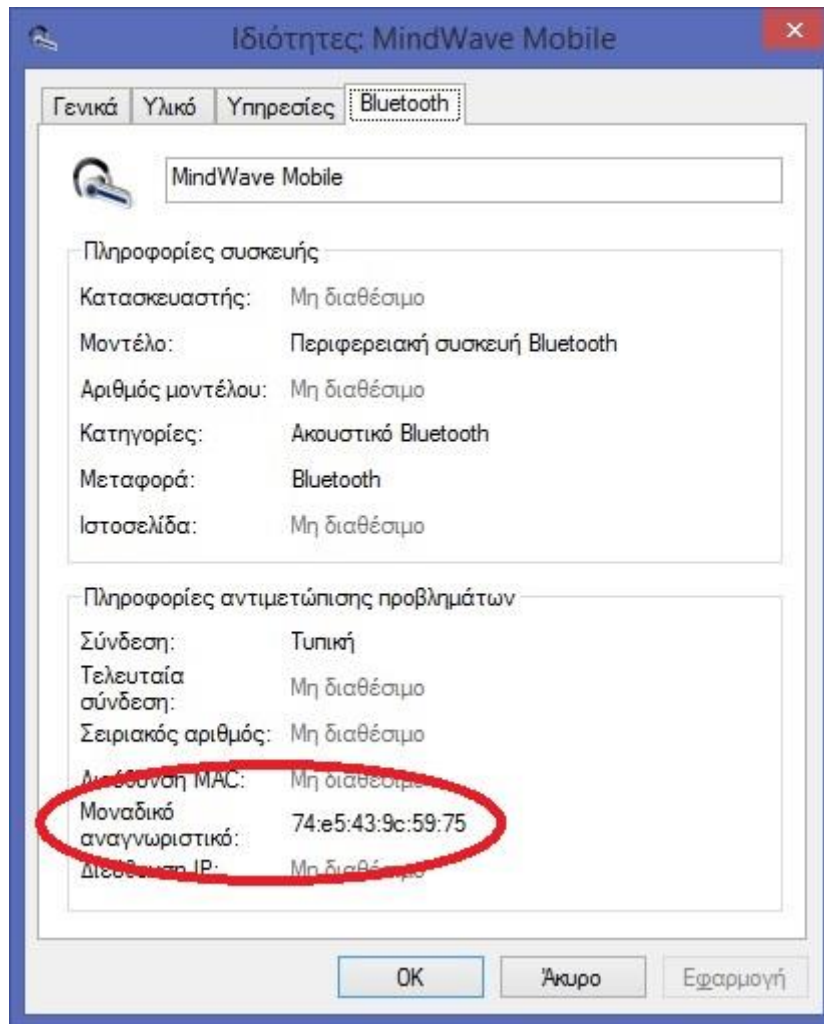


Εικόνα 50: Τελική μορφή σχεδίου

3.2 Ρυθμίσεις επικοινωνίας και σύνδεσης Mindwave Mobile και Arduino

Αφού γίνουν οι απαραίτητες συνδεσμολογίες του υλικού, το επόμενο βήμα είναι να γίνουν κάποιες ρυθμίσεις για να μπορέσει να επικοινωνήσει το Mindwave Mobile με το Bluesmirf και κατ' επέκταση με το Arduino. Η εταιρεία Neurosky στον παρακάτω υπερσύνδεσμο δίνει σαφείς οδηγίες τις οποίες ακολουθήσαμε για να επιτύχουμε την επικοινωνία. http://developer.neurosky.com/docs/doku.php?id=mindwave_mobile_and_arduino.

Σαν πρώτο βήμα πρέπει να γίνει λήψη από το διαδίκτυο του παρακάτω προγράμματος με όνομα Coolterm (<http://freeware.the-meiers.org/>). Το πρόγραμμα αυτό σκοπό έχει να θέσει τις σωστές παραμέτρους στο Mindwave Mobile πριν αυτό συνδεθεί στο Bluesmirf. Επειδή η επικοινωνία γίνεται μέσω της διεύθυνσης MAC του Mindwave Mobile, την πρώτη φορά θα χρειαστεί να υπάρχει διαθέσιμο ένα Bluetooth dongle μέσω του οποίου θα είναι δυνατή η ανεύρεση της διεύθυνσης MAC. Στην παρούσα εργασία η διεύθυνση MAC φαίνεται στην ακόλουθη εικόνα.

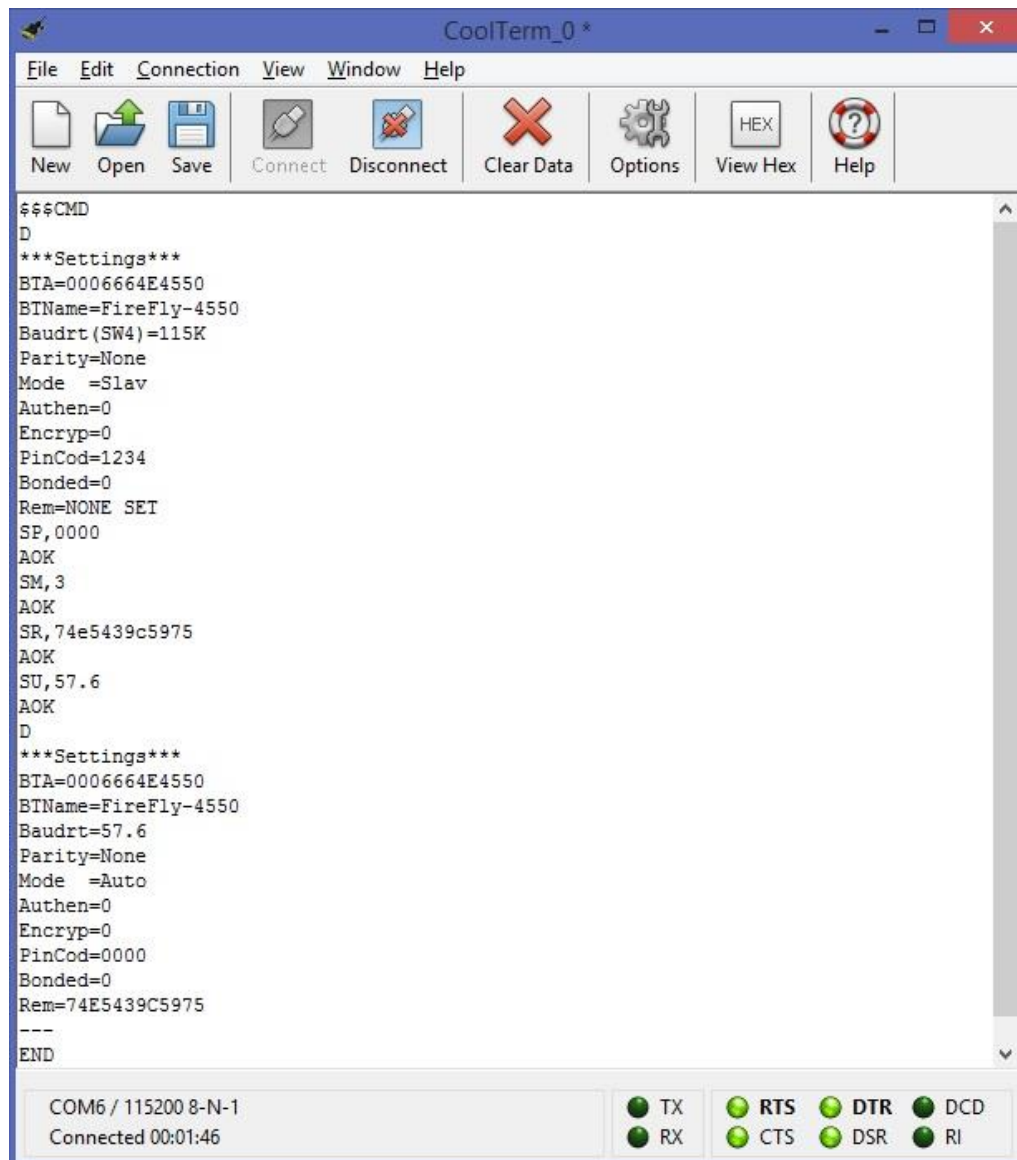


Εικόνα 51: Διεύθυνση MAC Mindwave Mobile

Έπειτα αφού έχει επιτευχθεί σύνδεση του Bluesmirf με το Arduino, το κόκκινο LED του Bluesmirf θα αρχίσει να αναβοσβήνει. Στο πρόγραμμα Coolterm, στην καρτέλα Serial Port είναι απαραίτητο να τεθεί Port στη θύρα που βρίσκεται το Arduino και το BaudRate στα 115200 ενώ στην καρτέλα Terminal πρέπει να ενεργοποιηθεί το «Local Echo» έτσι ώστε να είναι δυνατή η ανάγνωση των αποτελεσμάτων από τις όποιες ρυθμίσεις πραγματοποιηθούν. Στη συνέχεια από το κεντρικό περιβάλλον του Coolterm, πατώντας το πλήκτρο Connect πρέπει να ακολουθηθούν με τη σειρά τα παρακάτω βήματα (Εικόνα 50):

1. Πληκτρολόγηση \$\$\$ για την είσοδο σε περιβάλλον Command Mode. Αμέσως θα δούμε επιστρέφεται το μήνυμα CMD που επιβεβαιώνει την είσοδο σε Command Mode.
2. Πληκτρολόγηση D και Enter για την προβολή των τρεχουσών ρυθμίσεων του Bluesmirf.
3. Πληκτρολόγηση SP,0000 και Enter για την αλλαγή του τρέχοντος κωδικού από 1234 σε 0000. Αν έγινε σωστά η ρύθμιση θα εμφανιστεί το μήνυμα AOK.
4. Πληκτρολόγηση SM,3 και Enter για να τεθεί το Bluesmirf σε κατάσταση Auto – Connect. Αν έγινε σωστά η ρύθμιση θα εμφανιστεί το μήνυμα AOK.
5. Πληκτρολόγηση SR,Διεύθυνση MAC του Mindwave Mobile και Enter. Αν έγινε σωστά η ρύθμιση θα εμφανιστεί το μήνυμα AOK.
6. Πληκτρολόγηση SU,57.6 και Enter για την αλλαγή του Baud Rate από 115200 σε 57600. Αν έγινε σωστά η ρύθμιση θα εμφανιστεί το μήνυμα AOK.

7. Πληκτρολόγηση D και Enter για την απεικόνιση των νέων ρυθμίσεων του Bluesmirf.
8. Πληκτρολόγηση --- για την έξοδο από το Command Mode. Αν έγινε σωστά η ρύθμιση θα εμφανιστεί το μήνυμα END.



Εικόνα 52: Ρυθμίσεις BlueSmirf

Μετά την ολοκλήρωση των ρυθμίσεων και αφού το Mindwave Mobile είναι σε λειτουργία, θα πρέπει το LED με την ένδειξη **Connect** του Bluesmirf να γίνει σταθερά πράσινο. Αν συνεχίζει να αναβοσβήνει με κόκκινο χρώμα, τότε σημαίνει ότι πρέπει να επαναληφθούν οι παραπάνω ρυθμίσεις.

3.3 Υλοποίηση αλγορίθμου για τον έλεγχο του αμαξιδίου

Έχοντας πραγματοποιήσει τα προηγούμενα δύο βήματα, ακολουθεί η δημιουργία του κατάλληλου αλγορίθμου βάσει του οποίου θα γίνεται ο έλεγχος λειτουργίας του αμαξιδίου. Τα βασικά βήματα που ακολουθεί ο αλγόριθμος είναι:

- Ανάγνωση των δεδομένων
- Έλεγχος των bytes συγχρονισμού του ληφθέντος πακέτου
- Έλεγχος των bytes DataPayload που περιέχουν την πληροφορία
- Απεικόνιση – Οπτικοποίηση Εγκεφαλικών ρυθμών

- Ανίχνευση βλεφαρισμών
- Αναγνώριση βλεφαρισμών – Κίνηση Αμαξιδίου
- Αποφυγή εμποδίων

Ο πλήρης αλγόριθμος βρίσκεται στο [παράρτημα Α](#).

3.3.1 Αρχικές ρυθμίσεις και δηλώσεις μεταβλητών

Για τη δημιουργία αλγορίθμου ο οποίος θα ελέγχει τον Arduino μέσω του Mindwave Mobile, έγινε χρήση του δείγματος κώδικα που παρέχει η Neurosky στον ιστότοπό της και συγκεκριμένα σε αυτή τη διεύθυνση (http://developer.neurosky.com/docs/lib/exe/fetch.php?media=mindwave_mobile_arduino_sample_code.docx). Το κομμάτι που χρησιμοποιήθηκε ήταν εκείνο για τη δήλωση των μεταβλητών, την αρχικοποίησή τους στη συνάρτηση void Setup καθώς και η ανάγνωση μέσω της σειριακής θύρας των bytes που έρχονται από το Bluesmirf και κατ' επέκταση από το Mindwave Mobile.

Συγκεκριμένα, έγινε δήλωση των παρακάτω μεταβλητών:

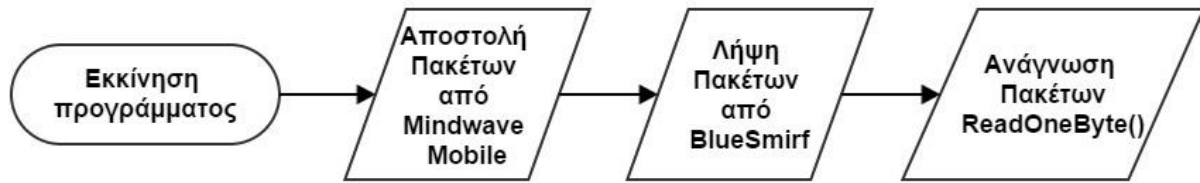
- **payloadLength** που δείχνει το μήκος των δεδομένων πληροφορίας που θα χρησιμοποιηθούν και δηλώνεται ως τύπου int λόγω του ότι μπορεί να λάβει τιμές, η αποθήκευση των οποίων μπορεί να χρειαστεί μνήμη μεγέθους μέχρι 4 bytes.
- **payloadChecksum** που αποτελεί το συμπλήρωμα ως προς 1 του υπολογισμένου αθροίσματος των bytes που είναι αποθηκευμένα στον πίνακα payloadData. Η μεταβλητή αυτή είναι τύπου byte αφού οι τιμές που λαμβάνει είναι από 0 έως 255.
- **checksum** το οποίο αντιστοιχεί στο τελευταίο byte του πακέτου που λαμβάνεται και δείχνει ποια πρέπει να είναι η τιμή του payloadChecksum για να είναι έγκυρο ένα πακέτο. Είναι τύπου byte γιατί η τιμή του μπορεί να είναι από 0 έως 255.
- **payloadData [64]** ο οποίος αποθηκεύει τα 64 bytes / sec (ή 512 bits / sec) που έρχονται από το Mindwave Mobile και αρχικοποιείται με μηδενικές τιμές.
- **poorQuality, attention, meditation** που δείχνουν την ποιότητα του σήματος και τα επίπεδα προσοχής και συγκέντρωσης του χρήστη. Δηλώνονται ως τύπου byte αφού η ποιότητα του σήματος παίρνει τιμές από 0 (άριστη ποιότητα) έως 255 (απουσία σήματος) και τα επίπεδα προσοχής και συγκέντρωσης παίρνουν τιμές 0 έως 100.

```
// Checksum variables
byte payloadChecksum = 0;
byte checksum = 0;
int payloadLength = 0;
byte payloadData[64] = { //512 bits/second or 64bytes/second streaming
from Mindwave Mobile stored in this matrix
0};
byte poorQuality = 0;
byte attention = 0;
byte meditation = 0;
```

3.3.2 Ανάγνωση δεδομένων

Η ανάγνωση των δεδομένων που στέλνει το Mindwave Mobile γίνεται μέσω μίας συνάρτησης με όνομα **ReadOneByte()** η οποία επιστρέφει τύπο byte και κάνει χρήση της εντολής Read() της κλάσης Serial του Arduino. Η εντολή Serial.Read() διαβάζει συνεχώς τα bytes που λαμβάνονται μέσω του Bluesmirf στον ακροδέκτη 0 (RX). Αφού ξεκινήσει η λήψη των δεδομένων, επιστρέφεται η μεταβλητή ByteRead που αντιστοιχεί στους

χαρακτήρες που έρχονται μέσω της σειριακής θύρας και ο καθένας από τους οποίους αποθηκεύεται στο σειριακό buffer.

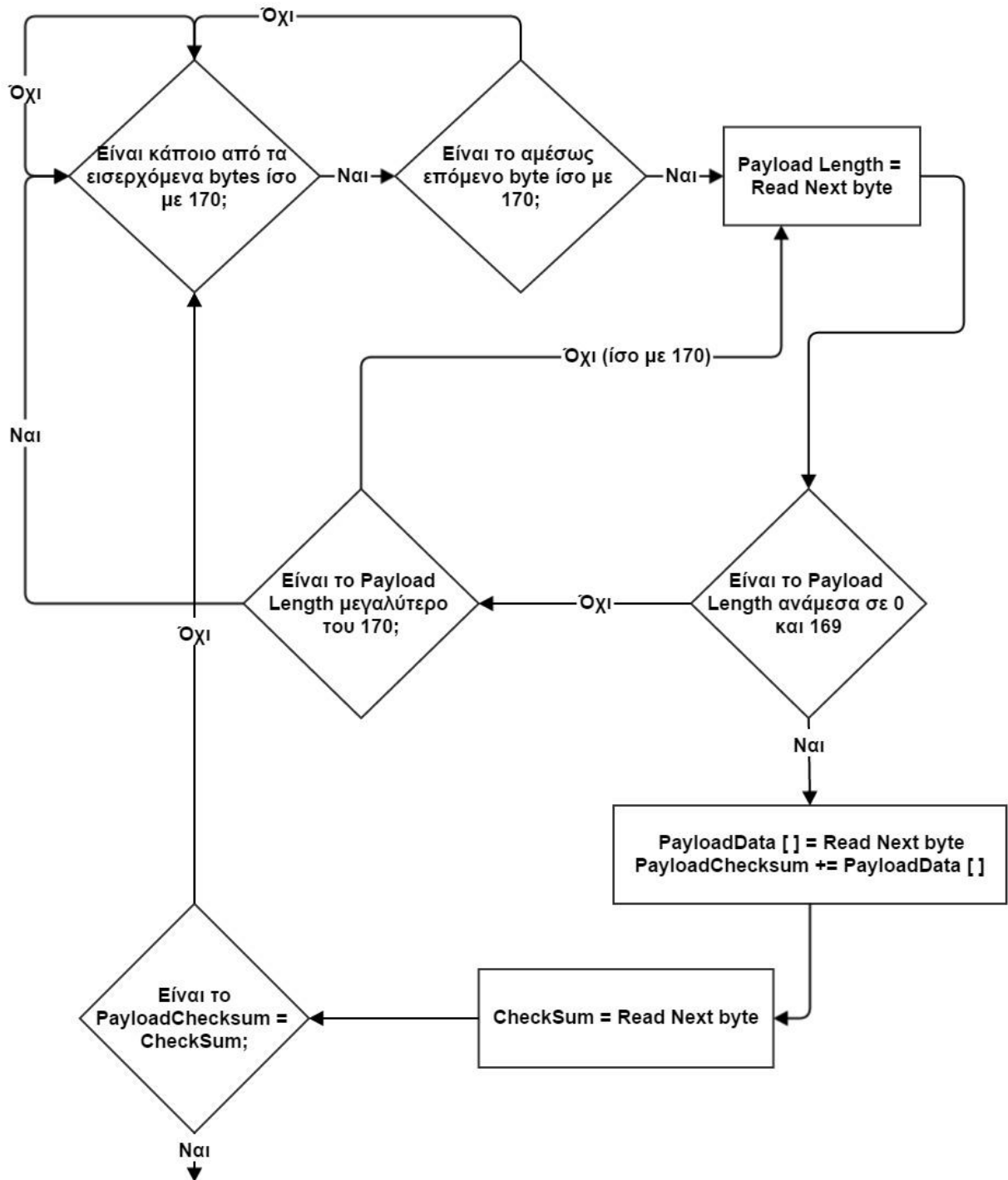


Σχήμα 6: Ανάγνωση λήψης δεδομένων από σειριακή θύρα

```
// Reading Bytes from Mindwave Mobile Device
byte ReadOneByte() {
  int ByteRead;
  // Wait until there is data
  while(!Serial.available());
  // Get the number of bytes (characters) available for reading from
  the serial port.
  // This is data that's already arrived and stored in the serial
  receive buffer (which holds 64 bytes)
  ByteRead = Serial.read();
  return ByteRead; // Read incoming serial data
}
```

3.3.3 Έλεγχος αρχικών bytes

Μέσα στη συνάρτηση void loop() αυτό που αρχικά γίνεται είναι ο έλεγχος των bytes που έρχονται από τη σειριακή θύρα του Arduino. Όπως προαναφέρθηκε στην παράγραφο 2.4, πρέπει να γίνει έλεγχος για τα bytes συγχρονισμού, τα bytes payloadLength και payloadData[64] καθώς επίσης και για το byte checksum. Παρακάτω φαίνεται το τμήμα εκείνο που κάνει αυτό τον έλεγχο.



Σχήμα 7: Έλεγχος bytes κεφαλίδας πακέτου

```

void loop() {
    // Look for sync bytes
    // Byte order: 0xAA, 0xAA, payloadLength, payloadData,
    // Checksum (sum all the bytes of payload, take lowest 8 bits, then
    // bit inverse on lowest

    if(ReadOneByte() == 0xAA) { // Read first sync byte and check
    if it's 0xAA
  
```

```
    if(ReadOneByte() == 0xAA) { // Read second sync byte and check
if it's also 0xAA in order not to make mistake
        payloadLength = ReadOneByte();
        if(payloadLength > 169){ // Payload length can not be
greater than 169
            return;
        }

        payloadChecksum = 0;
        for(int i = 0; i < payloadLength; i++) { // Loop until
payload length is complete
            payloadData[i] = ReadOneByte(); // Read payload
            payloadChecksum += payloadData[i];
        }

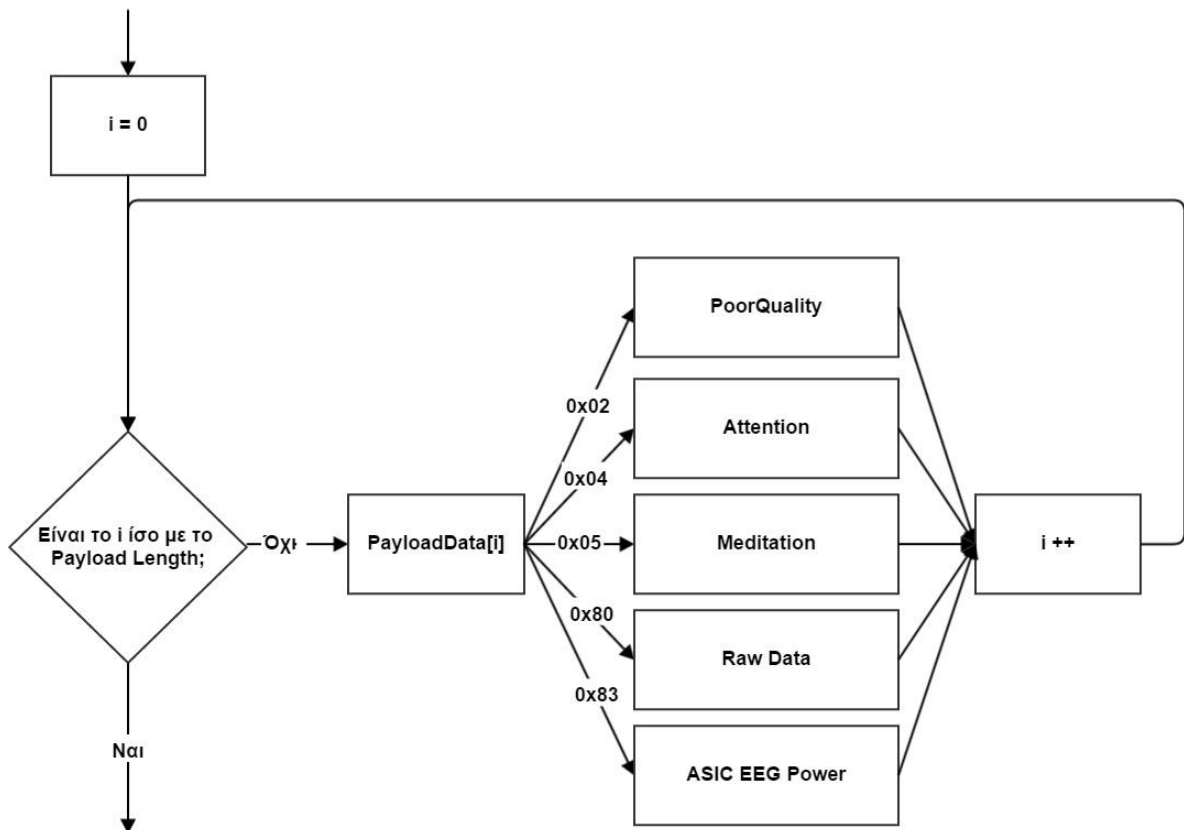
        checksum = ReadOneByte(); // Read checksum
byte from stream
        payloadChecksum = 255 - payloadChecksum; // Take one's
compliment of generated checksum

        if(checksum == payloadChecksum) {
            poorQuality = 200;
            attention = 0;
            meditation = 0;
        }
    }
```

3.3.4 Έλεγχος δεδομένων DataPayload

Αφού γίνει ο έλεγχος των bytes συγχρονισμού αλλά και των υπολοίπων που αποτελούν την κεφαλίδα του πακέτου, ο αλγόριθμος στη συνέχεια προχωράει στον έλεγχο των bytes που έχουν αποθηκευτεί στον πίνακα payloadData[64] και που αποτελούν την ουσία των δεδομένων μας αφού περιέχουν τις πληροφορίες για το τι αντιπροσωπεύει κάθε byte.

Με τη χρήση μίας δομής επανάληψης και συγκεκριμένα μέσω μίας εντολής for, γίνεται έλεγχος από το πρώτο στοιχείο του πίνακα payloadData[64] μέχρι και εκείνο που αντιστοιχεί στο μήκος payloadLength το οποίο είχε έρθει στο προηγούμενο βήμα, τι είδους πληροφορία περιέχεται στο πακέτο. Για κάθε ένα από τα bytes του πίνακα payloadData[64], γίνεται χρήση μίας δομής ελέγχου switch που δίνει την πληροφορία σχετικά με το αν το συγκεκριμένο byte του πίνακα αναφέρεται στην ποιότητα του σήματος, το επίπεδο προσοχής του χρήστη, το επίπεδο συγκέντρωσης του χρήστη, τα ακατέργαστα δεδομένα ή το ASIC EEG Power που περιέχει τους εγκεφαλικούς ρυθμούς.



Σχήμα 8: Έλεγχος bytes Payload Data

```

for(int i = 0; i < payloadLength; i++) { // Parse the payload
    switch (payloadData[i]) {
        case 02:
            i++;
            poorQuality = payloadData[i];
            break;
        case 04:
            i++;
            attention = payloadData[i];
            break;
        case 05:
            i++;
            meditation = payloadData[i];
            break;
        case 0x80: // Raw data
            i++;
            raw = (payloadData[i+1] << 8) | payloadData[i+2];
            //Serial.println(raw); //Uncomment to see the
values of raw data on Serial Monitor
            i += 2;
            break;
        case 0x83: // ASIC EEG POWER INT
            i++;
            byte vlen = payloadData[i];
            read_waves(i+1);
            i += vlen;
    }
}
    
```

```

        break;
    }
} // End of switch
// End of for loop

```

Κάθε φορά που ελέγχεται ένα από αυτά τα bytes, ανάλογα με το σε ποια κατηγορία ανήκει, γίνεται πρώτα αύξηση κατά μία θέση τον πίνακα payloadData και μετά γίνεται ανάθεση της τιμής στην αντίστοιχη μεταβλητή.

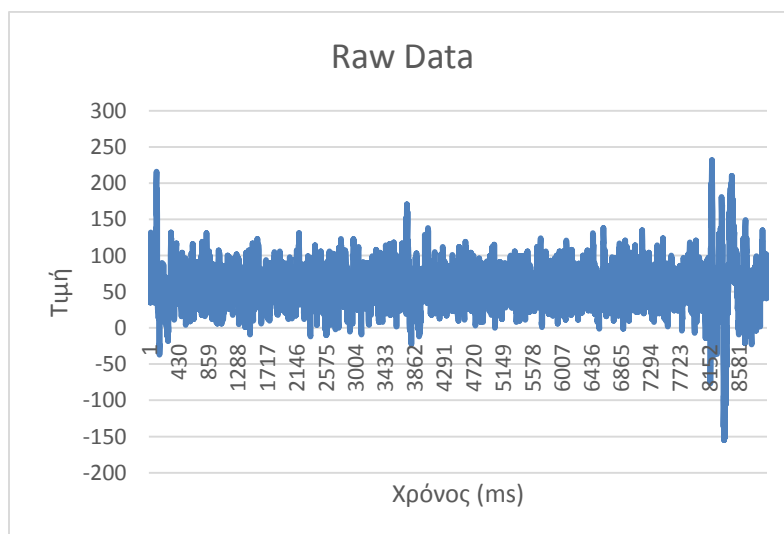
Στην περίπτωση που η τιμή ενός στοιχείου του πίνακα payloadData είναι ίση με 0x80 αυτό σημαίνει ότι έχουν ληφθεί bytes που σχετίζονται με τα ακατέργαστα δεδομένα (raw data). Στον παραπάνω τμήμα του αλγορίθμου, και πιο συγκεκριμένα στο case 0x80, όταν κάποιο byte πάρει την τιμή αυτή, τότε αμέσως γίνεται αύξηση του μετρητή i στον πίνακα όπου εκεί υπάρχει η πληροφορία για το πόσα bytes ακολουθούν. Επειδή τα ακατέργαστα δεδομένα αποτελούνται από δύο bytes η τιμή του payloadData[i] θα είναι 2. Για την εξαγωγή όμως της τιμής που λαμβάνουν τα ακατέργαστα δεδομένα, πρέπει σύμφωνα με το πρωτόκολλο λειτουργίας του Mindwave Mobile, να γίνει αριστερό bit shifting κατά 8 bits στο αμέσως επόμενο byte (High Order Byte) που ακολουθεί το byte που μας δίνει το πλήθος των bytes των ακατέργαστων δεδομένων. Στο αποτέλεσμα που προκύπτει κάνουμε bitwise OR με το αμέσως επόμενο byte (Low Order Byte). Ο πίνακας 3 παρουσιάζει την ανάλυση των ακατέργαστων δεδομένων.

Πίνακας 3: Ανάλυση ακατέργαστων δεδομένων (raw data)

payloadData[i]	i++	i++	i++	raw
0x80	payloadData [i] = Πλήθος bytes που ακολουθούν (είναι πάντα 2)	payloadData[i] = High Order Byte	payloadData[i] = Low Order Byte	((High Order Byte << 8) (Low Order Byte))

Σε περίπτωση που γίνει εκτύπωση των ακατέργαστων δεδομένων (Serial.println(raw);) στη σειριακή θύρα του Arduino με μετέπειτα χρήση του Matlab ή του Microsoft Office Excel για τη γραφική απεικόνιση των τιμών αυτών, τότε θα λάβουμε το παρακάτω γράφημα.

Πίνακας 4: Γράφημα ακατέργαστων δεδομένων



Πίνακας 5: Ανάλυση σήματος ASIC EEG Power

payloadData [i] = 0x83 _{HEX} ή 131 _{DEC}							
++i	131	++i	24 (Αριθμός bytes που ακολουθούν)	++i	Delta 1	++i	Delta 2
++i	Delta 3	++i	Theta 1	++i	Theta 2	++i	Theta 3
++i	Low Aplha 1	++i	Low Aplha 2	++i	Low Aplha 3	++i	High Aplha 1
++i	High Aplha 2	++i	High Aplha 3	++i	Low Beta 1	++i	Low Beta 2
++i	Low Beta 3	++i	High Beta 1	++i	High Beta 2	++i	High Beta 3
++i	Low Gamma 1	++i	Low Gamma 2	++i	Low Gamma 3	++i	High Gamma 1
++i	High Gamma 2	++i	High Gamma 3				

Στον παραπάνω πίνακα, αφού ελεγχθεί το byte [CODE] του εκάστοτε payloadData[i] και βρεθεί ότι είναι ίσο με 0x83, το αμέσως επόμενο byte αναφέρεται στο [VLENGTH] δηλαδή το μήκος των bytes που ακολουθούν. Στο συγκεκριμένο παράδειγμα είναι ίσο με 24 διότι το byte που αναφέρεται στην δεκαεξαδική τιμή 0x83, μας δίνει το σήμα ASIC EEG Power. Το σήμα αυτό έχει 24 bytes όπου κάθε ρυθμός καταλαμβάνει 3 από αυτά. Συνεπώς, η τιμή του [VLENGTH] θα αποθηκευτεί στο αμέσως επόμενο byte και για το λόγο αυτό αυξάνεται ο μετρητής κατά μία θέση ++i. Στα αμέσως επόμενα 3 bytes καταχωρείται η τιμή του ρυθμού Δέλτα και το ίδιο γίνεται για όλους τους επόμενους ρυθμούς με ανάλογη πρώτα αύξηση του μετρητή. Έτσι, μετά από 8 ρυθμούς των 3 bytes και 2 bytes από τις αρχικές τιμές το συνολικό πλήθος των bytes που λαμβάνει το σήμα αυτό είναι 26.

Για την ανάλυση των εγκεφαλικών ρυθμών και των τιμών που έχουν αυτοί, χρησιμοποιούνται οι ακόλουθες δύο συναρτήσεις.

```
// Function for each brain rhythm
void read_waves(int i) {
    deltaWave = bitShiftingBrainWaves(i);
    i+=3;
    thetaWave = bitShiftingBrainWaves(i);
    i+=3;
    lowAlphaWave = bitShiftingBrainWaves(i);
    i+=3;
    highAlphaWave = bitShiftingBrainWaves(i);
    i+=3;
}
```

```

    lowBetaWave = bitShiftingBrainWaves(i);
    i+=3;
    highBetaWave = bitShiftingBrainWaves(i);
    i+=3;
    lowGammaWave = bitShiftingBrainWaves(i);
    i+=3;
    midGammaWave = bitShiftingBrainWaves(i);
}

// Bit Shifting brain waves
int bitShiftingBrainWaves(int i) {
/* Each rhythm consists of three bytes. In order to take its value we
have to left bit shift all of them according to Mindset's
Communication Protocol */
    return ((payloadData[i] << 16) + (payloadData[i+1] << 8) +
payloadData[i+2]);
}

```

Η συνάρτηση `read_waves(int i)` καλείται μόνο στην περίπτωση που υπάρχει το σήμα ASIC EEG Power και παίρνει σαν όρισμα την επόμενη τιμή ($i + 1$) του μετρητή που δείχνει σε ποια θέση του πίνακα `payloadData [i]` βρίσκεται ο αλγόριθμος. Αυτό που κάνει είναι για οκτώ διαφορετικές μεταβλητές που έχουν οριστεί στην αρχή του προγράμματος ως unsigned integers, να καλεί με τη σειρά της τη συνάρτηση `bitShiftingBrainWaves(int i)`. Η συνάρτηση αυτή κάνει bit shifting για καθένα από τα τρία bytes που αποτελούν κάθε ρυθμό. Η χρήση του bit shifting είναι απαραίτητη σύμφωνα με το Mindset Communication Protocol της Neurosky για την εξαγωγή των τιμών κάθε ρυθμού. Στο πρώτο byte (`payloadData [i]`) γίνεται bit shifting κατά 16 bits αριστερά, στο δεύτερο (`payloadData [i+1]`) κατά 8 bits αριστερά ενώ το τρίτο (`payloadData [i+2]`) παραμένει ως έχει. Εν συνεχεία γίνεται το άθροισμα των αριθμών αυτών το οποίο επιστρέφεται στην καλούσα συνάρτηση `read_waves(int i)` η οποία με τη σειρά της το αποθηκεύει στη μεταβλητή εκείνη που χαρακτηρίζει τον εκάστοτε εγκεφαλικό ρυθμό.

3.3.5 Απεικόνιση – Οπτικοποίηση εγκεφαλικών ρυθμών

Αφού υπολογιστούν οι τιμές όλων των εγκεφαλικών ρυθμών καθώς επίσης και των επιπέδων προσοχής και συγκέντρωσης, μπορούμε να δούμε τις τιμές αυτές στη σειριακή οθόνη του Arduino (Εικόνα 53).

```

////////////////////////////////////
//////////////////////////////////// RHYTHMS VISUALIZATION //////////////////////////////////
////////////////////////////////////

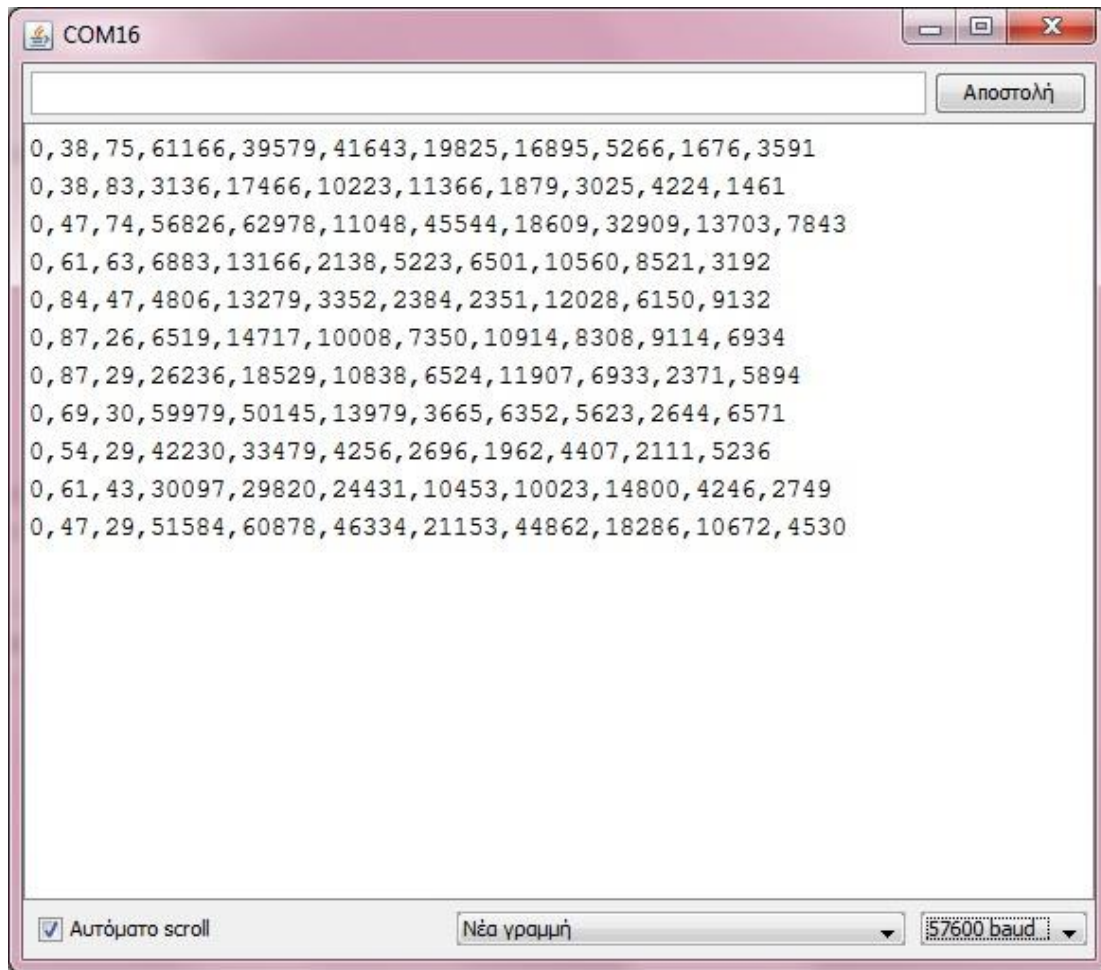
```

```

if(brainwave && attention > 0 && attention < 100) {
//Serial.print("Signal Quality value is: ");
Serial.print(poorQuality, DEC);
Serial.print(',');
//Serial.print("Attention value is: ");
Serial.print(attention, DEC);
Serial.print(',');
//Serial.print("Meditation value is: ");
Serial.print(meditation, DEC);
Serial.print(',');
}

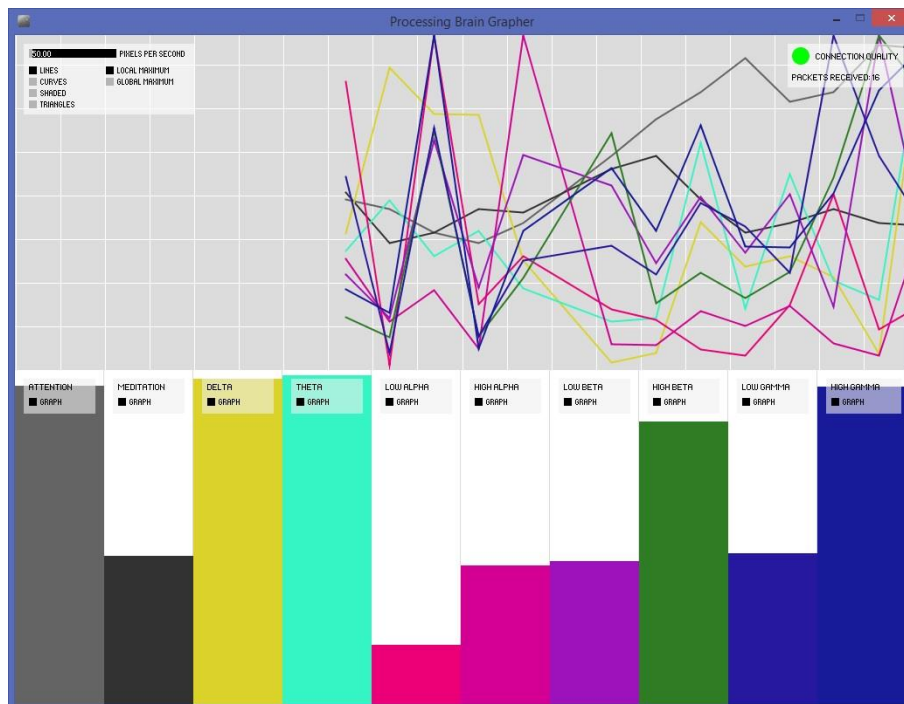
```

```
//Serial.print("Delta value is: ");
Serial.print(delta_wave, DEC);
Serial.print(',');
//Serial.print("Theta value is: ");
Serial.print(theta_wave, DEC);
Serial.print(',');
//Serial.print("Low Alpha value is: ");
Serial.print(low_alpha_wave, DEC);
Serial.print(',');
//Serial.print("High Alpha value is: ");
Serial.print(high_alpha_wave, DEC);
Serial.print(',');
//Serial.print("Low Beta is: ");
Serial.print(low_beta_wave, DEC);
Serial.print(',');
//Serial.print("High Beta is: ");
Serial.print(high_beta_wave, DEC);
Serial.print(',');
//Serial.print("Low Gamma Value is: ");
Serial.print(low_gamma_wave, DEC);
Serial.print(',');
//Serial.print("Mid Gamma Value is: ");
Serial.print(mid_gamma_wave, DEC);
Serial.println()
}
```

Εικόνα 53: Σειριακή Έξοδος Arduino

αλλά και να τις απεικονίσουμε γραφικά μέσω του γραφικού προγραμματιστικού περιβάλλοντος Processing (Εικόνα 54) και συγκεκριμένα κάνοντας χρήση της βιβλιοθήκης Brain Grapher που δημιούργησε ο Eric Mika [55] .



Εικόνα 54: Γραφική απεικόνιση εγκεφαλικών ρυθμών μέσω Processing

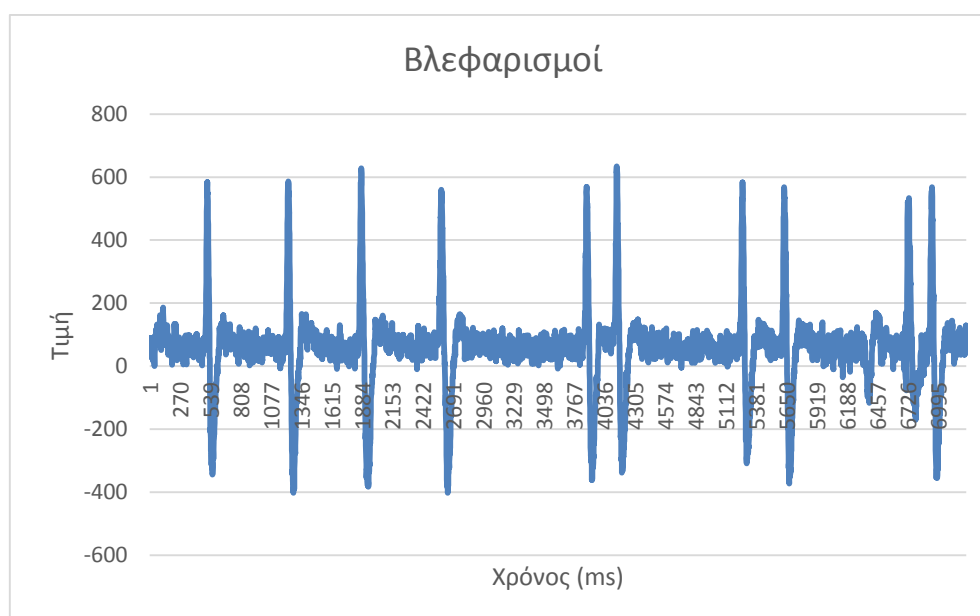
3.3.6 Συνάρτηση ανίχνευσης βλεφαρισμού ματιών

Στην εργασία αυτή, ο έλεγχος της κίνησης του αμαξιδίου έγινε με χρήση του βλεφαρισμού των ματιών. Επειδή η εταιρεία Neugosky είχε κλειδωμένη τη λειτουργία, με την οποία μπορούσε να γίνει ανίχνευση της ισχύος του σήματος του βλεφαρισμού μέσα από τα Data Values και συγκεκριμένα όταν το πεδίο [CODE] είχε την δεκαεξαδική τιμή 0x16, έπρεπε να επινοηθεί κάποιος τρόπος με τον οποίο θα γινόταν ανίχνευση των βλεφαρισμών καθώς επίσης και πως θα γινόταν η απομόνωσή τους.

Αυτό το οποίο πραγματοποιήθηκε, ήταν να γίνει λήψη των ακατέργαστων δεδομένων και μετά από αρκετές δοκιμές παρατηρήθηκε ότι οι βλεφαρισμοί συμβαίνουν για τιμές πάνω από 400, τη στιγμή που οι φυσιολογικές τιμές των ακατέργαστων δεδομένων εκτείνονται από -100 μέχρι + 150. Με τον τρόπο αυτό επιτεύχθηκε η απομόνωση εκείνων των τιμών των ακατέργαστων δεδομένων που αντιστοιχούσαν στους βλεφαρισμούς. Ένα πρόβλημα όμως που εμφανίστηκε αργότερα ήταν ότι γύρω από τον κύριο λοβό του σήματος εμφανίζονταν μικρότεροι πλευρικοί λοβοί με τιμή από 320 έως 370 και σε πολύ κοντινή απόσταση από τον κύριο λοβό, της τάξης περίπου των 40 ms. Το αποτέλεσμα ήταν να μετριούνται και αυτοί ως ξεχωριστοί βλεφαρισμοί ενώ στην πραγματικότητα ήταν παράσιτα.

Στο ακόλουθο γράφημα εμφανίζονται τα ακατέργαστα δεδομένα όπου οι κορυφές (peaks) υποδηλώνουν την ύπαρξη κάποιου βλεφαρισμού.

Πίνακας 6: Βλεφαρισμοί



Παρακάτω φαίνεται ο αλγόριθμος υλοποίησης της συνάρτησης που ανιχνεύει τον βλεφαρισμό των ματιών από τα ακατέργαστα δεδομένα. Συγκεκριμένα, στη συνάρτηση αυτή συγκρίνεται η τρέχουσα τιμή (current value) των ακατέργαστων δεδομένων με την ακριβώς προηγούμενη (previous value) και παίρνουμε την τρέχουσα (difference2) και την προηγούμενη (difference1) διαφορά τους. Αν η τρέχουσα διαφορά είναι αρνητική και η προηγούμενη διαφορά είναι θετική αυτό σημαίνει ότι υπάρχει αλλαγή της κλίσης, κάτι που σημαίνει ότι βρέθηκε τοπικό μέγιστο και κατά συνέπεια ανιχνεύθηκε ένας πλευρικός λοβός γύρω από τον κύριο. Έτσι θα πρέπει να γίνει μέτρηση των χρονικών διαφορών τους και αν είναι κάτω από 40 ms τότε θεωρείται ότι αποτελούν μέρος του κύριου λοβού και λαμβάνονται σαν ένας. Η τεχνική αυτή ονομάζεται **Debouncing**.

```

////////////////////////////////////
////////////////////          EYE BLINK FUNCTION          //////////////////////
////////////////////////////////////
boolean check_eye_blink(){

    previous_value = current_value;
    current_value = raw;
    difference1 = difference2;
    difference2 = current_value - previous_value;

    if (raw >= threshold && difference1 > 0 && difference2 < 0){ //
Local Maxima detected

        last_time = current_time;
        current_time = millis();
        time_difference = current_time - last_time;

        if (time_difference >= 40){                //Debouncing
            return true;
        }
        else{
            return false;
        }
    }
}
}

```

3.3.7 Αλγόριθμος αναγνώρισης βλεφαρισμών. Κίνηση κινητήρων - αμαξιδίου

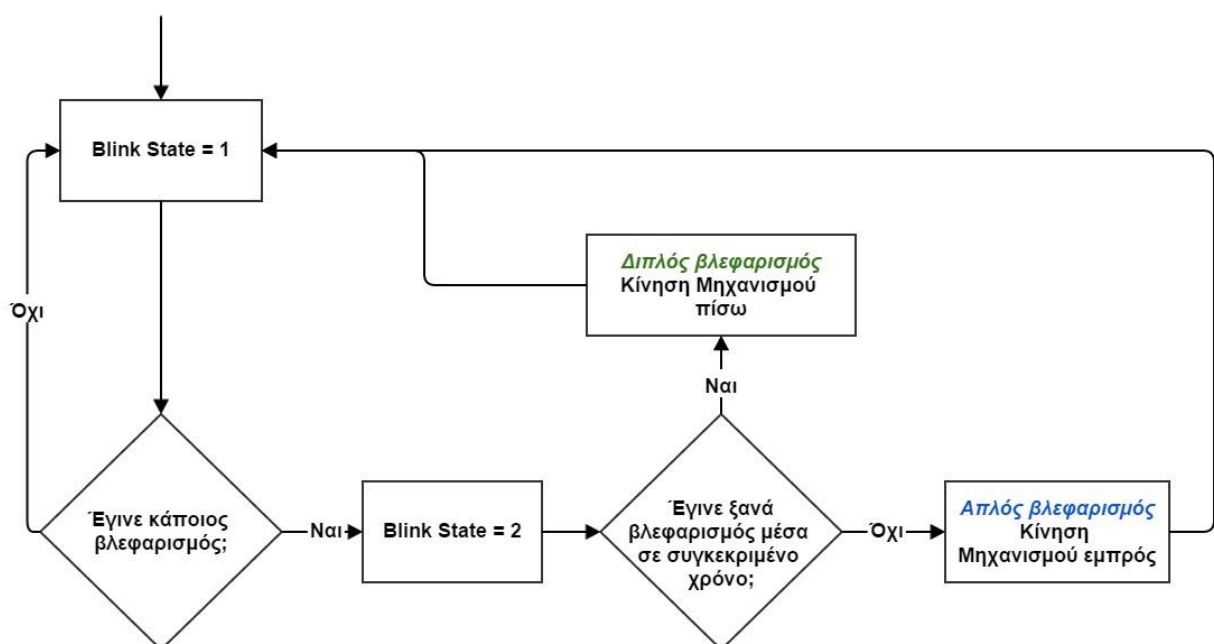
Από τη στιγμή που έγινε η ανίχνευση των βλεφαρισμών, το επόμενο βήμα ήταν να γίνει ο έλεγχος της κίνησης των κινητήρων ανάλογα με το αν κάθε βλεφαρισμός είναι απλός ή διπλός. Η κατηγοριοποίηση αυτή έγινε με κριτήριο ότι ένας βλεφαρισμός χαρακτηρίζεται ως **απλός (Single)** όταν η χρονική του απόσταση από το ακριβώς προηγούμενο είναι μεγαλύτερη των 400 ms, ενώ χαρακτηρίζεται ως **διπλός (Double)** όταν είναι μικρότερη από 400 ms. Το χρονικό αυτό παράθυρο λήφθηκε αυθαίρετα, βασισμένο στις μετρήσεις και στις δοκιμές που έγιναν στο εργαστήριο σε ένα μόνο χρήστη.

Ο αλγόριθμος που υλοποιεί την παραπάνω λογική (Σχήμα 9) είναι ο εξής: Αρχικά μέσω μιας δομής switch, ελέγχεται αν έχει συμβεί κάποιος βλεφαρισμός και όταν αυτό συμβεί, γίνεται λήψη σε μία μεταβλητή **lastBlinkTime** η χρονική στιγμή που αυτό συνέβη. Το προγραμματιστικό περιβάλλον του Arduino παρέχει τη δυνατότητα αυτή μέσω της συνάρτησης **millis()** που καταγράφει τον αριθμό των χιλιοστών του δευτερολέπτου που έχουν περάσει από τη στιγμή που ξεκίνησε να εκτελείται το πρόγραμμα. Αφού γίνει αυτό, ο αλγόριθμος μεταβαίνει σε μια δεύτερη κατάσταση στην οποία γίνεται επανέλεγχος για βλεφαρισμό. Όταν αυτό συμβεί, καταγράφεται εκ νέου η χρονική στιγμή που αυτό έγινε και υπολογίζεται η χρονική διαφορά του με την προηγούμενη καταγραφή μας. Αν η διαφορά αυτή είναι μεγαλύτερη των 400 ms, τότε ο βλεφαρισμός που έγινε χαρακτηρίζεται ως απλό και το αμαξίδιο κινείται εμπρός. Σε αντίθετη περίπτωση ο βλεφαρισμός χαρακτηρίζεται ως διπλό και το αμαξίδιο κινείται προς τα πίσω για 0,5 δευτερόλεπτο.

Αξίζει να σημειωθεί ότι κατά τη διάρκεια των δοκιμών, εμφανίστηκε ένα πρόβλημα το οποίο είχε να κάνει με την τυχαία εισαγωγή μια χρονικής καθυστέρησης χάρη στην οποία λαμβάνονταν υπόψη ένας τρίτος βλεφαρισμός κατά την επαναφορά της βλεφαρίδας από

την θέση του κλειστού ματιού στη θέση ανοικτού ματιού. Το αποτέλεσμα είναι ότι αυτός ο «επιπλέον» τρίτος βλεφαρισμός, καταγράφεται σαν απλός, αμέσως μετά την καταγραφή ενός διπλού βλεφαρισμού που είχε πράγματι συμβεί από το χρήστη. Έτσι το αμαξίδιο εκτελούσε κίνηση προς τα εμπρός, αμέσως μετά την κίνηση προς τα πίσω, κάτι το οποίο ήταν ανεπιθύμητο. Για την αντιμετώπιση του προβλήματος αυτού χρησιμοποιήθηκε ένα flag, έτσι ώστε όταν ο αλγόριθμος έμπαινε στην κατάσταση κατά την οποία είχε ανιχνευθεί ήδη ένας διπλός βλεφαρισμός, το flag αυτό γινόταν true. Στην περίπτωση λοιπόν που ανιχνευόταν ένας απλός βλεφαρισμός, γινόταν πρώτα έλεγχος για το αν αυτό το flag ήταν true ή false, αν είχε δηλαδή προηγηθεί ή όχι ένας διπλός βλεφαρισμός. Αν ήταν true που σημαίνει ότι έχει ήδη προηγηθεί ένας διπλός βλεφαρισμός, τότε ο απλός αυτός βλεφαρισμός που είχε ανιχνευθεί ήταν παράσιτο και ο αλγόριθμος μετέβαινε ξανά στην αρχική κατάσταση για τον εκ νέου έλεγχο για βλεφαρισμό.

Για κάθε απλό βλεφαρισμό που καταγράφεται, γίνεται απαρίθμηση αυτών και ανάλογα με το πόσοι απλοί βλεφαρισμοί έχουν καταγραφεί, το αμαξίδιο προσαρμόζει την ταχύτητά του αυξάνοντάς την. Πιο αναλυτικά, έχει χρησιμοποιηθεί μία μεταβλητή με όνομα **blinkcount** η οποία για κάθε βλεφαρισμό που συμβαίνει αυξάνεται κατά ένα κάθε φορά. Επειδή το αμαξίδιο σχεδιάστηκε έτσι ώστε να έχει τη δυνατότητα να κινείται με τρεις διαφορετικές ταχύτητες, χρησιμοποιήθηκε μία δομή switch η οποία έχει τρεις περιπτώσεις, μία για κάθε ταχύτητα. Μέσα σε κάθε case, καλείται η συνάρτηση εμπρόσθιας κίνησης του αμαξιδίου, `drive_forward()`, η οποία χρησιμοποιεί αναλογική έξοδο αντί για ψηφιακή. Ο λόγος που συμβαίνει κάτι τέτοιο είναι γιατί σύμφωνα με το σχεδιασμό του συστήματος, υπάρχουν διαφορετικές ταχύτητες στους κινητήρες και όχι μόνο δύο (μέγιστη κι ελάχιστη, ON και OFF) όπως στην ψηφιακή. Κάτι τέτοιο επιτυγχάνεται χρησιμοποιώντας την τεχνική **Διαμόρφωσης Εύρους Παλμών** (Pulse Width Modulation, P.W.M). Με την PWM μπορεί να χρησιμοποιηθεί μία οποιαδήποτε τιμή από 0-255 ως όρισμα στη συνάρτηση `analogWrite()` του Arduino. Έτσι μπορούν να επιτευχθούν διαφορετικές ταχύτητες στους κινητήρες μέσω διαφορετικών τιμών τάσεων που δίνονται στους ακροδέκτες που χαρακτηρίζονται ως PWM, εν αντιθέσει με την `digitalWrite`, που κάνει χρήση όλου του ποσού της τάσης των 5 V για τη θέση ON και 0V για τη θέση OFF.



Σχήμα 9: Αλγόριθμος κατηγοριοποίησης βλεφαρισμών

```

////////////////////////////////////
//                               EYE BLINK DETECTION                               //
////////////////////////////////////

```

```

switch (blink_state){
case 1:

    if (check_eye_blink() == true){
        lastBlinkTime = millis();
        blink_state = 2;
    }
    break;

case 2:

    if (millis() - lastBlinkTime >= 400){ // Timeout has passed
without another blink so the previous blink was a single blink

        if (doubleBlink == true){ // Checking for triple
blink (extra single blink when already a double blink has taken
place). If true then just go back
            doubleBlink = false; // to state 1 and
ignore extra single blink.
            blink_state = 1;
        }
        else{
            //Serial.println("Single Blink");
            blinkcount++;
            switch (blinkcount){
            case 1:
                //Serial.println("1st gear"); // 1st gear
                drive_forward(220);
                break;
            case 2:
                //Serial.println("2nd gear"); // 2nd gear
                drive_forward(238);
                break;
            case 3:
                //Serial.println("3rd gear"); // 3rd gear
                drive_forward(255);
                blinkcount = 0; // Reset blinkcount back
to 0
            }
            break;
        }
        blink_state = 1;
    }
}
else {
    if (check_eye_blink() == true){ // A second blink
before the timeout constitutes a double blink
        //Serial.println("Double Blink");

```

```
        doubleBlink = true;
        drive_backward(); // Move Backward for
1 second
        delay(1000);
        motor_stop(); // Stop Motor
        blink_state = 1;
    }
}
break;
} // End of switch loop
} //End of if (checksum)
} //End of "Read second sync byte" loop
} //End of "Read first sync byte" loop
} //End of loop function
```

3.3.8 Αλγόριθμοι κίνησης κινητήρων

Για την κίνηση των κινητήρων, έγινε ανάπτυξη πέντε διαφορετικών συναρτήσεων, μία για κάθε κίνηση προς τις τέσσερις κατευθύνσεις και μία για το σταμάτημα των κινητήρων.

Όλες οι συναρτήσεις για την κίνηση των κινητήρων κάνουν χρήση τεσσάρων μεταβλητών που έχουν αρχικά δηλωθεί ως δύο πίνακες δύο θέσεων. Οι μεταβλητές αυτές είναι οι `int motor_left[] = {10, 11}` και `int motor_right[] = {6, 9}`. Τα νούμερα στις αγκύλες αντιπροσωπεύουν οι ακροδέκτες του Arduino στα οποία συνδέονται οι ακροδέκτες του chip L293D. Αναλυτικότερα, η μεταβλητή `motor_right [6]` αντιπροσωπεύει την κίνηση του δεξιού κινητήρα εμπρός, η μεταβλητή `motor_right [9]` αντιπροσωπεύει την κίνηση του δεξιού κινητήρα πίσω, η μεταβλητή `motor_left [10]` αντιπροσωπεύει την κίνηση του αριστερού κινητήρα εμπρός και η μεταβλητή `motor_left [11]` αντιπροσωπεύει την κίνηση του αριστερού κινητήρα πίσω. Και οι τέσσερις αυτές μεταβλητές αρχικοποιούνται στη συνάρτηση `void setup()` ως έξοδοι με τη χρήση της εντολής `pinMode`.

Για την κάθε κίνηση ξεχωριστά, η υλοποίηση της κάθε συνάρτησης είναι η ακόλουθη:

- Για την κίνηση του αμαξιδίου προς τα εμπρός πρέπει και στον δεξιό και στον αριστερό κινητήρα να ενεργοποιηθούν οι μπροστινοί ακροδέκτες. Οι ακροδέκτες αυτοί ενεργοποιούνται με χρήση της `analogWrite()` γιατί γίνεται χρήση της εξόδου PWM για την επίτευξη διαφορετικών ταχυτήτων. Αυτό υλοποιείται με τη συνάρτηση

```
void drive_forward(int x){
    analogWrite(motor_left[0], x); // Analog output for different speed
    analogWrite(motor_right[0], x);
}
```

- Για την κίνηση του αμαξιδίου προς τα πίσω πρέπει και στον δεξιό και στον αριστερό κινητήρα να ενεργοποιηθούν οι ακροδέκτες που αντιστοιχούν στην κίνηση προς τα πίσω και να τεθούν σε κατάσταση HIGH, ενώ οι ακροδέκτες για την κίνηση προς τα εμπρός πρέπει να τεθούν σε κατάσταση LOW. Αυτό γίνεται με τη συνάρτηση

```
void drive_backward(){
    digitalWrite(motor_left[0], LOW);
    digitalWrite(motor_left[1], HIGH);
```

```
digitalWrite(motor_right[0], LOW);  
digitalWrite(motor_right[1], HIGH);  
}
```

- Για την κίνηση του αμαξιδίου προς τα αριστερά πρέπει στον δεξιό κινητήρα ο ακροδέκτης που αντιστοιχεί στην κίνηση προς τα εμπρός να τεθεί στην κατάσταση HIGH και ο ακροδέκτης που αντιστοιχεί στην κίνηση προς τα πίσω να τεθεί στην κατάσταση LOW, ενώ στον αριστερό κινητήρα ο ακροδέκτης που αντιστοιχεί στην κίνηση προς τα πίσω να τεθεί στην κατάσταση LOW και ο ακροδέκτης που αντιστοιχεί στην κίνηση προς τα εμπρός να τεθεί στην κατάσταση HIGH. Αυτό γίνεται με τη συνάρτηση:

```
void turn_left(){  
digitalWrite(motor_left[0], LOW);  
digitalWrite(motor_left[1], HIGH);  
  
digitalWrite(motor_right[0], HIGH);  
digitalWrite(motor_right[1], LOW);  
}
```

- Για την κίνηση του αμαξιδίου προς τα δεξιά πρέπει στον δεξιό κινητήρα ο ακροδέκτης που αντιστοιχεί στην κίνηση προς τα εμπρός να τεθεί στην κατάσταση LOW και ο ακροδέκτης που αντιστοιχεί στην κίνηση προς τα πίσω να τεθεί στην κατάσταση HIGH, ενώ στον αριστερό κινητήρα ο ακροδέκτης που αντιστοιχεί στην κίνηση προς τα πίσω να τεθεί στην κατάσταση HIGH και ο ακροδέκτης που αντιστοιχεί στην κίνηση προς τα εμπρός να τεθεί στην κατάσταση LOW. Αυτό γίνεται με τη συνάρτηση

```
void turn_right(){  
digitalWrite(motor_left[0], HIGH);  
digitalWrite(motor_left[1], LOW);  
  
digitalWrite(motor_right[0], LOW);  
digitalWrite(motor_right[1], HIGH);  
}
```

- Τέλος, για να σταματήσει το αμαξίδιο να κινείται όλοι οι ακροδέκτες πρέπει να τεθούν σε κατάσταση LOW. Αυτό γίνεται με τη συνάρτηση

```
void motor_stop(){  
digitalWrite(motor_left[0], LOW);  
digitalWrite(motor_left[1], LOW);  
  
digitalWrite(motor_right[0], LOW);  
digitalWrite(motor_right[1], LOW);  
}
```

3.3.9 Αλγόριθμος αποφυγής εμποδίων με χρήση αισθητήρα υπερήχων

Η υλοποίηση αλγορίθμου αποφυγής εμποδίων με χρήση του αισθητήρα HC-SR04 γίνεται συνήθως με χρήση της ενσωματωμένης συνάρτησης `pulseIn` του Arduino, η οποία παίρνει σαν ορίσματα τον ακροδέκτη που μας ενδιαφέρει και την κατάσταση στην οποία τίθεται αυτό αρχικά. Όταν ο ακροδέκτης αυτός φτάσει στην επιθυμητή κατάσταση θέλουμε (στην παρούσα εργασία HIGH), αμέσως η `pulseIn` ξεκινάει τη χρονομέτρηση μέχρι ο ακροδέκτης αυτός να βγει εκτός της τρέχουσας κατάστασης (κατάσταση LOW), οπότε και σταματάει η χρονομέτρηση. Αυτό που επιστρέφει η συνάρτηση είναι ο χρόνος σε μS .

Ένα κλασσικό παράδειγμα μέτρησης της απόστασης ενός εμποδίου από τον αισθητήρα είναι το ακόλουθο.

```
long obstacleDistance(){ // Create triggering pulse
  long duration;
  digitalWrite(trigger, LOW); // *****
  delayMicroseconds(2); // *
  digitalWrite(trigger, HIGH); // *
  delayMicroseconds(10); // *
  digitalWrite(trigger, LOW); // *
  delayMicroseconds(2); //*****

  duration = pulseIn(echo, HIGH); // The speed of sound is 340
  // m/s or 29 microseconds per centimeter.
  return duration / 29 / 2; // The ping travels out and
  // back, so to find the distance of the object we take half of the
  // distance travelled.
}
```

Στην παρούσα όμως εργασία, ο παραπάνω τρόπος αποδείχτηκε μη κατάλληλος διότι η συνάρτηση `pulseIn` εισάγει μικρές χρονικές καθυστερήσεις, οι οποίες δημιουργούν προβλήματα στην καταγραφή της σωστής χρονομέτρησης των βλεφαρισμών που συμβαίνουν.

Για το λόγο αυτό έγινε χρήση της βιβλιοθήκης *PulseInZero* που δημιούργησε ο Mike Almond (<https://github.com/mikedotalmond>). Ο λόγος που χρησιμοποιήθηκε η παραπάνω βιβλιοθήκη είναι ότι γίνεται χρήση της λειτουργίας **Διακοπών** (Interrupts) του Arduino. Τα interrupts προσφέρουν μεγάλη ευχέρεια στο να ελέγχονται πολλά πράγματα ταυτόχρονα χωρίς να εισάγονται μη αναγκαίες χρονικές καθυστερήσεις και να σταματάει τη λειτουργία του ο μικροεπεξεργαστής. Η βασική αρχή λειτουργίας των interrupts είναι ότι η ροή του κύριου προγράμματος εκτελείται κανονικά και όταν ικανοποιηθεί μία συνθήκη, γίνεται διακοπή για να εκτελεστεί κάποιο άλλο τμήμα του προγράμματος. Όταν τελειώσει το τμήμα αυτό, η ροή του προγράμματος ξεκινάει πάλι από εκεί που είχε σταματήσει.

Αυτό το οποίο επιτεύχθηκε με τη χρήση της συγκεκριμένης βιβλιοθήκης, είναι να εκτελείται το κυρίως πρόγραμμα κανονικά και να γίνεται έλεγχος για το πότε συμβαίνουν βλεφαρισμοί. Η συνθήκη της οποίας ο έλεγχος πρέπει να γίνεται ταυτόχρονα από τον μικροεπεξεργαστή, σχετίζεται με τον έλεγχο της απόστασης του αμαξιδίου από πιθανά εμπόδια που θα βρεθούν εντός της ακτίνας του αισθητήρα. Σε αυτή την περίπτωση, πρέπει να γίνει **διακοπή** του κυρίως προγράμματος και να εκτελεστεί το τμήμα εκείνο που βρίσκεται σε κάποια συνάρτηση που καλείται από τη διακοπή που έχουμε θέσει. Στην τρέχουσα εργασία, γινόταν έλεγχος της απόστασης του πιθανού εμποδίου και αν αυτή ήταν κάτω από 20 εκατοστά, τότε το αμαξίδιο σταματούσε να κινείται.

Για το λόγο αυτό κατά την εκκίνηση της συνάρτησης `loop()` του Arduino καταγράφεται η χρονική διαφορά της τρέχουσας από την προηγούμενη επανάληψη της `loop()` και στην περίπτωση που αυτή είναι μεγαλύτερη από 10ms, εκτελείται η συνάρτηση `ring()`. Η συνάρτηση αυτή αποστέλλει μέσω του ακροδέκτη διέγερσης (`trigger`) του αισθητήρα HC-SR04, έναν παλμό διάρκειας 10ms περιμένοντας για απάντηση από τον ακροδέκτη ηχούς (`echo`) του αισθητήρα. Η υλοποίηση της παραπάνω λογικής φαίνεται στο παρακάτω τμήμα κώδικα.

```
void loop() {
```

```
unsigned long time = millis();
unsigned long dt = time - lastTime;
lastTime = time;
pingTimer += dt; //Time difference between two successive pulses
if(pingTimer > pingDelay){
    pingTimer = 0;
    ping();
}
```

```
void ping(){
    digitalWrite(SIGNAL_PIN, HIGH); //Trigger the outward ping pulse(s)
    delayMicroseconds(10);
    digitalWrite(SIGNAL_PIN, LOW);
    PulseInZero::begin(); // Start listening out for the echo pulse on
interrupt 0
}
```

Κατά την αναμονή για λήψη απάντησης εκτελείται η συνάρτηση **begin()** της βιβλιοθήκης PulseInZero. Η συνάρτηση αυτή κάνει χρήση ενός interrupt στον ακροδέκτη 0 του Arduino το οποίο καλεί μία άλλη συνάρτηση **pinChange()** στην οποία γίνεται έλεγχος κάθε φορά που γίνεται αλλαγή του παλμού από αύξουσα (RISING) σε φθίνουσα (FALLING) κατάσταση. Όταν ολοκληρωθεί ο παλμός, λαμβάνεται η χρονική διάρκειά του σε μS .

```
void PulseInZero::begin(){
    state      = false;
    active     = true;
    attachInterrupt(0, pinChange, RISING);
}
```

```
void PulseInZero::pinChange(){ //Called whenever the interrupt pin
state changes
    static unsigned long pulseStart = 0;
    state = !state;

    if(state){ // Interrupt pin has changed, a pulse has started
        pulseStart = micros(); // Store the current microseconds
        attachInterrupt(0, pinChange, FALLING); // Now wait for the
falling edge
    }
    else { // Pulse complete, so we can detach the interrupt

        detachInterrupt(0);

        // Pin state changed again - pulse ended
        unsigned long duration = micros() - pulseStart; // Get the
pulse length
        active = false;
        onComplete(duration);
    }
}
```


Η χρονική διαφορά που διάνυσε ο παλμός από τη στιγμή που ξεκίνησε μέχρι την επιστροφή του, υπολογίζεται με βάση την ταχύτητα του ήχου. Ως γνωστόν, η ταχύτητα του ήχου είναι ίση με 343 m/s ή 1m ανά 0,0029s ή 1mm ανά 2,9 μS = 1 μS ανά 0,344mm σε θερμοκρασία 20 °C. Επειδή ο χρόνος αυτός περιλαμβάνει τη διαδρομή του παλμού τόσο μέχρι την ανίχνευση του εμποδίου όσο και την επιστροφή του, η απόσταση λαμβάνεται υπολογισμένη στο μισό του χρόνου, δηλαδή 1 μS ανά 0,172 mm ή 1 μS ανά 0,0172 cm.

Η συνάρτηση **pingPulseComplete** (unsigned long duration) παίρνει σαν όρισμα την τιμή της χρονικής απόστασης που διάνυσε ο παλμός σε μS και η οποία έχει υπολογιστεί στη συνάρτηση **pinChange()**. Στην εργασία αυτή, θέσαμε το ασφαλές όριο απόστασης που θα πρέπει να κρατάει το αμαξίδιο από ένα εμπόδιο στα 20cm. Με βάση ότι σε 1 μS το αμαξίδιο διανύει απόσταση 0,0172 cm, τότε η απόσταση των 20cm μεταφράζεται σε 1162 μS , περίπου 1200 μS . Έτσι, στη συνάρτηση pingPulseComplete, αν έχουμε ανίχνευση εμποδίου σε χρονική απόσταση μικρότερη των 1200 μS , το αμαξίδιο σταματάει την κίνησή του και περιμένει τον επόμενο βλεφαρισμό για να ξεκινήσει πάλι.

```
/**
 * Pulse complete callback handler for PulseInZero
 * @param duration - pulse length in microseconds
 */
void pingPulseComplete(unsigned long duration) {

    if (duration <= 1200){
        motor_stop();
        obstacle_detected = true;
    }
    else{
        obstacle_detected = false;
    }
}
```

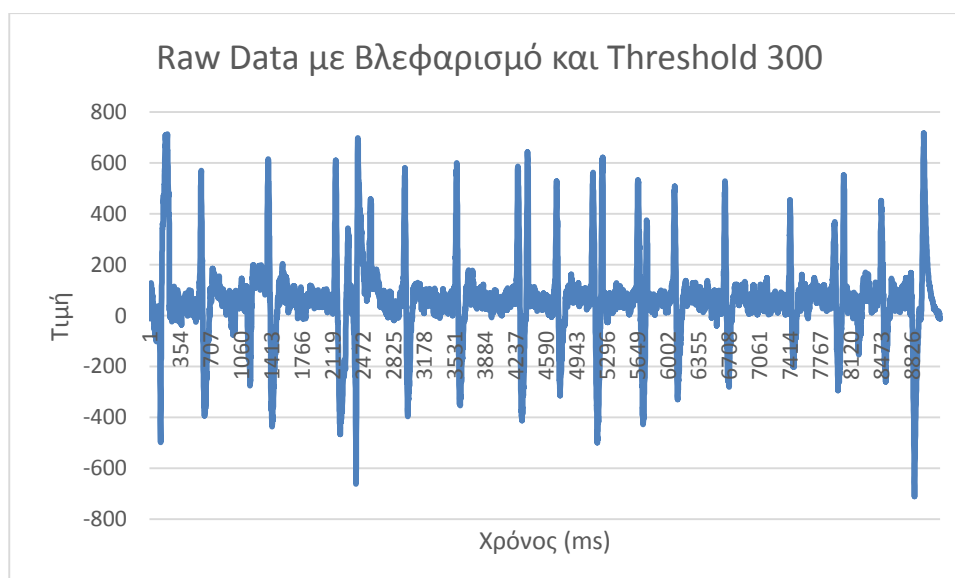
4. ΣΥΜΠΕΡΑΣΜΑΤΑ

Η παρούσα εργασία σκοπό είχε να αναδείξει τη δυνατότητα σχεδιασμού, υλοποίησης και ελέγχου ενός ρομποτικού μηχανισμού μέσω εγκεφαλικών σημάτων. Λαμβάνοντας υπόψη ότι τα βασικά χαρακτηριστικά ενός ρομπότ είναι η αυτονομία και η αποφυγή εμποδίων, κατασκευάστηκε ένα αμαξίδιο, το οποίο τροφοδοτείται από ένα σετ τεσσάρων μπαταριών, λαμβάνει σήμα μέσω πρωτοκόλλου Bluetooth από μία συσκευή καταγραφής ηλεκτροεγκεφαλογραφήματος, γίνεται ο έλεγχος μέσω ενός μικροεπεξεργαστή και όταν βρεθεί εμπόδιο σταματάει να κινείται.

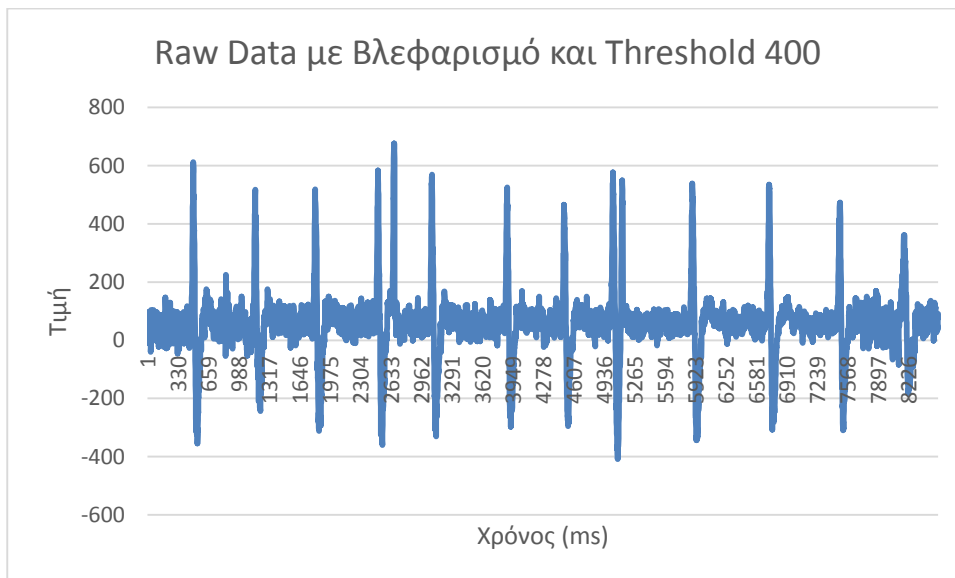
Με την ολοκλήρωσή της, προέκυψαν τα ακόλουθα χρήσιμα συμπεράσματα:

1. Με τις υπάρχουσες τεχνολογίες, είναι σχετικά εύκολο να υλοποιηθούν συστήματα και εφαρμογές χαμηλού κόστους, τα οποία θα ελέγχονται εν μέρει ή εξ' ολοκλήρου, από εγκεφαλικά σήματα. Τα συστήματα αυτά θα εξυπηρετούν τις βασικές ανάγκες κίνησης και προσβασιμότητας σε υπηρεσίες για άτομα που αντιμετωπίζουν κάποια δυσλειτουργία στον εγκέφαλο ή εν γένει στο νευρικό σύστημα, η οποία τους εμποδίζει να δραστηριοποιηθούν φυσιολογικά.
2. Προς το παρόν, το επιστημονικό ενδιαφέρον έχει επικεντρωθεί στην αντιμετώπιση προβλημάτων που σχετίζονται με τη σωματική κίνηση κυρίως παραπληγικών ανθρώπων. Έρευνες όμως δείχνουν ότι μπορούν να σχεδιαστούν συστήματα που μπορούν να δώσουν λύση σε άτομα με προβλήματα όρασης και ακοής.
3. Η κίνηση του αμαξιδίου γίνεται με βάση το βλεφαρισμό των ματιών. Η ανίχνευση του βλεφαρισμού στηρίζεται στα raw data που λαμβάνει ο Arduino και πάνω στα οποία γίνεται η επεξεργασία. Οι τιμές των raw data εκτείνονται από -150 έως περίπου 200, όπως φαίνεται στον πίνακα 4. Κατά το βλεφαρισμό όμως, οι τιμές αυτές φτάνουν μέχρι την τιμή 600 (βλ. πίνακα 6). Το κατώφλι που χρησιμοποιήθηκε για το φιλτράρισμα των raw data τέθηκε μετά από αρκετές δοκιμές στην τιμή 420. Στη συνέχεια παρουσιάζονται οι γραφικές παραστάσεις των τιμών των raw data συναρτήσει του χρόνου για διαφορετικά κατώφλια κάθε φορά και το αντίστοιχο ποσοστό επιτυχίας ανίχνευσης βλεφαρισμών.

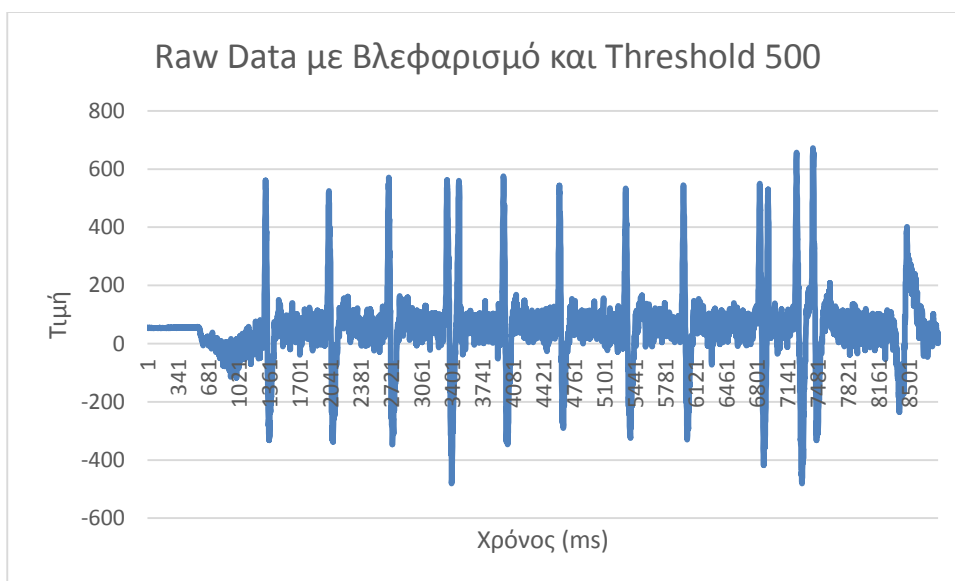
Στην παρακάτω εικόνα βλέπουμε τα raw data για κατώφλι στα 300 με ποσοστό ανίχνευσης 2/22 βλεφαρισμούς, εκ των οποίων όμως οι 3/9 είναι λανθασμένοι. Ενώ θα έπρεπε να ανιχνευτούν ως διπλοί βλεφαρισμοί, είτε δεν ανιχνεύτηκαν καθόλου, είτε ανιχνεύτηκαν ως απλοί, κάτι που οδηγεί σε λάθος κίνηση.



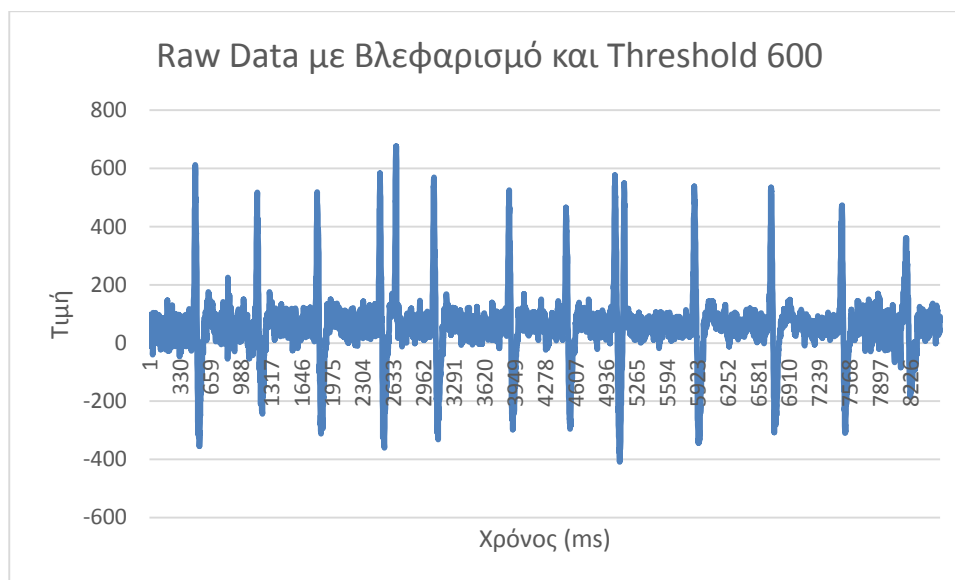
Στην ακόλουθη εικόνα βλέπουμε τα ακατέργαστα δεδομένα για κατώφλι στα 400 με ποσοστό ανίχνευσης 13/14 βλεφαρισμούς, εκ των οποίων μόνο ο 1/10 δεν ανιχνεύεται καθόλου. Το συγκεκριμένο κατώφλι είναι και αυτό που δίνει και τα καλύτερα αποτελέσματα.



Στην παρακάτω εικόνα βλέπουμε τα ακατέργαστα δεδομένα για κατώφλι στα 500 με ποσοστό ανίχνευσης 7/14 βλεφαρισμούς. Επίσης, με το συγκεκριμένο κατώφλι, πολλοί από τους διπλούς βλεφαρισμούς ανιχνεύονταν ως απλοί, με αποτέλεσμα να οδηγούμαστε σε λάθος κίνηση του αμαξιδίου



Στην τελευταία δοκιμή που πραγματοποιήσαμε φαίνονται τα ακατέργαστα δεδομένα για κατώφλι στα 600 με ποσοστό ανίχνευσης 2/14 βλεφαρισμούς, που μας οδήγησε αμέσως στην απόρριψη του συγκεκριμένου κατωφλίου.



Από τα παραπάνω γίνεται αντιληπτό ότι το κατώφλι εκείνο που μας δίνει τα καλύτερα δυνατά αποτελέσματα είναι στην τιμή 400 με μία απόκλιση της τάξης του $\pm 5\%$ από 380 έως 420. Για τιμές έξω από αυτό το εύρος, είτε θα έχουμε λανθασμένη ανίχνευση βλεφαρισμών, δηλαδή οι διπλοί θα ανιχνεύονται ως απλοί και το αντίστροφο, είτε δεν θα ανιχνεύονται αρκετοί, είτε ακόμη μπορεί να ανιχνεύονται κάποιοι χωρίς να έχουν συμβεί στην πραγματικότητα.

4. Η ευαισθησία του συγκεκριμένου συστήματος είναι ακόμα ένας αρκετά σημαντικός παράγοντας στον οποίο δόθηκε εξ' αρχής μεγάλη προσοχή. Ο αλγόριθμος είναι σε θέση να ανιχνεύει τους 9/10 βλεφαρισμούς με αποτέλεσμα η ευαισθησία να ανέρχεται στο 90%. Σημείο κλειδί για το υψηλό ποσοστό ευαισθησίας, ήταν η τεχνική του Debouncing (βλ. παρ. 3.3.6), χάρη στην οποία μπορούσαμε να χαρακτηρίσουμε μικρότερους πλευρικούς λοβούς γύρω από τον κύριο που συνέβαιναν σε χρονική απόσταση 40μs, ως ένα βλεφαρισμό. Χωρίς την εφαρμογή της τεχνικής αυτής, το σύστημα ανίχνευε τους πλευρικούς αυτούς λοβούς σαν κύριους και είχε μια εντελώς τυχαία συμπεριφορά, αφού οι λοβοί αυτοί αποτελούν στην ουσία θόρυβο που έπρεπε να αφαιρεθεί με κάποιο τρόπο.

5. Η εμπρόσθια κίνηση του αμαξιδίου βασίζεται στην τεχνική PWM για να μπορέσουμε να επιτύχουμε διαφορετικές ταχύτητες. Οι τιμές που μπορούν να χρησιμοποιηθούν στον Arduino για να δηλώσουμε διαφορετικές τιμές τάσεων εξόδου, και στην ουσία διαφορετικές ταχύτητες, έχουν εύρος 0 – 255. Λόγω του βάρους όμως του αμαξιδίου, τιμές μικρότερες του 220 αδυνατούν να το κινήσουν ή το κινούν με μεγάλη δυσκολία.

5. ΜΕΛΛΟΝΤΙΚΕΣ ΕΠΕΚΤΑΣΕΙΣ

Η εργασία πραγματοποιήθηκε πάνω σε ένα αρκετά νέο και ταχύτατα εξελισσόμενο τομέα, αυτόν του Brain Computer Interface, με συνέπεια τη δημιουργία ενός βασικού σκελετού πάνω στον οποίο μπορούν να προστεθούν μελλοντικά αρκετές νέες τροποποιήσεις .

Η σημαντικότερη από αυτές είναι η χρήση μίας διαφορετικής συσκευής καταγραφής ηλεκτροεγκεφαλογραφήματος, που θα προσφέρει μεγαλύτερη ακρίβεια στις μετρήσεις. Το Mindwave Mobile είναι μία εμπορική συσκευή χαμηλού κόστους, με συνέπεια η ακρίβεια των μετρήσεων και της καταγραφής των εγκεφαλικών ρυθμών να μην είναι η βέλτιστη. Αυτό κυρίως οφείλεται στο γεγονός ότι η συσκευή αυτή χρησιμοποιεί ένα μόνο ηλεκτρόδιο που λαμβάνει τις αναλογικές τιμές του ηλεκτροεγκεφαλογραφήματος από τον εγκέφαλο και εν συνεχεία τις μετατρέπει σε ψηφιακές μέσω ενός ολοκληρωμένου κυκλώματος. Το σημείο πάνω στο κεφάλι από το οποίο λαμβάνεται σήμα είναι το FP1 σύμφωνα με το διεθνές πρότυπο 10-20, πάνω από το φρύδι του αριστερού ματιού. Είναι λοιπόν προφανές ότι όλη η πληροφορία που μπορούσαμε να έχουμε προς επεξεργασία προερχόταν από το 1 από τα 19 σημεία που ιδανικά θα είχαμε αν χρησιμοποιούσαμε μία κάσκα καλωδίων σύμφωνα με τα διεθνή πρότυπα τα οποία εφαρμόζονται σε ερευνητικά εργαστήρια.

Μία χρήσιμη και αναγκαία προσθήκη που θα μπορούσε να εμπλουτίσει σημαντικά το παρόν project είναι η καταγραφή μετρήσεων σε ένα ικανοποιητικό αριθμών χρηστών για τη δημιουργία μίας βάσης δεδομένων και τη μετέπειτα στατιστική ανάλυση για εξαγωγή καλύτερων αποτελεσμάτων. Η εργασία αυτή πραγματοποιήθηκε εξ 'ολοκλήρου κάνοντας χρήση των δεδομένων ενός μόνο χρήστη, του γράφοντα την εργασία, με συνέπεια να λαμβάνονται αυθαίρετα τιμές κατωφλίων και ορίων, π.χ. στο χρονικό όριο που διαχωρίζει τον απλό από τον διπλό βλεφαρισμό, με βάση το καταγραφόμενο ηλεκτροεγκεφαλογράφημα και τις μετρήσεις του χρήστη αυτού.

Θα μπορούσαν επίσης να χρησιμοποιηθούν περισσότεροι του ενός αισθητήρες ανίχνευσης εμποδίων, ένας για κάθε κατεύθυνση, έτσι ώστε να ανιχνεύονται εμπόδια που πιθανόν να βρίσκονται εκεί. Στο ίδιο ακόμα πλαίσιο, όταν γίνεται ανίχνευση κάποιου εμποδίου, το αμαξίδιο θα μπορούσε να κινείται μόνο του μέχρι να αποφύγει τελείως το εμπόδιο χωρίς να σταματάει και χωρίς να περιμένει τον επόμενο βλεφαρισμό από το χρήστη. Επίσης, εκτός από αισθητήρες μέτρησης της απόστασης, θα μπορούσαν να χρησιμοποιηθούν αισθητήρες μέτρησης θερμοκρασίας και υγρασίας που να προειδοποιούν εγκαίρως το μικροελεγκτή για επικίνδυνο περιβάλλον στο οποίο υπάρχει φωτιά ή νερό.

Η πραγματοποίηση της παρούσας εργασίας έγινε με λογισμικό ανοικτού κώδικα και συγκεκριμένα σε γλώσσα C, με αποτέλεσμα η δυνατότητα εύρεσης ακόμα και των πιο βασικών πληροφοριών, που ήταν απαραίτητες για την επικοινωνία με το Mindwave Mobile, να είναι περιορισμένη ή ελλιπής. Οι περισσότερες εντολές που ήταν διαθέσιμες στο Διαδίκτυο, ήταν γραμμένες είτε σε Java, είτε σε Python, κάτι που θα μπορούσε να δώσει ιδέες για ανάπτυξη απλούστερων και γρηγορότερων εφαρμογών. Οι γλώσσες αυτές, διαθέτουν ένα αρκετά μεγάλο εύρος βιβλιοθηκών, που δίνουν τη δυνατότητα στον προγραμματιστή να συνδυάσει γρήγορα και αξιόπιστα, διαφορετικές τεχνικές που δίνουν καλύτερα αποτελέσματα.

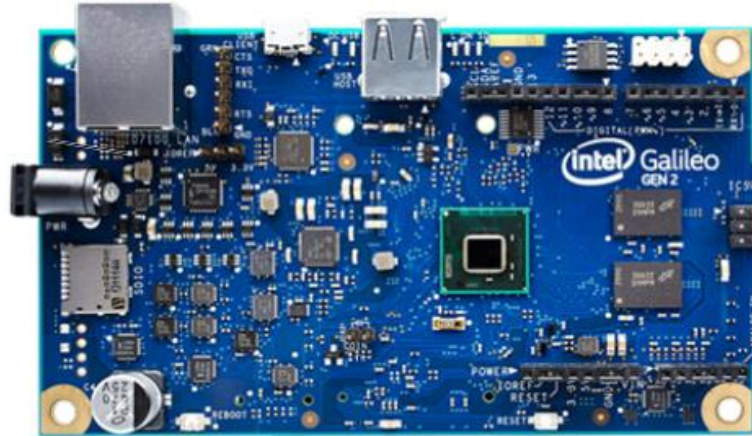
Τέλος, ο μικροελεγκτής που χρησιμοποιήθηκε στην εργασία είναι ο Arduino και συγκεκριμένα η πιο απλή έκδοση, Arduino Uno. Υπάρχει όμως μια πληθώρα επεξεργαστών που έχουν αρχίσει και κυκλοφορούν στην αγορά τα τελευταία χρόνια. Πολλοί από αυτούς προσφέρουν μεγαλύτερη ταχύτητα επεξεργασίας, θύρες USB, HDMI και Ethernet, ασύρματη επικοινωνία με πρωτόκολλα Wi-Fi και Bluetooth και λειτουργούν

Έλεγχος ρομποτικού μηχανισμού μέσω εγκεφαλικών σημάτων με χρήση πλατφόρμας ανοικτού υλικού

στα κυριότερα λειτουργικά συστήματα που χρησιμοποιούνται σήμερα παγκοσμίως. Οι κυριότεροι από αυτούς τους μικροελεγκτές είναι:

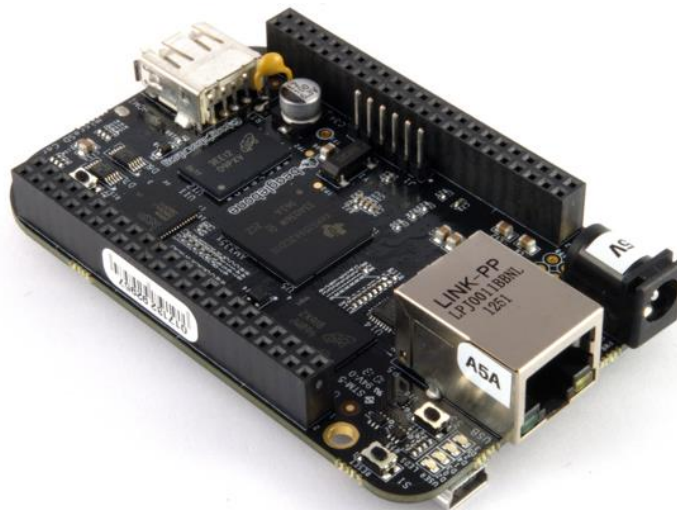
1. **Intel® Galileo Gen 2 Development Board**

(<http://www.intel.com/content/www/us/en/do-it-yourself/galileo-maker-quark-board.html>)



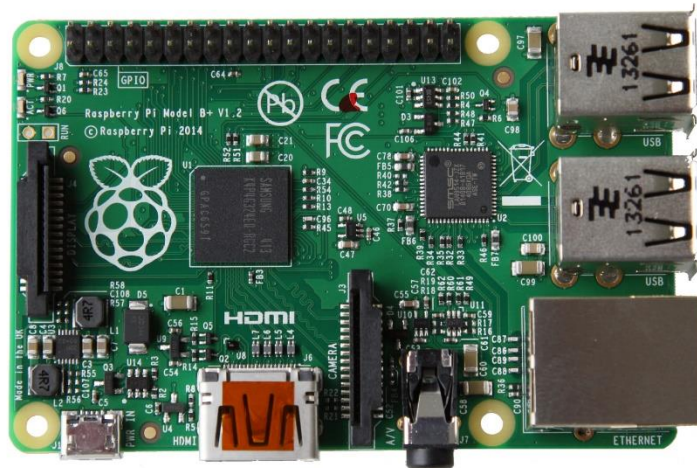
Εικόνα 55: Intel® Galileo Gen 2 Development Board [56]

2. **BeagleBone Black** (<http://beagleboard.org/BLACK>)



Εικόνα 56: Beagle Bone Black [57]

3. Raspberry Pi (<http://www.raspberrypi.org/>)



Εικόνα 57: Raspberry Pi [58]

4. Rabbit (<http://www.digi.com/products/rabbitprocessor/>)



Εικόνα 58: Rabbit [59]

ΠΑΡΑΡΤΗΜΑ Α

```
/*Created by Nikolaos Karagiannis and is License Free.
 I have used parts of code from Arduino Bluetooth Interface with
 Mindwave created by Sophi Kravitz (edit 11-4) and
 Shane Clements edit 11-5
 https://learn.sparkfun.com/tutorials/hackers-in-residence---hacking-
 mindwave-mobile/gathering-materials
 and part of code from Neurosky's Sample Code
 http://developer.neurosky.com/docs/doku.php?id=mindwave\_mobile\_and\_ard
 uino
 I have also used PulseInZero Library written from Mike Almond
 https://github.com/mikedotalmond */

#include <PulseInZero.h>

//UltraSensor
const int SIGNAL_PIN = 5; // Digital pin connected to the trigger port
on the module. The echo port to the pin for interrupt 0 (pin 2 on Uno)
unsigned long lastTime = 0;
int pingTimer = 0;
int pingDelay = 10; // Milliseconds between ping pulses
float millimetres = 0.0;
boolean obstacle_detected = false;

// DC Motors
int motor_left[] = { //PWM pins used for forward movement
  10, 11};
int motor_right[] = {
  6, 9};

// Checksum variables
byte payloadChecksum = 0;
byte checksum = 0;
int payloadLength = 0;
byte payloadData[64] = { //512 bits/second or 64bytes/second streaming
from Mindwave Mobile stored in this matrix
  0};
byte poorQuality = 0;
byte attention = 0;
byte meditation = 0;

// Raw and Eye blinks variables
int raw;
int threshold = 400; //Threshold for eye blinks
int previous_value;
int current_value;
int difference1;
int difference2;
int time_difference;
int current_time;
```

```
int last_time;
boolean blink_state = 1;
unsigned long lastBlinkTime;
boolean doubleBlink;
int blinkcount;

// Setting up system
void setup() {
  Serial.begin(57600);
  delay(50);
  previous_value = 0;
  current_value = 0;
  difference1 = 0;
  difference2 = 0;
  lastBlinkTime = 0;
  current_time = 0;
  doubleBlink = false;
  blinkcount = 0;
  // Setup motors
  int i;
  for(i = 0; i < 2; i++){
    pinMode(motor_left[i], OUTPUT);
    pinMode(motor_right[i], OUTPUT);
  }

  // Ultrasonic sensors
  pinMode(SIGNAL_PIN, OUTPUT);
  digitalWrite(SIGNAL_PIN, LOW);
  // Set up PulseInZero (pulseIn alternative using interrupt 0 - pin2
  on an Arduino Uno)
  // Use PulseInZero::begin() in place of pulseIn() and pulseComplete
  will fire when a pulse completes
  PulseInZero::setup(pingPulseComplete);
}

// Reading Bytes from Mindwave Mobile Device
byte ReadOneByte() {
  int ByteRead;
  // Wait until there is data
  while(!Serial.available());
  // Get the number of bytes (characters) available for reading from
  the serial port.
  // This is data that's already arrived and stored in the serial
  receive buffer (which holds 64 bytes)
  ByteRead = Serial.read();
  return ByteRead; // Read incoming serial data
}

// Initializing variables for each brain rhythm
unsigned int deltaWave = 0;
unsigned int thetaWave = 0;
unsigned int lowAlphaWave = 0;
```

```
unsigned int highAlphaWave = 0;
unsigned int lowBetaWave = 0;
unsigned int highBetaWave = 0;
unsigned int lowGammaWave = 0;
unsigned int midGammaWave = 0;

// Function for each brain rhythm
void read_waves(int i) {
    deltaWave = bitShiftingBrainWaves(i);
    i+=3;
    thetaWave = bitShiftingBrainWaves(i);
    i+=3;
    lowAlphaWave = bitShiftingBrainWaves(i);
    i+=3;
    highAlphaWave = bitShiftingBrainWaves(i);
    i+=3;
    lowBetaWave = bitShiftingBrainWaves(i);
    i+=3;
    highBetaWave = bitShiftingBrainWaves(i);
    i+=3;
    lowGammaWave = bitShiftingBrainWaves(i);
    i+=3;
    midGammaWave = bitShiftingBrainWaves(i);
}

// Bit Shifting brain waves
int bitShiftingBrainWaves(int i) {
    /* Each rhythm consists of three bytes. In order to take its value we
    have to left bit shift all of them according to Mindset's
    Communication Protocol */
    return ((payloadData[i] << 16) + (payloadData[i+1] << 8) +
    payloadData[i+2]);
}

////////////////////////////////////
//////////////////////////////////// MAIN LOOP //////////////////////////////////////
////////////////////////////////////

void loop() {
    // Begin sending trigger pulses
    unsigned long time = millis();
    unsigned long dt = time - lastTime;
    lastTime = time;
    pingTimer += dt; // Time difference between two successive pulses
    if(pingTimer > pingDelay){
        pingTimer = 0;
        ping();
    }

    // Look for sync bytes
```

```
// Byte order: 0xAA, 0xAA, payloadLength, payloadData,
// Checksum (sum all the bytes of payload, take lowest 8 bits, then
bit inverse on lowest

    if(ReadOneByte() == 0xAA) { // Read first sync byte and check
if it's 0xAA
        if(ReadOneByte() == 0xAA) { // Read second sync byte and check
if it's also 0xAA in order not to make mistake
            payloadLength = ReadOneByte();
            if(payloadLength > 169){ // Payload length can not be
greater than 169
                return;
            }

            payloadChecksum = 0;
            for(int i = 0; i < payloadLength; i++) { // Loop until
payload length is complete
                payloadData[i] = ReadOneByte(); // Read payload
                payloadChecksum += payloadData[i];
            }

            checksum = ReadOneByte(); // Read checksum
byte from stream
            payloadChecksum = 255 - payloadChecksum; // Take one's
compliment of generated checksum

            if(checksum == payloadChecksum) {
                poorQuality = 200;
                attention = 0;
                meditation = 0;

                for(int i = 0; i < payloadLength; i++) { // Parse the payload
                    switch (payloadData[i]) {
                        case 02:
                            i++;
                            poorQuality = payloadData[i];
                            break;
                        case 04:
                            i++;
                            attention = payloadData[i];
                            break;
                        case 05:
                            i++;
                            meditation = payloadData[i];
                            break;
                        case 0x80: // Raw data
                            i++;
                            raw = (payloadData[i+1] << 8) | payloadData[i+2];
                            //Serial.println(raw); //Uncomment to see the
values of raw data on Serial Monitor
                            i += 2;
                    }
                }
            }
        }
    }
}
```

```
        break;
    case 0x83:                                // ASIC EEG POWER INT
        i++;
        byte vlen = payloadData[i];
        read_waves(i+1);
        i += vlen;
        break;
    }                                          // End of switch
}                                             // End of for loop
```

```
////////////////////////////////////
////////////////////////////////////          RHYTHMS VISUALIZATION          //////////////////////////////////////
////////////////////////////////////
```

```
if(attention > 0 && attention < 100) {
    //Serial.print("Signal Quality value is: ");
    Serial.print(poorQuality, DEC);
    Serial.print(',');
    //Serial.print("Attention value is: ");
    Serial.print(attention, DEC);
    Serial.print(',');
    //Serial.print("Meditation value is: ");
    Serial.print(meditation, DEC);
    Serial.print(',');
    //Serial.print("Delta value is: ");
    Serial.print(deltaWave, DEC);
    Serial.print(',');
    //Serial.print("Theta value is: ");
    Serial.print(thetaWave, DEC);
    Serial.print(',');
    //Serial.print("Low Alpha value is: ");
    Serial.print(lowAlphaWave, DEC);
    Serial.print(',');
    //Serial.print("High Alpha value is: ");
    Serial.print(highAlphaWave, DEC);
    Serial.print(',');
    //Serial.print("Low Beta is: ");
    Serial.print(lowBetaWave, DEC);
    Serial.print(',');
    //Serial.print("High Beta is: ");
    Serial.print(highBetaWave, DEC);
    Serial.print(',');
    //Serial.print("Low Gamma Value is: ");
    Serial.print(lowGammaWave, DEC);
    Serial.print(',');
    //Serial.print("Mid Gamma Value is: ");
    Serial.print(midGammaWave, DEC);
    Serial.println();
}
```

```
//////////////////////////////////////
//////////////////////////////////////          EYE BLINK DETECTION          ////////////////////////////////////////
//////////////////////////////////////

switch (blink_state){
case 1:

    if (check_eye_blink() == true){
        lastBlinkTime = millis();
        blink_state = 2;
    }
    break;

case 2:

    if (millis() - lastBlinkTime >= 400){ // Timeout has passed
without another blink so the previous blink was a single blink

        if (doubleBlink == true){ // Checking for triple
blink (extra single blink when already a double blink has taken
place). If true then just go back
            doubleBlink = false; // to state 1 and
ignore extra single blink.
            blink_state = 1;
        }
        else{
            //Serial.println("Single Blink");
            blinkcount++;
            switch (blinkcount){
            case 1:
                //Serial.println("1st gear"); // 1st gear
                drive_forward(220);
                break;
            case 2:
                //Serial.println("2nd gear"); // 2nd gear
                drive_forward(238);
                break;
            case 3:
                //Serial.println("3rd gear"); // 3rd gear
                drive_forward(255);
                blinkcount = 0; // Reset blinkcount back
to 0
            }
            break;
        }
        blink_state = 1;
    }
}
else {
    if (check_eye_blink() == true){ // A second blink
before the timeout constitutes a double blink
```

```

        //Serial.println("Double Blink");
        doubleBlink = true;
        drive_backward(); // Move Backward for
1 second
        delay(1000);
        motor_stop(); // Stop Motor
        blink_state = 1;
    }
}
break;
} // End of switch loop
} //End of if (checksum)
} //End of "Read second sync byte" loop
} //End of "Read first sync byte" loop
} //End of loop function

////////////////////////////////////
////////////////////////////////////      ULTRASONIC SENSORS FUNCTION      //////////////////////////////////////
////////////////////////////////////

void ping(){
    digitalWrite(SIGNAL_PIN, HIGH); // Trigger the outward ping pulse(s)
    delayMicroseconds(10);
    digitalWrite(SIGNAL_PIN, LOW);
    PulseInZero::begin(); // Start listening out for the echo pulse on
interrupt 0
}

/*Pulse complete callback handler for PulseInZero
@param duration - pulse length in microseconds */
void pingPulseComplete(unsigned long duration) {

    if (duration <= 1200){
        motor_stop();
        obstacle_detected = true;
    }
    else{
        obstacle_detected = false;
    }
}

////////////////////////////////////
////////////////////////////////////      EYE BLINK FUNCTION      //////////////////////////////////////
////////////////////////////////////

boolean check_eye_blink(){

    previous_value = current_value;
    current_value = raw;
    difference1 = difference2;

```



```
difference2 = current_value - previous_value;

if (raw >= threshold && difference1 > 0 && difference2 < 0){ //
Local Maxima detected

    last_time = current_time;
    current_time = millis();
    time_difference = current_time - last_time;

    if (time_difference >= 40){ //Debouncing
        return true;
    }
    else{
        return false;
    }
}
}

////////////////////////////////////
//////////////////////////////////// MOTOR FUNCTIONS //////////////////////////////////////
////////////////////////////////////

void motor_stop(){
    digitalWrite(motor_left[0], LOW);
    digitalWrite(motor_left[1], LOW);

    digitalWrite(motor_right[0], LOW);
    digitalWrite(motor_right[1], LOW);
}

void drive_forward(int x){
    analogWrite(motor_left[0], x); // Analog output for different speed
    analogWrite(motor_right[0], x);
}

void drive_backward(){
    digitalWrite(motor_left[0], LOW);
    digitalWrite(motor_left[1], HIGH);

    digitalWrite(motor_right[0], LOW);
    digitalWrite(motor_right[1], HIGH);
}

void turn_left(){
    digitalWrite(motor_left[0], LOW);
    digitalWrite(motor_left[1], HIGH);

    digitalWrite(motor_right[0], HIGH);
    digitalWrite(motor_right[1], LOW);
}

void turn_right(){
```

Έλεγχος ρομποτικού μηχανισμού μέσω εγκεφαλικών σημάτων με χρήση πλατφόρμας ανοικτού υλικού

```
digitalWrite(motor_left[0], HIGH);  
digitalWrite(motor_left[1], LOW);  
  
digitalWrite(motor_right[0], LOW);  
digitalWrite(motor_right[1], HIGH);  
}
```

ΠΑΡΑΡΤΗΜΑ Β

Γενικά περί κινητήρων

Με τον όρο κινητήρας, εννοούμε κάθε σύστημα το οποίο μετατρέπει μέσω κατάλληλων διεργασιών, μία μορφή ενέργειας σε κινητική. Χαρακτηριστικό παράδειγμα αποτελεί ο κινητήρας των αυτοκινήτων (έχει επικρατήσει ο όρος Μηχανή Εσωτερικής Καύσης, Μ.Ε.Κ), ο οποίος μετατρέπει τη χημική ενέργεια του καυσίμου σε κινητική κυρίως ενέργεια και σε μικρότερο ποσοστό σε θερμική. Οι συνηθέστερες κατηγορίες κινητήρων που υπάρχουν είναι οι ακόλουθες [60] :

1. **Ηλεκτροκινητήρες** οι οποίοι μετατρέπουν την ηλεκτρική ενέργεια σε κινητική και διακρίνονται σε κινητήρες συνεχούς ρεύματος (DC Motors) και εναλλασσόμενου (AC Motors).
2. **Κινητήρες εσωτερικής καύσης** οι οποίοι μετατρέπουν τη χημική ενέργεια (πετρέλαιο και παράγωγά του π.χ. βενζίνη) σε κινητική και διακρίνονται σε κινητήρες diesel που χρησιμοποιούν σαν καύσιμο το πετρέλαιο και σε κινητήρες που χρησιμοποιούν σαν καύσιμο τη βενζίνη.
3. **Αερομηχανές** που χρησιμοποιούν σαν κινητήρια δύναμη τη δύναμη που παράγεται από την ανάφλεξη μείγματος ορισμένων αερίων και αέρα.
4. **Υδραυλικοί κινητήρες** οι οποίοι χρησιμοποιούν τη δύναμη που ασκεί το νερό όταν αυτό πέφτει πάνω σε μια ορισμένη επιφάνεια από ορισμένο ύψος που ονομάζεται "ύψος υδατόπτωσης".
5. **Ατμομηχανές** όπου την κινητήρια δύναμη παρέχει ο ατμός που παράγεται με τη θέρμανση νερού ή άλλου υγρού. Παλαιότερα χρησιμοποιήθηκαν για κίνηση κυρίως τραίνων.
6. **Αιολικοί κινητήρες ή ανεμογεννήτριες.** Είναι οι κινητήρες που χρησιμοποιούν τη δύναμη του ανέμου που για την παραγωγή ηλεκτρικής ενέργειας.
7. **Κινητήρες πυρηνικής ενέργειας.** Δεδομένου ότι η χρησιμοποίηση αυτών των κινητήρων βρίσκεται στη βρεφική της ηλικία και ότι οι δυνατότητές τους αναμένεται να είναι τεράστιες, πιστεύεται ότι η χρησιμοποίησή τους για παραγωγή ενέργειας θ' αυξηθεί σημαντικά.

Στην παρούσα εργασία, εξετάστηκε ο τρόπος λειτουργίας ηλεκτρικών κινητήρων και συγκεκριμένα των **Κινητήρων Συνεχούς Ρεύματος (DC Motors)**. Οι κινητήρες αυτοί παρέχουν τη δυνατότητα εύκολου ελέγχου της ταχύτητας και της ροπής, δύο βασικών χαρακτηριστικών κάθε κινητήρα σε βιομηχανικές εφαρμογές.

Ηλεκτρομαγνητική Επαγωγή

Η λειτουργία των DC κινητήρων βασίζεται στο φαινόμενο της «**Ηλεκτρομαγνητικής Επαγωγής**», η μαθητική περιγραφή του οποίου γίνεται στο «**Νόμο του Faraday**» προς τιμήν του Άγγλου επιστήμονα **Michael Faraday** (Εικόνα 59) (1791 - 1867) που πρώτος τον διατύπωσε το 1831.

Σύμφωνα με το φαινόμενο αυτό, όταν ένας μαγνήτης πλησιάσει το εσωτερικό ενός πηνίου, τότε το πηνίο αρχίζει να διαρρέεται από ηλεκτρικό ρεύμα. Το φαινόμενο αυτό εμφανίζεται μόνο κατά την κίνηση του μαγνήτη εντός του πηνίου ενώ εξαφανίζεται όταν υπάρχει απουσία κίνησης. Εάν συνδέσουμε ένα βολτόμετρο στα άκρα του πηνίου θα

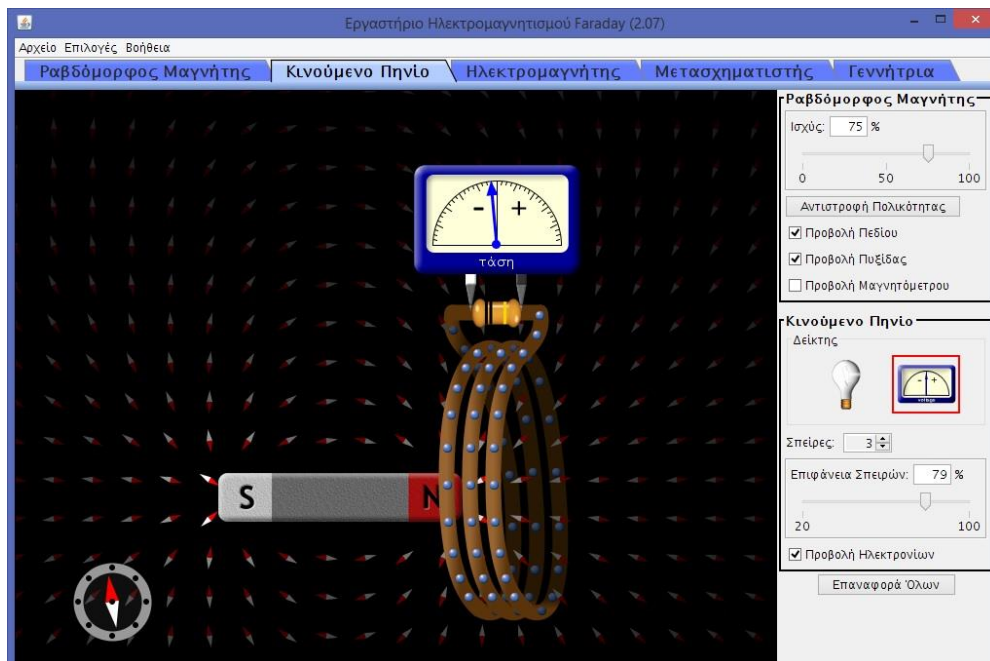


Εικόνα 59:
Michael Faraday
[61]

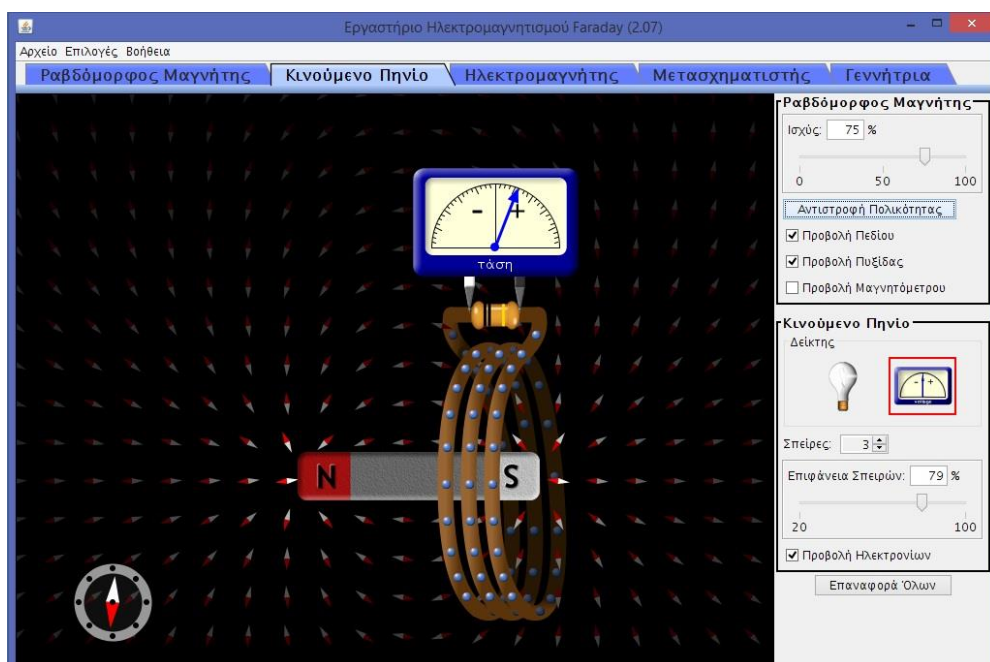
Έλεγχος ρομποτικού μηχανισμού μέσω εγκεφαλικών σημάτων με χρήση πλατφόρμας ανοικτού υλικού

παρατηρήσουμε ότι ο δείκτης κινείται προς μία κατεύθυνση όταν ο μαγνήτης κινείται προς το εσωτερικό του πηνίου, ενώ αλλάζει κατεύθυνση όταν ο μαγνήτης απομακρύνεται από το πηνίο .

Επίσης, η φορά του ηλεκτρικού πεδίου που δημιουργείται εξαρτάται από τη φορά των μαγνητικών γραμμών του μαγνητικού πεδίου. Στην περίπτωση που ο μαγνήτης εισέρχεται στο πηνίο με το βόρειο πόλο τότε η φορά του ηλεκτρικού πεδίου είναι από τον αρνητικό πόλο στο θετικό (Εικόνα 60) και αντίθετη όταν ο μαγνήτης εισέρχεται με το νότιο πόλο (Εικόνα 61).



Εικόνα 60: Αρνητική φορά ηλεκτρικού ρεύματος [62]



Εικόνα 61: Θετική φορά ηλεκτρικού ρεύματος [62]

Τέλος, η ένταση του ηλεκτρικού ρεύματος εξαρτάται από την ταχύτητα με την οποία εισέρχεται ο μαγνήτης στο πηνίο. Όσο μεγαλύτερη η ταχύτητα εισόδου του μαγνήτη, τόσο μεγαλύτερη η ένταση του δημιουργούμενου ηλεκτρικού.

Δομή DC Κινητήρα

Ένας DC κινητήρας αποτελείται από τα ακόλουθα μέρη:

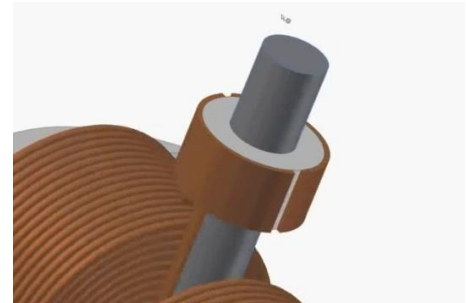
1. **Άξονας** (Shaft): Αποτελεί τον πυρήνα του κινητήρα και πάνω του στηρίζονται τα μεταλλικά μέρη τα οποία φέρουν τις περιελίξεις των πηνίων του κινητήρα.
2. **Ρότορας** (Rotor): Αποτελείται από τον πυρήνα και τα πηνία τα οποία είναι κατασκευασμένα από χαλκό. Ο πυρήνας μαζί με τα πηνία δημιουργούν τον ηλεκτρομαγνήτη που διαρρέεται από ηλεκτρικό ρεύμα. Αλλάζοντας τη φορά του ηλεκτρικού ρεύματος είναι δυνατόν να αλλάξει και η φορά περιστροφής του ρότορα.
3. **Συλλέκτης** (Commutator): Πρόκειται για ένα δαχτυλίδι τοποθετημένο πάνω στον άξονα αποτελούμενο από δύο ημικυκλικά κομμάτια φτιαγμένα από ειδικό υλικό, στο οποίο καταλήγουν αγώγιμες συνδέσεις, οι οποίες δεν εφάπτονται μεταξύ τους και οι οποίες ενώνονται με τον ηλεκτρομαγνήτη. Σκοπός του συλλέκτη είναι να τροφοδοτήσει με ρεύμα τα πηνία του ηλεκτρομαγνήτη.
4. **Στάτης** (Stator): Αποτελείται από δύο μόνιμους μαγνήτες τοποθετημένους αντίθετα ο ένας από τον άλλον και οι οποίοι έχουν αντίθετη πολικότητα.
5. **Ψήκτρες** (Brushes): Οι ψήκτρες ή καρβουνάκια εξασφαλίζουν την ηλεκτρική σύνδεση του συλλέκτη, παρέχοντας ρεύμα σε κάθε σύνδεση του συλλέκτη εναλλάξ, καθώς ο συλλέκτης περιστρέφεται.



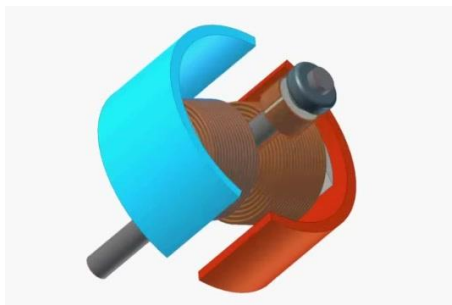
Εικόνα 62: Άξονας [63]



Εικόνα 63: Ρότορας [63]



Εικόνα 64: Συλλέκτης [63]



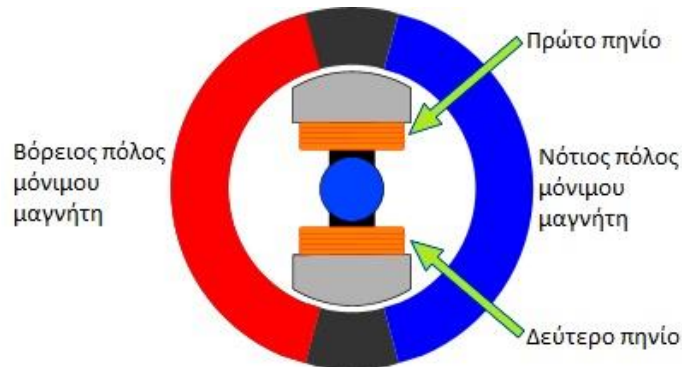
Εικόνα 65: Στάτης [63]



Εικόνα 66: Ψήκτρες [63]

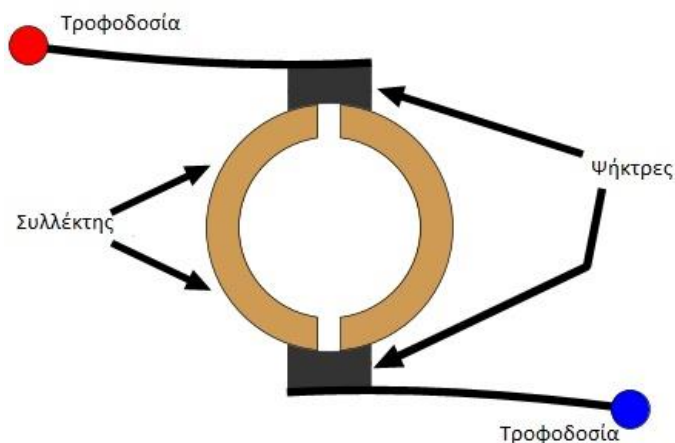
Βασική αρχή λειτουργίας DC κινητήρα

Ένας απλός DC κινητήρας βασικού τύπου αποτελείται από δύο πηνία και δύο μόνιμους μαγνήτες (Εικόνα 67).



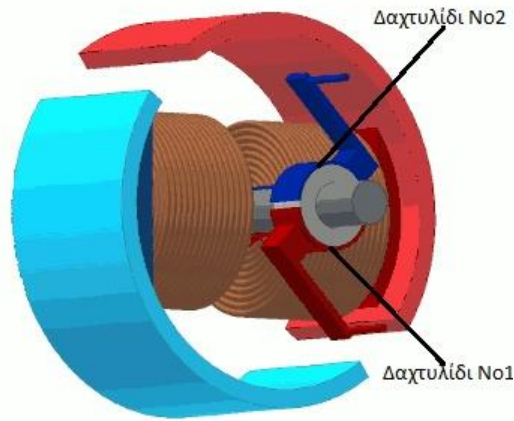
Εικόνα 67: Διάγραμμα DC κινητήρα [63]

Τα πηνία τροφοδοτούνται από το συλλέκτη και τις ψήκτρες. Το κάθε ημικυκλικό κομμάτι του συλλέκτη συνδέεται με το ένα από τα δύο πηνία με συνέπεια όταν τροφοδοτούνται τα δύο κομμάτια του συλλέκτη, να τροφοδοτούνται και τα πηνία (Εικόνα 68).

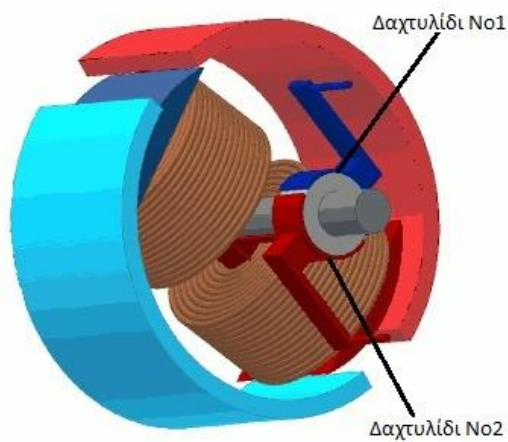


Εικόνα 68: Συλλέκτης – Ψήκτρες [63]

Κατά τη διάρκεια ενός πλήρους κύκλου του κινητήρα, ο άξονας του κινητήρα περιστρέφεται και μαζί του περιστρέφεται και ο συλλέκτης (Εικόνες 69 και 70). Αυτό προκαλεί αλλαγή της πολικότητας του ρεύματος που διαρρέει κάθε πηνίο με αποτέλεσμα να αλλάζει και η φορά του μαγνητικού πεδίου του ηλεκτρομαγνήτη. Δεδομένου, επίσης, ότι τα τυλίγματα των πηνίων που αποτελούν τον ηλεκτρομαγνήτη είναι τυλιγμένα σε αντίθετες φορές, όταν το ένα πηνίο αποτελεί το Νότιο Πόλο τότε το άλλο αποτελεί το Βόρειο Πόλο.



Εικόνα 69 Φόρτιση δαχτυλιδιών [63]



Εικόνα 70 Εναλλαγή φόρτισης δαχτυλιδιών [63]

Τέλος, η κίνηση του άξονα και συνεπώς του κινητήρα, εξαρτάται από την πολικότητα των πηνίων του ηλεκτρομαγνήτη. Η κίνηση προκαλείται μέσω των πηνίων τα οποία είτε έλκονται από το μόνιμο μαγνήτη (Εικόνα 71) όντας αντίθετα φορτισμένα,



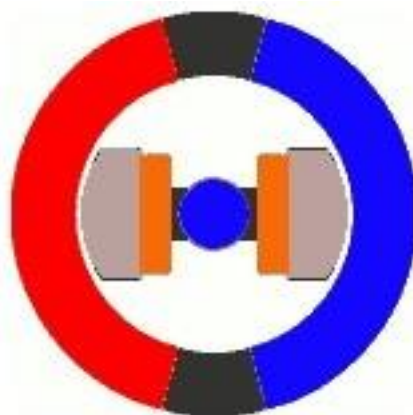
Εικόνα 71: Κίνηση κινητήρα με έλξη ηλεκτρομαγνήτη [63]

είτε απωθούνται από το μόνιμο μαγνήτη (Εικόνα 72) όντας όμοια φορτισμένοι.



Εικόνα 72: Κίνηση κινητήρα με απώθηση ηλεκτρομαγνήτη [63]

Υπάρχουν και στιγμές που ο ηλεκτρομαγνήτης απενεργοποιείται όταν συμβαίνει αλλαγή της φοράς του ηλεκτρικού ρεύματος. Στην περίπτωση αυτή (Εικόνα 73) ο κινητήρας παύει στιγμιαία να λειτουργεί και να δημιουργεί ροπή.

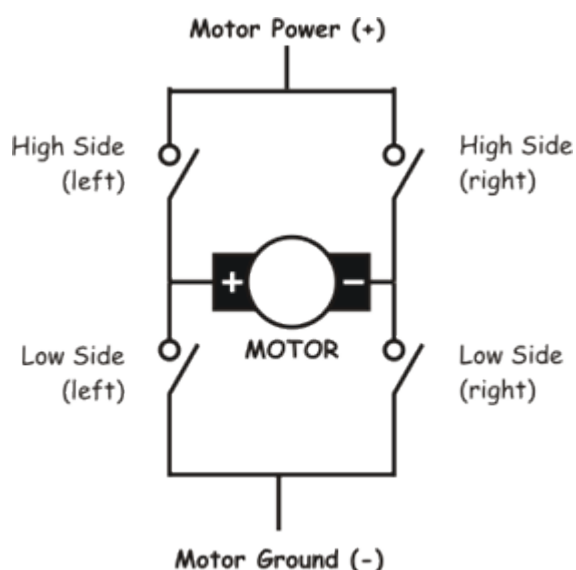


Εικόνα 73: Παύση κινητήρα [63]

Γέφυρα H (H - Bridge)

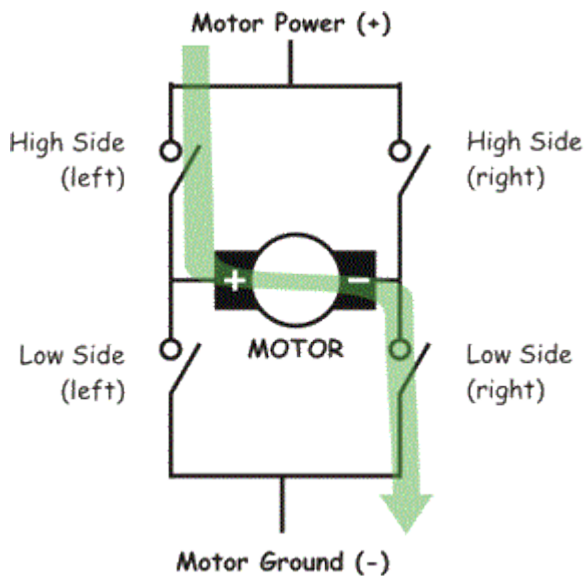
Για τον έλεγχο και την οδήγηση ενός DC κινητήρα είναι απαραίτητο να γνωρίζουμε τη λειτουργία ενός κυκλώματος που ονομάζεται **Γέφυρα H** (H – Bridge). Οφείλει την ονομασία του στο ότι αποτελείται από τέσσερα στοιχεία, συνήθως transistors, που παίζουν το ρόλο διακοπών, τα οποία τοποθετούνται στις γωνίες ενός σχήματος που έχει τη μορφή του γράμματος H και στη μέση τοποθετείται ο κινητήρα που πρόκειται να οδηγηθεί.

Τα τέσσερα transistors που έχουν το ρόλο διακοπών, λειτουργούν ανά ζεύγη, είτε το πάνω αριστερά με το κάτω δεξιά, είτε το πάνω δεξιά με το κάτω αριστερά αλλά ποτέ δεν πρέπει να ενεργοποιηθούν ταυτόχρονα τα transistors που ανήκουν στον ίδιο κλάδο το δεξί



Εικόνα 74: Διάγραμμα Γέφυρας H [64]

ή τον αριστερό. Κάτι τέτοιο θα δημιουργούσε βραχυκύκλωμα μεταξύ του θετικού και του αρνητικού πόλου της πηγής.



Εικόνα 75: Δεξιόστροφη λειτουργία κινητήρα [64]

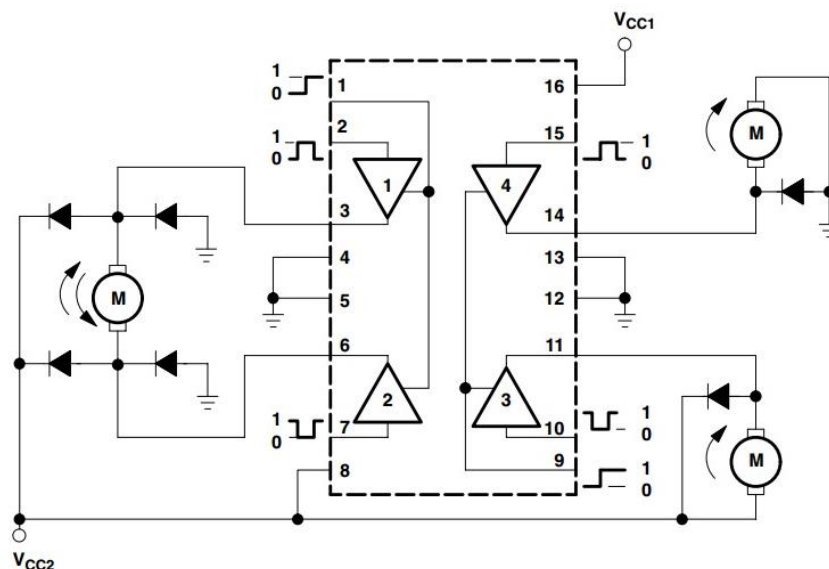
Για τη λειτουργία του κινητήρα χρειάζεται να ενεργοποιηθεί ένα από τα διαγώνια ζεύγη διακοπών. Στην εικόνα 75, ενεργοποιώντας τον πάνω αριστερά διακόπτη και τον κάτω δεξιά διακόπτη, αρχίζει η ροή ρεύματος από τον θετικό πόλο της πηγής προς τον αρνητικό. Το αποτέλεσμα είναι η περιστροφή του κινητήρα προς δεξιόστροφα. Αντίστοιχα στην περίπτωση που ενεργοποιηθούν οι διακόπτες πάνω δεξιά και κάτω αριστερά, ο κινητήρας θα κινηθεί αριστερόστροφα.

Η δημιουργία μίας γέφυρας H μπορεί να δημιουργηθεί με διακριτά εξαρτήματα που έχουν χαρακτηριστικά διακοπών, όπως οι διόδοι και τα τρανζίστορ. Στις σύγχρονες και πιο απαιτητικές εφαρμογές όμως, υπάρχουν έτοιμα **Ολοκληρωμένα Κυκλώματα**

(Integrated Circuits, ICs) που δίνουν ευχέρεια στο σχεδιασμό, πρόσθετη ασφάλεια και προσφέρουν μείωση του κόστους. Δύο ολοκληρωμένα κυκλώματα που χρησιμοποιούνται ευρέως και υλοποιούν τη γέφυρα H είναι:

- Το L293D (<http://www.ti.com/lit/ds/symlink/l293d.pdf>)
- Το SN754410 ne (<http://www.ti.com/lit/ds/symlink/sn754410.pdf>)

Και τα δύο παραπάνω ολοκληρωμένα, κατασκευάζονται από την εταιρεία Texas Instruments (www.ti.com). Η βασική τους διαφορά είναι ότι το L293D έχει μέγιστο ρεύμα εξόδου 600mA ενώ το SN754410ne έχει 1A.



Εικόνα 76: Block diagram L293D [65]

Είδη DC κινητήρων

Οι κινητήρες DC ανάλογα με τον τρόπο τροφοδοσίας του τυλίγματος διέγερσης χωρίζονται στις ακόλουθες κατηγορίες [66]:

1. **Ξένης Διέγερσης:** Το τύλιγμα πεδίου τροφοδοτείται από ανεξάρτητη πηγή τάσης/ρεύματος δίνοντας τη δυνατότητα για πολύ εύκολο έλεγχο του πεδίου διέγερσης.
2. **Διέγερσης σε Σειρά:** Το τύλιγμα πεδίου τροφοδοτείται σε σειρά με το τύλιγμα τυμπάνου δίνοντας τη δυνατότητα για κατασκευή κινητήρων υψηλής ροπής σε μικρές ταχύτητες.
3. **Παράλληλης Διέγερσης:** Το τύλιγμα πεδίου τροφοδοτείται παράλληλα με το τύλιγμα τυμπάνου δίνοντας τη δυνατότητα για κατασκευή κινητήρων υψηλών ταχυτήτων.
4. **Σύνθετης Διέγερσης:** Το τύλιγμα πεδίου τροφοδοτείται εν μέρει σε σειρά και εν μέρει παράλληλα με το τύλιγμα τυμπάνου δίνοντας τη δυνατότητα για κατασκευή κινητήρων περίπου σταθερής ταχύτητας παρά τις μεταβολές της ροπής.

Πλεονεκτήματα – Μειονεκτήματα DC Κινητήρων

Μερικά από τα κυριότερα **πλεονεκτήματα** της χρήσης DC κινητήρων είναι τα εξής [67]:

1. Ο έλεγχός τους είναι πολύ απλός
2. Μικρό κόστος
3. Είναι πολύ καλοί για εφαρμογές που απαιτούν μεταβαλλόμενη ταχύτητα

Στην αντίπερα όχθη τα βασικότερα **μειονεκτήματα** των DC κινητήρων είναι [67]:

1. Μικρότερη διάρκεια ζωής
2. Μεγαλύτερο κόστος συντήρησης
3. Έχουν περιορισμό σε χαμηλής ισχύος εφαρμογές

Αναφορές

- [1] Niedermeyer E. and da Silva F.L. (2004), «*Electroencephalography: Basic Principles, Clinical Applications, and Related Fields*», LippincotWilliams & Wilkins. ISBN: 0781751268.
- [2] http://www.quazoo.com/q/Richard_Caton
- [3] Mary Agnes Burniston Brazier, «*A History of the Electrical Activity of the Brain: The First Half Century*», London: Pitman Medical Publishing Company, 1961.
- [4] V. Pravdich-Neminsky, «*Ein Versuch der Registrierung der elektrischen Gehirnerscheinungen*», Zbl Physiol, 1912.
- [5] <http://www.findagrave.com/cgi-bin/fg.cgi?page=gr&GRid=68147693>
- [6] R. Ellingson, «*Introduction: The first Hans Berger lecture*», pp. 76(4) 294-295, 1990.
- [7] http://en.wikipedia.org/wiki/Edgar_Adrian,_1st_Baron_Adrian
- [8] Thomas F. Collura, «*History and Evolution of Electroencephalographic Instruments and Techniques*», Journal of Clinical Neurophysiology, pp. 10(4):476-504, 1993.
- [9] <http://beckerarchives.wustl.edu/index.php?p=collections/findingaid&id=8515>
- [10] <http://www.cmaj.ca/site/100/medleaders.xhtml>
- [11] <http://www.nae.edu/27592.aspx>
- [12] <http://www.ilae.org/visitors/Farewells/Goldensohn.cfm>
- [13] <https://wiki.engr.illinois.edu/display/BIOE414/EEG+-+History>
- [14] Arthur C. Guyton, John. E. Hall, «*Textbook of Medical Physiology*», Philadelphia, Pennsylvania: Elsevier Saunders, 2006.
- [15] <http://www.rhodes.aegean.gr/sxedia/grafdaskalou/anatomy/sub1/thought/pcns.htm>
- [16] D. N. Sehati, «*Nervous System Anatomy and Physiology*», <http://sehati.org/index/patientresources/normalanatomy.html>
- [17] <http://www.adsi.info/forebraintechnique.htm>
- [18] <http://brainanatomy.tk/tag/diagram-of-the-hindbrain-forebrain-midbrain>
- [19] <https://dandelion199.wordpress.com/2014/11/22/awesome-facts-about-the-human-brain/>
- [20] M. Teplan, «*Fundamentals of EEG Measurement*», MEASUREMENT SCIENCE REVIEW, 2002.
- [21] Harold L. Atwood, Ph.D. Mackay W.A, «*Essentials of Neurophysiology*», B.C Decker, Hamilton, Canada: Mosby-Year Book, 1989.
- [22] P. L. Nunez, «*Neocortical Dynamics and Human EEG Rhythms*», New York: Oxford University Press, 1995.
- [23] <http://en.wikipedia.org/wiki/Electroencephalography>
- [24] <http://www.lucidsage.com/ep-4-the-lucid-dreaming-podcast/>
- [25] Jonathan R. Wolpaw, Niels Birbaumer, Dennis J. McFarlanda, Gert Pfurtschellere, Theresa M. Vaughan, «*Brain-computer interfaces for communication and control*», ELSEVIER, p. 113 (2002) 767–791, 2 March 2002.

- [26] Jeremiah D. Wander, Rajesh PN Rao, «*Brain–computer interfaces: a powerful tool for scientific inquiry*», *ELSEVIER*, p. 70–75, April 2014.
- [27] Gert Pfurtscheller, Allison BZ, Brunner C, Bauernfeind G, Solis-Escalante T, Scherer R, Zander TO, Mueller-Putz G, Neuper C, Birbaumer N, «*The Hybrid BCI*», *Frontiers in Neuroscience*, p. 4: 42, 21 April 2010.
- [28] <http://www.cs.ucla.edu/~vidal/vidal.html>
- [29] Jacques J. Vidal, *Toward Direct Brain-Computer Communication*, University of California, Los Angeles, 1973.
- [30] Anirudh Vallabhaneni, Tao Wang, Bin He, «*Brain—Computer Interface*», *Neural Engineering*, Springer US, 2005, pp. 85-121.
- [31] <http://www.averybiomedical.com/memorial-william-h-dobelle-ph-d-1941-2004/>
- [32] http://neuroscience.berkeley.edu/users/users_profile.php?id=5
- [33] BBC, «*Looking through cats' eyes*» 11 October 1999: <http://news.bbc.co.uk/2/hi/science/nature/471786.stm>.
- [34] <https://pni.princeton.edu/directory/ray-lee>
- [35] Md. Ibrahim Arafat, «*BrainComputer Interface: Past, Present & Future*». Dept. of Computer Science & Engineering, International Islamic University Chittagong (IIUC), Chittagong, Bangladesh.
- [36] G. H. Melis Oner, «*Analyzing One-Channel EEG Signals for Detection of Close and Open Eyes Activities*», 2013 Second IIAI International Conference on Advanced Applied Informatics, Los Alamitos, CA, Aug. 31 2013-Sept. 4 2013.
- [37] <http://www.um.edu.mt/think/moving-wheelchairs-with-your-thoughts/>
- [38] http://www.bci2000.org/wiki/index.php/User_Tutorial:EEG_Measurement_Setup#The_10-20_International_System
- [39] <http://www.edfplus.info/specs/edftexts.html>
- [40] <http://techreport.com/review/14957/ocz-neural-impulse-actuator>
- [41] http://www.coroflot.com/chris_thomson/Emotiv-Epoc
- [42] <http://press.neurosky.com/MindWaveMobile.html>
- [43] <http://www.choosemuse.com/what-does-it-measure/>
- [44] <https://openbci.myshopify.com/collections/frontpage/products/openbci-8-bit-board-kit>
- [45] <https://learn.sparkfun.com/tutorials/hackers-in-residence---hacking-mindwave-mobile/all>
- [46] http://download.neurosky.com/support_page_files/MindWaveMobile/docs/mindwave_mobile_user_guide.pdf
- [47] W. Salabun, «*Processing and spectral analysis of the raw EEG signal from the MindWave*», Szczecin: West Pomeranian University of Technology, 2014.
- [48] <http://www.seeedstudio.com/document/pdf/TGAM%20Ddatasheet.pdf>
- [49] http://wearingcam.org/ece516/mindset_communications_protocol.pdf
- [50] <https://www.sparkfun.com/datasheets/Wireless/Bluetooth/rn-42-ds.pdf>

- [51] <http://users.ece.utexas.edu/~valvano/Datasheets/HCSR04b.pdf>
- [52] <http://arduino.cc/en/Guide/Introduction>
- [53] <http://arduino.cc/en/Main/ArduinoBoardUno>
- [54] <http://deltahacker.gr/arduino-intro/>
- [55] <https://github.com/kitschpatrol/BrainGrapher>
- [56] <http://linuxgizmos.com/intel-unveils-second-generation-galileo-hacker-sbc/>
- [57] <http://www.linuxuser.co.uk/reviews/beaglebone-black-review>
- [58] <http://www.raspberrypi.org/forums/viewtopic.php?t=4751>
- [59] <http://www.digi.com/products/wireless-wired-embedded-solutions/single-board-computers/bl4s200>
- [60] <http://el.wikipedia.org/wiki/%CE%9A%CE%B9%CE%BD%CE%B7%CF%84%CE%AE%CF%81%CE%B1%CF%82>
- [61] <http://www.rare-earth-magnets.com/t-michael-faraday.aspx>
- [62] <http://phet.colorado.edu/el/simulation/faraday>
- [63] http://www.pcbheaven.com/wikipages/How_DC_Motors_Work/
- [64] <http://www.mcmanis.com/chuck/robotics/tutorial/h-bridge/>
- [65] <http://users.ece.utexas.edu/~valvano/Datasheets/L293d.pdf>
- [66] Α. Κ. Σεργιάδης, «Διερεύνηση λειτουργίας και κατασκευή οδήγησης βηματικού κινητήρα μέσω παράλληλης θύρας», Αθήνα: Εθνικό Μετσόβιο Πολυτεχνείο, 2009.
- [67] <https://engineering.purdue.edu/ME588/LectureNotes/Unit5--DCMotors.pdf>