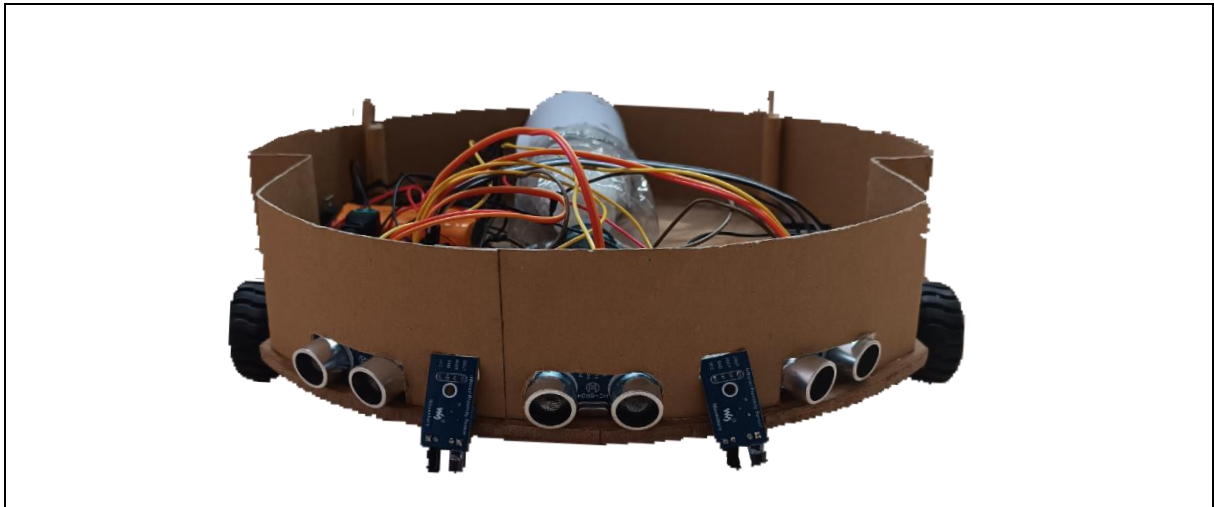


ΣΧΟΛΗ ΜΗΧΑΝΙΚΩΝ
ΤΜΗΜΑ ΜΗΧΑΝΙΚΩΝ ΠΛΗΡΟΦΟΡΙΚΗΣ
ΚΑΙ ΗΛΕΚΤΡΟΝΙΚΩΝ ΣΥΣΤΗΜΑΤΩΝ

ΔΙΠΛΩΜΑΤΙΚΗ ΕΡΓΑΣΙΑ
«Ηλεκτρική σκούπα ελεγχόμενη από μικροελεγκτή»



Του φοιτητή
Παντελεήμων Κυριακίδη
Αρ. Μητρώου: 517067

Επιβλέπων
Άγγελος Γιακουμής
Επίκουρος Καθηγητής

Τίτλος Δ.Ε. Ηλεκτρική σκούπα ελεγχόμενη από μικροελεγκτή

Κωδικός Δ.Ε. 23264

Όνοματεπώνυμο φοιτητή Παντελεήμων Κυριακίδη

Όνοματεπώνυμο εισηγητή Άγγελος Γιακουμής

Ημερομηνία ανάληψης Δ.Ε. 06-10-2023

Ημερομηνία περάτωσης Δ.Ε 06-05-2024

Βεβαιώνω ότι είμαι ο συγγραφέας αυτής της εργασίας και ότι κάθε βοήθεια την οποία είχα για την προετοιμασία της είναι πλήρως αναγνωρισμένη και αναφέρεται στην εργασία. Επίσης, έχω καταγράψει τις όποιες πηγές από τις οποίες έκανα χρήση δεδομένων, ιδεών, εικόνων και κειμένου, είτε αυτές αναφέρονται ακριβώς είτε παραφρασμένες. Επιπλέον, βεβαιώνω ότι αυτή η εργασία προετοιμάστηκε από εμένα προσωπικά, ειδικά ως διπλωματική εργασία, στο Τμήμα Μηχανικών Πληροφορικής και Ηλεκτρονικών Συστημάτων του ΔΙ.ΠΑ.Ε.

Η παρούσα εργασία αποτελεί πνευματική ιδιοκτησία του φοιτητή Παντελεήμων Κυριακίδη που την εκπόνησε. Στο πλαίσιο της πολιτικής ανοικτής πρόσβασης, ο συγγραφέας/δημιουργός εκχωρεί στο Διεθνές Πανεπιστήμιο της Ελλάδος άδεια χρήσης του δικαιώματος αναπαραγωγής, δανεισμού, παρουσίασης στο κοινό και ψηφιακής διάχυσης της εργασίας διεθνώς, σε ηλεκτρονική μορφή και σε οποιοδήποτε μέσο, για διδακτικούς και ερευνητικούς σκοπούς, άνευ ανταλλάγματος. Η ανοικτή πρόσβαση στο πλήρες κείμενο της εργασίας, δεν σημαίνει καθ' οιονδήποτε τρόπο παραχώρηση δικαιωμάτων διανοητικής ιδιοκτησίας του συγγραφέα/δημιουργού, ούτε επιτρέπει την αναπαραγωγή, αναδημοσίευση, αντιγραφή, πώληση, εμπορική χρήση, διανομή, έκδοση, μεταφόρτωση (downloading), ανάρτηση (uploading), μετάφραση, τροποποίηση με οποιονδήποτε τρόπο, τμηματικά ή περιληπτικά της εργασίας, χωρίς τη ρητή προηγούμενη έγγραφη συναίνεση του συγγραφέα/δημιουργού.

Η έγκριση της διπλωματικής εργασίας από το Τμήμα Μηχανικών Πληροφορικής και Ηλεκτρονικών Συστημάτων του Διεθνούς Πανεπιστημίου της Ελλάδος, δεν υποδηλώνει απαραίτητως και αποδοχή των απόψεων του συγγραφέα, εκ μέρους του Τμήματος.

«Για τους αγαπημένους μου ανθρώπους»

Πρόλογος

Η επιλογή της παρούσης διπλωματικής εργασίας έγινε γιατί συνδυάζει έως έναν βαθμό την ρομποτική με τις μαθηματικές συναρτήσεις που οδηγούν στο χάος. Οι συγκεκριμένοι κλάδοι, συγκαταλέγονται στο ερευνητικό μου ενδιαφέρον και ήταν χαρά μου να διευρύνω τις γνώσεις μου σε αυτά τα θέματα. Με την υλοποίηση της κατασκευής κατάφερα να ελέγξω γνώσεις μου και τις ικανότητές μου να συνδυάσω πληροφορίες από διάφορες πηγές και να δημιουργήσω κάτι που λειτουργεί σε ικανοποιητικό βαθμό.

Περίληψη

Με τη διεκπεραίωση της διπλωματικής εργασίας, υλοποιήθηκε μια αυτόματη σκούπα ρομπότ η οποία έχει την ικανότητα πλοήγησης σε έναν άγνωστο χώρο, χωρίς καμία πληροφορία γι αυτόν, πέρα από αυτές που συλλέγει από τους αισθητήρες της. Για την κατασκευή χρησιμοποιήθηκαν πέντε αισθητήρες, τρεις για τον έλεγχο αντικειμένου στο μπροστινό μέρος της και δύο για τον έλεγχο ύπαρξης δαπέδου, ώστε να μπορεί να κινηθεί χωρίς τον φόβο πτώσης από σκάλες ή κάποιο κενό στο πάτωμα. Η επιλογή της κατεύθυνσης γίνεται από μια ψευδοτυχαία γεννήτρια αριθμών, τα αποτελέσματα της οποίας αντιστοιχίζονται με μια από τις επτά κατευθύνσεις των δύο αξόνων. Φυσικά, για την υλοποίηση έγινε πρώτα η απαραίτητη μελέτη και η επιλογή των εξαρτημάτων όπως παρουσιάζονται και στην συνέχεια. Σημαντικό κομμάτι ήταν επίσης και η επιλογή της συνάρτησης, διεξήχθησαν τα απαραίτητα τεστ για τον προσδιορισμό της καλύτερης επιλογής, με τα αποτελέσματά τους να αναλύονται στην συνέχεια της εργασίας. Τέλος πραγματοποιήθηκαν διάφορα πειράματα για τον προσδιορισμό της αποτελεσματικότητάς της, ενώ έγινε και η σύγκριση μεταξύ τους.

« Vacuum cleaner controlled by microcontroller»

«Panteleimon Kyriakidis»

Abstract

With the completion of the thesis, an automatic vacuum cleaner was implemented which has the ability to navigate in an unknown space, without any information about it, apart from the information it collects from its sensors. Five sensors were used for the construction, three to control if there is an object in the front of the vacuum cleaner and two to control the existence of a floor, so that it can move without the fear of falling down stairs or a gap in the floor. The direction is selected by a pseudo-random number generator, the results of which are mapped to one of the seven directions of the two axes. Of course, for the implementation the necessary study and selection of the components was done first as presented below. An important part was also the selection of the function, the necessary tests were carried out to determine the best choice and their results are discussed later in the paper. Finally, several experiments were carried out to determine its effectiveness and a comparison between them was made.

Ευχαριστίες

Πριν ξεκινήσει η εργασία, θα ήθελα να ευχαριστήσω όλους εκείνους που συνέβαλαν και με βοήθησαν για την πραγματοποίησή της. Κυρίως, θα ήθελα να ευχαριστήσω τον επιβλέποντα καθηγητή κ. Γιακουμή Άγγελο που μου έδωσε την ευκαιρία να μελετήσω και να υλοποιήσω ένα τόσο ενδιαφέρον, για εμένα, θέμα. Επίσης να τον ευχαριστήσω για την βοήθειά του, την καθοδήγησή του καθώς και για την υπομονή του καθ' όλη τη διάρκεια εκπόνησης της διπλωματικής μου εργασίας. Πέρα και πάνω από όλα όμως, να ευχαριστήσω τους γονείς μου, για την αμέριστη και ανιδιοτελή ψυχολογική και οικονομική υποστήριξη μου όλα αυτά τα χρόνια.

Περιεχόμενα

Πρόλογος.....	v
Περίληψη.....	vi
Abstract	vii
Ευχαριστίες	viii
Περιεχόμενα	ix
Κατάλογος Σχημάτων	xii
Κατάλογος Πινάκων.....	xiii
Κεφάλαιο 1ο: Εισαγωγή.....	i
Κεφάλαιο 2ο: Ιστορική Αναδρομή.....	1
2.1 Εισαγωγή.....	1
2.2 Κυριότερα μοντέλα	1
2.3 Επίλογος.....	5
Κεφάλαιο 3ο: Υλικά.....	5
3.1 Εισαγωγή.....	5
3.2 Renesas RA4M1.....	6
3.2.1 Arm® Cortex®-M4.....	6
3.2.2 Μνήμη	12
3.2.3 Συνδεσιμότητα.....	13
3.2.4 Μετατροπές και λειτουργίες.....	14
3.2.5 Χρονιστές.....	14
3.3 Κινητήρες.....	15
3.4 Κύκλωμα οδήγησης κινητήρων με L298N	16
3.5 Αισθητήρας μέτρησης απόστασης με υπέρηχους.....	18
3.6 Αισθητήρας ανίχνευσης αντικειμένου με υπέρυθρες ακτίνες.....	19
3.7 Ψηφιακό Βολτόμετρο.....	20
3.8 Συσσκευή αναρρόφησης.....	21
3.9 Επίλογος.....	22
Κεφάλαιο 4ο: Κίνηση.....	22
4.1 Εισαγωγή.....	22
4.2 Κατηγορίες τροχών ανά χρήση	22
4.3 Κατηγορίες ανά ενέργειες τροχών.....	23

4.4	Επίλογος.....	25
Κεφάλαιο 5ο: Πλοήγηση.....		25
5.1	Εισαγωγή.....	25
5.2	Πρόγνωση.....	25
5.3	Σάρωση.....	26
5.4	Τυχαίες κινήσεις.....	27
5.5	Επίλογος.....	28
Κεφάλαιο 6ο: Χάος.....		28
6.1	Εισαγωγή.....	28
6.2	Εισαγωγή στα μαθηματικά του Χάους.....	28
6.3	Έλεγχος τυχειότητας.....	30
6.4	Βασικές συναρτήσεις.....	32
6.5	Έλεγχος συναρτήσεων.....	37
6.6	Επίλογος.....	39
Κεφάλαιο 7ο: Υλοποίηση.....		39
7.1	Εισαγωγή.....	39
7.2	Σχεδίαση και μελέτη σε πρόγραμμα.....	41
7.3	Τυχαίες κινήσεις.....	42
7.3.1	Επιλογή συνάρτησης.....	43
7.3.2	Αντιστοίχιση κατευθύνσεων.....	46
7.4	Ροή του προγράμματος.....	47
7.5	Υλοποίηση κατασκευής.....	48
7.6	Επίλογος.....	53
Κεφάλαιο 8ο: Πειράματα.....		53
8.1	Εισαγωγή.....	53
8.2	Πρώτο σετ δοκιμών.....	54
8.3	Δεύτερο σετ δοκιμών.....	57
8.4	Επίλογος.....	60
Κεφάλαιο 9ο: Συμπεράσματα και προτάσεις βελτίωσης.....		61
ΒΙΒΛΙΟΓΡΑΦΙΑ.....		62
ΠΑΡΑΡΤΗΜΑ Α : Κώδικας προγραμματισμού Arduino για τον έλεγχο αισθητήρων, υπολογισμού συνάρτησης και κίνησης.....		65
ΠΑΡΑΡΤΗΜΑ Β : Κώδικας προγραμματισμού Arduino για τον υπολογισμό του χρόνου εκτέλεσης της κάθε συνάρτησης.....		74

ΠΑΡΑΡΤΗΜΑ C : Κώδικας προγραμματισμού Arduino για τον υπολογισμό τυχαιότητας των συναρτήσεων.....	77
ΠΑΡΑΡΤΗΜΑ D : Κώδικας προγραμματισμού Arduino για τον υπολογισμό της τυχαιότητας των αποτελεσμάτων μετά την αντιστοίχιση.....	90

Κατάλογος Σχημάτων

Σχήμα 2.1. Whirlpool Automatic Floor Cleaner	2
Σχήμα 2.2. Electrolux Trilobite	3
Σχήμα 2.3. Dyson DC-06	4
Σχήμα 2.4. iRobot Roomba (αριστερά), Roomba Pro (κέντρο), Roomba Pro Elite (δεξιά)	4
Σχήμα 2.5. Neato XV-11	5
Σχήμα 3.1. Χαρακτηριστικά και λειτουργίες που διαθέτει το ολοκληρωμένο κύκλωμα RA4M1 της Renesas	6
Σχήμα 3.2. Λειτουργίες του μικροεπεξεργαστή Cortex M4	7
Σχήμα 3.3. Κινητήρας συνεχούς ρεύματος - πίσω όψη	15
Σχήμα 3.4. Κινητήρας συνεχούς ρεύματος - μπροστινή όψη	15
Σχήμα 3.5. Κυκλωμα οδήγησης κινητήρων με L298N - κάτω όψη	16
Σχήμα 3.6. Κυκλωμα οδήγησης κινητήρων με L298N - πάνω όψη	16
Σχήμα 3.7. Αισθητήρας μέτρησης απόστασης με υπέρηχους - πίσω όψη	18
Σχήμα 3.8. Αισθητήρας μέτρησης απόστασης με υπέρηχους - μπροστινή όψη	18
Σχήμα 3.9. Αισθητήρας ανίχνευσης αντικειμένου με υπέρυθρες ακτίνες - μπροστινή όψη	19
Σχήμα 3.10. Αισθητήρας ανίχνευσης αντικειμένου με υπέρυθρες ακτίνες - πίσω όψη	19
Σχήμα 3.11. Πισινή όψη ψηφιακού βολτόμετρου	21
Σχήμα 3.12. Μπροστινή όψη ψηφιακού βολτόμετρου	21
Σχήμα 4.1. Κυκλικός τροχός (αριστερά), τροχός με έξι άκρα (κέντρο), πανκατευθυντικός τροχός (δεξιά)	23
Σχήμα 5.1. Χάρτης εκτέλεσης βημάτων για προγνωστική συσκευή	26
Σχήμα 5.2. Τελικός χάρτης μετά την σάρωση του χώρου (πάνω αριστερά), Σημείο εκκίνησης συσκευής (πάνω δεξιά), Υπολογισμός βέλτιστης διαδρομής (κάτω αριστερά), Τέλος εκτέλεσης προγράμματος και μετακίνηση σε τελική θέση (κάτω δεξιά)	27
Σχήμα 6.1. Διάγραμμα διακλάδωσης λογιστικού χάρτη	33
Σχήμα 6.2. Διάγραμμα εκθέτη Lyapunov για λογιστικό χάρτη	33
Σχήμα 6.3. Διάγραμμα εκθέτη Lyapunov για χάρτη ημιτόνου	34
Σχήμα 6.4. Διάγραμμα διακλάδωσης χάρτη ημιτόνου	34
Σχήμα 6.5. Διάγραμμα διακλάδωσης χάρτη Renyi	34
Σχήμα 6.6. Διάγραμμα εκθέτη Lyapunov για χάρτη Renyi	35
Σχήμα 6.7. Διάγραμμα διακλάδωσης χάρτη τροποποιημένης σκηνής	36
Σχήμα 6.8. Διάγραμμα εκθετών Lyapunov χάρτη τροποποιημένης σκηνής	36
Σχήμα 6.9. Διαγράμματος διακλάδωσης στρεβλώμενου χάρτη σκηνής	37
Σχήμα 6.10. Διάγραμμα εκθετών Lyapunov στρεβλώμενου χάρτη σκηνής	37
Σχήμα 7.1. Κύκλωμα ρυθμιστή Im7805 σταθερής τάσης εξόδου	39
Σχήμα 7.2. Σχηματικό ηλεκτρονικό κύκλωμα αυτόματης σκούπας	41
Σχήμα 7.3. Συνδέσεις των υπέρηχων αισθητήρων με τον επεξεργαστή	41
Σχήμα 7.4. Συνδέσεις των υπέρυθρων αισθητήρων με τον επεξεργαστή	41
Σχήμα 7.5. Συνδέσεις κυκλώματος οδήγησης κινητήρων	42
Σχήμα 7.6. Διάγραμμα απεικόνισης εμφάνισης κάθε αριθμού στον στρεβλωμένο χάρτη σκηνής	45
Σχήμα 7.7. Διάγραμμα απεικόνισης εμφάνισης κάθε αριθμού στον τροποποιημένο χάρτη σκηνής	45
Σχήμα 7.8. Διάγραμμα απεικόνισης εμφάνισης κάθε αριθμού στον χάρτη Renyi	45
Σχήμα 7.9. Διάγραμμα απεικόνισης εμφάνισης κάθε αριθμού στον χάρτη ημιτόνου	45
Σχήμα 7.10. Διάγραμμα απεικόνισης εμφάνισης κάθε αριθμού στον λογιστικό χάρτη	45

Σχήμα 7.11. Ποσοστά εμφάνισης αποτελεσμάτων μετά από αντιστοίχιση σε 7 και 6 τιμές	47
Σχήμα 7.12. Διάγραμμα ροής ελέγχου του προγράμματος	47
Σχήμα 7.13. Καπάκι κατασκευής	49
Σχήμα 7.14. Βάσης κατασκευής.....	49
Σχήμα 7.15. Δεξιό πλαϊνό μέρος μέρος.....	49
Σχήμα 7.16. Επάνω όψη κατασκευής.....	49
Σχήμα 7.17. Ηλεκτρονική πλακέτα	50
Σχήμα 7.18. Κάτω όψη της βάσης.....	50
Σχήμα 7.19. Δοκιμές επιλογής θέσης των υπέρηχων.....	50
Σχήμα 7.20. Κάτω όψη σκούπας μετά από τρύπες για αναρρόφηση	51
Σχήμα 7.21. Μπροστινό μέρος σκούπας με αισθητήρες υπέρηχων	51
Σχήμα 7.22. Δεξιο πλαϊνό μέρος κατασκευής.....	52
Σχήμα 7.23. Μπροστινό μέρος τελικής κατασκευής.....	52
Σχήμα 7.24. Πισινό μέρος τελικής κατασκευής	52
Σχήμα 7.25. Αριστερό πλαϊνό μέρος κατασκευής.....	52
Σχήμα 7.26. Μεγέθυνση στον επεξεργαστή για την απεικόνιση των συνδέσεων	53
Σχήμα 7.27. Επάνω όψη τελικής κατασκευής.....	53
Σχήμα 8.1. Ελεγχόμενος εργασιακός χώρος	54
Σχήμα 8.2. Πείραμα 1 ^ο σε κενό χώρο.....	54
Σχήμα 8.3. Πείραμα 2 ^ο σε κενό χώρο.....	54
Σχήμα 8.4. Πείραμα 4 ^ο σε κενό χώρο.....	55
Σχήμα 8.5. Πείραμα 3 ^ο σε κενό χώρο.....	55
Σχήμα 8.6. Πείραμα 5 ^ο σε κενό χώρο.....	56
Σχήμα 8.7. Πείραμα 6 ^ο σε κενό χώρο.....	56
Σχήμα 8.8. Πείραμα 8 ^ο σε κενό χώρο.....	56
Σχήμα 8.9. Πείραμα 7 ^ο σε κενό χώρο.....	56
Σχήμα 8.10. Πείραμα 9 ^ο σε κενό χώρο.....	57
Σχήμα 8.11. Πείραμα 10 ^ο σε κενό χώρο.....	57
Σχήμα 8.12. Πείραμα 2 ^ο με αντικείμενο στο χώρο	58
Σχήμα 8.13. Πείραμα 1 ^ο με αντικείμενο στο χώρο	58
Σχήμα 8.14. Πείραμα 3 ^ο με αντικείμενο στο χώρο	58
Σχήμα 8.15. Πείραμα 4 ^ο με αντικείμενο στο χώρο	58
Σχήμα 8.16. Πείραμα 6 ^ο με αντικείμενο στο χώρο	59
Σχήμα 8.17. Πείραμα 5 ^ο με αντικείμενο στο χώρο	59
Σχήμα 8.18. Πείραμα 7 ^ο με αντικείμενο στο χώρο	59
Σχήμα 8.19. Πείραμα 8 ^ο με αντικείμενο στο χώρο	59
Σχήμα 8.20. Πείραμα 9 ^ο με αντικείμενο στο χώρο	60
Σχήμα 8.21. Πείραμα 10 ^ο με αντικείμενο στο χώρο	60

Κατάλογος Πινάκων

Πίνακας 3.1. Τεχνικά χαρακτηριστικά εικονιζόμενων κινητήρων.....	16
Πίνακας 3.2. Τεχνικά χαρακτηριστικά του κυκλώματος οδήγησης κινητήρων με χρήση L298N.....	17
Πίνακας 3.3. Τεχνικά χαρακτηριστικά του αισθητήρα μέτρησης απόστασης με χρήση υπέρηχων	19

Πίνακας 3.4. Τεχνικά χαρακτηριστικά του αισθητήρα μέτρησης απόστασης με χρήση υπέρυθρων ακτίνων.....	20
Πίνακας 3.5. Τεχνικά χαρακτηριστικά ψηφιακού βολτόμετρου	21
Πίνακας 3.6. Τεχνικά χαρακτηριστικά κινητήρα αναρρόφησης.....	22
Πίνακας 4.1. Όλοι οι πιθανοί συνδυασμοί υλοποιήσεις της κατασκευής με τρεις τροχούς και η χρήση τους.....	24
Πίνακας 6.1. Αποτελέσματα τεστ NIST για λογιστικού, ημιτόνου, Renyi και τροποποιημένου χάρτη	38
Πίνακας 7.1. Κατευθύνσεις κίνησης κατασκευής.....	40
Πίνακας 7.2. Τιμές μεταβλητών για τον υπολογισμό των συναρτήσεων.....	43
Πίνακας 7.3. Αποτελέσματα χρόνου εκτέλεσης των συναρτήσεων.....	43
Πίνακας 7.4. Αποτελέσματα ποσοστών εμφάνισης του κάθε αριθμού.....	44
Πίνακας 7.5. Αντιστοίχιση σετ κατευθύνσεων	46

Κεφάλαιο 1ο: Εισαγωγή

Στο πλαίσιο της συγκεκριμένης διπλωματικής εργασίας ορίστηκε η μελέτη και η υλοποίηση μιας αυτόματης σκούπας ρομπότ, η οποία θα είχε την δυνατότητα πλοήγησης στον χώρο μέσω μιας γεννήτριας ψευδοτυχαίων αριθμών. Ο έλεγχος του περιβάλλοντος χώρου θα γινόταν μέσω διαφόρων αισθητήρων στο μπροστινό μέρος της κατασκευής και θα λαμβάνονταν οι απαραίτητες αποφάσεις από ένα μικροεπεξεργαστή για την πλοήγησή της στον χώρο εργασίας. Στόχος της είναι η ανεμπόδιστη πλοήγησή της, καθώς θα είναι εφικτή η αποφυγή των αντικειμένων και η κάλυψη όσο τον δυνατόν περισσότερου χώρου. Τα κεφάλαια της εργασίας ξεκινούν με μια ιστορική αναδρομή για την εξέλιξη των αυτόματων σκουπών μέχρι την τελική τους μορφή, όπως είναι την σημερινή εποχή. Στην συνέχεια γίνεται η αναφορά των υλικών όπου χρησιμοποιήθηκαν και παρουσιάζονται οι λειτουργίες και τα χαρακτηριστικά τους. Μετά έγινε μια κατηγοριοποίηση του μέσου κίνησης τέτοιων αυτόματων συσκευών. Ακολουθούν οι διαφορετικές επιλογές πλοήγησης σε ένα χώρο εργασίας. Γίνεται μια εισαγωγή στην έννοια του χάους και παρουσιάζονται μερικές συναρτήσεις και οι τρόποι ελέγχου τους. Στην συνέχεια γίνεται η αναφορά στα στάδια υλοποίησης της κατασκευής και τέλος αναγράφονται τα αποτελέσματα των πειραμάτων και μια μικρή σύγκρισή τους.

Κεφάλαιο 2ο: Ιστορική Αναδρομή

2.1 Εισαγωγή

Σε αυτό το κεφάλαιο διεξήχθη ιστορική αναφορά στα διάφορα στάδια που δρομολογήθηκαν μέχρι την εξέλιξη των αυτόματων σκουπών όπως είναι γνωστές σήμερα, καθώς επίσης και στις τεχνολογίες που απαιτήθηκαν για τον σκοπό αυτό. Τα ρομπότ οικιακής χρήσης αποτελούν αυτόματα ρομπότ, που βρήκαν χρήση σε οικιακές δουλειές, όπως στην καθαριότητα, στο μαγείρεμα, στην οργάνωση ή ακόμα στη φροντίδα των ανθρώπων. Η κύρια λειτουργία τους ήταν η αντικατάσταση των ανθρώπων που δεν μπορούσαν ή δεν είχαν τον χρόνο για να εκτελέσουν τις δουλειές αυτές μόνοι τους. Επίσης, ένας λόγος που οδήγησε στην επιτυχία τους, αποτέλεσε και η επιλογή της από άτομα τρίτης ηλικίας, λόγω έλλειψης επικινδυνότητας, η οποία που θα μπορούσε να προκληθεί στην περίπτωση που προτιμούσαν ανθρώπινη βοήθεια για την καθαριότητα του σπιτιού. Τέλος, καθώς προγραμματίζονταν και σχεδιάζοντουσαν για μια συγκεκριμένη εργασία, ήταν και πιο αποτελεσματικές συγκριτικά με τον άνθρωπο, πράγμα που απέφερε οικονομικά οφέλη. Όπως δείχνουν και έρευνες στα άρθρα [1] και [2], η ένταξη των αυτόματων σκουπών στην ζωή των ανθρώπων κάθε ηλικίας κάνει αισθητή την διαφορά και προτιμάται από τους χρήστες τους.

2.2 Κυριότερα μοντέλα

Αν και σχετικά καινούργια και άγνωστη για την καθημερινότητα των ανθρώπων, η ιδέα της ρομποτικής / αυτόματης σκούπας ξεκίνησε από το 1956 [3]. Ήταν πατέντα του Donald G. Moore, Αμερικάνου μηχανικού της Whirlpool, η ιδέα του οποίου αποτελεί υλοποιήσιμη βάση που εφαρμόζεται στις

Κεφάλαιο 2

σημερινές ρομποτικές σκούπες. Η συσκευή του ελεγχόταν από ένα κουμπί που έστελνε την σκούπα σε ένα προκαθορισμένο μονοπάτι οδήγησης για τον καθαρισμό ενός δωματίου. Αξιοσημείωτο είναι το γεγονός ότι είχε μια συσκευή ρομπότ για το σκούπισμα, αλλά και για το σφουγγάρισμα, πράγμα που άργησε να υλοποιηθεί και να ενταχθεί στις συσκευές. Η πρώτη παρουσίαση έγινε το 1957, σε ένα περιοδικό τεύχος της τότε Whirlpool RCA, το The RCA Whirlpool Miracle Kitchen, που διαφημίστηκε με την ονομασία Automatic Floor Cleaner. Το πρόβλημά όμως ήταν ότι ποτέ δεν κυκλοφόρησε στην αγορά.



Σχήμα 2.1. Whirlpool Automatic Floor Cleaner

Το 1996 διαφημίστηκε για πρώτη φορά στην τηλεόραση από την Electrolux το πρωτότυπο της συσκευής της, Trilobite. Στην αγορά κυκλοφόρησε το 2001 όπου ήταν η πρώτη διαθέσιμη αυτόματη σκούπα. Μπορούσε να κινηθεί κάτω από τα περισσότερα κρεβάτια, τραπέζια και άλλα έπιπλα καθώς είχε ύψος μόλις 13 εκατοστά και διάμετρο 35 εκατοστά. Χρησιμοποιούσε υπέρηχους αισθητήρες για την πλοήγηση ενώ είχε και την δυνατότητα χαρτογράφησης του χώρου. Οι αισθητήρες της επέτρεπαν να σταματήσει σε απόσταση 2,54 εκατοστών μακριά από τα αντικείμενα χωρίς να συγκρουστεί, με μεγάλη αξιοπιστία και συχνότητα αλλά μερικές φορές αποτύγχανε, όταν το αντικείμενο ήταν πολύ μικρό ή είχε αιχμηρή γωνία. Λόγω της μικρής απόστασης, η σκούπα άφηνε σημεία μη καθαρισμένα γύρω από τα αντικείμενα και τους τοίχους του χώρου. Η συσκευή γυρνούσε αυτόματα στον σταθμό φόρτισης, όταν τελείωνε η μπαταρία ή όταν τελείωνε την εργασία της. Εάν δεν είχε ολοκληρωθεί ο καθαρισμός και χρειαζόνταν φόρτιση, τότε επέστρεφε στη βάση της και ολοκλήρωνε τον καθαρισμό μετά, αφού είχε φορτιστεί πλήρως [4],[5].



Σχήμα 2.2. Electrolux Trilobite

Το 2001 έγινε μια προσπάθεια της Dyson να μπει στην αγορά των ρομποτικών σκουπών με το μοντέλο της DC06. Η συσκευή μπορούσε να καθαρίσει κάθε είδος δωματίου που θα εισέρχονταν, καθώς δεν βασιζόταν σε προκαθορισμένη πορεία, αλλά αντιθέτως μπορούσε να παρατηρήσει τον χώρο και να οργανώσει μια διαδρομή που θα περνούσε απ' όλα τα σημεία. Η κίνησή της ήταν ελικοειδής προς τα μέσα, δηλαδή ξεκινούσε από μια άκρη του δωματίου και πόρευε προς το κέντρο, χωρίς πολλές επαναλήψεις στην διαδρομή και αποφεύγοντας τα αντικείμενα και τα έπιπλα. Για όλα τα παραπάνω όμως, η συσκευή χρειάστηκε πάνω από 50 αισθητήρες και 3 μικροεπεξεργαστές, οι οποίοι είχαν την δυνατότητα λήψης 16 αποφάσεων το δευτερόλεπτο. Χάριν τους αισθητήρες, υπήρχε η δυνατότητα αναγνώρισης κινούμενου κινδύνου, παραδείγματος χάριν, κατοικίδια ή παιδιά, αλλά και σταθερού κινδύνου όπως οι σκάλες ή ένα κενό στο πάτωμα. Άλλη δυνατότητα ήταν επίσης και η επιλογή μεταξύ των δύο ταχυτήτων, μια των 0,25 μέτρων ανά δευτερόλεπτο, για αργό και σχολαστικό καθάρισμα και μια των 0,45 μέτρων ανά δευτερόλεπτο για ένα γρήγορο καθάρισμα. Τέλος, η συσκευή διέθετε και υποδοχή για σωλήνα όπου με την βοήθεια ανθρώπων θα γινόταν και χειροκίνητη καθαριότητα σε δυσπρόσβατες για την σκούπα περιοχές. Οι μηχανικοί της Dyson μετά από την έρευνα περί της υλοποίησης της σκούπας, αποφάσισαν να μην κυκλοφορήσει στην αγορά καθώς έπρεπε να βελτιώσουν ορισμένα κομμάτια στην πολυπλοκότητα της κατασκευής. Αυτά αποτέλεσαν οι πολλοί αισθητήρες και το μέγεθος της κατασκευής, καθώς ήταν αρκετά ογκώδης και δεν μπορούσε να καθαρίσει κάτω από τα περισσότερα έπιπλα [6], [7], [8], [9].



Σχήμα 2.3. Dyson DC-06

Η πρώτη επιτυχία στην αγορά ήρθε το 2002 από την iRobot και την συσκευή της, Roomba. Η σκούπα διέθετε στο άνω μέρος δύο υπέρηχους αισθητήρες για εντοπισμό αντικειμένων, έναν αισθητήρα απόστασης από τον τοίχο, τοποθετημένος στο δεξιό μέρος της συσκευής, έναν αισθητήρα στο μπροστινό μέρος για την αναγνώριση του εικονικού τοίχου και τέσσερις υπέρυθρους αισθητήρες για τον εντοπισμό γκρεμού, σκαλών ή λακούβων. Στην κυκλοφορία βγήκαν τρεις εκδόσεις της συσκευής, η iRobot Roomba, η iRobot Roomba Pro και η iRobot Roomba Pro Elite, δίχως κάποια αλλαγή στον τρόπο κατασκευής τους. Όλες είχαν διάμετρο 34 εκατοστά και ύψος 9,1 εκατοστά, ενώ και οι τρεις πρόσφεραν τα ίδια χαρακτηριστικά και λειτουργίες καθαρίσματος. Η έκδοση Pro πρόσφερε συν μια λειτουργία για συγκεντρωμένο καθαρισμό σε περιπτώσεις όπου χρειαζόταν σχολαστικό καθάρισμα σε ένα σημείο του χώρου. Η έκδοση Pro Elite πρόσφερε την λειτουργία σχολαστικού καθαρισμού, συν μια έξτρα λειτουργία για βαθύ καθαρισμό, με την οποία, η σκούπα θα διένυε πολλές φορές το ίδιο δωμάτιο. Και στις τρεις εκδόσεις, η επιλογή της λειτουργίας γινόταν μέσω ενός πάνελ στο άνω μέρος της συσκευής. Εκεί βρίσκεται το μενού με τα κουμπιά που επιλέγουν τις λειτουργίες S για δωμάτια 3.05×3.66 μέτρα, M για δωμάτια 4.27×4.88 μέτρα και L για δωμάτια 4.57×6.01 μέτρα. Στις εκδόσεις όπου υπάρχουν οι έξτρα λειτουργίες, υπάρχουν κι έξτρα κουμπιά, συγκεκριμένα, ένα SPOT για σχολαστικό καθάρισμα κι ένα MAX για το βαθύ καθαρισμό [10], [11], [12], [13], [14].



Σχήμα 2.4. iRobot Roomba (αριστερά), Roomba Pro (κέντρο), Roomba Pro Elite (δεξιά)

Πέρα από μερικές καινούργιες κυκλοφορίες, η επόμενη επιτυχία έγινε το 2010 από την Neato με το μοντέλο της XV-11. Η καινοτομία και αυτό που το ξεχώρισε από τα υπόλοιπα ήταν το καινούργιο σύστημα πλοήγησης που δημιούργησε η Neato με τον πρωτοποριακό αισθητήρα laser, γυρνούσε και σκάνανε σε 360° τον χώρο δημιουργώντας ένα χάρτη του δωματίου. Ξεκινώντας από την εξωτερική εμφάνιση, η XV-11 ξεχώρισε από τις υπόλοιπες συσκευές λόγω του σχήματός της, όπως φαίνεται στο Σχήμα 2.5, είναι σαν το αγγλικό γράμμα D και όχι η κλασική στρογγυλή εμφάνιση. Από αισθητήρες, εκτός από laser διαθέτει στο μπροστινό μέρος της έναν αισθητήρα πίεσης και από κάτω, μπροστά από τις ρόδες, 2 αισθητήρες υπερύθρων για τον εντοπισμό γκρεμού, σκαλών ή λακούβων. Διαθέτει δύο λειτουργίες καθαρίσματος, μια κανονική λειτουργία όπου θα περάσει μια φορά το δωμάτιο σκανάροντας και χαρτογραφώντας το και στην συνέχεια, ξαναπερνάει σχολαστικά από κάθε μεριά, καθαρίζοντάς την. Η δεύτερη λειτουργία μπορούσε να χρησιμοποιηθεί για τοπικό καθαίρισμα ενός χώρου 1.22×1.83 μέτρα, [15], [16], [17], [18].



Σχήμα 2.5. Neato XV-11

2.3 Επίλογος

Στο παραπάνω Κεφάλαιο έγινε μια ιστορική αναδρομή στην εξέλιξη της αυτόματης σκούπας, όπως είναι γνωστή τη σημερινή εποχή. Ξεκίνησε η αναφορά από την πρώτη της εμφάνιση ως πατέντα και αναφέρθηκαν τα κυριότερα τεχνολογικά επιτεύγματα όπου αναπτύχθηκαν για την κατάληξή της στον τρόπο όπου λειτουργούν τα τελευταία μοντέλα της αγοράς.

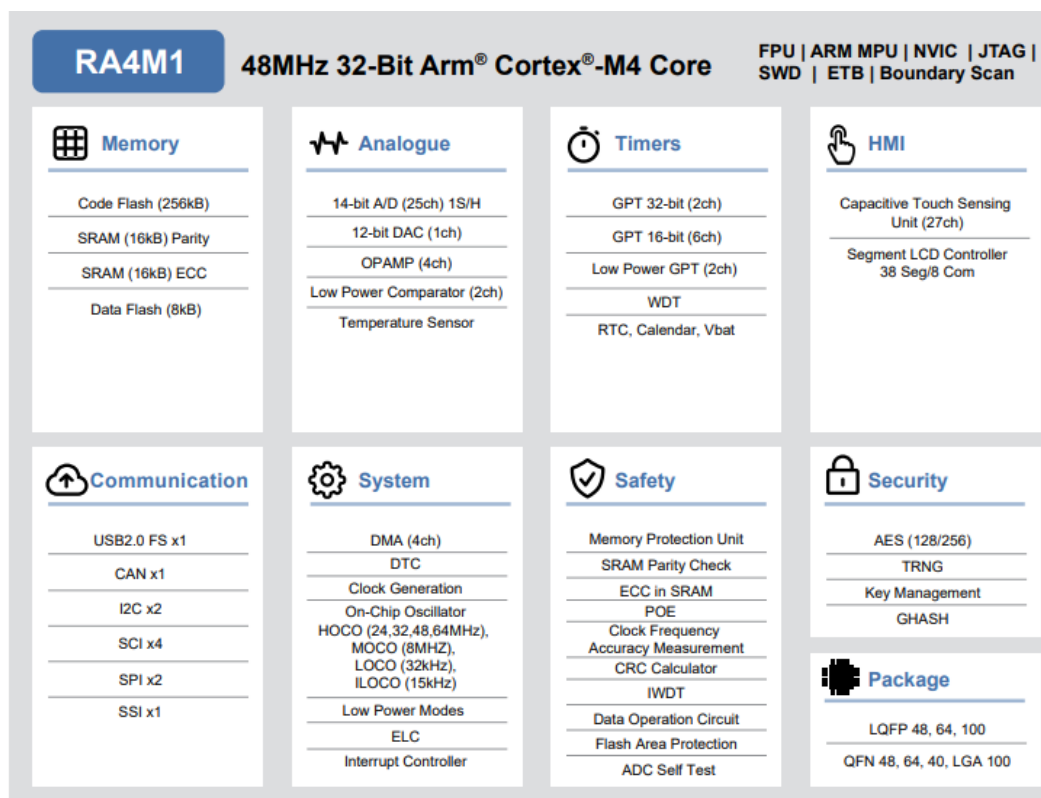
Κεφάλαιο 3ο: Υλικά

3.1 Εισαγωγή

Στο παρόν κεφάλαιο έγινε η παρουσίαση όλων των υλικών όπου χρησιμοποιήθηκαν για την κατασκευή της αυτόματης σκούπας. Αναφέρθηκαν οι δυνατότητες, τα χαρακτηριστικά και οι διάφορες λειτουργίες που διαθέτουν.

3.2 Renesas RA4M1

Ένα πολύ σημαντικό κομμάτι μιας κατασκευής, αν όχι το πιο σημαντικό, είναι ο μικροεπεξεργαστής. Στο παρόν υποκεφάλαιο αναλύθηκε το ολοκληρωμένο κύκλωμα της Renesas με όνομα RA4M1 το οποίο χρησιμοποιήθηκε και για την υλοποίηση της κατασκευής. Στον πυρήνα του διαθέτει τον μικροεπεξεργαστή Arm® Cortex®-M4, ενώ προσφέρει ενσωματωμένη μνήμη και διάφορες χρήσιμες λειτουργίες, όπως αναλογικούς σε ψηφιακούς μετατροπείς (Analog to Digital Converter), ψηφιακούς σε αναλογικούς μετατροπείς (Digital to Analog Converter), ενισχυτές, χρονιστές και υποστηρίζει σύνδεση χωρητικής μονάδας ανίχνευσης αφής (Capacitive Touch Sensing Unit) και ελεγκτή LCD επτά τμημάτων. Επίσης, περιλαμβάνει μονάδα σύνδεσης USB 2.0 πλήρης ταχύτητας, διεπαφή σειριακής επικοινωνίας (Serial Communications Interface) και διεπαφή διαύλου I²C (Inter-Integrated Circuit bus interface). Μπορεί να ανιχνεύσει και να διορθώσει λάθη ενός ψηφίου και να εντοπίσει λάθη δύο ψηφίων, διαθέτει κύκλωμα ανίχνευσης ακρίβειας της συχνότητας του ρολογιού (Clock Frequency Accuracy Measurement Circuit) και κύκλωμα χειρισμού δεδομένων (Data Operation Circuit) [19].

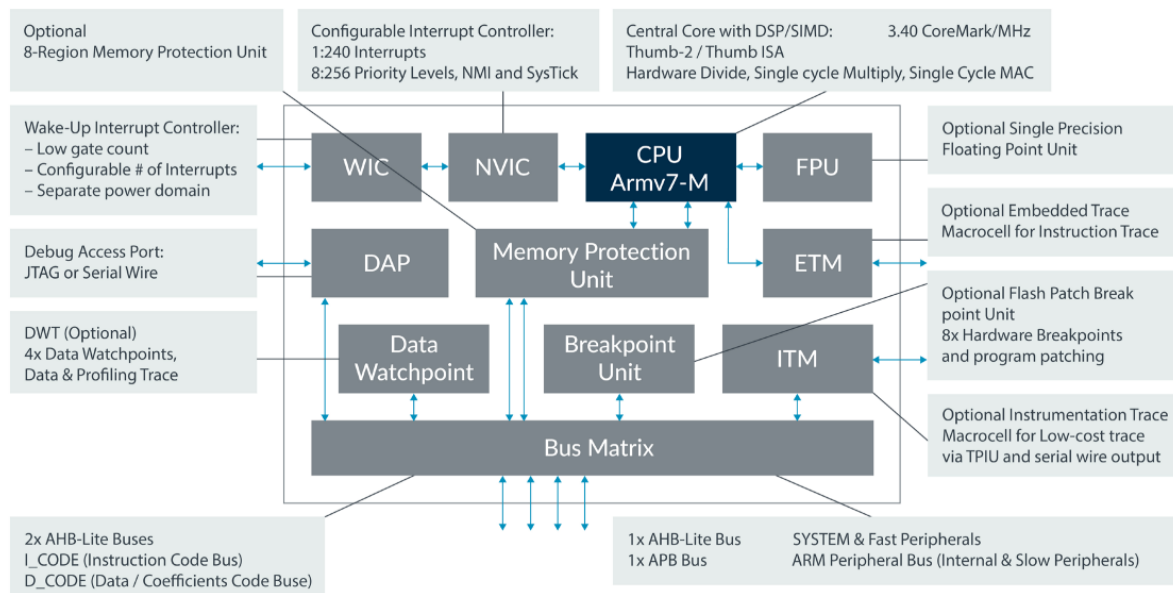


Σχήμα 3.1. Χαρακτηριστικά και λειτουργίες που διαθέτει το ολοκληρωμένο κύκλωμα RA4M1 της Renesas

3.2.1 Arm® Cortex®-M4

Πιο συγκεκριμένα, ο μικροεπεξεργαστής της Arm® Cortex®-M4 είναι ένας επεξεργαστής χαμηλής κατανάλωσης ενέργειας που διαθέτει χαμηλό αριθμό πυλών, χαμηλή καθυστέρηση διακοπών, χαμηλό κόστος αποσφαλμάτωση και περιλαμβάνει λειτουργίες αριθμητικής κινητής υποδιαστολής, μέσω της μονάδας κινητής υποδιαστολής (Floating-Point Unit). Ο επεξεργαστής περιέχει τον πυρήνα του, έναν ελεγκτή διακοπών με διανυσματικά στοιχεία (Nested Vectored Interrupt Controller), ο οποίος είναι

στενά ενσωματωμένος για την επίτευξη της επεξεργασίας διακοπών με πολύ χαμηλή καθυστέρηση, μία μονάδα προστασίας μνήμης (Memory Protection Unit) και πολλαπλές διεπαφές διαύλου υψηλής απόδοσης [20].



Σχήμα 3.2. Λειτουργίες του μικροεπεξεργαστή Cortex M4

3.2.1.1 Επεξεργαστής

Τα τμήματα του πυρήνα προβάλλονται στο Σχήμα 3.2. Ξεκινώντας από τον επεξεργαστή συχνότητας 48MHz με 32bit που σχεδιάστηκε με την αρχιτεκτονική της Armv7-M και χρησιμοποιεί ένα υποσύνολο εντολών Thumb, Thumb-2 και DSP. Είναι ικανός να διαχειρίζεται πολλαπλές διεργασίες παράλληλα λόγω της χρήσης του δείκτη αποθηκευμένων στοιβάδων (Banked Stack Pointer). Η κάθε διεργασία συσχετίζεται με μια συγκεκριμένη λειτουργία εκτέλεσης ή επίπεδο εξαίρεσης. Αυτό επιτρέπει στον επεξεργαστή να διαχειρίζεται αποτελεσματικά τις στοίβες σε διαφορετικά περιβάλλοντα, απλοποιώντας την εναλλαγή τους και βελτιώνοντας τη σταθερότητα του συστήματος. Μπορεί να εκτελέσει εντολές διαίρεσης, είτε με πρόσημο, είτε χωρίς, με τη χρήση εξειδικευμένου υλικού βελτιώνοντας την απόδοση, την ακρίβεια και τη ταχύτητα υλοποίησης της διαίρεσης ακεραίων. Διαθέτει δύο βασικές λειτουργίες, την λειτουργία νήματος και την λειτουργία χειρισμού (Thread and Handler). Η πρώτη χρησιμοποιείται για την εκτέλεση εφαρμογών επιπέδου χρήστη με μειωμένα προνόμια και περιορισμένη πρόσβαση στους πόρους του συστήματος, ενώ η λειτουργία χειρισμού προορίζεται για την εκτέλεση προνομιακού, σημαντικού, κώδικα συστήματος και το χειρισμό εξαιρέσεων με πρόσβαση σε όλους τους πόρους που διαθέτει το σύστημα. Η αρχιτεκτονική του συστήματος επιτρέπει στον επεξεργαστή να αποθηκεύει και να επαναφέρει αυτόματα την κατάσταση όπου βρισκόταν πριν την εμφάνιση κάποιας ρουτίνας εξυπηρέτησης διακοπής. Αυτό το χαρακτηριστικό επιτρέπει στον επεξεργαστή να εναλλάσσει γρήγορα το πλαίσιο μεταξύ του τρέχοντος κώδικα που εκτελείται και του χειριστή διακοπής, διασφαλίζοντας παράλληλα την ακεραιότητα της κατάστασης του κώδικα που έχει διακοπεί. Υποστηρίζει εντολές με διακοπές, όπως η πολλαπλή αποθήκευση και η πολλαπλή φόρτωση, για αποθήκευση και φόρτωση

καταχωρητών σε έναν κύκλο λειτουργίας. Ένα άλλο είδος εντολών με δυνατότητα διακοπής είναι η προσθήκη και η αφαίρεση, που χρησιμοποιούνται για τον χειρισμό των στοιβών.

3.2.1.2 Μονάδα κινητής υποδιαστολής

Η μονάδα κινητής υποδιαστολής είναι μία περιοχή υλικού εντός του επεξεργαστή, η οποία έχει σχεδιαστεί ειδικά για να εκτελεί αριθμητικές πράξεις κινητής υποδιαστολής. Μία τέτοια μονάδα εξειδικεύεται στην υλοποίηση πράξεων με κλασματικά μέρη και με πολύ μεγάλους ή με πολύ μικρούς αριθμούς, που δεν μπορούν να αναπαρασταθούν με ακρίβεια μόνο με την αριθμητική σταθερής υποδιαστολής. Χρησιμοποιεί κωδικοποιημένες, έως 32-bit μήκος, εντολές όπου εκτελούν μια συγκεκριμένη αριθμητική πράξη, όπως πρόσθεση, αφαίρεση, πολλαπλασιασμό, διαίρεση ή άλλες μαθηματικές λειτουργίες, σε αριθμούς κινητής υποδιαστολής μονής ακρίβειας. Αυτές οι εντολές παρέχουν αποδοτικές και ταχύτερες υπολογιστικές δυνατότητες για πράξεις κινητής υποδιαστολής σε εφαρμογές όπως οι επιστημονικοί υπολογισμοί, η ψηφιακή επεξεργασία σήματος και η απόδοση γραφικών. Διαθέτει υλικό υποστήριξης για χειρισμό μη κανονικοποιημένων αριθμών και τρόπους στρογγυλοποίησης σύμφωνα με το πρότυπο IEEE 754, εξασφαλίζοντας ακριβή και συνεπή αριθμητική. Περιλαμβάνει 32 καταχωρητές σχεδιασμένους για πράξεις κινητής υποδιαστολής μονής ακρίβειας, ενώ παράλληλα προσφέρει την επιλογή για την προσπέλασή τους ως ζεύγη των 16. Η χρήση τους ως 32-bit καταχωρητές γίνεται συνήθως για την αποθήκευση ενδιάμεσων αποτελεσμάτων, τελεστών και άλλων δεδομένων κατά την διάρκεια αριθμητικών υπολογιστικών πράξεων. Τα ζεύγη των καταχωρητών προσδίδουν ευελιξία στην πρόσβαση και τον χειρισμό των δεδομένων, ιδίως όταν πρόκειται για μεγαλύτερου τύπου δεδομένων ή εκτελούνται λειτουργίες μίας εντολής με πολλά δεδομένα εισόδου. Η μονάδα κινητής υποδιαστολής χειρίζεται τη διαδικασία εκτέλεσης των εντολών σε τρία διακριτά στάδια, την ανάκληση, την αποκωδικοποίηση και την εκτέλεση των εντολών, τα οποία έχουν τη δυνατότητα να λειτουργούν ανεξάρτητα μεταξύ τους για τη βελτίωση της απόδοσης και της ροής του προγράμματος. Κάθε στάδιο της εκτέλεσης επεξεργάζεται ταυτόχρονα μια διαφορετική εντολή, επιτρέποντας την παράλληλη επεξεργασία πολλαπλών εντολών βελτιώνοντας τη συνολική απόδοση.

3.2.1.3 Ελεγκτής διακοπών με διανυσματικά στοιχεία

Ένα κομμάτι του επεξεργαστή κοντά στον πυρήνα του είναι αφοσιωμένο στην διαχείριση των διαφόρων διακοπών. Η βασική δουλειά του είναι να συντονίζει τον χειρισμό των διακοπών από τις πηγές όπως τα διάφορα περιφερειακά, οι χρονιστές και τα υπόλοιπα εξωτερικά συμβάντα που πιθανό να προκύψουν. Καθήκον του είναι η αποθήκευση της κατάστασης του προγράμματος πριν την έναρξη της εκτέλεσης διακοπής, η εξασφάλιση ότι οι διακοπές εξυπηρετούνται εγκαίρως και με την σωστή σειρά προτεραιότητας και μετά την λήξη τους, η ομαλή επαναφορά στη ροή του προγράμματος. Η σειρά προτεραιότητας διαμορφώνονται συνήθως από το λογισμικό ενώ μπορούν να προσαρμόζονται και δυναμικά για τη βελτιστοποίηση της απόκρισης του συστήματος σε διάφορα συμβάντα. Υποστηρίζει εμφωλευμένες διακοπές επιτρέποντας στις διακοπές υψηλότερης προτεραιότητας να προηγούνται των χαμηλότερων όταν είναι απαραίτητο. Χρησιμοποιεί έναν πίνακα διανυσμάτων διακοπής για να αποθηκεύει τις διευθύνσεις των ρουτινών εξυπηρέτησης για κάθε πηγή διακοπής. Όταν συμβαίνει μια διακοπή, συμβουλευεται τον πίνακα για να προσδιορίσει τη διεύθυνσή της και μεταβαίνει σε αυτήν για να ξεκινήσει την επεξεργασία της διακοπής. Ο συγκεκριμένος εκλεκτής μπορεί να διαχειρίζεται από 1 μέχρι 240 διαφορετικές εξωτερικές διακοπές, όπου μπορεί να είναι περιφερειακές συσκευές, αισθητήρες ή εξωτερικά συμβάντα, τα οποία μπορούν να ενεργοποιήσουν τον μικροελεγκτή για να

εκτελέσει συγκεκριμένες ενέργειες ή να χειριστεί συμβάντα. Μπορεί να ρυθμίσει από 0 μέχρι και 255 διαφορετικά επίπεδα προτεραιότητας καθώς χρησιμοποιεί από 3 έως και 8 bit για την αναπαράσταση. Πέρα από ξεχωριστό αριθμό προτεραιότητας, είναι εφικτό να ομαδοποιηθούν μερικές διακοπές μαζί και να υπάρξει διαχωρισμός σε ομάδες, όπου σε κάθε ομάδα γίνεται η ταξινόμηση των διακοπών.

3.2.1.4 Μονάδα προστασίας μνήμης

Η μονάδα προστασίας μνήμης είναι ένα υλικό στοιχείο που παρέχει λειτουργίες ελέγχου πρόσβασης στη μνήμη και ασφάλειας σωστής λειτουργίας. Η κύρια χρήση της είναι να επιβάλλει την προστασία της μνήμης, περιορίζοντας την πρόσβαση σε συγκεκριμένες περιοχές μνήμης με βάση διαμορφώσιμους κανόνες και δικαιώματα. Χωρίζει τη μνήμη σε 8 περιοχές ή ζώνες και επιτρέπει στους προγραμματιστές να καθορίζουν δικαιώματα πρόσβασης όπως η ανάγνωση, η εγγραφή και η εκτέλεση για κάθε περιοχή ξεχωριστά. Κάθε περιοχή μπορεί να αντιπροσωπεύει ένα ξεχωριστό τμήμα της μνήμης, όπως μνήμη για αποθήκευση κώδικα, μνήμη δεδομένων, περιφερειακή μνήμη ή μνήμη συστήματος. Καθορίζοντας τους κανόνες πρόσβασης και τα δικαιώματα, μπορεί να αποτραπεί η μη εξουσιοδοτημένη πρόσβαση σε κρίσιμους πόρους του συστήματος και ευαίσθητα δεδομένα, ενισχύοντας την ασφάλεια και την αξιοπιστία του συστήματος. Στις περιπτώσεις παραβίαση πρόσβασης στη μνήμη, όπως προσπάθεια πρόσβασης σε προστατευόμενη περιοχή μνήμης χωρίς τα κατάλληλα δικαιώματα, η μονάδα προστασίας μνήμης παράγει σφάλμα διαχείρισης μνήμης. Τα σφάλματα αυτά συνήθως αντιμετωπίζονται από τον μηχανισμό χειρισμού εξαιρέσεων του μικροελεγκτή, επιτρέποντας στο σύστημα να αντιδράσει στις παραβιάσεις και να λάβει τα κατάλληλα μέτρα, όπως ο τερματισμός της παραβατικής διεργασίας ή η ενεργοποίηση ενός χειριστή σφαλμάτων. Ένας άλλος μηχανισμός είναι απενεργοποίηση υπο περιοχών όπου επιτρέπει τον λεπτομερή έλεγχο της προστασίας της μνήμης με την ενεργοποίηση ή την απενεργοποίηση συγκεκριμένων υπο περιοχών εντός μιας ευρύτερης περιοχής μνήμης. Το χαρακτηριστικό αυτό βελτιώνει την αποδοτικότητα της χρήσης της μνήμης, διατηρώντας παράλληλα την ασφάλεια και τον έλεγχο πρόσβασης.

3.2.1.5 Μονάδα διασύνδεσης πολλαπλής ενεργοποίησης (Cross Trigger Interface Unit)

Η μονάδα διασύνδεσης πολλαπλής ενεργοποίησης κατέχει καθοριστικό ρόλο στην αλληλεπίδραση μεταξύ της λογικής αποσφαλμάτωσης, όπως η θύρα πρόσβασης αποσφαλμάτωσης ή η θύρα αποσφαλμάτωσης και ανίχνευσης, και άλλων στοιχείων του προγράμματος όπως της ενσωματωμένη μακροκυψέλη ιχνών (Embedded Trace Macrocell). Τα στοιχεία της λογικής αποσφαλμάτωσης συχνά ελέγχουν τις λειτουργίες αποσφαλμάτωσης και ανίχνευσης του μικροελεγκτή, συμπεριλαμβανομένου του καθορισμού σημείων διακοπής, της διακοπής πυρήνων και της πρόσβασης σε καταχωρητές πυρήνων. Μέσω της διασύνδεσης πολλαπλής ενεργοποίησης, η λογική αποσφαλμάτωσης μπορεί να ενεργοποιεί συμβάντα ή ενέργειες με βάση συγκεκριμένες συνθήκες που εντοπίζονται κατά την ανάλυση δεδομένων ιχνολάτησης, όπως ο εντοπισμός σημείων διακοπής ή σημείων παρακολούθησης επιτρέποντας συντονισμένες δραστηριότητες αποσφαλμάτωσης και ανίχνευσης σε όλο το σύστημα. Με την αλληλεπίδραση της μονάδα διασύνδεσης πολλαπλής ενεργοποίησης και της ενσωματωμένη μακροκυψέλη ιχνών, η δεύτερη, μπορεί να λαμβάνει σήματα ενεργοποίησης, να ελέγχει τις ενέργειες αποσφαλμάτωσης και να συγχρονίζει τη σύλληψη δεδομένων ιχνών με άλλες δραστηριότητες αποσφαλμάτωσης, βελτιώνοντας την αποτελεσματικότητα και την αποδοτικότητα της ανάλυσης του συστήματος. Η διαμόρφωση της μονάδα διασύνδεσης πολλαπλής ενεργοποίησης είναι συνήθως προσαρμόσιμη, επιτρέποντας στους προγραμματιστές να ορίζουν πηγές ενεργοποίησης, συνθήκες και

ενέργειες με βάση τις συγκεκριμένες απαιτήσεις αποσφαλμάτωσης και ανίχνευσης των εφαρμογών τους. Οι διαμορφώσιμοι παράμετροι μπορεί να περιλαμβάνουν εισόδους σκανδαλισμού, εξόδους σκανδαλισμού, δρομολόγηση σκανδαλισμού, επίπεδα προτεραιότητας και μηχανισμούς φιλτραρίσματος για την προσαρμογή της συμπεριφοράς της στην αρχιτεκτονική του συστήματος και τους στόχους σχεδιασμού.

3.2.1.6 Αποσφαλμάτωση και ανίχνευση

Η αποσφαλμάτωση και η ανίχνευση είναι βασικές λειτουργίες που παρέχονται από τις αρχιτεκτονικές μικροελεγκτών για την υποστήριξη της ανάπτυξης λογισμικού, της αποσφαλμάτωσης συστημάτων, της ανάλυσης επιδόσεων και της βελτιστοποίησης τους. Αυτές οι δυνατότητες είναι ζωτικής σημασίας για την κατανόηση της συμπεριφοράς των συστημάτων, τη διάγνωση προβλημάτων και τη βελτίωση της συνολικής αξιοπιστίας και της απόδοσης. Η αποσφαλμάτωση γίνεται με διάφορους τρόπους, ένας από αυτούς είναι με τα σημεία διακοπής. Είναι σημεία στον κώδικα όπου η εκτέλεση σταματά, επιτρέποντας στους προγραμματιστές να ελέγξουν την κατάσταση του συστήματος και να αναλύσουν τη συμπεριφορά του προγράμματος. Ένα άλλο είδος είναι τα σημεία παρακολούθησης όπου στην ουσία πρόκειται για ενεργοποιητές που σταματούν την εκτέλεση όταν προσπελούνται ή τροποποιούνται ορισμένες θέσεις μνήμης, επιτρέποντας την ανίχνευση ζητημάτων που σχετίζονται με τη μνήμη. Πιο εξειδικευμένος τρόπος είναι η μονή εκτέλεση, όπου επιτρέπει στους προγραμματιστές να εκτελούν τον κώδικα μία εντολή τη φορά, διευκολύνοντας τη λεπτομερή επιθεώρηση της ροής του προγράμματος και των τιμών των μεταβλητών. Τέλος γίνεται και με την πρόσβαση και την τροποποίηση των καταχωρητών του επεξεργαστή, επιτρέποντας στους προγραμματιστές να επιθεωρούν και να χειρίζονται την κατάσταση του επεξεργαστή κατά τη διάρκεια συνεδριών εντοπισμού σφαλμάτων. Η ανίχνευση παρέχει την ορατότητα στη συμπεριφορά εκτέλεσης των εντολών στο σύστημα συλλέγοντας και καταγράφοντας λεπτομερείς πληροφορίες κατά τη διάρκεια της εκτέλεσης. Η ανίχνευση γίνεται συνήθως καταγράφοντας τα ίχνη εκτέλεσης εντολών, τα ίχνη πρόσβασης σε δεδομένα και τα ίχνη συμβάντων. Ξεκινώντας από τα ίχνη εντολών, καταγράφουν την ακολουθία των εντολών που εκτελούνται από τον επεξεργαστή, επιτρέποντας την ανάλυση της ροής του προγράμματος και των διαδρομών εκτέλεσης. Στην συνέχεια υπάρχουν και τα ίχνη πρόσβασης σε δεδομένα όπου καταγράφουν λειτουργίες πρόσβασης στη μνήμη, όπως αναγνώσεις και εγγραφές, διευκολύνοντας την ανάλυση των εξαρτημένων δεδομένων και των προτύπων χρήσης της μνήμης. Τέλος, τα ίχνη συμβάντων καταγράφουν συμβάντα συστήματος, διακοπές, εξαιρέσεις και άλλα ασύγχρονα γεγονότα, βοηθώντας τους προγραμματιστές να κατανοήσουν τη συμπεριφορά και την απόδοση του συστήματος. Οι λειτουργίες αποσφαλμάτωσης και ιχνηλάτησης ενσωματώνονται συνήθως σε εργαλεία αποσφαλμάτωσης, όπως ολοκληρωμένα περιβάλλοντα ανάπτυξης, ανιχνευτές αποσφαλμάτωσης και υλικό καταγραφής ιχνών. Τα δεδομένα ιχνών είναι πολύτιμα για την ανάλυση επιδόσεων, τη βελτιστοποίηση, τη σκιαγράφηση και την αντιμετώπιση προβλημάτων ενσωματωμένων εφαρμογών, ιδίως σε συστήματα με κρίσιμες επιδόσεις σε πραγματικό χρόνο. Τα δεδομένα τους βασίζονται σε ειδικά στοιχεία υλικού, όπως η θύρα πρόσβασης αποσφαλμάτωσης, η ενσωματωμένη μακροκυψέλη ιχνών και η μονάδα διασύνδεσης θύρας ιχνηλάτησης (Trace Port Interface Unit). Η θύρα αποσφαλμάτωσης με σειριακό καλώδιο (Serial Wire Debug Port) και η θύρα αποσφαλμάτωσης JTAG (Serial Wire JTAG Debug Port) είναι δύο κοινές διεπαφές που χρησιμοποιούνται σε αρχιτεκτονικές μικροελεγκτών για τη διευκόλυνση του εντοπισμού σφαλμάτων και του προγραμματισμού ενσωματωμένων συστημάτων. Η καταγραφή των ιχνών γίνεται μέσω της διεπαφής ενσωματωμένης μακροκυψέλης ιχνών όπου συλλέγει λεπτομερείς πληροφορίες σχετικά με τη ροή εκτέλεσης του προγράμματος, επιτρέποντας στους

προγραμματιστές να αποκτήσουν εικόνα της συμπεριφοράς του συστήματος, να εντοπίσουν τα σημεία συμφόρησης και να βελτιστοποιήσουν την απόδοση του λογισμικού. Τα εργαλεία εντοπισμού σφαλμάτων παρέχουν μια, φιλική προς το χρήστη, διεπαφή για τους προγραμματιστές ώστε να αλληλοεπιδρούν με τον μικροελεγκτή, να θέτουν σημεία διακοπής, να διαμορφώνουν σημεία παρακολούθησης, να αναλύουν δεδομένα ιχνών και να αποσφαλματώνουν προβλήματα λογισμικού. Τα δεδομένα ιχνών που λαμβάνονται κατά τη διάρκεια του χρόνου εκτέλεσης συχνά οπτικοποιούνται με τη χρήση εξειδικευμένων εργαλείων ανάλυσης, επιτρέποντας στους προγραμματιστές να εντοπίζουν σημεία συμφόρησης, να βελτιστοποιούν τον κώδικα και να βελτιώνουν την απόδοση του συστήματος.

3.2.1.7 Δίαυλοι διεπαφών

Οι διεπαφές διαύλου είναι κρίσιμα στοιχεία στην αρχιτεκτονική μικροελεγκτών καθώς επιτρέπουν την επικοινωνία μεταξύ των διαφόρων τμημάτων του συστήματος, συμπεριλαμβανομένων των επεξεργαστών, της μνήμης, και των περιφερειακών συσκευών. Αυτές οι διεπαφές καθορίζουν τα πρωτόκολλα, τα σήματα και τους μηχανισμούς που χρησιμοποιούνται για τη μεταφορά δεδομένων και τον έλεγχο εντός του συστήματος. Η επιλογή της κατάλληλης διεπαφής εξαρτάται από παράγοντες όπως η ταχύτητα μεταφοράς δεδομένων, η απόσταση, η κατανάλωση ενέργειας και η συμβατότητα με περιφερειακές συσκευές και πρωτόκολλα του συστήματος. Η διεπαφή του συστήματος που περιγράφεται περιέχει 3 προηγμένους διαύλους υψηλής απόδοσης, τον διάδρομο μεταφοράς εντολών, τον διάδρομο μεταφοράς δεδομένων και τον διάδρομο συστήματος. Ο δίαυλος μεταφοράς εντολών είναι υπεύθυνος για τη σύνδεση του πυρήνα του επεξεργαστή με το υποσύστημα μνήμης εντολών. Η δουλειά του είναι η διευκόλυνση της λήψης εντολών από τη μνήμη προς τον πυρήνα του επεξεργαστή κατά την εκτέλεση του προγράμματος. Μεταφέρει συνήθως εντολές ανάγνωσης που ξεκινούν από τον πυρήνα του επεξεργαστή για την ανάκτηση δεδομένων από τη μνήμη εντολών. Ο διάδρομος μεταφοράς δεδομένων συνδέει τον πυρήνα του επεξεργαστή με το υποσύστημα μνήμης δεδομένων. Χειρίζεται συναλλαγές ανάγνωσης και εγγραφής που ξεκινούν από τον πυρήνα του επεξεργαστή για πρόσβαση σε δεδομένα που είναι αποθηκευμένα στη μνήμη. Επιτρέπει στον πυρήνα του επεξεργαστή να εκτελεί πράξεις ανάγνωσης και εγγραφής σε δεδομένα που είναι αποθηκευμένα στη μνήμη δεδομένων. Ο δίαυλος συστήματος χρησιμεύει ως η κύρια διασύνδεση εντός της αρχιτεκτονικής του μικροελεγκτή, συνδέοντας διάφορα στοιχεία όπως επεξεργαστές, μνήμη και περιφερειακά μεταξύ τους. Παρέχει μια δίοδο επικοινωνίας για τη μεταφορά δεδομένων και σημάτων ελέγχου μεταξύ των διαφόρων τμημάτων του συστήματος. Ο δίαυλος συστήματος υποστηρίζει συνήθως μια ποικιλία συναλλαγών, συμπεριλαμβανομένων των μεταφορών ανάγνωσης, εγγραφής και πολλαπλών μεταβιβάσεων, για να διευκολύνει την αποτελεσματική μεταφορά δεδομένων και την πρόσβαση στους πόρους του συστήματος. Ένα άλλο είδος διαδρόμου είναι ο ιδιωτικός περιφερειακός δίαυλος, όπου είναι μια εξειδικευμένη διεπαφή διαδρόμου. Λειτουργεί ως ένα αφιερωμένο κανάλι επικοινωνίας για σύνδεση περιφερειακών συσκευών και του πυρήνα του επεξεργαστή, επιτρέποντας αποτελεσματική πρόσβαση σε πόρους κατά την εκτέλεση προγραμμάτων. Λειτουργεί ως διασύνδεση για την πρόσβαση σε καταχωρητές ελέγχου και τον διαμορφωτή ενότητας που σχετίζονται με λειτουργίες πυρήνα, συμπεριλαμβανομένης της διαχείρισης διακοπών, των ρυθμίσεων ελέγχου συστήματος και των δυνατοτήτων αποσφαλμάτωσης. Με τη χρήση του ιδιωτικού περιφερειακού διαύλου, ο πυρήνας μπορεί να έχει αποδοτική πρόσβαση και έλεγχο σε περιφερειακά ειδικά στοιχεία και καταχωρητές ελέγχου χωρίς να ανταγωνίζεται για εύρος ζώνης με περιφερειακά υψηλής ταχύτητας που συνδέονται με διαδρόμους όπως τους προηγμένους διαύλους υψηλής απόδοσης. Η διεπαφή υποστηρίζει εντολές που εκτελούνται σε επίπεδο ψηφίου με ατομικές λειτουργίες. Οι εντολές αυτές έχουν την δυνατότητα να

χειρίζονται ένα μεμονωμένο ψηφίο σαν να κατείχε μια θέση μνήμης και είναι αδιάκοπες. Στην εγγραφή ενός μεμονωμένου ψηφίου με ατομική εντολή, είναι εγγυημένο ότι θα πραγματοποιηθεί η καταγραφή ή η διαγραφή του ψηφίου χωρίς παρεμβολές ή διακοπές από άλλες λειτουργίες. Ομοίως στην εντολή της ανάγνωσης ψηφίου με ατομική λειτουργία, το σύστημα διασφαλίζει ότι η τιμή επιστρέφεται από την εκτέλεση της εντολής αντικατοπτρίζει την τρέχουσα τιμή του ψηφίου στην θέση μνήμης χωρίς επιρροές από τυχόν άλλες εντολές. Διαθέτει προσωρινή μνήμη εγγραφής όπου χρησιμοποιείται για την προσωρινή διατήρηση των δεδομένων εγγραφής προτού εγγραφούν στην θέση μνήμης προορισμού ή στη συσκευή αποθήκευσης.

3.2.2 Μνήμη

Η μνήμη που συνοδεύει τον επεξεργαστή στο ολοκληρωμένο κύκλωμα RA4M1 της Renesas είναι ένας συνδυασμός από διάφορους τύπους μνήμης.

3.2.2.1 Flash μνήμη

Αρχικά υπάρχει η flash μνήμη των 256kB όπου χρησιμοποιείται για την αποθήκευση του κώδικα που τρέχει στο σύστημα. Αξίζει να σημειωθεί ότι ο χάρτης διευθύνσεων προβάλλει την μνήμη ως 128 τεμάχια των 2kB. Οι πραγματικές διευθύνσεις των θέσεων ανάγνωσης και εγγραφής/διαγραφής είναι από 00000000h μέχρι και την 0003FFFFh. Οι ταχύτητές της για τον χρόνο ανάγνωσης αλλάζουν ανάλογα με το αν υπάρχει η πληροφορία στην προσωρινή μνήμη ή όχι και ανάλογα με την ρύθμιση του εσωτερικού χρονιστή. Στην περίπτωση ο χρονιστής είναι ρυθμισμένος από 32MHz μέχρι και τα 48MHz, αν η πληροφορία προς ανάγνωση υπάρχει στην προσωρινή μνήμη, ο χρόνος εκτέλεσης της εντολής θα είναι 1 κύκλος εργασίας ενώ αν δεν υπάρχει ήδη φορτωμένη η πληροφορία θα χρειαστούν 2-3 κύκλοι για την ανάγνωσή της. Στην περίπτωση όπου ο ρυθμιστής είναι κάτω από 32 MHz, η ανάγνωση θα γίνει σε 1 κύκλο, άσχετα με το αν η πληροφορία υπάρχει ή όχι στην προσωρινή μνήμη. Ίδιο τύπο μνήμης flash υπάρχει και με 8kB χωρητικότητας για την αποθήκευση παραμέτρων διαμόρφωσης, δεδομένων διακρίβωσης και άλλων παραμέτρων. Σε αυτήν οι θέσεις προβάλλονται ως 8 τεμάχια του 1kB με τις διευθύνσεις ανάγνωσης να ξεκινούν από την θέση 4010 0000h μέχρι και την 4010 1FFFh, με την διαφορά ότι οι διευθύνσεις για την εγγραφή ή την διαγραφή να βρίσκονται από FE00 0000h μέχρι και FE00 1FFFh. Η ταχύτητά της παραμένει σταθερή στους 6 κύκλους για την ανάγνωση των δεδομένων της κάθε θέσης. Και στις 2 μνήμες η τιμή τους μετά από την ολική διαγραφή του περιεχομένου τους θα είναι FFh. Διαθέτουν δικό τους τμήμα ελέγχου flash για την αναγνώριση των εντολών και την επικοινωνία τους με τον επεξεργαστή. Υποστηρίζει ανάγνωση, εγγραφή ή διαγραφή δεδομένων από την μνήμη κώδικα παράλληλα με την μνήμη δεδομένων. Η προσωρινή μνήμη μπορεί διαχωριστεί σε 3 υποκατηγορίες, η προσωρινή μνήμη για την εκτέλεση εντολών από τον επεξεργαστή, η προσωρινή μνήμη για την πρόσβαση σε τελεστές του επεξεργαστή και της μονάδας άμεσης πρόσβασης μνήμης και η προσωρινή μνήμη για προανάκληση προσπέλασης εντολών του επεξεργαστή. Και οι 3 υποκατηγορίες στοχεύουν τις διευθύνσεις 00000000h έως και 007FFFFFFh. Η χωρητικότητά τους ποικίλη από 128byte για την εκτέλεση εντολών από τον επεξεργαστή, 8byte για την πρόσβαση σε τελεστές του επεξεργαστή και της μονάδας άμεσης πρόσβασης στη μνήμη και 16byte για την προανάκληση προσπέλασης εντολών του επεξεργαστή.

3.2.2.2 Στατική μνήμη τυχαίας προσπέλασης (SRAM)

Για την στατική μνήμη τυχαίας προσπέλασης το ολοκληρωμένο κύκλωμα διαθέτει 32kB χωρητικότητας όπου χωρίζονται στην μέση για τις 2 λειτουργίες αναγνώρισης λαθών που προσφέρει. Το πρώτο μισό στην σειρά των διευθύνσεων χρησιμοποιείται για την κωδικοποίηση διόρθωσης σφαλμάτων (ECC) και καταλαμβάνει τις διευθύνσεις από 20000000h έως 20003FFFh. Προσθέτει επιπλέον bit σε κάθε λέξη ώστε να είναι εφικτή η παρατήρηση των 2bit λαθών ενώ διορθώνει τα λάθη σε 1bit. Το δεύτερο μισό την μνήμης χρησιμοποιείται τις περιπτώσεις όπου δεν είναι ενεργοποιημένη η κωδικοποίηση διόρθωσης σφαλμάτων και γίνεται χρήση του ελέγχου ισοτιμίας. Η αποθήκευση της πληροφορίας γίνεται από τις διευθύνσεις 2000 4000h έως και 2000 7FFFh. Η τεχνική του ελέγχου ισοτιμίας προσθέτει 1bit ισοτιμίας σε κάθε 8bit δεδομένων για να διασφαλίσει ότι το αποτέλεσμα της άθροισης είναι ζυγός αριθμός.

3.2.3 Συνδεσιμότητα

Η επικοινωνία του επεξεργαστή με τις περιφερειακές συσκευές εισόδου και εξόδου γίνεται με διάφορους τρόπους. Ο προγραμματισμός της συσκευής όπως και η αποστολή δεδομένων από τον επεξεργαστή γίνεται μέσω μιας θήρας USB 2.0 όπου υποστηρίζει μέγιστες ταχύτητες μεταφοράς της πληροφορίας. Περιέχει 5 διεπαφές σειριακής επικοινωνίας, ένα γενικό ασύγχρονο δέκτη / πομπό (Universal Asynchronous Receiver / Transmitter), ένα σύγχρονο γενικό δέκτη/πομπό (Universal Synchronous and Asynchronous Receiver / Transmitter), ένα απλό διασυνδεδεμένο κύκλωμα (simple PC), μία απλή σειριακή περιφερειακή διεπαφή (simple Serial Peripheral Interface) και μία διεπαφή έξυπνης κάρτας. Στην περίπτωση την ασύγχρονης επικοινωνίας, δίνεται η επιλογή του μήκους των δεδομένων από 7, 8 ή 9 bit και η επιλογή ανάμεσα από επαλήθευση με bit ισοτιμίας σε μονό άθροισμα, σε ζυγό άθροισμά ή καθόλου επαλήθευση. Στην σύγχρονη επικοινωνία η πληροφορία μεταδίδεται με μήκος 8 bit σταθερά ενώ δίνεται η επιλογή για την ρύθμιση της πηγής χρόνου είτε από το εσωτερικό ρολόι ως master, είτε από εξωτερική πηγή ως slave. Για το simple PC χρησιμοποιείται το PC πρωτόκολλο επικοινωνίας όπου η πληροφορία στέλνεται πρώτα από το πιο σημαντικό bit και στο τέλος το λιγότερο σημαντικό bit, είναι εφικτή μόνο η χρήση του ως master και επιτυγχάνονται ταχύτητες έως 400Kbps. Στην περίπτωση του simple SPI η πληροφορία μεταφέρεται ανά 8 bit με χρόνο ρολογιού είτε το εσωτερικό ρολόι για λειτουργία ως master είτε με εξωτερικό ρολόι σε λειτουργία slave. Δίνεται η επιλογή για την διαμόρφωση μεταξύ τεσσάρων ρυθμίσεων φάσης και της πολικότητας του ρολογιού που επιλέχθηκε. Διαθέτει επιπλέον 2 ανεξάρτητα κανάλια για SPI και 2 για PC επικοινωνία με πολλαπλές περιφερειακές συσκευές. Είναι δυνατή η επικοινωνία μέσω μονάδας ελέγχου περιοχής δικτύου (Control Area Network) για τη λήψη και τη μετάδοση δεδομένων μεταξύ πολλαπλών master και slave βασισμένες σε μηνύματα. Η μονάδα τηρεί το πρότυπο ISO 11898-1 (CAN 2.0A/CAN 2.0B) και υποστηρίζει έως και 32 συνδέσεις, οι οποίες μπορούν να διαμορφωθούν για μετάδοση ή λήψη δεδομένων. Υποστηρίζονται τόσο οι τυπικές 11bit όσο και οι διευρυμένες 29bit μορφές μηνυμάτων. Τέλος, υπάρχει και η επιλογή της ενισχυμένης σειριακής διεπαφής ήχου (Serial Sound Interface Enhanced) όπου παρέχει δυνατότητες μετάδοσης και λήψης δεδομένων από και προς πολλαπλές συσκευές που υποστηρίζουν διαφορετικές μορφές δεδομένων ήχου, όπως τον ενσωματωμένο διάυλο ήχου (Integrated Inter-IC Sound Bus) ή μονοφωνικές μορφές (monaural formats).

3.2.4 Μετατροπές και λειτουργίες

Το ολοκληρωμένο κύκλωμα διαθέτει και ενσωματωμένη μονάδα μετατροπής αναλογικού σε ψηφιακού σήματος διαδοχικής προσέγγισης 14 bit (A/D C 14bit). Ενώ χωρίζεται σε 25 αναλογικά κανάλια εισόδου όπου επιλέγονται μέσω προγράμματος. Είναι δυνατή η επιλογή ανάλυσης μετατροπής A/D είτε 14-bit είτε 12 bit, καθιστώντας δυνατή τη βελτιστοποίηση μεταξύ ταχύτητας και ανάλυσης κατά τη δημιουργία μίας ψηφιακής τιμής. Μπορεί να εκτελέσει λειτουργίες όπως, η μονή σάρωση όπου ο μετατροπέας σαρώνει τις εισόδους του και παράγει την αντίστοιχη ψηφιακή έξοδο μια φορά, η συνεχόμενη σάρωση όπου ο μετατροπέας σαρώνει και μετατρέπει τα δεδομένα των εισόδων του συνεχόμενα, η λειτουργία ομάδων όπου οι εισόδοι του μετατροπέα χωρίζονται σε ομάδες και η σάρωση γίνεται ανεξάρτητα για κάθε ομάδα και τέλος, στην λειτουργία ομάδων γίνεται να υπάρξει προτεραιότητα σε κάποια ομάδα κι έτσι να προηγείται των υπολοίπων στην σειρά εκτέλεσης της σάρωσης. Υπάρχει επίσης κι ένας μετατροπέας ψηφιακού σε αναλογικού σήματος (D/A C 12bit). Διαθέτει 1 κανάλι και μπορεί να εκτελέσει λειτουργίες όπως η μονή σάρωση και η σάρωση γεγονότων όπου ο μετατροπέας είναι σε αναμονή μέχρι να εμφανιστεί κάποιο σήμα προς μετατροπή. Προσφέρει και ένα αναλογικό συγκριτή χαμηλής κατανάλωσης (Analog Comparator Low-Power) όπου συγκρίνει 2 αναλογικές τάσεις εισόδου. Τα κανάλια του συγκριτή ACMPLP0 και ACMPLP1 είναι ανεξάρτητα μεταξύ τους και προέρχονται από 2 D/A 8-bit μετατροπείς. Το αποτέλεσμα της σύγκρισης των τάσεων εισόδου μπορεί να διαβαστεί από το λογισμικό ή και να εξαχθεί εξωτερικά. Η ταχύτητα απόκρισης σύγκρισης μπορεί να ρυθμιστεί πριν από την έναρξη μιας λειτουργίας. Η ρύθμιση της λειτουργίας υψηλής ταχύτητας μειώνει το χρόνο καθυστέρησης απόκρισης, αλλά αυξάνει την κατανάλωση ρεύματος. Η ρύθμιση λειτουργίας χαμηλής ταχύτητας αυξάνει το χρόνο καθυστέρησης απόκρισης, αλλά μειώνει την κατανάλωση ρεύματος. Τέλος, παρέχει 4 τελεστικούς ενισχυτές με 2 εισόδους και 1 έξοδο όπου μπορούν να χρησιμοποιηθούν για την ενίσχυση μικρών αναλογικών τάσεων εισόδου. Τα σήματα εξόδου από όλες τις μονάδες μπορούν να χρησιμοποιηθούν για τα σήματα εισόδου στον μετατροπέα A/D. Υποστηρίζεται η λειτουργία υψηλής ταχύτητας με κόστος την υψηλή κατανάλωση ρεύματος και η λειτουργία χαμηλής ισχύος με κόστος την αργή ταχύτητα απόκρισης. Μπορεί να γίνει η επιλογή οποιαδήποτε από τις δύο λειτουργίες με βάση τους συμβιβασμούς που χρειάζονται για την κάθε εφαρμογή μεταξύ της ταχύτητας και της κατανάλωσης ρεύματος. Η λειτουργία μπορεί να ενεργοποιηθεί με κάθε σκανδάλη από τον ασύγχρονο χρονομετρητής γενικής χρήσης (AGT) χαμηλής ισχύος, ενώ μπορεί να διακοπεί από ένα σκανδαλισμό τέλους.

3.2.5 Χρονιστές

Οι απαραίτητοι χρόνοι για τις διάφορες λειτουργίες του συστήματος μετρούνται από τους χρονιστές που υπάρχουν. Ξεκινώντας από τον χρονοδιακόπτη γενικού σκοπού (General Purpose Timer) είναι ένας χρονοδιακόπτης 32bit με δύο κανάλια και ένας χρονοδιακόπτης 16bit με έξι κανάλια. Οι κυματομορφές Pulse Width Modulation που μπορούν να παραχθούν είναι, η ανηφορική μέσω του ανοδικού μετρητή, η καθοδική μέσω του καθοδικού μετρητή και η τριγωνική μέσω του ανοδικού και καθοδικού μετρητή. Ο GPT μπορεί επίσης να χρησιμοποιηθεί ως χρονοδιακόπτης γενικής χρήσης. Είναι εφικτή η σύγχρονη εκκίνηση, διακοπή και εκκαθάριση μετρητών για αυθαίρετα κανάλια. Επίσης, κάθε ρολόι μπορεί να λειτουργήσει ανεξάρτητα και αυτόνομα από τα υπόλοιπα ώστε να παραχθούν οι διάφορες κυματομορφές που χρειάζεται το σύστημα. Διαθέτει κι ένα ασύγχρονο χρονομετρητή γενικής χρήσης (Asynchronous General purpose Timer) χαμηλής κατανάλωσης ισχύος των 16 bit που μπορεί να χρησιμοποιηθεί για την έξοδο παλμών, τη μέτρηση του πλάτους ή της περιόδου των εξωτερικών παλμών και την καταμέτρηση διάφορων συμβάντων στις εισόδους του. Ο χρονομετρητής αποτελείται

από έναν καταχωρητή επαναφόρτωσης και έναν καθοδικό μετρητή. Ο καταχωρητής επαναφόρτισης και ο μετρητής καθόδου βρίσκονται στην ίδια διεύθυνση και είναι προσβάσιμοι με τον καταχωρητή του AGT.

3.3 Κινητήρες

Οι κινητήρες είναι ένα αναπόσπαστο κομμάτι για την κατασκευή της αυτόματης σκούπας αλλά και γενικότερα σε πολλές άλλες εφαρμογές. Γενικά υπάρχουν δύο βασικοί τύποι κινητήρων ανάλογα με την τάση τροφοδοσίας που χρειάζονται, αυτοί που λειτουργούν με συνεχές ρεύμα και αυτοί που λειτουργούν με εναλλασσόμενο ρεύμα. Στην παρούσα εργασία θα γίνει αναφορά στους συνεχούς ρεύματος κινητήρες καθώς οι συγκεκριμένοι χρησιμοποιήθηκαν για την κατασκευή. Αυτού του είδους κινητήρες χρησιμοποιούν συνεχές ρεύμα για την παραγωγή μηχανικής δύναμης. Οι πιο συνηθισμένοι τύποι βασίζονται σε μαγνητικές δυνάμεις που παράγονται από ρεύματα στα πηνία. Χρησιμοποιήθηκαν ευρέως καθώς υπάρχουν σε μεγάλη ποικιλία χαρακτηριστικών όπως η ονομαστική τάση τροφοδοσίας, οι μέγιστες στροφές και η ροπή του κινητήρα. Ο έλεγχος της ταχύτητας γίνεται σε μεγάλη κλίμακα τιμών, είτε με τη χρήση μεταβλητής τάσης τροφοδοσίας του, είτε με τη μεταβολή της έντασης του ρεύματος στα τυλίγματα πεδίου του. Παρακάτω, στο Σχήμα 3.3 και 3.4 παρουσιάζεται ένας τυπικός μικρός κινητήρας συνεχούς ρεύματος ο οποίος θα χρησιμοποιηθεί και για την κατασκευή [21].



Σχήμα 3.4. Κινητήρας συνεχούς ρεύματος
- μπροστινή όψη



Σχήμα 3.3. Κινητήρας συνεχούς ρεύματος
- πίσω όψη

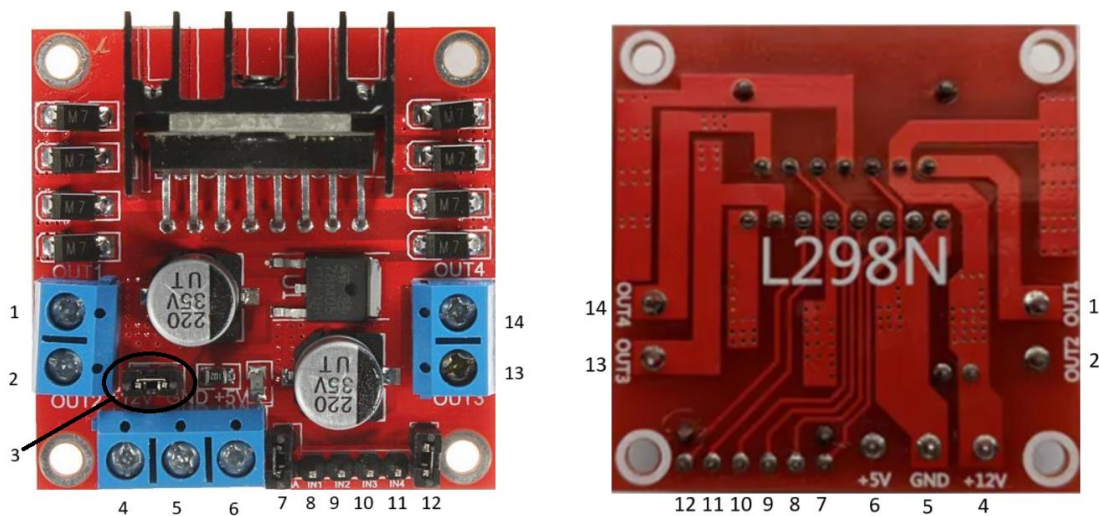
Για την φορά της περιστροφής σημασία έχει πως θα γίνει η σύνδεση. Οι κινητήρες έχουν 2 ακροδέκτες όπου συνδέονται, είτε στην μπαταρία, είτε σε κάποιο κύκλωμα οδήγησης. Ανάλογα με το πως θα συνδεθούν οι ακροδέκτες, ο κινητήρας θα έχει και διαφορετική φορά περιστροφής, στην μία περίπτωση δεξιόστροφα ενώ αν αλλάξουν μεταξύ τους οι δύο ακροδέκτες θα γυρίζει αριστερόστροφα. Τα τεχνικά χαρακτηριστικά των κινητήρων κίνησης της κατασκευής παρατηρούνται στον Πίνακα 3.1.

Πίνακας 3.1. Τεχνικά χαρακτηριστικά εικονιζόμενων κινητήρων.

Τύπος	Κινητήρας
Βάρος	9g
Χρήση	Κίνηση
Τάση τροφοδοσίας	Από +1,5V έως +12V
Ονομαστική τάση	6V
Μέγιστο ρεύμα	3A

3.4 Κύκλωμα οδήγησης κινητήρων με L298N

Ο έλεγχος των κινητήρων είναι αδύνατον να γίνει κατευθείαν από τον μικροεπεξεργαστή καθώς δεν μπορεί να παρέχει το ανάλογο ρεύμα που χρειάζεται για την κίνησή τους [22]. Γι αυτό τον λόγο μπαίνει ενδιάμεσά τους ένα κύκλωμα οδήγησης το οποίο στην ουσία είναι μια γέφυρα σε συνδεσμολογία Η. Μπορεί να χρησιμοποιηθεί για τον έλεγχο βηματικών ή/και DC κινητήρων. Στους DC κινητήρες, ο έλεγχος των στροφών γίνεται μέσω παλμών ρυθμιζόμενου πλάτους όπου τους προσαρμόζει ο μικροεπεξεργαστής. Στα παρακάτω Σχήματα 3.5 και 3.6 φαίνονται αναλυτικά οι ακροδέκτες του κυκλώματος οδήγησης.



Σχήμα 3.6. Κυκλωμα οδήγησης κινητήρων με L298N - πάνω όψη
Σχήμα 3.5. Κυκλωμα οδήγησης κινητήρων με L298N - κάτω όψη

1. Ακροδέκτης εξόδου «+» για DC κινητήρα 1 ή A+ για βηματικό κινητήρα
2. Ακροδέκτης εξόδου «-» για DC κινητήρα 1 ή A- για βηματικό κινητήρα
3. Βραχυκυκλωμένοι ακροδέκτες όπου επιτρέπουν την σύνδεση του ακροδέκτη 4 με τον 5V ρυθμιστή τάσης της πλακέτας. Θα πρέπει να αποσυνδεθεί αν η τάση τροφοδοσίας είναι μεγαλύτερη των 12V
4. Ακροδέκτης τάσης τροφοδοσίας, μέγιστη τάση 35V
5. Ακροδέκτης γείωσης

6. Ακροδέκτης εξόδου 5V στην περίπτωση όπου ο βραχυκυκλωτήρας ακροδεκτών 3 είναι συνδεδεμένος
7. Βραχυκυκλωμένοι ακροδέκτες. Συνδεδεμένοι επιτρέπουν την χρήση για βηματικό κινητήρα 1, αλλιώς μπορεί να συνδεθεί το σήμα ελέγχου στροφών του κινητήρα 1
8. Ακροδέκτης εισόδου για έλεγχο φοράς κινητήρα 1
9. Ακροδέκτης εισόδου για έλεγχο φοράς κινητήρα 1
10. Ακροδέκτης εισόδου για έλεγχο φοράς κινητήρα 2
11. Ακροδέκτης εισόδου για έλεγχο φοράς κινητήρα 2
12. Βραχυκυκλωμένοι ακροδέκτες. Συνδεδεμένοι επιτρέπουν την χρήση για βηματικό κινητήρα 2, αλλιώς μπορεί να συνδεθεί το σήμα ελέγχου στροφών του κινητήρα 2
13. Ακροδέκτης εξόδου «+» για DC κινητήρα 2 ή B+ για βηματικό κινητήρα
14. Ακροδέκτης εξόδου «-» για DC κινητήρα 2 ή B- για βηματικό κινητήρα

Για τον έλεγχο ενός DC κινητήρα χρειάζονται οι ακόλουθες ενέργειες. Αρχικά συνδέονται οι ακροδέκτες του κινητήρα στους ακροδέκτες εξόδου 1 και 2 της μονάδας ελέγχου. Στην συνέχεια συνδέεται η τάση τροφοδοσίας στον ακροδέκτη 4 και η γείωση στον 5. Αν η τροφοδοσία γίνεται με μεγαλύτερη τάση από τα 12V, τότε είναι απαραίτητο να αφαιρεθεί ο βραχυκυκλωτήρας στο σημείο 3. Τέλος, για την ταχύτητα των τροχών, εφαρμόζεται το κατάλληλο σήμα παλμών από κάποιο μικροεπεξεργαστή στον ακροδέκτη 7, ενώ συνδέονται σε κατάσταση HIGH και LOW ή LOW και HIGH οι ακροδέκτες 8 και 9. Με τους ακροδέκτες 8 και 9 ελέγχεται η φορά της κίνησης των τροχών δηλαδή για παράδειγμα με HIGH τον ακροδέκτη 8 και LOW τον 9 μπορεί να ο κινητήρας να κινείται με την φορά του ρολογιού, ενώ με LOW και HIGH να κινείται αντίστροφα. Για την κίνηση ενός DC κινητήρα είναι απαραίτητο να συνδεθεί ένα σήμα παλμών στο ποδαράκι 7 και να τεθεί ένα από τα 8 ή 9 σε κατάσταση HIGH ενώ το άλλο σε κατάσταση LOW. Τα τεχνικά χαρακτηριστικά του αναγράφονται στον παρακάτω Πίνακα 3.2.

Πίνακας 3.2. Τεχνικά χαρακτηριστικά του κυκλώματος οδήγησης κινητήρων με χρήση L298N

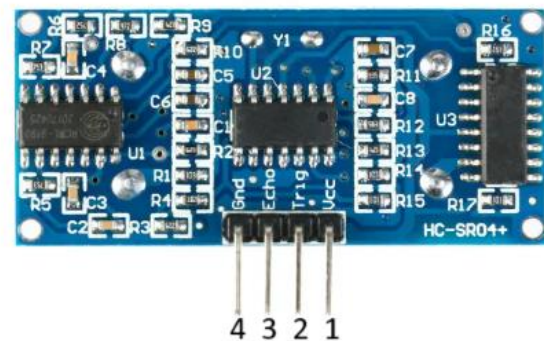
Τύπος	Κύκλωμα οδήγησης κινητήρων
Βάρος	25g
Χρήση	DC κινητήρες
	Βηματικοί κινητήρες
Οδήγηση	L298N Dual H Bridge DC Motor Driver IC
Τάση τροφοδοσίας	Από +5V έως +35V
Μέγιστο ρεύμα εξόδου	3A
Μέγιστη απώλεια ισχύος	20W σε θερμοκρασία 75°C

3.5 Αισθητήρας μέτρησης απόστασης με υπέρηχους

Υπάρχουν αρκετοί τρόποι για να μετρηθεί η απόσταση από ένα σημείο μέχρι ένα άλλο, όπου ο καθένας έχει τα πλεονεκτήματα και τα μειονεκτήματά του. Σε αυτή την υποενότητα εξετάζεται ο αισθητήρας μέτρησης απόστασης με χρήση υπερήχων[23]. Η βασική αρχή λειτουργίας του είναι η μέτρηση του χρόνου από την εκτόξευση του σήματος από τον αισθητήρα μέχρι την αντανάκλασή του, όταν συναντήσει ένα αντικείμενο. Αποτελείται από δύο βασικά μέρη, τον αποστολέα όπου στέλνει το σήμα και από τον παραλήπτη όπου δέχεται το σήμα προς μέτρηση. Η διαδικασία μέτρησης ξεκινάει εκπέμποντας ένα σήμα συχνότητας ~40KHz για 10 μs και τελειώνει όταν ληφθεί το σήμα πίσω ή μετά το πέρασμα 38ms όπου τότε σταματάει ο παραλήπτης να αναμένει το σήμα. Στην περίπτωση που λάβει το σήμα ο δέκτης, η απόσταση υπολογίζεται από την σχέση

$$\text{Απόσταση} = \frac{\text{ταχύτητα ήχου} * \text{χρόνο ακρόασης}}{2}$$

Ένας τέτοιος αισθητήρας απεικονίζεται στα Σχήματα 3.7 και 3.8 παρακάτω.



Σχήμα 3.8. Αισθητήρας μέτρησης απόστασης με υπέρηχους - μπροστινή όψη

Σχήμα 3.7. Αισθητήρας μέτρησης απόστασης με υπέρηχους - πίσω όψη

1. Ακροδέκτης τάσης τροφοδοσίας 5V
2. Ακροδέκτης εισόδου
3. Ακροδέκτης εξόδου
4. Ακροδέκτης γείωσης

Για την αρχικοποίηση και λειτουργία του αισθητήρα γίνονται οι ακόλουθες ενέργειες. Αρχικά συνδέεται η τάση τροφοδοσίας και η γείωση στους αντίστοιχους ακροδέκτες, ενώ ο Trig και ο Echo σε οποιοδήποτε ψηφιακό ακροδέκτη του μικροεπεξεργαστή. Για να ξεκινήσει η μέτρηση, μέσω του προγράμματος τίθεται ο ακροδέκτης με τον Trig σε κατάσταση HIGH για 10μs όπου για την διάρκεια αυτή θα σταλθεί το σήμα υπέρηχων προς μέτρηση. Μόλις τεθεί σε κατάσταση LOW ο Trig, δίνει λογικό HIGH ο ακροδέκτης Echo και μετράει τον χρόνο μέχρι να επιστρέψει το σήμα μετά την αντανάκλασή του σε κάποιο αντικείμενο. Μόλις λάβει την μέτρηση του σήματος επιστρέφει σε κατάσταση LOW. Στην συνέχεια γίνεται ο υπολογισμός της απόστασης και εκτελείται η συνέχεια του προγράμματος. Τα τεχνικά χαρακτηριστικά του αναγράφονται στον παρακάτω Πίνακα 3.3.

Πίνακας 3.3. Τεχνικά χαρακτηριστικά του αισθητήρα μέτρησης απόστασης με χρήση υπέρηχων

Τύπος	Αισθητήρας μέτρησης απόστασης
Μέσο μέτρησης	Υπέρηχοι
Τάση τροφοδοσία	Από +3,3V έως +5V
Κατανάλωση ρεύματος	15mA
Συχνότητα εκπομπής	40KHz
Εμβέλεια	Από 2cm έως 4m για τροφοδοσία +3,3V Από 2cm έως 4,5m για τροφοδοσία +5V
Ανάλυση μέτρησης	3mm
Γωνία ανίχνευσης σήματος	$\pm 15^\circ$
Χρόνος εκπομπής	10μs
Διαστάσεις	4,5×2×1,5cm (x×y×z)

3.6 Αισθητήρας ανίχνευσης αντικειμένου με υπέρυθρες ακτίνες

Σε αυτή την υποενότητα εξετάζεται η δομή και λειτουργία του αισθητήρα ανίχνευσης αντικειμένων με χρήση υπέρυθρων ακτίνων [24]. Η βασική λειτουργία του αισθητήρα αυτού είναι η χρήση του σαν έναν διακόπτη on/off με οπτικό ερέθισμα. Αποτελείται από τρία βασικά μέρη, τον εκπομπέα υπέρυθρων ακτίνων, τον δέκτη τους μετά την ανάκλαση και ένα ποτενσιόμετρο για την ρύθμιση της ευαισθησίας. Αρχικά ο αποστολέας όπου είναι ένα LED εκπομπής υπέρυθρης ακτινοβολίας, εκπέμπει ένα σήμα. Ο δέκτης που λαμβάνει το σήμα, είναι ένας ημιαγωγός όπου αλλάζει την αντίστασή του ανάλογα με την ένταση του σήματος μετά την ανάκλαση και άρα αλλάζει το ρεύμα που διέρχεται από τον ημιαγωγό σύμφωνα με την απόσταση του. Το ποτενσιόμετρο ρυθμίζει ένα ρεύμα όπου θα συγκριθεί με αυτό του ημιαγωγού του δέκτη και ανάλογα το αποτέλεσμα θα κρίνει ο αισθητήρας εάν υπάρχει, σε κάποια συγκεκριμένη απόσταση, αντικείμενο μπροστά του. Στα Σχήματα 3.9 και 3.10 απεικονίζεται ένας αισθητήρας ανίχνευσης αντικειμένου με υπέρυθρες ακτίνες.



Σχήμα 3.9. Αισθητήρας ανίχνευσης αντικειμένου με υπέρυθρες ακτίνες - μπροστινή όψη



Σχήμα 3.10. Αισθητήρας ανίχνευσης αντικειμένου με υπέρυθρες ακτίνες - πίσω όψη

1. Ακροδέκτης ψηφιακής εξόδου
2. Ακροδέκτης αναλογικής εξόδου
3. Ακροδέκτης γείωσης
4. Ακροδέκτης τάσης τροφοδοσίας 5V

Η διαδικασία χρήσης του αισθητηρίου έχει ως εξής, αρχικά συνδέονται οι ακροδέκτες γείωσης και τροφοδοσίας τάσης στους αντίστοιχους ακροδέκτες της μπαταρίας ή του μικροεπεξεργαστή. Στην συνέχεια, ανάλογα με την χρήση, επιλέγεται είτε η αναλογική είτε η ψηφιακή έξοδος. Η ψηφιακή έξοδος του αισθητήρα δίνει λογικές τιμές, δηλαδή LOW στην περίπτωση που δεν υπάρχει αντικείμενο μπροστά από τον αισθητήρα και HIGH στην περίπτωση ανάκλασης του σήματος μέσα στην εμβέλεια που έχει οριστεί από το ποτενσιόμετρο. Η αναλογική έξοδος παράγει τιμές από 0 μέχρι 1023 έτσι ώστε να εισέλθουν σε ένα οκταψήφιο μετατροπέα από αναλογικό σε ψηφιακό σήμα και να επεξεργαστούν μετά από τον μικροελεγκτή σύμφωνα με το πρόγραμμα. Τα τεχνικά χαρακτηριστικά του αναγράφονται στον παρακάτω Πίνακα 3.4.

Πίνακας 3.4. Τεχνικά χαρακτηριστικά του αισθητήρα μέτρησης απόστασης με χρήση υπέρυθρων ακτίνων

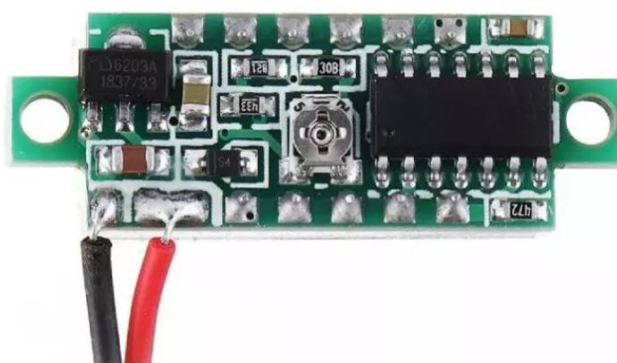
Τύπος	Αισθητήρας ανίχνευσης αντικειμένου
Μέσο μέτρησης	Υπέρυθρες ακτίνες
Τάση τροφοδοσίας	Από +3,3V έως +5V
Κατανάλωση ρεύματος	15mA
Εμβέλεια	Από 2cm έως 30cm, ανάλογα το υλικό ανάκλασης μπορεί παραπάνω ή λιγότερο
Γωνία ανίχνευσης σήματος	$\pm 17,5^\circ$

3.7 Ψηφιακό Βολτόμετρο

Σε μια κατασκευή είναι χρήσιμη πληροφορία η γνώση της τάσης της μπαταρίας καθώς έτσι γνωρίζει χρήστης πότε να φορτίσει της συσκευή. Για τον έλεγχο της τάσης είναι συνηθισμένη η χρήση ενός βολτόμετρου καθώς είναι εύκολα στην ένταξή τους στο κύκλωμα προς έλεγχο και οι μετρήσεις τους είναι αρκετά ακριβείς και αξιόπιστες. Στα παρακάτω Σχήματα 3.11 και 3.12 απεικονίζεται ένα πολύ μικρό ψηφιακό βολτόμετρο το οποίο συνδέεται παράλληλα με το κύκλωμα προς μέτρηση και εμφανίζει την τάση μέσω τριών led επτά τμημάτων [25]. Στον Πίνακα 3.5. αναγράφονται όλα τα βασικά χαρακτηριστικά του συγκεκριμένου βολτόμετρου,.



Σχήμα 3.12. Μπροστινή όψη ψηφιακού βολτόμετρου



Σχήμα 3.11. Πισινή όψη ψηφιακού βολτόμετρου

Πίνακας 3.5. Τεχνικά χαρακτηριστικά ψηφιακού βολτόμετρου

Τύπος	Ψηφιακό βολτόμετρο συνεχούς τάσης
Χρήση	Ένδειξη τάσης στο σημείο σύνδεσης
Τάση λειτουργίας	Από +2,5 έως +30 V
Ένδειξη μέτρησης	Από + 2,5V έως +30V
Ρεύμα λειτουργίας	<30mA
Ρυθμός ανανέωσης	200ms
Ακρίβεια μετρήσεων	0,3V
Διαστάσεις	3,1×1,2×0,9cm (x×y×z)

3.8 Συσκευή αναρρόφησης

Για την αναρρόφηση της κατασκευής χρησιμοποιήθηκε ένα ασύρματο σκουπάκι χειρός το οποίο έχει τη δυνατότητα καθαρισμού στερεών. Τα τεχνικά του χαρακτηριστικά πάρθηκαν από τον κατασκευαστή και απεικονίζονται στον παρακάτω Πίνακα 3.6.

Πίνακας 3.6. Τεχνικά χαρακτηριστικά κινητήρα αναρρόφησης

Τύπος	Σκουπάκι χειρός
Τάση λειτουργίας	Από μπαταρία +7,4V Από καλώδιο +12V
Ονομαστική ισχύς	120W
Ισχύς αναρρόφησης	6000PA
Χωρητικότητα δοχείου	0,1 λίτρα
Βάρος	0,5Kg
Διαστάσεις	19×6×7cm (x×y×z)

3.9 Επίλογος

Στο παρόν Κεφάλαιο έγινε η αναφορά των βασικών υλικών όπου χρησιμοποιούνται σε τέτοιου είδους κατασκευές. Αναλύθηκαν τα χαρακτηριστικά τους, οι λειτουργίες που προσφέρουν και έγινε μια περίληψη του τρόπου σύνδεσης του καθενός .

Κεφάλαιο 4ο: Κίνηση

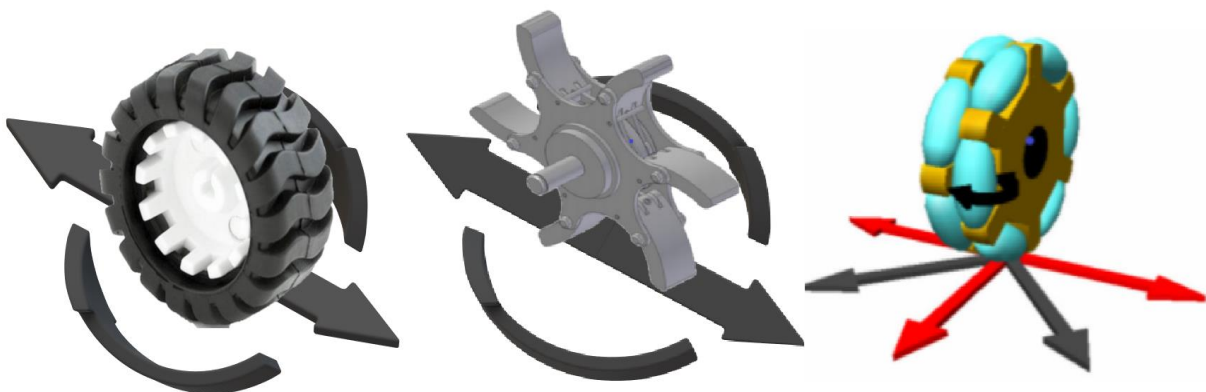
4.1 Εισαγωγή

Ένα ρομπότ συνήθως χρειάζεται κάποιο μηχανισμό για την ανεμπόδιστη μετακίνησή του στο περιβάλλον όπου κινείται. Καθώς η τεχνολογία εξελίσσεται και το αντικείμενο μελετάται βαθύτερα, αναπτύσσονται όλο και περισσότεροι τρόποι για την επίτευξη της μετακίνησης και άρα η σωστή επιλογή του τρόπου της είναι όλο και πιο σημαντική. Οι κινήσεις τους είναι συνήθως εμπνευσμένες από την ανθρώπινη φύση όπως είναι το περπάτημα, το τρέξιμο, η ολίσθηση και τα άλματα αλλά υπάρχουν και οι περιπτώσεις όπως το πέταγμα ή η κίνηση με τροχούς, με την τελευταία να είναι και η πιο διαδεδομένη. Οι τροχοί προσδίδουν λιγότερους βαθμούς ελευθερίας σε σύγκριση με τα πόδια και επομένως η μελέτη και η ανάπτυξή της είναι πιο απλή.

4.2 Κατηγορίες τροχών ανά χρήση

Η επιλογή ανάμεσα από τους διάφορους τρόπους μετακίνησης έχει να κάνει με βάση τον τύπο και την κλίση της επιφάνειας κίνησης. Οι τροχοί συνιστώνται για χρήση σε σκληρές επιφάνειες χωρίς ιδιαίτερη σημασία η κλίση της. Από την άλλη τα χρήση ποδιών δεν επηρεάζεται από την αντίσταση που θα προβάλλει η επιφάνεια αλλά έχει ιδιαίτερη σημασία η κλίση και η ομαλότητά της. Η χρήση τροχών ενδείκνυται για χρήση σε σκληρό και επίπεδο έδαφος όπως και είναι κατά βάση το πάτωμα των σπιτιών. Μπορούν να παρέχουν πολύ καλές αποδόσεις και μάλιστα με μια σχετικά απλή υλοποίηση. Επιπλέον, η ισορροπία δεν αποτελεί πρόβλημα στις κατασκευές ρομπότ με τροχούς, επειδή σχεδιάζονται έτσι ώστε σχεδόν πάντα όλοι οι τροχοί να βρίσκονται σε επαφή με το έδαφος ανά πάσα στιγμή. Ανάλογα με τον αριθμό των τροχών, μπορεί σε κάποιες περιπτώσεις να χρειαστούν περαιτέρω μελέτες από κάποιες

άλλες, για την σωστή υλοποίηση και λειτουργία της κατασκευής. Στην περίπτωση των τεσσάρων ή και παραπάνω τροχών εξασφαλίζεται η ισορροπία αλλά χρειάζεται η μελέτη ενός συστήματος αναρτήσεων για την διασφάλιση ότι οι τροχοί όπου προσδίδουν την κίνηση έρχονται σε επαφή με την επιφάνεια κάθε χρονική στιγμή. Από την άλλη, στην περίπτωση των δύο τροχών είναι σίγουρο ότι θα έρχονται σε επαφή έτσι ώστε η συσκευή να κινείται αποτελεσματικά αλλά σχεδόν αδύνατο να ισορροπήσει η συσκευή όσο είναι ακίνητη. Η χρήση των τριών τροχών επιτυγχάνει καλές αποδόσεις και είναι απλούστερη σε σχέση με της υπόλοιπες επιλογές καθώς δεν υπάρχει πρόβλημα με την ισορροπία αλλά ούτε χρειάζεται κάποιο σύστημα που θα επιτρέπει την ταυτόχρονη επαφή των τροχών με το έδαφος. Μπορεί επίσης να γίνει ένας διαχωρισμός των τροχών και ανάλογα με την λειτουργία που προσφέρουν καθώς ένας τροχός μπορεί να είναι ένα παθητικό στήριγμα είτε να ενεργεί προσφέροντας κίνηση παράλληλα με την στήριξη της κατασκευής. Πέρα από τον αριθμό των τροχών και την λειτουργία τους, μια άλλη διαφοροποίηση είναι και ανάλογα το σχήμα τους, με ποιο γνωστό να είναι το κυκλικό σχήμα όπου χρησιμοποιείται σχεδόν σε όλες της κατασκευές, όπως και στη κατασκευή που υλοποιήθηκε, καθώς προσφέρει ομαλή πορεία στις περισσότερες επιφάνειες με δυνατότητα κίνησης δύο αξόνων σε οποιαδήποτε κατεύθυνση. Ένα άλλο είδος είναι κι ο πανκατευθυντικός, είναι ένας κυκλικός τροχός όπου επιτρέπει την κίνηση σε δύο άξονες προς όλες τις κατευθύνσεις αλλά επιτρέπει επίσης και την ολίσθηση. Τρίτο είδος μπορεί να θεωρηθεί ένας τροχός με τρία άκρα, με τέσσερα, ή και οποιοδήποτε αριθμό άκρων, σε κατάλληλο σχεδιασμό, όπου είτε προσομοιάζει το ανθρώπινο βήμα είτε δίνει την δυνατότητα αναρρίχησης σε μη προσβάσιμα μέρη [26], [27], [28], [29].



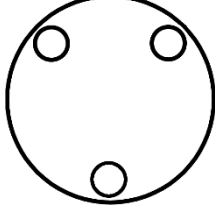
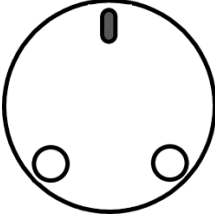
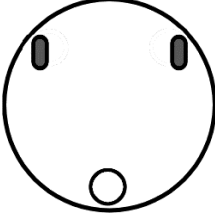
Σχήμα 4.1. Κυκλικός τροχός (αριστερά), τροχός με έξι άκρα (κέντρο), πανκατευθυντικός τροχός (δεξιά)

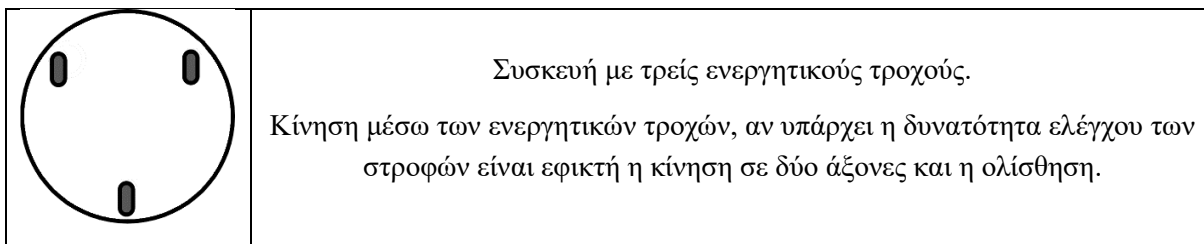
4.3 Κατηγορίες ανά ενέργειες τροχών

Επίσης, η κίνηση όπου προσφέρουν οι τροχοί πρέπει να μελετηθεί και να αναλυθεί. Κάθε τροχός έχει ένα ρόλο που επιτρέπει την κίνηση ολόκληρου του ρομπότ όπως επίσης προσθέτει και κάποιους περιορισμούς. Για την σωστή υλοποίηση του συστήματος θα πρέπει να κατανοηθεί η μηχανική συμπεριφορά του ρομπότ για να σχεδιαστούν κατάλληλα τα κινητά μέρη αλλά και για να δημιουργηθεί το λογισμικό ελέγχου της κίνησης. Οι παθητικοί τροχοί προσφέρουν στήριξη ενώ επιτρέπουν την κίνηση είτε προς έναν άξονα είτε προς όλες τις κατευθύνσεις με το μειονέκτημα ότι χρειάζεται κάποιος εξωτερικός παράγοντας για την εκκίνηση της κατασκευής. Οι ενεργητικοί τροχοί είναι σταθεροί προσφέροντας στήριξη και επιτρέποντας την κίνηση συνήθως σε έναν άξονα με το πλεονέκτημα ότι συνδέονται με κινητήρες ώστε να προσφέρουν, χωρίς την εξωτερική παρεμβολή, την εκκίνηση της

κατασκευής. Σύμφωνα με τον αριθμό και την αναλογία ενεργητικών και παθητικών τροχών υπάρχουν διάφοροι τρόποι περιήγησης στον χώρο. Παρακάτω παρουσιάζονται οι διαφορετικές περιπτώσεις αναλογιών των τριών τροχών και οι δυνατότητές τους. Η απλούστερη είναι οι τρεις παθητικοί τροχοί όπου προσφέρουν την απαραίτητη στήριξη, ενώ ανάλογα τον τύπο τους μπορεί να γίνει μετακίνηση σε έναν άξονα, είτε προς όλες τις κατευθύνσεις, με την εκκίνηση γίνεται μόνο από κάποιο εξωτερικό παράγοντα. Στην συνέχεια υπάρχει η αναλογία του ενός ενεργού τροχού με άλλους δύο παθητικούς, όπου η κατασκευή είναι σταθερή και ισορροπεί στάσιμη και επίσης επιτρέπει την κίνηση σε έναν άξονα, χωρίς την δυνατότητα στροφής ή ολίσθησης δίχως εξωτερική βοήθεια. Η περίπτωση των δύο ενεργών τροχών και του ενός παθητικού είναι η πιο απλή όπου η συσκευή έχει την δυνατότητα να ισορροπεί και να κινείται και στους δύο άξονες αυτόνομα. Το επιτυγχάνει όταν είναι εφικτή η διαφορετική ταχύτητα περιστροφής των δύο τροχών καθώς στην ίδια ταχύτητα, η συσκευή κινείται ευθεία στον άξονα, στην περίπτωση όπου ο δεξιά τροχός κινείται γρηγορότερα από τον αριστερό, η κατασκευή στρίβει αριστερά καθώς ο τροχός δεν προλαβαίνει να ευθυγραμμίσει την συσκευή, ενώ αντίστοιχα αν ο αριστερός τροχός κινείται ταχύτερα, το ρομπότ στρίβει δεξιά. Εάν υπάρχει η δυνατότητα για αντίθετη περιστροφή, είναι εφικτή η στροφή της κατασκευής ενώ παραμένει στάσιμη. Για την περίπτωση των τριών ενεργών τροχών είναι πάλι δυνατή η ισορροπία, η κίνηση και στους δύο άξονες καθώς επίσης και η ολίσθηση της κατασκευής. Η στροφή γίνεται με την χρήση των δύο τροχών και εκτελείται η ίδια διαδικασία όπως με τους δύο ενεργούς τροχούς ενώ για την ολίσθηση χρειάζονται και οι τρεις τροχοί, οι δύο περιστρέφονται προς την κατεύθυνση ολίσθησης ενώ ο τρίτος περιστρέφεται προς την αντίθετη [30].

Πίνακας 4.1. Όλοι οι πιθανοί συνδυασμοί υλοποιήσεις της κατασκευής με τρεις τροχούς και η χρήση τους

	<p>Συσκευή με τρεις παθητικούς τροχούς. Χρειάζεται βοήθεια για την εκκίνηση, δυνατότητα κίνησης προς όλους τους άξονες</p>
	<p>Συσκευή με δύο παθητικούς κι ένα ενεργητικό τροχό. Κίνηση μέσω του ενεργητικού τροχού, μόνο προς έναν άξονα</p>
	<p>Συσκευή με δύο ενεργητικούς κι ένα παθητικό τροχό. Κίνηση μέσω των ενεργητικών τροχών, αν υπάρχει η δυνατότητα ελέγχου των στροφών είναι εφικτή η κίνηση σε δύο άξονες.</p>



Να σημειωθεί ότι για την υλοποίηση της αυτόματης σκούπας χρησιμοποιήθηκαν σταθεροί, στρογγυλοί τροχοί καθώς το πάτωμα όπου κατά βάση θα μετακινείται είναι επίπεδο και σκληρό και επιζητείται ομαλή πορεία. Ενώ επιλέχθηκαν τρεις τροχοί, μεταξύ των οποίων οι δύο ενεργοί κι ένας παθητικός γιατί προσφέρουν την απαραίτητη στήριξη και ισορροπία όπου χρειάζεται η συσκευή καθώς επίσης λόγω της εύκολης πορείας και στροφής με βάση την ταχύτητα περιστροφής των τροχών.

4.4 Επίλογος

Στον παραπάνω Κεφάλαιο έγινε η αναφορά και η σύγκριση των δυνατοτήτων όπου προσφέρει το κάθε είδος τροχού σε μια κατασκευή. Έγινε η ανάλυσή τους και δόθηκαν μερικά παραδείγματα. Στην συνέχεια παρουσιάστηκαν οι συνδυασμοί υλοποίησης στο παράδειγμα των τριών τροχών για την κίνηση και συγκρίθηκαν μεταξύ τους για τις λειτουργίες και τον έλεγχο όπου προσφέρουν.

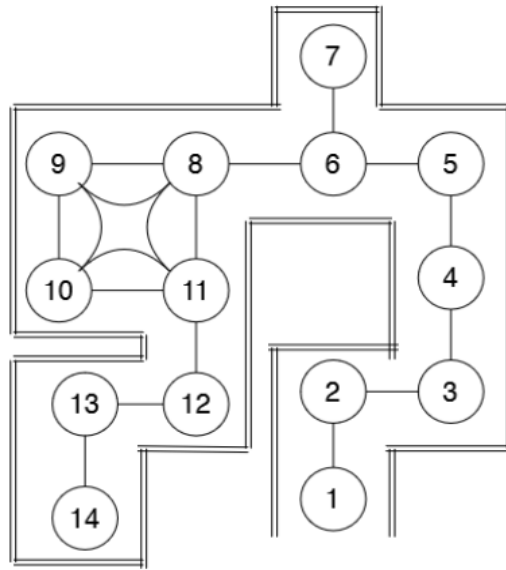
Κεφάλαιο 5ο: Πλοήγηση

5.1 Εισαγωγή

Πέρα από την κατανόηση της λειτουργίας κίνησης της κατασκευής, εξίσου σημαντική είναι η μελέτη και η πλοήγηση στον χώρο όπου θα κινείται. Γενικά έχουν μελετηθεί τρεις βασικοί τρόποι πλοήγησης σε κάποιο χώρο εργασίας, η πρόγνωση, η σάρωση και οι τυχαίες κινήσεις.

5.2 Πρόγνωση

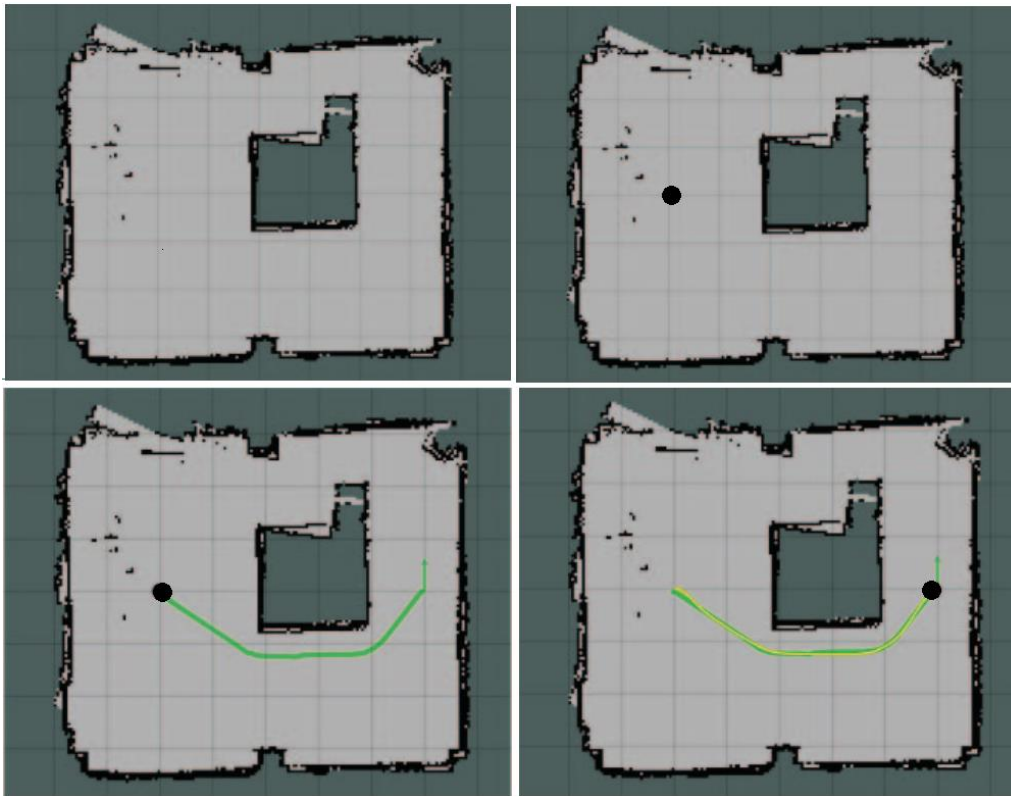
Στην περίπτωση της πρόγνωσης, το πρόβλημα είναι γνωστό και αντιμετωπίζεται μέσω του προγράμματος της συσκευής. Η κατασκευή γνωρίζει τον χώρο κίνησης, τις απαγορευμένες ζώνες και τα εμπόδια όπου θα συναντήσει στον δρόμο της. Κατά την αρχικοποίηση του προγράμματος, ο προγραμματιστής ορίζει τον χώρο εργασίας και συντάσσει τις εντολές όπου θα εκτελέσει η κατασκευή. Οι εντολές αυτές περιλαμβάνουν ένα-ένα και αναλυτικά τα βήματα όπου θα πρέπει να πραγματοποιήσει το ρομπότ για την γρηγορότερη πλοήγηση και την αποφυγή των αντικειμένων στο χώρο. Η περίπτωση της πρόγνωσης του χώρου είναι η ευκολότερη και η γρηγορότερη για την εκτέλεση, αλλά με το μειονέκτημα ότι δεν είναι δυνατή η εκτέλεση του ίδιου προγράμματος σε διαφορετικό χώρο, ο οποίος δεν είναι όμοιος με τον αρχικό. Ακόμα, στην περίπτωση μετακίνησης κάποιου αντικειμένου, το πρόγραμμα αδυνατεί να εκτελεσθεί ορθά, ούτε στον ίδιο χώρο [31].



Σχήμα 5.1. Χάρτης εκτέλεσης βημάτων για προγνωστική συσκευή

5.3 Σάρωση

Μία άλλη μέθοδος είναι αυτή της σάρωσης του χώρου και στην συνέχεια η εκτέλεση της λειτουργίας. Στην προκειμένη περίπτωση, η συσκευή ξεκινώντας από ένα αρχικό σημείο διανύει όλο τον χώρο εργασίας, σκανάρώντας τον για τυχόν αντικείμενα που εμποδίζουν. Αμέσως μετά την λήξη της σάρωσης η συσκευή επιστρέφει στην αρχική της θέση και υπολογίζει την βέλτιστη διαδρομή που θα ακολουθήσει για την καλύτερη κάλυψη του χώρου. Στην συνέχεια ακολουθεί την πορεία όπου παρήγαγε, εκτελώντας τις εργασίες όπου χρειάζεται να πραγματοποιήσει. Η συσκευή πρέπει να είναι σε θέση να ελέγξει παράλληλα τον χώρο για τυχόν αλλαγές και να λαμβάνει αποφάσεις στιγμής για την αποφυγή κάποιου καινούργιου αντικειμένου. Η μέθοδος αυτή είναι σε θέση να εκτελεσθεί σε διάφορους χώρους, ανεξαρτήτως μεγέθους και πολυπλοκότητας, καθώς υπολογίζει κάθε φορά από την αρχή την βέλτιστη διαδρομή όπου θα ακολουθήσει. Το γεγονός ότι η συσκευή πρέπει να διανύσει τον χώρο δύο φορές είναι μειονέκτημα για τους μικρούς ή απλούς χώρους όπου με άλλους τρόπους θα εκτελούνταν η διεργασία πιο γρήγορα. Από την άλλη, η γνώση της βέλτιστης διαδρομής από την αρχή της πορείας είναι πλεονέκτημα για μεγάλους ή πολύπλοκους χώρους, όπου υπο άλλες συνθήκες η συσκευή μπορεί να χαθεί ή να εγκλωβιστεί σε κάποιο σημείο [32], [33].



Σχήμα 5.2. Τελικός χάρτης μετά την σάρωση του χώρου (πάνω αριστερά),
 Σημείο εκκίνησης συσκευής (πάνω δεξιά),
 Υπολογισμός βέλτιστης διαδρομής (κάτω αριστερά),
 Τέλος εκτέλεσης προγράμματος και μετακίνηση σε τελική θέση (κάτω δεξιά)

5.4 Τυχαίες κινήσεις

Η τρίτη περίπτωση τρόπου πλοήγησης είναι αυτή των τυχαίων κινήσεων. Η συσκευή είναι σε θέση να πλοηγηθεί στο χώρο εργασίας χωρίς καμία πληροφορία για αυτόν. Μέσα από το πρόγραμμα είναι γραμμένες οι εντολές όπου εκτελεί κάθε φορά που συναντάει κάποιο αντικείμενο ή τοίχο. Οι εντολές αυτές ορίζονται με τρόπο ώστε να γίνεται όσο το δυνατόν καλύτερη η αποφυγή των αντικειμένων. Μπορεί να είναι προκαθορισμένες τιμές όπου εκτελούνται με την σειρά είτε τυχαίες κινήσεις. Η μέθοδος αυτή μπορεί να εκτελείται σε διάφορους χώρους ανεξαρτήτως διαστάσεων και πολυπλοκότητας. Για την περίπτωση των προκαθορισμένων εντολών, η συσκευή έρχεται εύκολα σε επαναλαμβανόμενες κινήσεις με αποτέλεσμα να εγκλωβίζεται σε ένα κομμάτι του συνολικού χώρου, χωρίς να διανύει τον υπόλοιπο. Στην περίπτωση χρήσης των τυχαίων τιμών, χρειάζεται η καλή μελέτη του τρόπου άντλησης των δεδομένων ώστε να υποστηρίζεται η τυχειότητα και να μην υπάρξει πρόβλημα εγκλωβισμού. Η συσκευή πρέπει να είναι σε θέση να εντοπίζει καλά τα αντικείμενα του χώρου και να λαμβάνει αποφάσεις εκείνη την στιγμή για την επιλογή της κατεύθυνσης όπου θα στρίψει [34], [35].

5.5 Επίλογος

Στο Κεφάλαιο αυτό αναφέρθηκαν τα τρία είδη πλοήγησης όπου μπορεί να εκτελέσει μια συσκευή. Ξεκίνησε με την πρόγνωση του χώρου, ορίζοντας την σημασία της και αναφέροντας την λειτουργία της. Στην συνέχεια έγινε η αναφορά στην σάρωση, αναπτύχθηκε ο τρόπος λειτουργίας της και τα πλεονεκτήματά της. Τέλος πραγματοποιήθηκε η εισαγωγή στις τυχαίες κινήσεις και πως αυτές λειτουργούν.

Κεφάλαιο 6ο: Χάος

6.1 Εισαγωγή

Η θεωρία του χάους είναι από τις μεγαλύτερες ανακαλύψεις του 20^{ου} αιώνα. Σύμφωνα με τους μαθηματικούς, είναι δύσκολο να οριστεί το χάος αλλά είναι εύκολο να αναγνωριστεί. Γενικά σημαίνει ότι για τη συμπεριφορά ενός φυσικού συστήματος υπάρχει μια απρόβλεπτη κατάσταση ή με άλλα λόγια υπάρχει μια κατάσταση πλήρους σύγχυσης. Η θεωρία του χάους δηλώνει ότι μικρές διαφορές στις αρχικές συνθήκες, όπως αυτές όπου οφείλονται σε σφάλματα στρογγυλοποίησης, μπορούν να αποδώσουν μεγάλους αποκλεισμούς στα αποτελέσματα των χαοτικών συστημάτων, καθιστώντας αδύνατη τη μακροπρόθεσμη πρόβλεψη. Είναι ένας κλάδος μελέτης των μαθηματικών με εφαρμογές σε διάφορα πεδία όπως η φυσική, η μηχανική, η οικονομία, η βιολογία και η πληροφορική. Στην φυσική βρίσκει χρήση στην πρόγνωση του καιρού, στην εξήγηση της τυρβώδους ροής για τη ρευστοδυναμική, και στην πρόβλεψη της συμπεριφοράς των ουράνιων σωμάτων του ηλιακού συστήματος. Η ενσωμάτωση του χάους στην μηχανική βοηθά στην ανάλυση των δονήσεων σε κατασκευές κι έτσι γίνεται καλύτερος σχεδιασμός όπου αντέχει περισσότερο σε απρόβλεπτες καταστάσεις. Το χρηματιστήριο παρουσιάζει κι αυτό χαοτικές συμπεριφορές καθώς σε μικρές μεταβολές των τιμών, υπάρχει η πιθανότητα να παρουσιαστούν σημαντικές διακυμάνσεις. Για τον τομέα της ιατρικής και της βιολογίας συνεισφέρει στην πρόβλεψη της εξάπλωσης των ασθενειών, βοηθά τους καρδιολόγους να κατανοήσουν τους ακανόνιστους παλμούς της καρδιάς, βελτιώνοντας την θεραπεία των καρδιακών διαταραχών. Στην πληροφορική βρήκε χρήση στην ανάπτυξη αλγορίθμων επικοινωνίας όπου επιτυγχάνεται η κρυπτογράφηση των δεδομένων με χαοτικό τρόπο ώστε να μην είναι εφικτή η γνώση τους, κι έπειτα να γίνεται η αποκρυπτογράφηση τους από το αντίστοιχο σύστημα του δέκτη. Το χάος στην ρομποτική επιλύει το πρόβλημα της κάλυψης ενός άγνωστου εργασιακού χώρου, όπου γίνεται εφικτή η πλοήγηση μιας συσκευής χωρίς την πρόγνωση της τοποθεσίας της. Περισσότερες πληροφορίες σχετικά με τα μαθηματικά αλλά και την χρήση του χάους σε διάφορες εφαρμογές μπορούν να βρεθούν στα παρακάτω άρθρα [36], [37], [38], [39], [40], [41], [43], [44].

6.2 Εισαγωγή στα μαθηματικά του Χάους

Πιο συγκεκριμένα, πρόκειται για την μελέτη πολύπλοκων, μη γραμμικών δυναμικών συστημάτων όπου φαίνονται να είναι προβλέψιμα αλλά, στην πραγματικότητα, υποκρύπτουν χαοτικές συμπεριφορές. Τα χαρακτηριστικά τέτοιων συστημάτων αποτελούνται από τη μη περιοδική συμπεριφορά, τη μη γραμμικότητα, την ευαισθησία στις αρχικές συνθήκες και την απρόβλεπτη κίνηση. Υπάρχουν αρκετές συναρτήσεις όπου οδηγούν στο χάος με τις βασικότερες να είναι ο λογιστικός χάρτης, ο χάρτης του ημίτονου, ο χάρτης του Renyi, ο τροποποιημένος χάρτης σκηνης και ο στρεβλωμένος χάρτης σκηνης.

Για την ανάλυση και την σύγκριση των παραπάνω συναρτήσεων, πρέπει πρώτα να γίνει μια μικρή εισαγωγή σε μερικές σημαντικές έννοιες των μαθηματικών. Από τη μαθηματική πλευρά, τα δυναμικά συστήματα είναι αιτιοκρατικά μοντέλα τα οποία περιγράφουν την κατάσταση ενός συστήματος κατά τη διάρκεια του χρόνου. Ο υπολογισμός των μεταβλητών του συστήματος για μια ορισμένη χρονική στιγμή παρέχει την πληροφορία για τη κατάσταση του συστήματος. Ο χρόνος όπου υπολογίζεται η κατάσταση μπορεί να είναι είτε συνεχής, είτε να παίρνει διακριτές τιμές. Η δημιουργία του διανυσματικού χώρου είναι απαραίτητη προϋπόθεση για την επαρκή γνώση της συμπεριφοράς του συστήματος. Ο διανυσματικός χώρος ονομάζεται χώρος των φάσεων και κατασκευάζεται από ένα σύστημα καρτεσιανών συντεταγμένων, της κάθε μεταβλητή του συστήματος. Αντίστοιχα το φασικό πορτραίτο αποτελεί τη γεωμετρική αναπαράσταση της τροχιάς του δυναμικού συστήματος στο χώρο των φάσεων. Τα δυναμικά συστήματα συνήθως περιγράφονται από ένα σύνολο διαφορικών εξισώσεων πρώτης τάξης, η λύση των οποίων καθορίζει την ύπαρξη ή μη των ελκυστών. Ένα φραγμένο ελκτικό σύνολο είναι το σημείο που περιέχει μια κλειστή τροχιά η οποία γεμίζει πυκνά ένα σημείο στο διάγραμμα. Παρακάτω αναφέρονται οι κατηγορίες των τροχιών ενός δυναμικού συστήματος:

1. Ευθεία τροχιά ή σημείο ισορροπίας

Σε ένα αυτόνομο δυναμικό σύστημα δηλαδή σε σύστημα χωρίς εξάρτηση από το χρόνο, το σημείο ισορροπίας είναι μια σταθερή και ανεξάρτητη από το χρόνο λύση των διαφορικών εξισώσεων του συστήματος. Αντιθέτως στα μη αυτόνομα συστήματα δεν εμφανίζονται σημεία ισορροπίας λόγω της εξάρτησης τους από το χρόνο. Μια τροχιά που ξεκινά από ένα σημείο ισορροπίας παραμένει σε αυτό το σημείο. Στα γραμμικά συστήματα συνήθως εμφανίζεται ένα σημείο ισορροπίας ενώ αντιθέτως στα μη-γραμμικά μπορεί εμφανίζονται πολλαπλά σημεία ισορροπίας.

2. Περιοδική τροχιά

Η περιοδική τροχιά ή οριακός κύκλος είναι μια λύση ενός μη αυτόνομου δυναμικού συστήματος. Ονομάζονται έτσι γιατί οι τιμές των μεταβλητών κατάστασης επαναλαμβάνονται με περίοδο T και δημιουργείται μια κλειστή τροχιά στο χώρο κατάστασης.

3. Ψευδοπεριοδική Τροχιά

Μια συνηθισμένη συμπεριφορά που εμφανίζεται σε μη-γραμμικά συστήματα υψηλής τάξης είναι η ημιπεριοδική ή ψευδοπεριοδική τροχιά. Η χρονική εξέλιξη του συστήματος αναπαρίσταται από μια ελικοειδής κίνηση στην επιφάνειά ενός τοροειδούς. Επίσης, η τροχιά δεν θα τέμνει ποτέ τον εαυτό της και θα καλύψει όλη την επιφάνεια του τοροειδούς.

4. Χαοτική τροχιά

Οι χαοτικές τροχιές έχουν ένα κύριο χαρακτηριστικό, δεν τέμνονται στο χώρο των φάσεων και είναι φραγμένες σε ένα υποχώρο του χώρου των φάσεων. Δύο γειτονικές τροχιές έχουν το χαρακτηριστικό της εκθετικής απόκλισης, οπότε στην περίπτωση που οι αρχικές τιμές έχουν μια ελάχιστη απόκλιση, η απόσταση μεταξύ τους αυξάνεται καθώς εξελίσσεται στο χρόνο. Λόγω της εκθετικής απόκλισης των τροχιών του συστήματος, το σφάλμα από ένα χρονικό σημείο και έπειτα γίνεται τόσο μεγάλο που καθιστά μη προβλέψιμη τη συμπεριφορά του. Στις τροχιές αυτές επειδή εμφανίζονται περίεργες ιδιότητες υπάρχουν και οι παράξενοι ελκυστές. Οι χαοτικοί ελκυστές δεν είναι ένα απλό γεωμετρικό σχήμα αλλά έχουν τη μορφή ενός μορφοκλασματικού σχήματος (fractal) και έχουν μια μη ακέραια κλασματική διάσταση που ονομάζεται φρακταλική διάσταση. Με βάση τη διάσταση αυτή μπορούν να εξαχθούν χρήσιμες πληροφορίες, όπως είναι οι βαθμοί ελευθερίας του δυναμικού συστήματος.

Καθώς μεταβάλλονται οι τιμές της παραμέτρου ελέγχου του συστήματος τόσο ο αριθμός, όσο και η ευστάθεια των σημείων ισορροπίας μπορεί να αλλάξει. Η τιμή της παραμέτρου, στην οποία εμφανίζεται μια τέτοια αλλαγή ονομάζεται σημείο διακλάδωσης. Για τη παρουσίαση των διακλαδώσεων χρησιμοποιούνται τα διαγράμματα διακλάδωσης, τα οποία απεικονίζουν τη θέση και την ευστάθεια του σημείου ισορροπίας συναρτήσει της παραμέτρου διακλάδωσης ή ελέγχου του συστήματος. Τα διαγράμματα διακλάδωσης απεικονίζουν τη συμπεριφορά του δυναμικού συστήματος. Για την επαλήθευση των συμπερασμάτων από τα διαγράμματα, πρέπει να παραχθεί μια ποσότητα, η οποία θα δίνει την πληροφορία για την εξέλιξη του συστήματος. Αυτή η ποσότητα ονομάζεται εκθέτης Lyapunov και αποτελεί το μέτρο της ευαισθησίας του δυναμικού συστήματος από τις αρχικές συνθήκες. Ο εκθέτης Lyapunov εκφράζει τη εκθετική απόκλιση γειτονικών τροχιών στο χώρο των φάσεων. Ένα σύστημα έχει αριθμό εκθετών Lyapunov ίσο με τη διάστασή του. Σύμφωνα με το πρόσημο του εκθέτη εκφράζεται ο ρυθμός σύγκλισης ή απόκλισης των γειτονικών τροχιών. Η χαοτική συμπεριφορά εμφανίζεται με την ύπαρξη ενός θετικού εκθέτη Lyapunov, ενώ στην περίπτωση που δύο ή περισσότεροι εκθέτες είναι θετικοί τότε το σύστημα εμφανίζει υπερχαοτική συμπεριφορά.

Στα δυναμικά συστήματα η χαοτική συμπεριφορά τους μπορεί να αναλυθεί με βάση, είτε τους εκθέτες Lyapunov, είτε με τη μορφοκλασματική διάσταση (fractal dimension) η οποία όμως στηρίζεται στους εκθέτες Lyapunov. Η μορφοκλασματική διάσταση χρησιμοποιείται για να παρέχει έναν ορθολογικό στατιστικό δείκτη της λεπτομέρειας πολυπλοκότητας σε ένα μοτίβο. Σε γενικές γραμμές fractal ονομάζονται τα πολύπλοκα γεωμετρικά σχήματα με λεπτή δομή σε αυθαίρετα μικρές κλίμακες όπου έχουν κάποιο βαθμό αυτοομοιότητας. Δηλαδή αν μεγεθύνουμε με ένα μικροσκοπικό σε ένα τμήμα ενός fractal, θα δούμε χαρακτηριστικά που θυμίζουν το σύνολο, μερικές φορές η ομοιότητα είναι ακριβής, ενώ άλλες είναι μόνο κατά προσέγγιση ή στατιστική αλλά και πάλι θυμίζουν το αρχικό σχήμα. Τα fractals παρουσιάζουν μεγάλο ενδιαφέρον λόγω του εξαιρετικού συνδυασμού ομορφιάς, πολυπλοκότητας και ατελείωτης δομής.

6.3 Έλεγχος τυχαιότητας

Για την χρήση του χάους σε εφαρμογές χρειάζονται οι γεννήτριες ψευδοτυχαίων αριθμών. Οι γεννήτριες αυτές έχουν την ικανότητα να παράγουν, μέσω προκαθορισμένων τεχνικών, αριθμούς, οι οποίοι στη συνέχεια μετατρέπονται στα ζητούμενα δεδομένα, ανάλογα με την εφαρμογή. Η υλοποίηση των γεννητριών της καθιστά να είναι ντετερμινιστικές, δηλαδή οι έξοδοι τους εξαρτώνται από τις εισόδους και για αυτό το λόγο ονομάστηκαν ψευδοτυχαίες γεννήτριες. Συνεπώς η χρήση ενός συστήματος εισόδου, το οποίο δεν είναι προβλέψιμο είναι σημαντικό ζητούμενο για την τυχαιότητα της γεννήτριας. Η πιο συνηθισμένη μορφή εισόδων των συστημάτων είναι τα χαοτικά συστήματα. Αυτό συμβαίνει διότι με τη χρήση των χαοτικών συστημάτων επιτυγχάνεται η πολυπλόθητη τυχαιότητα, με αποτέλεσμα αυτές οι γεννήτριες να μπορούν χρησιμοποιηθούν σε ποικίλες εφαρμογές. Ένα απαραίτητο στάδιο μελέτης για την υλοποίηση μιας γεννήτριας ψευδοτυχαίων αριθμών είναι η επαλήθευση της τυχαιότητας των παραγόμενων αριθμών. Για τον έλεγχο αυτό είναι ευρέως διαδεδομένο να χρησιμοποιείται το πακέτο στατιστικών τεστ, από το National Institute of Standards and Technology (NIST), των Ηνωμένων Πολιτειών, που είναι γνωστό με την ονομασία Federal Information Processing Standards (FIPS) [42]. Το συγκεκριμένο πακέτο περιέχει 15 τεστ τα οποία ελέγχουν την τυχαιότητα της ακολουθίας των αριθμών στην έξοδο της γεννήτριας. Είναι σχεδιασμένα να ελέγξουν ολόκληρη την ακολουθία, να τη χωρίσουν σε υπακολουθίες, είτε να ελέγξουν ξεχωριστά κάθε μία από αυτές, αν πληρούνται τα κριτήρια

των στατιστικών τεστ. Καθένα από αυτά τα τεστ έχει σχεδιαστεί ώστε να ελέγχει, αν ισχύει, μια μηδενική υπόθεση ή όχι. Τα στατιστικά πακέτα του FIPS είναι σχεδιασμένα με τέτοιο τρόπο ώστε να παράγουν μια τιμή, η οποία στην περίπτωση όπου είναι μεγαλύτερη από 0.001, τότε η μηδενική υπόθεση ισχύει και συνεπώς η εξεταζόμενη ακολουθία είναι τυχαία. Σε διαφορετική περίπτωση η ακολουθία λαμβάνεται ως μη τυχαία.

Υπάρχουν διάφορα τεστ που χρησιμοποιούνται για την επαλήθευση της τυχειότητας, ένα από αυτά είναι το πακέτο NIST όπου περιέχει 15 δοκιμές:

1. Το τεστ συχνότητας ενός ψηφίου

Σκοπός του τεστ είναι ο έλεγχος της αναλογίας των μονάδων και των μηδενικών σε μια ακολουθία δυαδικών ψηφίων. Σκοπός είναι ο προσδιορισμός της αναλογίας των μηδενικών και των μονάδων όπου παράγει η γεννήτρια, με αυτά να πρέπει να είναι σχεδόν ίσα ώστε να βγει αποτέλεσμα περίπου 50% και για τα δύο.

2. Τεστ συχνότητας σε ένα κομμάτι της ακολουθίας

Παρόμοιο με το πρώτο αλλά ο έλεγχος γίνεται ανά μερικά ψηφία της ακολουθίας και αυτά θα πρέπει να είναι ίσα με το μισό του πλήθους όπου μετράται για να είναι τυχαία η γεννήτρια.

3. Δοκιμή εκτελέσεων

Σκοπός του τεστ είναι να ελέγξει αν το μήκος των ακολουθιών όπου εμφανίζουν ίδια δυαδικά ψηφία είναι ο αναμενόμενος για μια τυχαία ακολουθία

4. Τεστ μεγαλύτερης σειράς ίδιου ψηφίου

Σκοπός του τεστ είναι η αξιολόγηση της μεγαλύτερης σειράς μονάδων σε ένα κομμάτι της ακολουθίας και η σύγκρισή του με άλλα κομμάτια. Συνήθως οι πολύ μεγάλες ή οι πολύ μικρές ακολουθίες ίδιου ψηφίου υποδηλώνουν τη μη τυχειότητα την γεννήτριας

5. Τεστ κατάταξης δυαδικού πίνακα

Σκοπός του τεστ είναι ο έλεγχος της γραμμικής εξάρτησης μεταξύ κομματιών σταθερού μεγέθους της ακολουθίας.

6. Τεστ διακριτού μετασχηματισμού Fourier

Σκοπός του τεστ είναι η ανίχνευση περιοδικών μοτίβων εφαρμόζοντας τον διακριτό μετασχηματισμό Fourier στην ακολουθία και εξετάζοντας τα ύψη των κορυφών στο φάσμα.

7. Τεστ δοκιμής αντιστοίχισης μη επικαλυπτόμενων προτύπων

Σκοπός του τεστ είναι η καταμέτρηση των εμφανίσεων ενός προκαθορισμένου προτύπου ακολουθίας σε μια σειρά μη επικαλυπτόμενων κομματιών της εξόδου της γεννήτριας.

8. Τεστ δοκιμής αντιστοίχισης επικαλυπτόμενων προτύπων

Σκοπός του τεστ είναι ο ίδιος με του μη επικαλυπτόμενων προτύπων αλλά αυτό επιτρέπει την επικάλυψη της ακολουθίας

9. Καθολικό στατιστικό τεστ

Σκοπός του τεστ είναι η μέτρηση της συμπεστότητας της ακολουθίας, η οποία σχετίζεται με την προβλεψιμότητά της. Η υψηλή συμπεστότητα σημαίνει πολύ προβλέψιμη ακολουθία και υποδηλώνει μη τυχειότητα.

10. Τεστ γραμμικής πολυπλοκότητας

Σκοπός του τεστ είναι η αξιολόγηση της πολυπλοκότητας της ακολουθίας με τον προσδιορισμό του μήκους του συντομότερου καταχωρητή μετατόπισης γραμμικής ανάδρασης ((Linear Feedback Shift Register) που μπορεί να παράγει την ακολουθία τυχαία. Στις τυχαίες ακολουθίες ο καταχωρητής εμφανίζει μεγάλο μήκος.

11. Τεστ σειριακής δοκιμής

Σκοπός του τεστ είναι η επέκταση του τεστ συχνότητας, σε μοτίβα προκαθορισμένων ψηφίων για την ανίχνευση μοτίβων ή κανονικοτήτων και σύγκριση των μετρήσεων με την αναμενόμενη ομοιόμορφη κατανομή.

12. Τεστ προσεγγιστικής εντροπίας

Σκοπός του τεστ είναι η μέτρηση της εντροπίας της ακολουθίας συγκρίνοντας τη συχνότητα των επικαλυπτόμενων μοτίβων.

13. Τεστ συσσωρευτικών αθροισμάτων (Cumulative Sums)

Σκοπός του τεστ είναι η ανίχνευση αποκλίσεων στην ακολουθία από τον αναμενόμενο αριθμό μονάδων και μηδενικών με τη χρήση συσσωρευμένων αθροισμάτων.

14. Τεστ τυχαίων περιηγήσεων

Σκοπός του τεστ είναι η αξιολόγηση του αριθμού και της κατανομής των επισκέψεων για μια ομάδα καταστάσεων κατά τη διάρκεια ενός τυχαίου περιπάτου που προκύπτει από την ακολουθία.

15. Τεστ τυχαίων περιηγήσεων με παραλλαγή

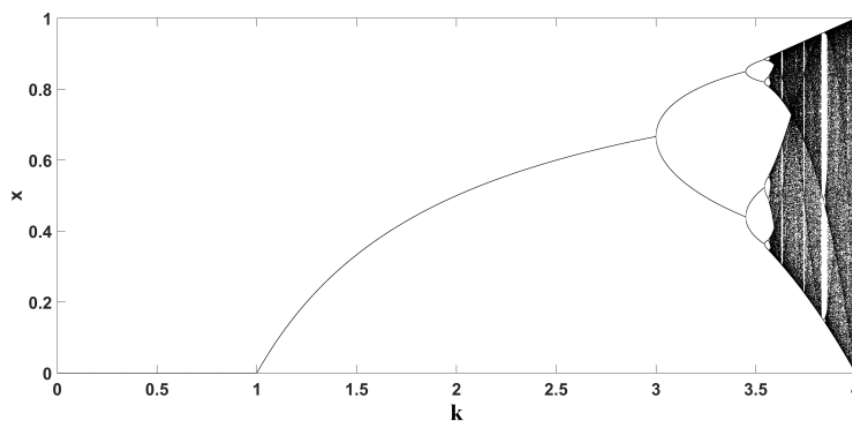
Σκοπός του τεστ είναι παρόμοιος με τη δοκιμή τυχαίων περιηγήσεων, αλλά εξετάζει κάθε μεμονωμένη κατάσταση στον τυχαίο περίπατο.

6.4 Βασικές συναρτήσεις

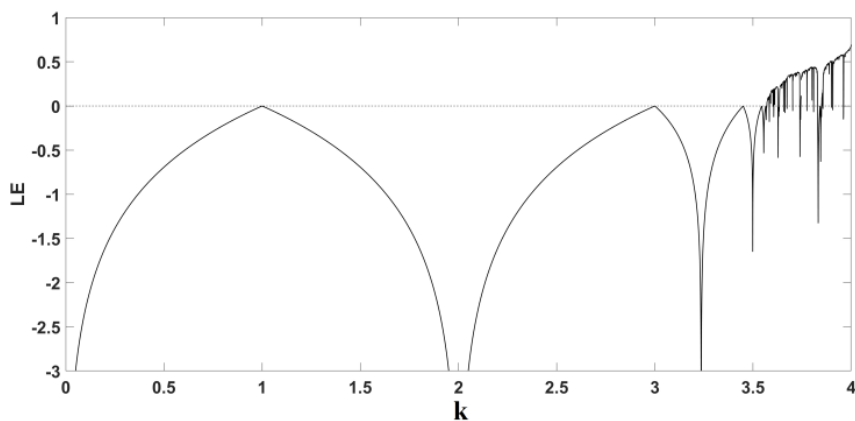
Όπως αναφέρθηκε, αρκετές συναρτήσεις είναι δυνατό να οδηγηθούν προς το χάος και όσο περνάει ο καιρός, εξετάζεται εκτενέστερα το θέμα και βρίσκονται ακόμα περισσότεροι τρόποι. Στο παρόν υποκεφάλαιο θα εξεταστούν μερικές βασικές συναρτήσεις, και θα συγκριθούν κάποια χρήσιμα αποτελέσματα τους. Ίσως μια από τις πιο διαδεδομένες και μελετημένες συναρτήσεις είναι αυτή του λογιστικού χάρτη (logistic map). Η συνάρτηση έχει την μορφή:

$$x_i = k * x_{i-1} * (1 - x_{i-1})$$

Είναι ένας χάρτης μονής μεταβλητής όπου για $k=4$ παράγει τιμές $[0, 1]$, παρακάτω απεικονίζεται, στο Σχήμα 6.1 το διάγραμμα διακλάδωσης και στο Σχήμα 6.2 το διάγραμμα του εκθέτη Lyapunov για διάφορες τιμές του k .



Σχήμα 6.1. Διάγραμμα διακλάδωσης λογιστικού χάρτη

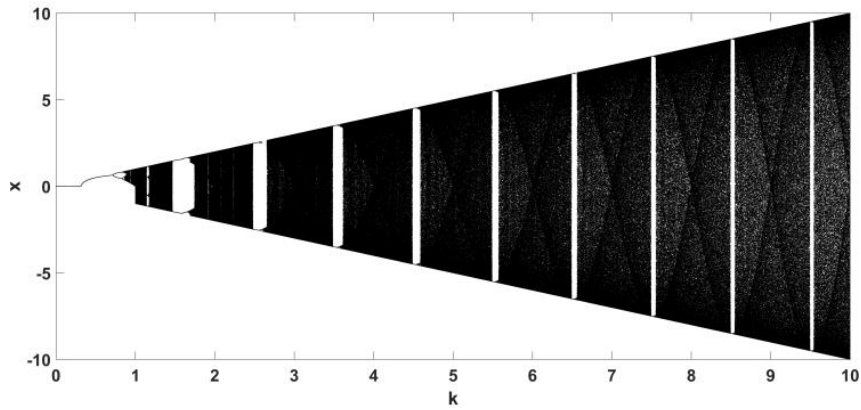


Σχήμα 6.2. Διάγραμμα εκθέτη Lyapunov για λογιστικό χάρτη

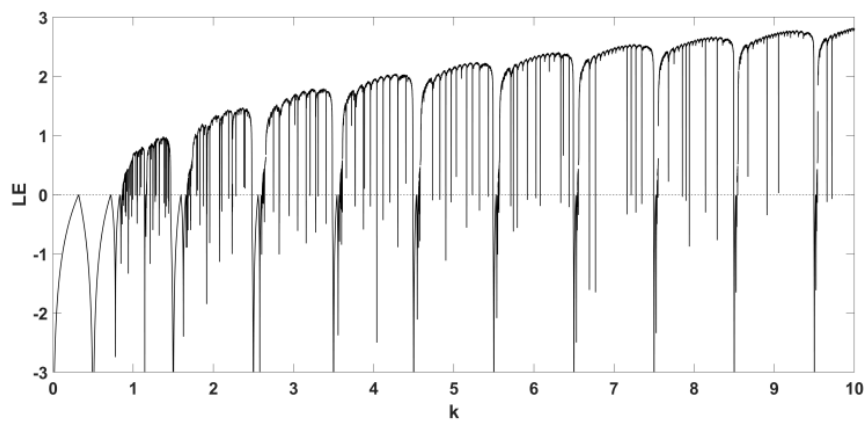
Στην συνέχεια εξετάζεται ο χαοτικός χάρτης του ημιτόνου (Sine map) με την συνάρτησή του να είναι:

$$x_i = k * \sin * (\pi * x_i - 1)$$

Όπως παρατηρείται είναι κι αυτός ένας χάρτης μονής μεταβλητής και παράγει τιμές από $[0, 1]$ για $k=1$. Στα Σχήματα 6.3 και 6.4 απεικονίζονται τα διαγράμματα διακλάδωσης και εκθέτη Lyapunov του χάρτη ημιτόνου.



Σχήμα 6.4. Διάγραμμα διακλάδωσης χάρτη ημιτόνου

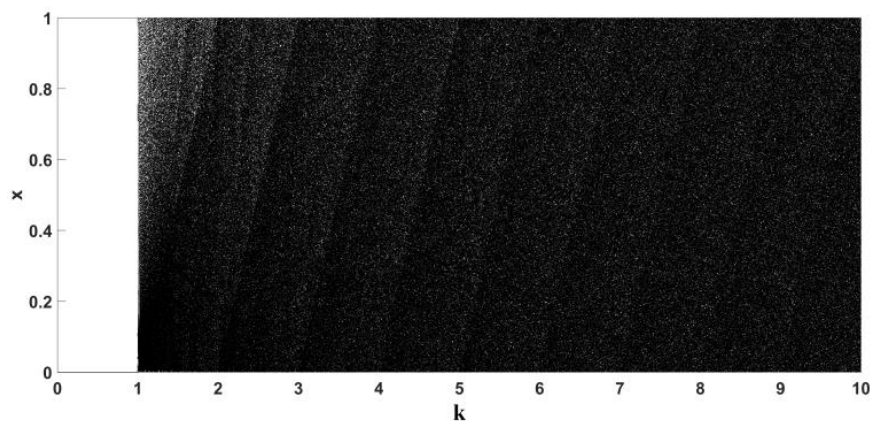


Σχήμα 6.3. Διάγραμμα εκθέτη Lyapunov για χάρτη ημιτόνου

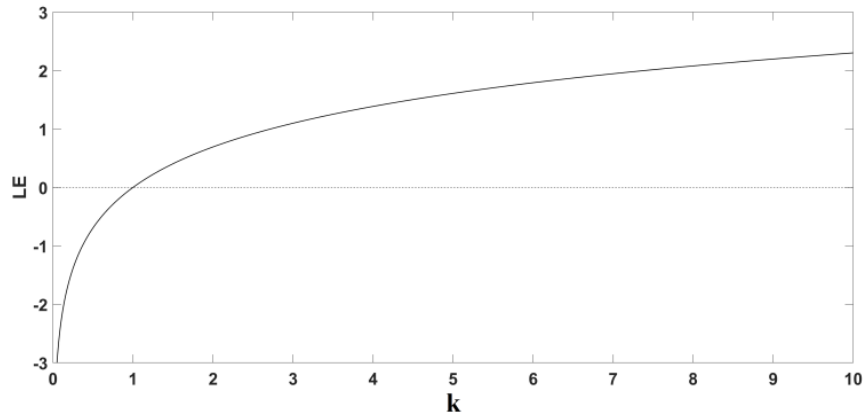
Τρίτος στην σειρά είναι ο χάρτης του Renyi (Renyi map). Η συνάρτησή του είναι:

$$x_i = \text{mod} * (k * x_{i-1}, 1)$$

Κι αυτός είναι μονής μεταβλητής και για τις τιμές όπου $k > 1$ ο χάρτης παράγει έξοδο $[0, 1]$. Παρακάτω, στο Σχήμα 6.5 απεικονίζεται το διάγραμμα διακλάδωσης του χάρτη Renyi. Στο Σχήμα 6.6 προβάλλεται το διάγραμμα εκθέτη Lyapunov και αξίζει να παρατηρηθεί ότι ο εκθέτης ξεπερνάει τη τιμή 2 για $k=10$.



Σχήμα 6.5. Διάγραμμα διακλάδωσης χάρτη Renyi



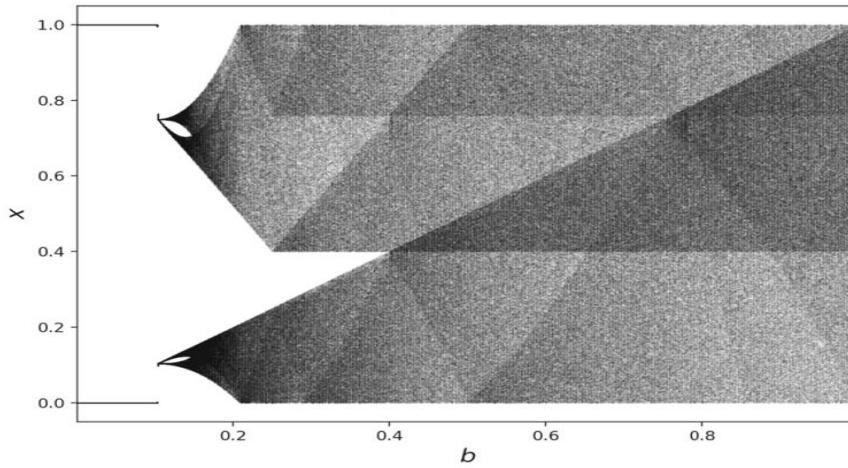
Σχήμα 6.6. Διάγραμμα εκθέτη Lyapunov για χάρτη Renyi

Οι παραπάνω συναρτήσεις υλοποιήθηκαν και αναλύθηκαν καλύτερα από τους συγγραφείς στο [36], [44].

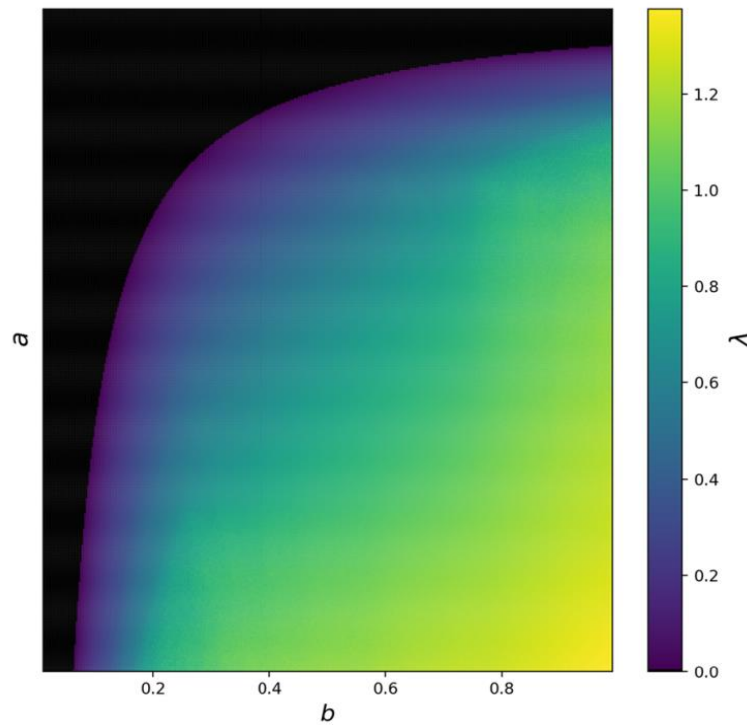
Επόμενος είναι ο τροποποιημένος χάρτης σκηνή (Modified tent map), ο οποίος έχει την συνάρτηση:

$$x_{k+1} = \begin{cases} -\frac{1-a}{p*r} * x_k + 1, & 0 \leq x_k \leq p*r \\ \frac{1-a}{p*(1-r)} * (x_k - p) + 1, & p*r < x_k \leq p \\ \frac{b}{r*(1-p)} * (x_k - p), & p < x_k \leq (1-p)*r + p \\ \frac{b}{(1-r)*(1-p)} * (1 - x_k), & (1-p)*r + p < x_k \leq 1 \end{cases}$$

Όπως παρατηρείται από την συνάρτηση, ο συγκεκριμένος χάρτης παράγει στην έξοδό του $[0, 1]$ ενώ περιέχει και 4 μεταβλητές για την υλοποίησή του. Για την εξαγωγή των διαγραμμάτων θέτονται οι τιμές $a=0.4$, $p=0.5$, $r=0.5$ και λύνεται ως προς b ώστε να προκύψει το σχετικό διάγραμμα διακλάδωσης του τροποποιημένου χάρτη σκηνής που παρουσιάζεται στο Σχήμα 6.7. Για το διάγραμμα εκθετών Lyapunov θέτονται $p=r=0.5$ και λύνεται ως a και b για την εξαγωγή του διαγράμματος που απεικονίζεται στο Σχήμα 6.8. Να σημειωθεί ότι πρέπει να ισχύει $a > p*r$ και $b < (1-p)*r + p$, ώστε η συνάρτηση να μην έχει σταθερό σημείο.



Σχήμα 6.7. Διάγραμμα διακλάδωσης χάρτη τροποποιημένης σκηνής



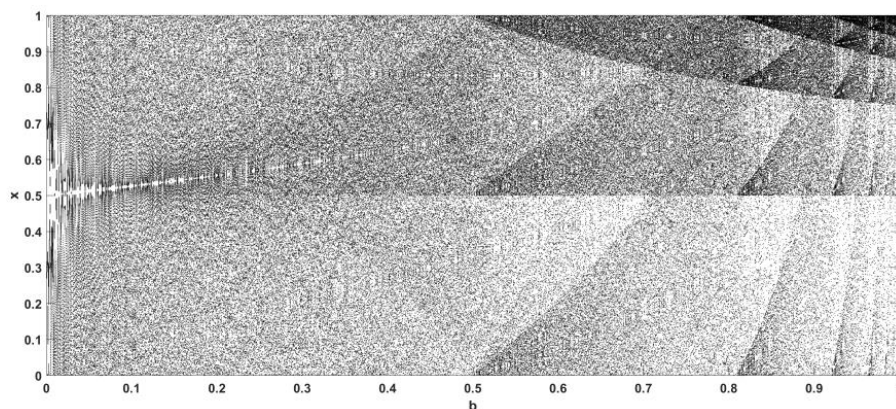
Σχήμα 6.8. Διάγραμμα εκθετών Lyapunov χάρτη τροποποιημένης σκηνής

Η παραπάνω συνάρτηση αναλύθηκε και από τους συγγραφείς στο [37].

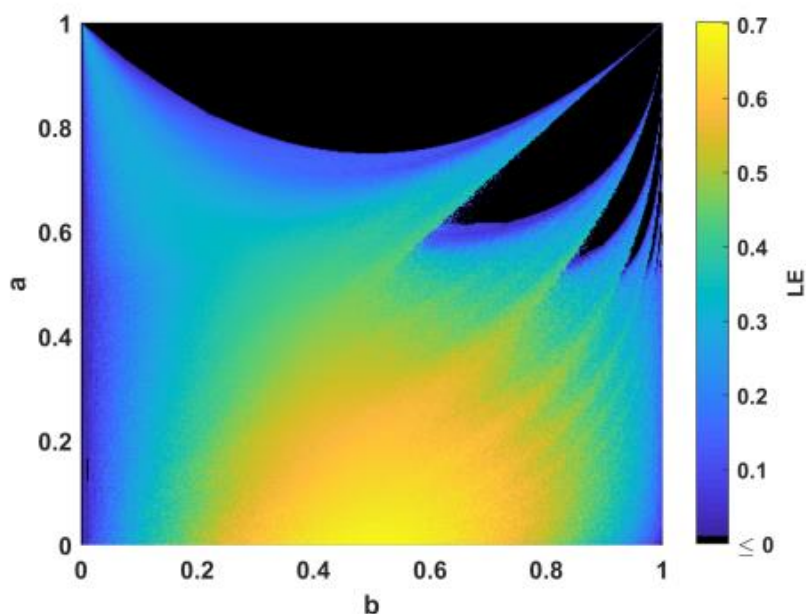
Ο τελευταίος χάρτης όπου θα εξεταστεί είναι ο στρεβλωμένος χάρτης σκηνής (Skewed tent map) με την συνάρτησή του να έχει την μορφή:

$$x_i = \begin{cases} \frac{1-a}{b} * x_{i-1} + a, & 0 \leq x_{i-1} \leq b \\ \frac{1-c}{b-1} * x_{i-1} + \frac{b*c-1}{b-1}, & b < x_{i-1} \leq 1 \end{cases}$$

Όπου όπως φαίνεται από τη συνάρτηση, ο χάρτης περιέχει 3 μεταβλητές. Θέτοντας την τιμή $c=0$ η έξοδος της συνάρτησης εγκλωβίζεται στα όρια $[0, 1]$. Για την εξαγωγή του διαγράμματος διακλάδωσης στρεβλωμένου χάρτη σκινης, η τιμή $c=0$ παραμένει και θέτεται $a=0.5$ ώστε να προκύψει το παρακάτω Σχήμα 6.9. Το διάγραμμα των εκθετών Lyapunov απεικονίζεται στο Σχήμα 6.10.



Σχήμα 6.9. Διαγράμματος διακλάδωσης στρεβλωμένου χάρτη σκινης



Σχήμα 6.10. Διάγραμμα εκθετών Lyapunov στρεβλωμένου χάρτη σκινης

Η παραπάνω συνάρτηση αναλύθηκε και από τους συγγραφείς στο [38]

6.5 Έλεγχος συναρτήσεων

Στο παρόν υποκεφάλαιο έγινε αναφορά στα αποτελέσματα των τεστ FIPS από μερικές συναρτήσεις στην περίπτωση όπου αυτές χρησιμοποιηθούν για τη δημιουργία ψευδοτυχαίας γεννήτριας ψηφίων ή

αριθμών. Η υλοποίηση των εξισώσεων για την επίλυση των τεστ FIPS έγινε από τους συγγραφείς στο άρθρο [37] και στο [44]. Στην περίπτωση του πρώτου άρθρου, ο τροποποιημένος λογιστικός χάρτης υλοποιήθηκε και δοκιμάστηκε ως προς την τυχαιότητά του. Η γεννήτρια δημιουργεί 15 ψηφία αν επανάληψη, χρησιμοποιώντας τον τελεστή modulo για τον κατακερματισμό τους, έναν μετατροπέα για την δυαδική αναπαράσταση και μια πράξη XOR μεταξύ του περισσότερου και λιγότερου σημαντικού ψηφίου της δυαδικής κατακερματισμένης σειράς. Για τις δοκιμές δημιουργήθηκαν σύνολα των 1000 ψηφίων μήκους 10^6 για τυχαίες αρχικές τιμές x_k . Στο δεύτερο άρθρο η υλοποίηση των ψευδοτυχαίων γεννητριών ψηφίων ή αριθμών έγινε υπολογίζοντας την συνάρτηση για τυχαίο αριθμό x_k , στην συνέχεια μετατοπίζοντας τον αριθμό κατά δέκα θέσεις δεξιά και πολλαπλασιάζοντας τον αριθμό με 10^{10} . Το ακαριαίο μέρος του αποτελέσματος περνάει και διαχωρίζεται από έναν τελεστή modulo των 256, ο οποίος επιστρέφει έναν ακέραιο αριθμό στο διάστημα $[0, 255]$. Ο τελικός αριθμός θα περάσει και από ένα μετατροπέα ώστε να μετασχηματιστεί σε δυαδικό σύστημα αρίθμησης. Οι δοκιμές πραγματοποιήθηκαν για 10^9 επαναλήψεις, με τα αποτελέσματά τους να παρουσιάζονται στον Πίνακα 6.1.

Πίνακας 6.1. Αποτελέσματα τεστ NIST για λογιστικού, ημιτόνου, Renyi και τροποποιημένου χάρτη

Τεστ	Λογιστικός	Ημιτόνου	Renyi	Τροποποιημένος
Συχνότητας	Περνάει	Περνάει	Περνάει	Περνάει
Συχνότητας σε κομμάτι	Περνάει	Περνάει	Περνάει	Περνάει
Δοκιμή εκτελέσεων	Περνάει	Περνάει	Περνάει	Περνάει
Μεγαλύτερης σειράς ίδιου ψηφίου	Περνάει	Αποτυχία	Περνάει	Περνάει
Κατάταξη δυαδικού πίνακα	Περνάει	Περνάει	Περνάει	Περνάει
Fourier	Περνάει	Περνάει	Περνάει	Περνάει
Αντιστοίχισης μη επικαλυπτόμενων προτύπων	Περνάει	Αποτυχία	Περνάει	Περνάει
Αντιστοίχισης επικαλυπτόμενων προτύπων	Περνάει	Αποτυχία	Περνάει	Περνάει
Καθολικό στατιστικό	Περνάει	Περνάει	Περνάει	Περνάει
Γραμμικής πολυπλοκότητας	Περνάει	Περνάει	Περνάει	Περνάει
Σειριακής δοκιμής	Αποτυχία	Περνάει	Περνάει	Περνάει
Προσεγγιστικής εντροπίας	Αποτυχία	Περνάει	Περνάει	Περνάει
Συσσωρευτικών αθροισμάτων	Περνάει	Περνάει	Περνάει	Περνάει
Τυχαίων περιηγήσεων	Περνάει	Περνάει	Περνάει	Περνάει
Τυχαίων περιηγήσεων με παραλλαγή	Περνάει	Περνάει	Περνάει	Περνάει

Από τα παραπάνω αποτελέσματα προκύπτει το συμπέρασμα ότι σύμφωνα με την πολυπλοκότητα των συναρτήσεων είναι πιθανό να μην βγουν απολύτως αποτελεσματικοί για την αυτούσια χρήση τους σε γεννήτριες ψευδοτυχαίων αριθμών.

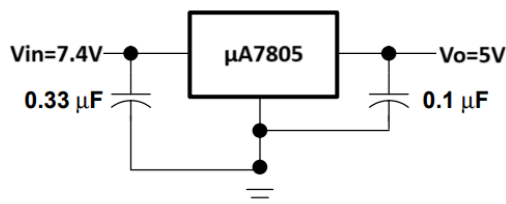
6.6 Επίλογος

Από το παραπάνω Κεφάλαιο είναι σημαντική η κατανόηση των βασικών εννοιών για την θεωρία του χάους. Οι διάφορες συναρτήσεις μπορούν με την σωστή υλοποίηση να οδηγήσουν σε απρόβλεπτες καταστάσεις το σύστημα και ανάλογα με την κατανόηση υπάρχει η επιλογή της χρήσης τους με διάφορους τρόπους. Έγινε η αναφορά και μια μικρή εξήγηση βασικών συναρτήσεων και παρουσιάστηκε ένας τρόπος ανάλυσης των αποτελεσμάτων τους. Τέλος, υλοποιήθηκαν σε εφαρμογές οι συναρτήσεις και συγκρίθηκαν ως προς την τυχειότητα που προσφέρουν μέσω των τεστ NIST.

Κεφάλαιο 7ο: Υλοποίηση

7.1 Εισαγωγή

Στο παρόν Κεφάλαιο αναφέρονται λεπτομερώς οι διαδικασίες που πραγματοποιήθηκαν για την υλοποίηση της αυτόματης σκούπας, με αυτόματη πλοήγηση στο χώρο, μέσω χαοτικής συνάρτησης. Αρχικά, ξεκίνησε η επιλογή των υλικών, καθώς αυτά καθόρισαν και τις λειτουργίες της κατασκευής. Για την τροφοδοσία της, χρησιμοποιήθηκε μια μπαταρία ιόντων λιθίου με ονομαστική τάση 7,4Volt. Καθώς όμως τα αισθητήρια και ο επεξεργαστής δεν λειτουργούν με 7,4V αλλά με 5V, χρειάστηκε κι ένα κύκλωμα step down από συνεχή σε συνεχή τάση για την τροφοδοσία τους. Γι αυτή την δουλειά επιλέχθηκε ο ρυθμιστής τάσης lm7805 σε συνδεσμολογία σταθερής εξόδου όπως φαίνεται και στο Σχήμα 7.1.



Σχήμα 7.1. Κύκλωμα ρυθμιστή lm7805 σταθερής τάσης εξόδου

Στην περίπτωση των αισθητηρίων, χρησιμοποιήθηκαν δύο είδη, οι αισθητήρες υπερήχων και οι αισθητήρες υπέρυθρων ακτίνων, τα χαρακτηριστικά των οποίων αναλύθηκαν στο Κεφάλαιο 3. Η επιλογή των αισθητήρων υπέρυθρων έγινε γιατί έχουν μελετηθεί αρκετά για την μέτρηση απόστασης και κρίθηκε ότι είναι ιδανικοί και για τη συγκεκριμένη κατασκευή. Έπειτα από δοκιμές, τοποθετήθηκαν τρεις στο μπροστινό μέρος, σε τέτοια απόσταση και κλίση, έτσι ώστε να επιτρέπει την κατασκευή να σκανάρει τον χώρο για αντικείμενα χωρίς κενά και ούτε πολλές επικαλύψεις στην εμβέλειά τους. Μια σημαντική λειτουργία των σύγχρονων αυτόματων σκουπών είναι αυτή της επίβλεψης δαπέδου. Η πληροφορία της ύπαρξης του πατώματος ή όχι είναι υψηλής σημασίας για την κατασκευή καθώς έτσι γνωρίζει ότι υπάρχει ή όχι μέρος μπροστά της για να κινηθεί. Αυτό επιτυγχάνεται με την χρήση των αισθητήρων ανίχνευσης αντικειμένων με υπέρυθρες ακτίνες όταν ο αισθητήρας στρέφεται προς τα κάτω και επιβλέπει αν υπάρχει ή δεν υπάρχει το πάτωμα. Επιλέχθηκε να τοποθετηθούν δύο, ένας σε κάθε πλευρά, για την βέλτιστη ανίχνευση.

Η κίνηση, όπως αναφέρθηκε στο Κεφάλαιο 4 επιλέχθηκε να γίνει με δύο κινητήρες συνεχούς ρεύματος όπως αυτοί που παρουσιάστηκαν στο Κεφάλαιο 3. Τα χαρακτηριστικά που δεν αναφέρθηκαν προηγουμένως είναι η ταχύτητα περιστροφής χωρίς φορτίο όπου φτάνει τα 30rpm και η ροπή τους όπου φτάνει το $1.5\text{kg}\cdot\text{cm}$. Κρίθηκαν επαρκή για την μεταφορά του βάρους της κατασκευής και για να μην επιβαρυνθεί η πολυπλοκότητά της, τοποθετήθηκε κι ένας παθητικός τροχός για την στήριξη. Ο έλεγχός τους γίνεται μέσω προγράμματος στον μικροεπεξεργαστή και η οδήγηση από το κύκλωμα οδήγησης L298N του Κεφαλαίου 3. Για την πλοήγηση στον χώρο μελετήθηκε και γράφτηκε κώδικας ώστε να επιτρέπει την συσκευή να κινείται και να στρίβει και στις οκτώ κατευθύνσεις των δύο αξόνων. Πρακτικά, η πλοήγηση σε ευθεία πραγματοποιείται με την κίνηση του δεξιά κινητήρα με την φορά του ρολογιού ενώ του αριστερού κινητήρα με την αντίθετη φορά. Η στροφές επιτυγχάνονται σε δύο στάδια, πρώτα γίνεται η στροφή της κατασκευής προς την θέση όπου χρειάζεται να στρίψει, ενώ βρίσκεται σε σταθερό σημείο, κι έπειτα πραγματοποιείται η κίνηση σε ευθείας προς την κατεύθυνση όπου κατέληξε μετά την στροφή. Στον παρακάτω Πίνακα 7.1 παρουσιάζονται οι κατευθύνσεις και ο τρόπος υλοποίησής τους.

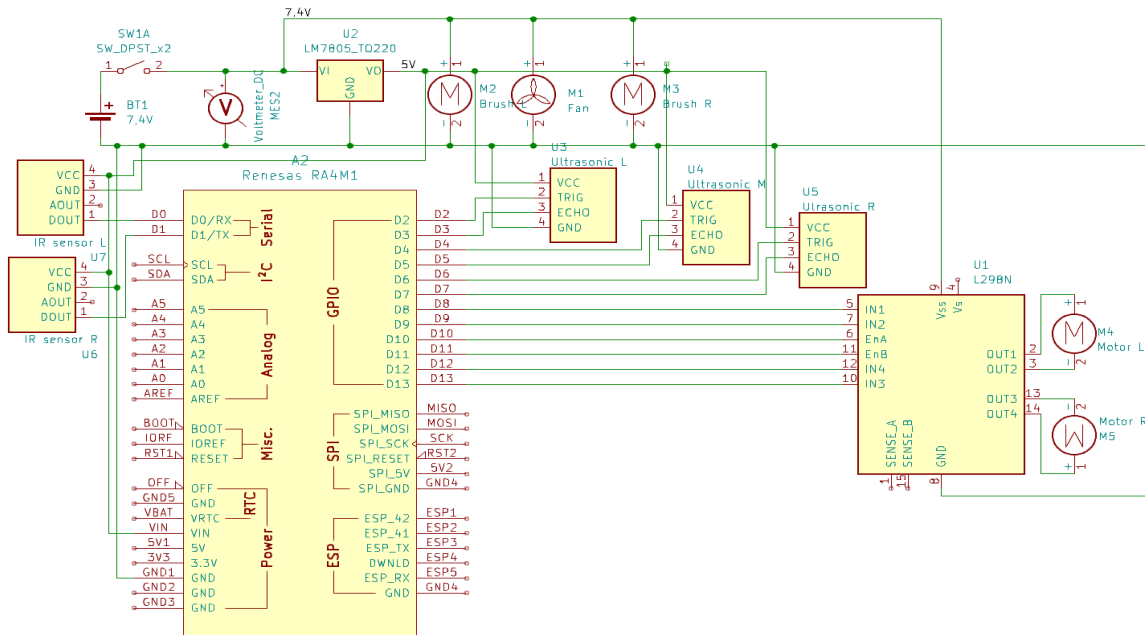
Πίνακας 7.1. Κατευθύνσεις κίνησης κατασκευής

Κατεύθυνση πλοήγησης	Αριστερός κινητήρας	Δεξιός κινητήρας	Χρόνος
Ευθεία	Αντίθετη Φορά του ρολογιού	Φορά του ρολογιού	-
Μπροστά-Δεξιά	Αντίθετη Φορά του ρολογιού	Αντίθετη Φορά του ρολογιού	1.5 sec
Δεξιά	Αντίθετη Φορά του ρολογιού	Αντίθετη Φορά του ρολογιού	3 sec
Πίσω-Δεξιά	Αντίθετη Φορά του ρολογιού	Αντίθετη Φορά του ρολογιού	4.5 sec
Πίσω	Φορά του ρολογιού	Αντίθετη Φορά του ρολογιού	-
Πίσω-Αριστερά	Φορά του ρολογιού	Φορά του ρολογιού	4.5 sec
Αριστερά	Φορά του ρολογιού	Φορά του ρολογιού	3 sec
Μπροστά-Αριστερά	Φορά του ρολογιού	Φορά του ρολογιού	1.5 sec

Φυσικά σε μια αυτόματη σκούπα, πέρα από την πλοήγηση, σημασία έχει και το σύστημα αναρρόφησης που διαθέτει. Για τη συγκεκριμένη κατασκευή χρησιμοποιήθηκε ένα μικρό σκουπάκι τα χαρακτηριστικά του οποίου εξετάστηκαν στο Κεφάλαιο 3. Τοποθετήθηκε πάνω στην βάση, ενώ το στόμιο από όπου γίνεται η αναρρόφηση περνάει μέσα από μια τρύπα και φτάνει μερικά χιλιοστά πάνω από το πάτωμα για την καλύτερη δυνατή αποτελεσματικότητα. Η θέση της τρύπας επιλέχθηκε λίγο πάνω από το κέντρο της κατασκευής γιατί προσφέρει τον απαραίτητο χώρο για να τοποθετηθεί το σκουπάκι στην βάση. Για την καλύτερη κάλυψη και απόδοση της συσκευής, επιλέχθηκε να τοποθετηθεί κι ένα σύστημα βοήθειας για την αναρρόφηση όπου περιέχει δύο κινητήρες με βουρτσάκια στην άκρη του άξονα περιστροφής τους. Οι κινητήρες που επιλέχθηκαν έχουν τα ίδια χαρακτηριστικά τροφοδοσίας και διαστάσεων με αυτούς που περιγράφονται στο Κεφάλαιο 3 και ταχύτητα περιστροφής τα 200rpm. Η φορά περιστροφής τους είναι αντίθετη, με το αριστερό να γυρίζει με τη φορά του ρολογιού και με το δεξί να γυρίζει αντίθετα από το ρολόι. Η λειτουργία που προσφέρουν είναι το σκούπισμα στην ακτίνα όπου φτάνει το βουρτσάκι του καθενός. Λόγο της φοράς τους μεταφέρουν τις σκόνες και τα σκουπίδια προς το κέντρο της κατασκευής κι έτσι καταφέρνει να τα ρουφήξει το σύστημα αναρρόφησης.

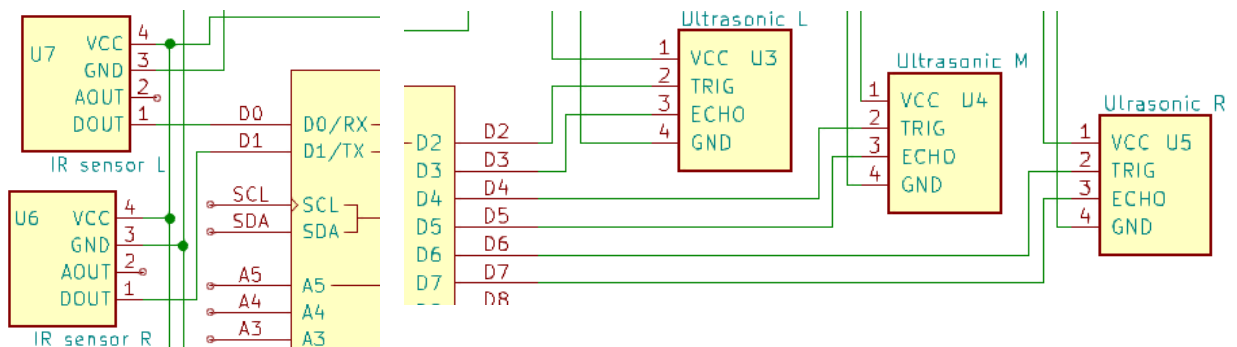
7.2 Σχεδίαση και μελέτη σε πρόγραμμα

Η μελέτη του ηλεκτρονικού κυκλώματος σχεδιάστηκε αρχικά σε πρόγραμμα προσομοίωσης.



Σχήμα 7.2. Σχηματικό ηλεκτρονικό κύκλωμα αυτόματης σκούπας

Στο Σχήμα 7.2 παρουσιάζεται το ολοκληρωμένο ηλεκτρονικό κύκλωμα της αυτόματης σκούπας και όλες οι συνδέσεις όπου πρέπει να γίνουν για την λειτουργία της. Όπως αναφέρθηκε και στην αρχή του Κεφαλαίου, υπάρχουν δύο πηγές τάσης για την τροφοδοσία όλων των εξαρτημάτων που χρησιμοποιήθηκαν και φαίνονται στην πάνω αριστερή γωνία, με ονομασία BT1 είναι η μπαταρία όπου προσφέρει τα 7,4V στο κύκλωμα και συνδέεται με το βολτόμετρο, το κύκλωμα οδήγησης των κινητήρων L298N, το σκουπάκι αναρρόφησης M1, τους δύο κινητήρες βοήθειας της αναρρόφησης M2, M3 και τον ρυθμιστή της τάσης LM7805, ώστε να παρέχει κι αυτός τα 5V με την σειρά του. Σε αυτόν είναι συνδεδεμένα τα υπόλοιπα υλικά όπου είναι, οι πέντε αισθητήρες που χρησιμοποιήθηκαν, οι 2 υπέρυθροι U6, U7 και οι 3 υπέρηχοι U3, U4, U5, και ο επεξεργαστής RA4M1 της κατασκευής. Πέραν από τους κινητήρες M1, M2, M3, όπου χρειάζονται μόνο τροφοδοσία και γείωση, κάθε εξάρτημα συνδέεται και με τον επεξεργαστή ώστε να επιτευχθεί η επικοινωνία μεταξύ τους και να γίνει ο σωστός έλεγχος της κατασκευής.

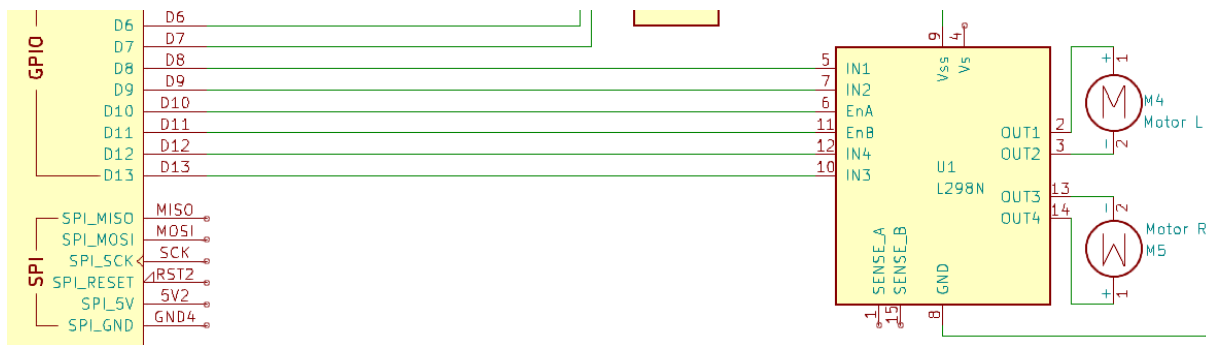


Σχήμα 7.3. Συνδέσεις των υπέρηχων αισθητήρων με τον επεξεργαστή

Σχήμα 7.4. Συνδέσεις των υπέρυθρων αισθητήρων με τον επεξεργαστή

Στα Σχήματα 7.3 και 7.4 παρατηρούνται οι συνδέσεις των αισθητηρίων με υπέρηχους και με υπέρυθρες ακτίνες με τον επεξεργαστή. Ξεκινώντας από τους πρώτους, οι υπέρηχοι αισθητήρες χρησιμοποιούν δύο ακροδέκτες για την επικοινωνία τους με τον επεξεργαστή. Όπως αναφέρθηκε και στο Κεφάλαιο 3, χρειάζονται ένα σήμα σκανδαλισμού όπου το λαμβάνουν από τον ακροδέκτη TRIG για να ξεκινήσει η διαδικασία μέτρησης της απόστασης. Μετά την λήξη της, μεταφέρουν την μέτρηση στον επεξεργαστή μέσω του ακροδέκτη ECHO. Από την μεριά του, πρέπει να γίνει η αρχικοποίηση των ακροδεκτών D2, D4, D6 σε ακροδέκτη εξόδου και των D3, D5, D7 σε ακροδέκτη εισόδου ώστε να ολοκληρωθεί η προετοιμασία και να λειτουργήσει ανάλογα με το πρόγραμμα. Για τους υπέρυθρους αισθητήρες τα πράγματα είναι πιο απλά, πέρα από τροφοδοσία και γείωση, το μόνο που χρειάζονται είναι ένας ακροδέκτης για την μεταφορά της πληροφορίας. Μέσω του DOUT επικοινωνούν με το D0 και D1, όπου πρέπει να αρχικοποιηθούν ως εισοδοί, κι έτσι ολοκληρώνεται η συνδεσμολογία τους και εκτελούνται κανονικά οι λειτουργίες του προγράμματος.

Τέλος, μένει και η σύνδεση του κυκλώματος οδήγησης για την ολοκλήρωση της σχεδίασης στο πρόγραμμα.



Σχήμα 7.5. Συνδέσεις κυκλώματος οδήγησης κινητήρων

Όπως προαναφέρθηκε, για τον έλεγχο των κινητήρων χρειάζεται ένα κύκλωμα οδήγησης καθώς ο επεξεργαστής δεν είναι ικανός να παρέχει αρκετό ρεύμα για την απευθείας κίνησή τους. Το ολοκληρωμένο κύκλωμα με L298N που χρησιμοποιήθηκε, χρειάζεται έξι εξόδους από τον επεξεργαστή, με τις δύο από αυτές να είναι ικανές να παράγουν PWM σήματα. Η συνδεσμολογία έγινε όπως φαίνεται στο Σχήμα 7.5, δηλαδή οι τρεις ακροδέκτες εξόδου για τον έλεγχο του πρώτου κινητήρα είναι οι D8, D9 και D10 με τον τελευταίο να παράγει τα PWM σήματα για τον έλεγχο και συνδέονται με τα IN1, IN2 και ENA αντίστοιχα. Ο δεύτερος κινητήρας συνδέεται με τον επεξεργαστή μέσω των ακροδεκτών D11 με ENB, D12 με IN4 και D13 με IN3. Ο D11 είναι αυτός που ελέγχει τον κινητήρα μέσω των PWM. Μέσω του προγράμματος ο επεξεργαστής κατευθύνει την φορά των κινητήρων, αλλάζοντας την έξοδο των D8 και D9 για τον πρώτο κινητήρα και των D12, D13 για τον δεύτερο. Από τα PWM στους ακροδέκτες D10 και D11 ο επεξεργαστής ορίζει την ταχύτητα περιστροφής των κινητήρων αντίστοιχα.

7.3 Τυχαίες κινήσεις

Για την επιλογή της συνάρτησης, πέρα από τα τεστ του Κεφαλαίου 6, πραγματοποιήθηκαν και περαιτέρω δοκιμές. Ο υπολογισμός των τεστ NIST γίνεται με την μετατροπή των ψηφίων της εξόδου

της κάθε συνάρτησης σε μηδέν και άσους, πράγμα που δεν χρειάζεται στην περίπτωση της συγκεκριμένης κατασκευής καθώς, όπως θα αναλυθεί και παρακάτω, η επιλογή της κατεύθυνσης γίνεται από ένα φάσμα τιμών μηδέν έως οκτώ. Οι υλοποίηση και η μέτρηση των συναρτήσεων στις παρακάτω δοκιμές έγινε με τις τιμές παραμέτρων όπου παρουσιάζονται στον Πίνακα 7.2.

Πίνακας 7.2. Τιμές μεταβλητών για τον υπολογισμό των συναρτήσεων

Συνάρτηση	Μεταβλητές
Λογιστικός χάρτης	$k=4$
Χάρτης ημιτόνου	$k=1$
Χάρτης Renyi	$k=19.9$
Τροποποιημένος χάρτης σκηνής	$a=0.4$, $b=0.41$, $p=r=0.5$
Στρεβλώμενος χάρτης σκηνής	$a=0.23$, $b=0.65$, $c=0$

7.3.1 Επιλογή συνάρτησης

Ο πρώτος έλεγχος που πραγματοποιήθηκε αφορούσε τον χρόνο εκτέλεσης των συναρτήσεων καθώς είναι σημαντικό να μην υπάρχει καθυστέρηση στην επιλογή την κατεύθυνσης όπου θα στρίψει η κατασκευή. Για τις μετρήσεις, έγινε η υλοποίηση των συναρτήσεων σε πρόγραμμα, στην συνέχεια εκτελέστηκαν για τον αριθμό εκτελέσεων που ορίστηκε και μετρήθηκε ο συνολικός χρόνος όπου χρειάστηκε από την αρχή μέχρι και την ολοκλήρωσή των επαναλήψεων. Στον Πίνακα 7.3 απεικονίζονται οι μετρούμενοι χρόνοι εκτέλεσης μετά από 1000 και 1000000 επαναλήψεις.

Πίνακας 7.3. Αποτελέσματα χρόνου εκτέλεσης των συναρτήσεων

Συναρτήσεις	Χρόνοι εκτέλεσης 1000 επαναλήψεις	Χρόνοι εκτέλεσης 1000000 επαναλήψεις
Logistic map	2 microsec / 0.00 sec	2 microsec / 0.00 sec
Renyi map	9762 microsec / 0.01 sec	9758226 microsec / 9.76 sec
Sine map	48223 microsec / 0.05 sec	47522204 microsec / 47.52 sec
Modified tent map	2 microsec / 0.00 sec	2 microsec / 0.00 sec
Skewed tent map	2 microsec / 0.00 sec	2 microsec / 0.00 sec

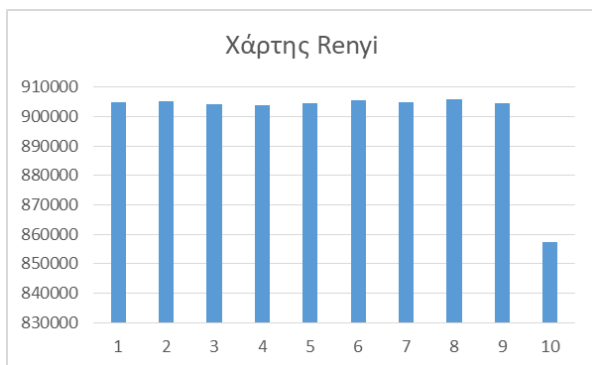
Από τα αποτελέσματα φαίνεται το λογικό συμπέρασμα, ότι η συσκευή, όσο περισσότερες φορές εκτελεί και υπολογίζει την συνάρτηση που πρέπει, τόσο ο χρόνος ολοκλήρωσης των εκτελέσεων μεγαλώνει. Σημαντική όμως είναι η παρατήρηση ότι ανάλογα με την συνάρτηση, μπορεί να προστεθεί είτε επιπλέον μετρήσιμος χρόνος, όπου θα καθυστερήσει την ολοκλήρωση της εργασίας, είτε μπορεί να είναι μηδαμινός. Αυτό σημαίνει ότι για την ελαχιστοποίηση του χρόνου εκτέλεσης της εργασίας, καλό θα ήταν να ληφθούν υπόψη τα παραπάνω αποτελέσματα στην επιλογή της συνάρτησης.

Ο δεύτερος έλεγχος είχε να κάνει με την μέτρηση των ποσοστών εμφάνισης του κάθε ψηφίου στην έξοδο της συνάρτησης, καθώς αυτά είναι που θα καθορίζουν την κατεύθυνση. Οι δοκιμές έγιναν εκτελώντας την κάθε συνάρτηση, για κάθε αρχική τιμή της από $[0,0, 1,0]$ με βήμα $0,1$. Να σημειωθεί όμως, ότι για αρχική τιμή $0,5$ κάποιες συναρτήσεις δεν υλοποιούνται σωστά κι έτσι τα αποτελέσματά τους δεν συμπεριλήφθηκαν στην μέτρηση. Στην συνέχεια υπολογίστηκε το ποσοστό εμφάνισης του κάθε αριθμού και το εύρος της απόστασης από την μεγαλύτερη τιμή επαναλήψεων, μέχρι και την μικρότερη. Στην συνέχεια έγινε και η απεικόνιση των δεδομένων, ώστε να γίνει η διευκόλυνση της κατανόησής τους.

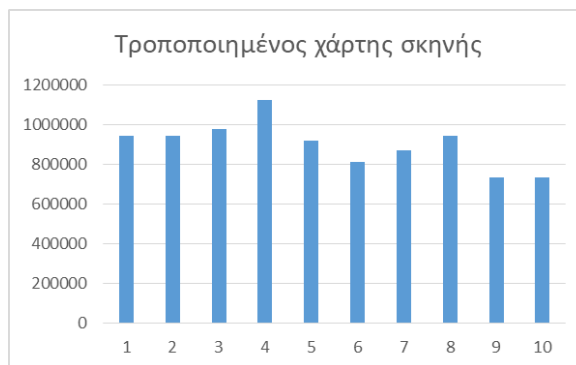
Πίνακας 7.4. Αποτελέσματα ποσοστών εμφάνισης του κάθε αριθμού

	0	1	2	3	4	5	6	7	8	9	Εύρος (max-min)
logistic	1843829	814346	663914	601263	577963	577514	601425	663208	812674	1843864	1266350
	20.49%	9.05%	7.38%	6.68%	6.42%	6.42%	6.68%	7.37%	9.03%	20.49%	14.07%
Renyi	904783	905144	904173	903650	904487	905526	904965	905681	904364	857227	48454
	10.05%	10.06%	10.05%	10.04%	10.05%	10.06%	10.06%	10.06%	10.05%	9.52%	0.54%
Sine	2088604	884478	700262	615641	574975	561484	571983	618535	740744	1643294	1527120
	23.21%	9.83%	7.78%	6.84%	6.39%	6.24%	6.36%	6.87%	8.23%	18.26%	16.97%
Modified	943683	941761	979914	1123001	920053	814055	869807	942894	732526	732306	390695
	10.49%	10.46%	10.89%	12.48%	10.22%	9.05%	9.66%	10.48%	8.14%	8.14%	4.34%
Skewed	450765	450226	715109	830278	826075	1063002	1077519	1076885	1224773	1285368	835142
	5.01%	5.00%	7.95%	9.23%	9.18%	11.81%	11.97%	11.97%	13.61%	14.28%	9.28%

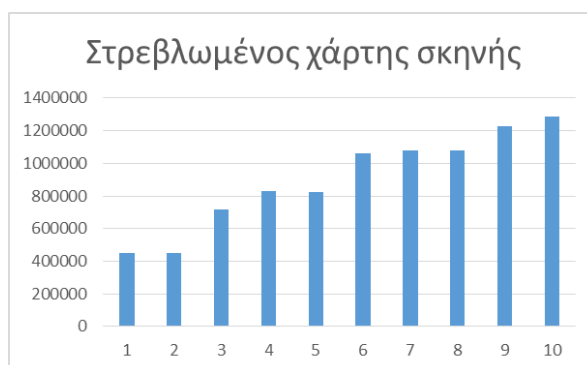
Στον Πίνακα 7.4 απεικονίζονται οι τιμές των αριθμών όπου προκύπτουν μετά από 900000 εκτελέσεων, 1000 για κάθε αρχική τιμή από $0,0$ μέχρι και $1,0$ εκτός από $0,5$ όπως προαναφέρθηκε. Πολύ σημαντική στήλη είναι αυτή του εύρους, καθώς δείχνει πόσο απλωμένα είναι τα ποσοστά των εμφανίσεων του κάθε αριθμού. Όσο μικρότερο είναι το εύρος, τόσο πιο μαζεμένα είναι τα ποσοστά κι αυτό σημαίνει ότι οι αριθμοί εμφανίζονται παρόμοιες φορές, πράγμα καλό για την τυχαιότητα των κινήσεων. Όπως φαίνεται από την τελευταία στήλη, οι συναρτήσεις με το καλύτερο εύρος είναι ο χάρτης Renyi, δεύτερος στην σειρά είναι ο τροποποιημένος χάρτης σκηνης, αμέσως μετά ο στρεβλωμένος χάρτης σκηνης, στην συνέχεια είναι ο λογιστικός χάρτης και στο τέλος ο χάρτης ημιτόνου. Παρακάτω προβάλλονται σε διαγράμματα οι τιμές των εμφανίσεων για την καλύτερη ανάγνωση των αποτελεσμάτων.



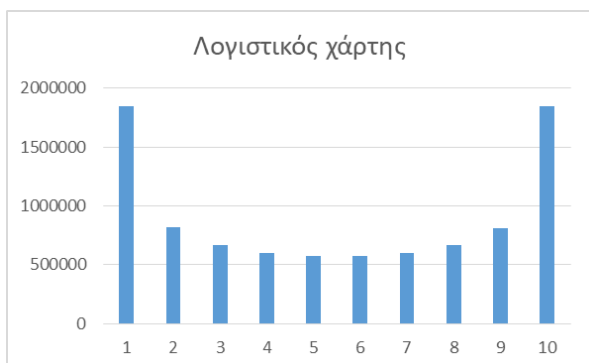
Σχήμα 7.8. Διάγραμμα απεικόνισης εμφάνισης κάθε αριθμού στον χάρτη Renyi



Σχήμα 7.7. Διάγραμμα απεικόνισης εμφάνισης κάθε αριθμού στον τροποποιημένο χάρτη σκηνης



Σχήμα 7.6. Διάγραμμα απεικόνισης εμφάνισης κάθε αριθμού στον στρεβλωμένο χάρτη σκηνης



Σχήμα 7.10. Διάγραμμα απεικόνισης εμφάνισης κάθε αριθμού στον λογιστικό χάρτης



Σχήμα 7.9. Διάγραμμα απεικόνισης εμφάνισης κάθε αριθμού στον χάρτη ημιτόνου

Το συμπέρασμα παρατηρείται και από τα διαγράμματα καθώς είναι ξεκάθαρη η ομοιομορφία του χάρτη Renyi με τις εμφανίσεις του κάθε αριθμού να είναι σχεδόν ίσες πέρα από το δέκα όπου κι αυτό όμως είναι πολύ κοντά, αν συγκριθεί με τις αποστάσεις των άλλων χαρτών. Στην συνέχεια είναι ο τροποποιημένος χάρτης σκηνης όπου και πάλι είναι κοντά μεταξύ τους οι εμφανίσεις των τιμών αλλά παρατηρείται μια ανύψωση στο τέσσερα και μια πτώση στο εννέα και δέκα, συγκριτικά με τα υπόλοιπα νούμερα. Στον στρεβλωμένο χάρτη σκηνης είναι φανερή η αύξηση των εμφανίσεων όσο μεγαλώνουν

τα νούμερα, με το δέκα να κάνει τις περισσότερες και το μηδέν να φτάνει περίπου το $1/3$ των εμφανίσεων. Ο λογιστικός χάρτης και ο χάρτης ημιτόνου κατέχουν παρόμοια διαγράμματα όπου και τα δύο εμφανίζουν τις ακραίες περιπτώσεις τους περισσότερες φορές απ' ότι στις μεσαίες.

Από τις παραπάνω συγκρίσεις έγινε η επιλογή του τροποποιημένου χάρτη σκηνης ως την συνάρτηση που θα εκτελείται για τον υπολογισμό της κατεύθυνσης της κατασκευής. Βάση δόθηκε στο εύρος και τα ποσοστά εμφάνισης των τιμών όπου εξάγουν αλλά, το γεγονός ότι ο χάρτης του Remyi ήταν αρκετά πιο αργός από τον τροποποιημένο χάρτη σκηνης, οδήγησε στην επιλογή του δεύτερου στην σειρά.

7.3.2 Αντιστοίχιση κατευθύνσεων

Όπως αναφέρθηκε και προηγουμένως η κίνηση της κατασκευής γίνεται και στις οκτώ κατευθύνσεις των δύο αξόνων. Για την επιλογή της φοράς περιστροφής, χρησιμοποιούνται τα αποτελέσματα της συνάρτησης, στα οποία γίνεται μια αντιστοιχία με τις κατευθύνσεις. Πιο συγκεκριμένα, στο πρόγραμμα μετά την εκτέλεση της συνάρτησης, γίνεται η αντιστοιχία των τιμών 0,0 μέχρι 1,0 όπου εξάγονται, σε ένα σετ των επτά και των έξι κατευθύνσεων. Για το σετ των επτά κατευθύνσεων γίνεται η αντιστοίχιση από $[0,0, 1,0]$ σε τιμές $[1, 8)$, ενώ παρομοίως, για το σετ των 6 κατευθύνσεων γίνεται η αντιστοίχιση από $[0,0, 1,0]$ σε $[1, 7)$. Τα δύο διαφορετικά σετ υπάρχουν γιατί χρησιμοποιούνται σε διαφορετικές περιπτώσεις. Στην περίπτωση όπου η συνάρτηση κληθεί από τους αισθητήρες υπέρυθρων ακτίνων γιατί δεν υπάρχει πάτωμα μπροστά για να συνεχιστεί η κίνηση, η κατασκευή κινείται προς τα πίσω για λίγο και στην συνέχεια καλείται η συνάρτηση με το σετ των έξι τιμών καθώς οι κατευθύνσεις όπου μπορεί να συνεχίσει η κατασκευή είναι η μπροστά-δεξιά, δεξιά, πίσω-δεξιά, πίσω-αριστερά, αριστερά και μπροστά-αριστερά. Από την στιγμή όπου η σκούπα πήγε πίσω για να αποφύγει το κενό, δεν υπάρχει λόγος να ξανά πάει πίσω κι έτσι αυτή η κατεύθυνση δεν χρειάζεται να υπάρχει σε αυτό το σετ. Για την περίπτωση όπου η συνάρτηση κληθεί από τους αισθητήρες υπέρυθρων, το σετ όπου θα χρειαστεί, μπορεί να περιέχει και την πισινή κατεύθυνση, η οποία όμως αν επιλεγθεί θα τρέξει και πάλι το σετ των έξι. Στον παρακάτω Πίνακα 7.5 απεικονίζονται οι κατευθύνσεις των δύο σετ και η αντιστοιχία τους με τις τιμές των εξόδων της συνάρτησης μετά την μετατροπή τους στο σωστό εύρος.

Πίνακας 7.5. Αντιστοίχιση σετ κατευθύνσεων

Σετ 7 κατευθύνσεων		Σετ 6 κατευθύνσεων	
Αριθμός	Κατεύθυνση	Αριθμός	Κατεύθυνση
1	Μπροστά-δεξιά	1	Μπροστά-δεξιά
2	Αριστερά	2	Δεξιά
3	Δεξιά	3	Αριστερά
4	Μπροστά-αριστερά	4	Πίσω-δεξιά
5	Πίσω-αριστερά	5	Μπροστά-αριστερά
6	Πίσω-δεξιά	6	Πίσω-αριστερά
7	Πίσω		

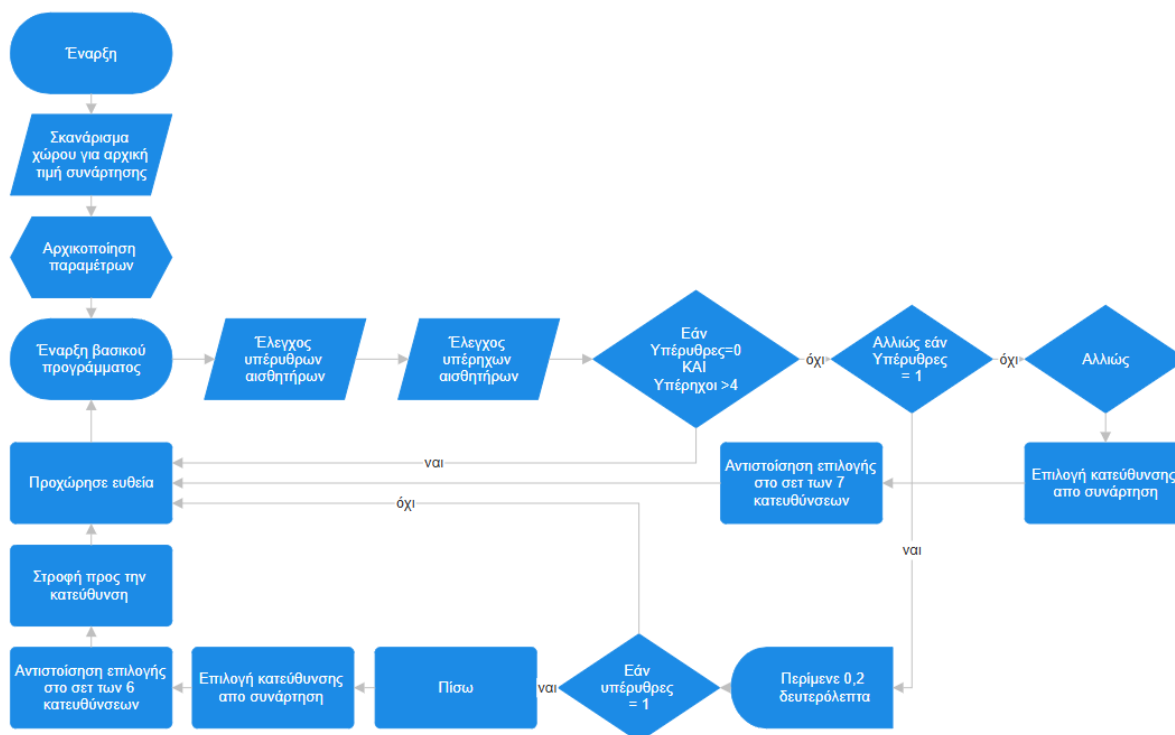
Η συγκεκριμένη αντιστοίχιση έγινε με βάση τις πιθανότητες εμφάνισης του κάθε αριθμού, καθώς είναι προτιμότερο να κληθούν οι αριθμοί με τις περισσότερες πιθανότητες και να προκαλέσουν την κίνηση στη δεξιά ή την αριστερή μεριά, μετά στην μπροστά-δεξιά και μπροστά-αριστερά κατεύθυνση και τέλος στις τρεις πίσω κατευθύνσεις. Τα αποτελέσματα με τις πιθανότητες των εμφανίσεων για 1000000 επαναλήψεις παρουσιάζονται στο Σχήμα 7.11.

map 7 τιμές	1	2	3	4	5	6	7	Σύνολο	max-min
Modified	1414573	1349540	1574733	1221120	1217687	1197276	1025071	9000000	549662
	15.72%	14.99%	17.50%	13.57%	13.53%	13.30%	11.39%	100.00%	6.11%
map 6 τιμές	1	2	3	4	5	6	Σύνολο	max-min	
Modified	1602890	1666660	1638862	1405872	1513752	1171964	9000000	494696	
	17.81%	18.52%	18.21%	15.62%	16.82%	13.02%	100.00%	5.50%	

Σχήμα 7.11. Ποσοστά εμφάνισης αποτελεσμάτων μετά από αντιστοίχιση σε 7 και 6 τιμές

7.4 Ροή του προγράμματος

Στο παρόν υποκεφάλαιο έγινε αναφορά στα κομμάτια και τις λειτουργίες του προγράμματος, ενώ, για την καλύτερη κατανόησή του δημιουργήθηκε και ένα διάγραμμα ροής όπως φαίνεται και στο Σχήμα 7.12.

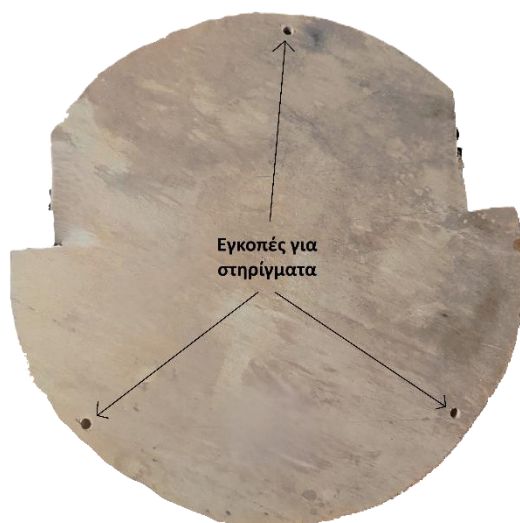


Σχήμα 7.12. Διάγραμμα ροής ελέγχου του προγράμματος

Το πρώτο πράγμα όπου κάνει το πρόγραμμα μετά την έναρξη της λειτουργίας του, είναι να σκανάρει με τον κεντρικό αισθητήρα υπερήχων, την απόσταση από το κοντινότερο αντικείμενο και στην συνέχεια ανάλογα την τιμή θα αρχικοποιήσει τις παραμέτρους του. Στην αρχικοποίηση ορίζει τους ακροδέκτες ως εισόδους κι εξόδους και θέτει την αρχική τιμή της συνάρτησης σύμφωνα με την απόσταση όπου υπολόγισε, ενώ στην συνέχεια ξεκινάει το συνεχής εκτελούμενο κομμάτι του προγράμματος. Αρχικά ελέγχει τους αισθητήρες υπέρυθρων ακτίνων και τους υπέρηχους και ανάλογα με τις τιμές τους και κάνει τις απαραίτητες διαδικασίες. Στην περίπτωση όπου οι αισθητήρες υπέρυθρων δώσουν μηδέν και οι αισθητήρες υπέρηχων έχουν τιμή μεγαλύτερη του τέσσερα σημαίνει ότι υπάρχει πάτωμα για να κινηθεί στην ευθεία και δεν υπάρχει αντικείμενο τουλάχιστον για τέσσερα εκατοστά μπροστά, οπότε η κατασκευή προχωράει ευθεία. Στην περίπτωση όπου ένας από τους αισθητήρες υπέρυθρων ακτίνων δώσουν τιμή άσσο, σημαίνει ότι δεν βρίσκουν πάτωμα μπροστά, όπως φαίνεται και από το διάγραμμα ροής, το πρόγραμμα περιμένει για 0,2 δευτερόλεπτα και ξαναελέγχει την τιμή όπου δίνουν οι αισθητήρες καθώς καμιά φορά μπορεί να δώσουν άσσο για λάθος πράγματα. Όταν λοιπόν ο αισθητήρας δείχνει άσσο και μετά τα 0,2 δεύτερα, η κατασκευή κινείται με όπισθεν πίσω, στην συνέχεια εκτελεί την συνάρτηση, αντιστοιχεί την τιμή με μια κατεύθυνση από τις έξι διαθέσιμες, στρίβει προς την κατεύθυνση και τέλος συνεχίζει ευθεία ενώ το πρόγραμμα ξαναξεκινάει από την αρχή. Αν τυχόν ο άσσος ήταν μόνο στον πρώτο έλεγχο και εξαφανιστεί στον δεύτερο, η σκούπα συνεχίζει κανονικά την πορεία της ευθεία. Στην τελευταία περίπτωση, οι αισθητήρες υπέρυθρων δίνουν μηδέν οπότε υπάρχει πάτωμα στην ευθεία, αυτό σημαίνει ότι ο έλεγχος ενεργοποιήθηκε από τους αισθητήρες υπέρηχων και πρέπει να υπάρχει αντικείμενο που να εμποδίζει μπροστά από την κατασκευή. Για να το αποφύγει η σκούπα σταματάει, τρέχει την συνάρτηση, αντιστοιχεί την τιμή όπου έλαβε σε μία από τις επτά που μπορεί να κινηθεί, στην συνέχεια στρίβει προς την κατεύθυνση που επιλέχθηκε και προχωράει ευθεία ενώ το πρόγραμμα γυρίζει στην αρχική του θέση.

7.5 Υλοποίηση κατασκευής

Το τελευταίο βήμα μετά την μελέτη και σχεδίαση του κυκλώματος σε πρόγραμμα σχεδίασης και τη συγγραφή του προγράμματος, είναι η δημιουργία του. Για την βάση της χρησιμοποιήθηκε ένα σκληρό κι ελαφρύ φύλλο ξύλου με στρογγυλό σχήμα διαμέτρου 30 εκατοστών και πάχους 3 χιλιοστών όπως φαίνεται στο Σχήμα 7.14, όπου του έγιναν οι κατάλληλες διεργασίες για την στερέωση των εξαρτημάτων. Σε τρία σημεία τρυπήθηκε έτσι ώστε να τοποθετηθεί μπαμπού ύψους 8,5 εκατοστών για την στερέωση του καπακιού. Το επάνω μέρος φτιάχτηκε από διαφανές πλαστικό στρόγγυλου σχήματος με διάμετρο 30 εκατοστών όπου τρυπήθηκε, όπως απεικονίζεται στο Σχήμα 7.13, ώστε να μπορεί να στερεωθεί του με το υπόλοιπο σώμα της κατασκευής.



Σχήμα 7.14. Βάσης κατασκευής

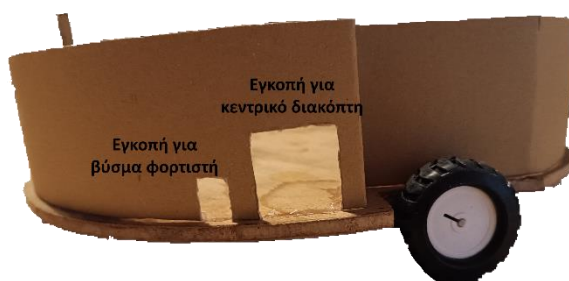


Σχήμα 7.13. Καπάκι κατασκευής

Το πλαϊνό μέρος της παρουσιάζεται στα Σχήματα 7.16 και 7.15, αποτελείται από χοντρό χαρτόνι ύψους 6 εκατοστών και διαθέτει ανοίγματα για τον έλεγχο του κεντρικού διακόπτη της συσκευής, για την σύνδεση του φορτιστή της μπαταρίας, τον έλεγχο της τάσης της μπαταρίας μέσω του βολτόμετρου.

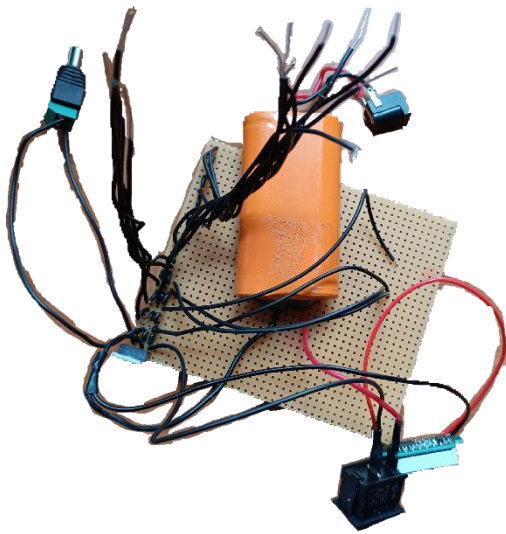


Σχήμα 7.16. Επάνω όψη κατασκευής

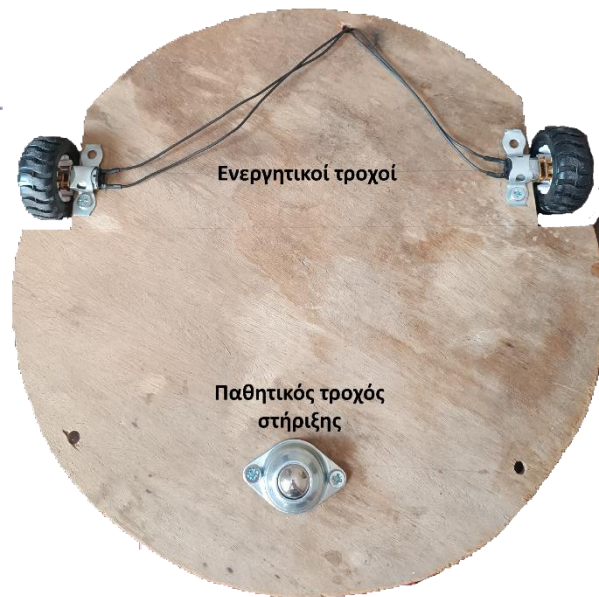


Σχήμα 7.15. Δεξιό πλαϊνό μέρος

Το επόμενο στάδιο της κατασκευής περιλαμβάνει την κόλληση καλωδίων όπως φαίνεται στο Σχήμα 7.17, για την παροχή τροφοδοσίας όλων των ηλεκτρονικών εξαρτημάτων που χρειάζονται στην διάτρητη πλακέτα και η τοποθέτηση των κινητήρων στην βάση όπως παρουσιάζονται στο Σχήμα 7.18.

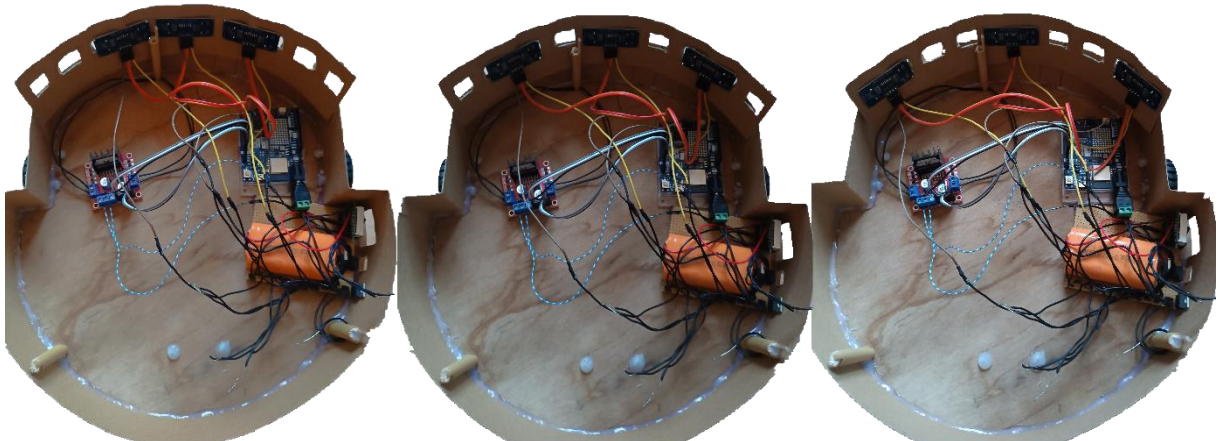


Σχήμα 7.17. Ηλεκτρονική πλακέτα



Σχήμα 7.18. Κάτω όψη της βάσης

Στην συνέχεια έγιναν οι δοκιμές όπου προαναφέρθηκαν, για την τοποθέτηση των αισθητήρων υπερήχων. Ο μεσαίος αισθητήρας ήταν δεδομένο ότι θα τοποθετηθεί στο κέντρο της κατασκευής και έπρεπε να γίνουν οι απαραίτητες δοκιμές ώστε να βρεθεί η κατάλληλη θέση και για τους άλλους δύο, έτσι ώστε να μην υπάρχουν ούτε μεγάλες επικαλύψεις αλλά ούτε κενά ενδιάμεσα από τους αισθητήρες. Στο παρακάτω σχήμα 7.19 φαίνονται οι τρεις δοκιμές για την τελική επιλογή των θέσεων



Σχήμα 7.19. Δοκιμές επιλογής θέσης των υπερήχων αισθητήρων

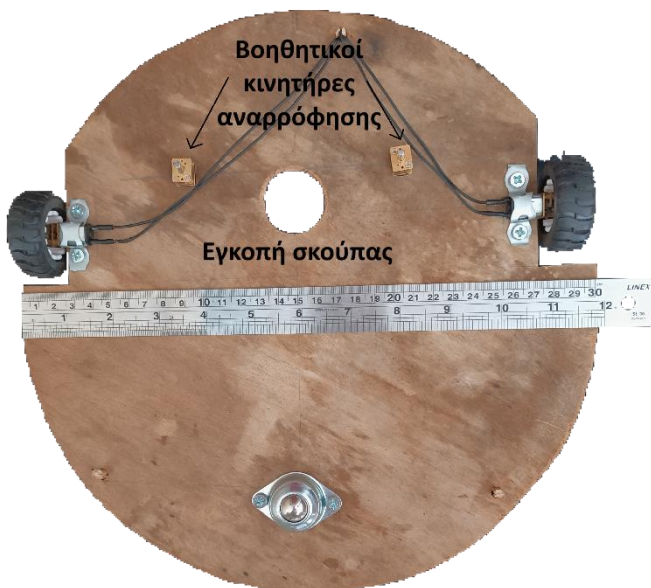
Από τις επιλογές αυτές, η πρώτη είχε το πλεονέκτημα ότι δεν αφήνει καθόλου κενά ενδιάμεσα από τους αισθητήρες αλλά οι άκρες της κατασκευής βρισκόταν έξω από την εμβέλειά τους με αποτέλεσμα να δημιουργούνται τυφλά σημεία και στις πλαϊνές κινήσεις η σκούπα να τρακάρει σε αντικείμενο που δεν έβλεπε. Η τρίτη περίπτωση μετέφερε τους αισθητήρες στην μεγαλύτερη δυνατή απόσταση μεταξύ τους με αποτέλεσμα να εξαλείφει τα τυφλά σημεία στο πλάι. Λόγο της μεγάλης απόστασης υπήρχε η

περίπτωση, ένα αντικείμενο να περάσει ανάμεσα από την εμβέλεια των αισθητήρων χωρίς να το προσέξουν κι έτσι η σκούπα μπορεί να τρακάρει και να κολλήσει σε ένα αντικείμενο χωρίς να το γνωρίζει. Η δεύτερη περίπτωση είναι η πιο ουδέτερη καθώς οι αισθητήρες βρίσκονται σε τέτοια απόσταση έτσι ώστε εκεί που τελειώνει η εμβέλεια του ενός, ξεκινούσε η εμβέλεια του άλλου. Αυτό έχει ως αποτέλεσμα να μην υπάρχουν τυφλά σημεία μεταξύ τους και ταυτόχρονα η απόσταση είναι η μεγαλύτερη δυνατή ώστε να καλυφθεί όσο το δυνατόν περισσότερος χώρος. Από τα πειράματα που διεξήχθη όμως, ακόμα και η δεύτερη είχε κάποια προβλήματα όταν το αντικείμενο ήταν πολύ λεπτό ή είχε αιχμηρή γωνία η οποία αντανακλούσε το σήμα με διαφορετική φορά.

Αφού έγινε η επιλογή της πιο ουδέτερης, δεύτερης θέσης, το επόμενο στάδιο ήταν να κοπεί το χαρτόνι έτσι ώστε να μπορέσουν να σταθεροποιηθούν όπως φαίνεται και στο Σχήμα 7.21. Από αυτή τη θέση μετρήθηκε ότι τα αντικείμενα όπου ελέγχονται από τους αισθητήρες θα πρέπει να είναι ψηλότερα από 3 εκατοστά και να ξεκινούν τουλάχιστον από τα επτά εκατοστά καθώς στην περίπτωση όπου το αντικείμενο είναι χαμηλότερο από τα 3, η εμβέλεια των αισθητήρων θα περνάει από πάνω του και στην περίπτωση όπου το αντικείμενο αρχίζει μετά τα πρώτα 7 εκατοστά, η εμβέλεια των αισθητήρων θα περνάει από κάτω του. Να σημειωθεί ότι στην περίπτωση όπου το αντικείμενο ξεκινάει μετά τα 10,5 εκατοστά, ολόκληρη η σκούπα θα περάσει από κάτω κι έτσι δεν θα υπάρχει πρόβλημα. Στην ίδια περίοδο έγιναν και οι τρύπες για τους βοηθητικούς κινητήρες στο σύστημα αναρρόφησης και το μέρος απ' όπου θα περάσει το στόμιο της σκούπας για να έρθει σε κοντινή απόσταση με το πάτωμα, όπως απεικονίζεται και στο Σχήμα 7.20.

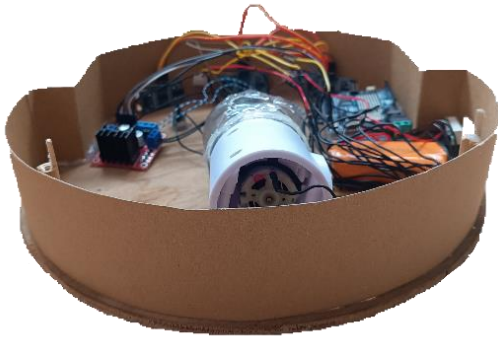


Σχήμα 7.21. Μπροστινό μέρος σκούπας με αισθητήρες υπέρηχων

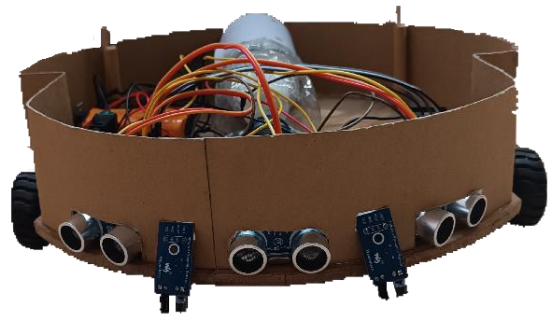


Σχήμα 7.20. Κάτω όψη σκούπας μετά από τρύπες για αναρρόφηση

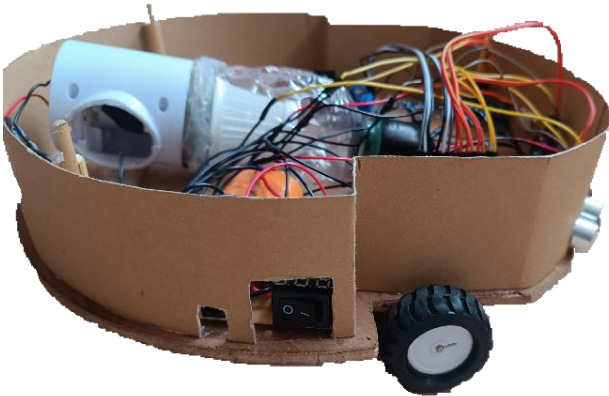
Το τελικό βήμα για την ολοκλήρωση της κατασκευής έγινε με την κόλληση όλων των εξαρτημάτων και την τοποθέτησή τους στην βάση της όπως απεικονίζεται και παρακάτω.



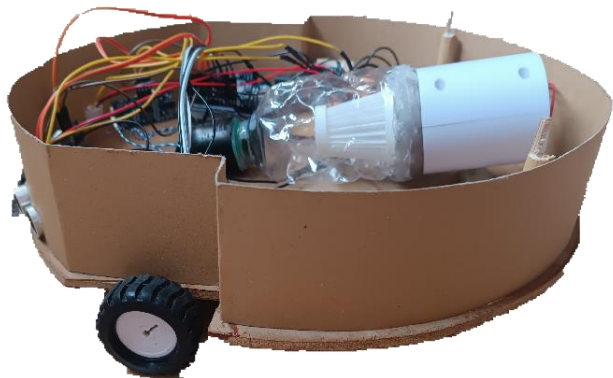
Σχήμα 7.24. Πισινό μέρος τελικής κατασκευής



Σχήμα 7.23. Μπροστινό μέρος τελικής κατασκευής



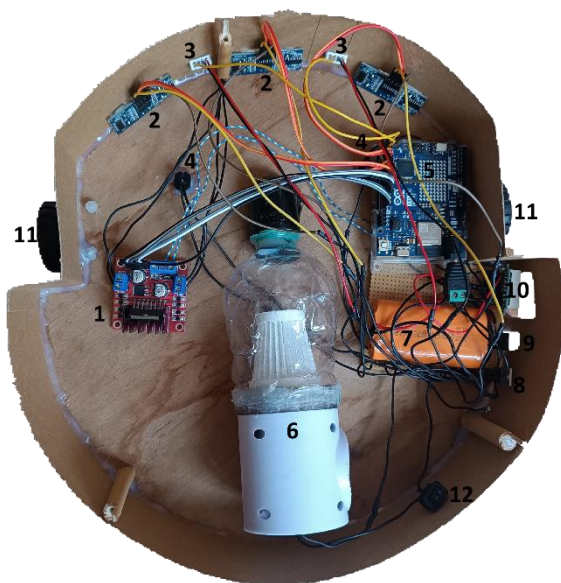
Σχήμα 7.22. Δεξιο πλαϊνό μέρος κατασκευής



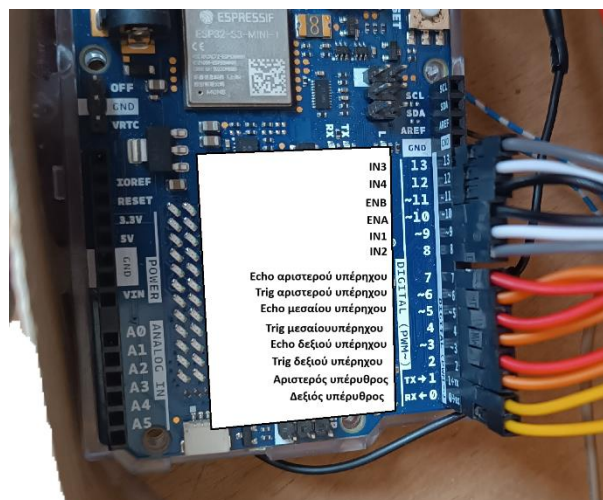
Σχήμα 7.25. Αριστερό πλαϊνό μέρος κατασκευής

Στο Σχήμα 7.26 εμφανίζονται οι συνδέσεις του επεξεργαστή με τα υλικά όπου χρειάζονται οδήγηση από αυτόν, ενώ στο Σχήμα 7.27 μπορούν να παρατηρηθούν καλύτερα όλα τα υλικά που χρησιμοποιήθηκαν για την συγκεκριμένη υλοποίηση:

1. Κύκλωμα οδήγησης ενεργητικών κινητήρων με L298N
2. Αισθητήρες απόστασης υπερήχων
3. Αισθητήρας ανίχνευσης αντικειμένου με υπέρυθρες ακτίνες
4. Βοηθητικοί κινητήρες στο σύστημα αναρρόφησης
5. Επεξεργαστής RA4M1
6. Σκουπάκι αναρρόφησης
7. Μπαταρία ιόντων λιθίου 7,4V
8. Ρυθμιστής τάσης 5V LM7805
9. Βύσμα φόρτισης μπαταρίας
10. Κεντρικός διακόπτης και βολτόμετρο
11. Ενεργητικοί κινητήρες
12. Διακόπτης συστήματος αναρρόφησης



Σχήμα 7.27. Επάνω όψη τελικής κατασκευής



Σχήμα 7.26. Μεγέθυνση στον επεξεργαστή για την απεικόνιση των συνδέσεων

7.6 Επίλογος

Στο παραπάνω Κεφάλαιο συνδυάστηκαν όλες οι πληροφορίες όπου αναφέρθηκαν κατά την διάρκεια της εργασίας, για την παραγωγή της κατασκευής όπου απεικονίστηκε. Έγινε αναφορά στην μελέτη και την ροή της σκέψης, από την αρχική ιδέα, πως πέρασε στο στάδιο της σχεδίασης, η συγγραφή του προγράμματος και τέλος η υλοποίηση με τα εξαρτήματα που αναλύθηκαν στα προηγούμενα Κεφάλαια.

Κεφάλαιο 8ο: Πειράματα

8.1 Εισαγωγή

Στο παρόν κεφάλαιο έγινε αναφορά στον τρόπο διεξαγωγής των πειραμάτων και στα αποτελέσματα των δοκιμών της λειτουργίας της κατασκευής. Αρχικά να αναφερθεί ότι για τα πειράματα χρησιμοποιήθηκε ο παρακάτω ελεγχόμενος χώρος με διαστάσεις 180×180 εκατοστά και 60×150 εκατοστά, συνολικά στα 4,14m².



Σχήμα 8.1. Ελεγχόμενος εργασιακός χώρος

8.2 Πρώτο σετ δοκιμών

Διεξήχθησαν δοκιμές σε χώρο χωρίς αντικείμενα, μετρώντας την κάλυψη όπου πραγματοποίησε η κατασκευή, τον χρόνο ολοκλήρωσης της κάλυψης όπου μετά το πέρασμα των 17 λεπτών το πείραμα σταματούσε και οι επισκέψεις της σε κάθε τετράγωνο του χώρου. Πραγματοποιήθηκαν 10 πειράματα από διαφορετικές αρχικές θέσεις σε αυτόν το χώρο, όπου και καταγράφηκαν τα ζητούμενα χαρακτηριστικά. Στα παρακάτω σχήματα απεικονίζονται οι φορές όπου πέρασε από το κάθε τετράγωνο, έγινε η αρίθμηση και μια κατάταξη με χρώματα ώστε να είναι ευκολότερη η απικόνισή τους

Πείραμα 1ο κενός χώρος

5	5	4	4	2	← 3		
9	7	3	4	3	4	2	1
5	9	2	4	2	4	2	1
2	7	7	5	1	2	1	2
2	5	8	2	4	3	1	1
3	3	1	1	3	2	2	1

Σχήμα 8.2. Πείραμα 1^ο σε κενό χώρο

Πείραμα 2ο κενός χώρος

2	3	2	4	4	↓ 3		
5	6	3	2	6	3	5	2
4	4	3	3	2	7	3	5
4	5	2	2	4	8	3	2
3	3	3		3	5	6	5
1	2	3	1	3	2	4	2

Σχήμα 8.3. Πείραμα 2^ο σε κενό χώρο

Στο πρώτο πείραμα παρατηρείται ότι η κάλυψη του χώρου ήταν επιτυχής στο 100% με αρκετές επαναλήψεις σε μερικά τετράγωνα του χώρου. Η κατασκευή ξεκίνησε από το τετράγωνο με το βελάκι, με την φορά όπου δείχνει και ο χρόνος που μετρήθηκε για την ολοκλήρωση της κάλυψης του χώρου ήταν τα 13:20 λεπτά. Στο δεύτερο πείραμα, η κατασκευή ξεκίνησε πάλι από την ίδια αρχική θέση, με άλλη κατεύθυνση. Σημαντική παρατήρηση είναι ότι δεν ολοκληρώθηκε η κάλυψη του χώρου μετά το πέρασμα των 17 λεπτών αλλά παρέμεινε στο 97,8%.

Πείραμα 3ο κενός χώρος							
	3	6	2	2			
7	6	2	2	3	3	4	← 3
5	4	3	1	3	6	6	3
3	10	2	1	3	2	4	3
3	5	7	2	1	2	2	4
3	3	3	2	2	2	2	2

Σχήμα 8.5. Πείραμα 3^ο σε κενό χώρο

Πείραμα 4ο κενός χώρος							
2	2	4	1	3	4		
4	5	4	5	4	7	2	↓ 2
5	3	3	2	1	2	1	2
3	3	3	1	1	3	2	4
3	5	4	2	1	3	5	4
1	3	2	1	2	5	4	3

Σχήμα 8.4. Πείραμα 4^ο σε κενό χώρο

Το τρίτο πείραμα ξεκίνησε από την αρχική θέση με το βελάκι και είχε την φορά όπου υποδεικνύει. Επίσης δεν πραγματοποίησε ολόκληρη την κάλυψη του χώρου μετά το πέρασμα των 17 λεπτών, αλλά έφτασε στο 95,6% της επιφάνειας που έπρεπε να διανύσει. Στο τέταρτο πείραμα, με την φορά όπου παρατηρείται, η κατασκευή κατάφερε και διένυσε ολόκληρο τον εργασιακό χώρο σε 13:55 λεπτά. Είναι καλό επίσης το γεγονός ότι δεν υπήρξε τόση επικάλυψη όση στο πρώτο πείραμα όπως φαίνεται καθώς σε αυτό η περισσότερες φορές όπου πέρασε από ένα τετράγωνο ήταν οι επτά σε σχέση με τα δύο τετράγωνα όπου πέρασε εννέα φορές στο πρώτο πείραμα. Το αρνητικό είναι ο χρόνος όπου ήταν περισσότερος κι αυτό οδηγεί στο συμπέρασμα ότι η κατασκευή δεν πέρασε από τα ίδια τετράγωνα γιατί είχε «κολλήσει» στις γωνίες κι έτσι περνούσε οι χρόνος χωρίς να παράγει δουλειά.

Πείραμα 5ο κενός χώρος							
4	4	5	3	5	5		
5	6	5	5	3	5	2	
6	7	3	1	3	4	3	1
3	5	3	1		3	2	3
2	4	4		1	2	2	1
3	3	4	2	1	3	3	← 2

Σχήμα 8.6. Πείραμα 5^ο σε κενό χώρο

Πείραμα 6ο κενός χώρος							
7	5	3	2	1	1		
9	7	3	1	2	2	4	3
9	7	1	1	3	4	5	5
7	7	3		2	2	1	5
3	9	5	3	3	1	1	5
2	4	4	3	2	3	6	↑ 3

Σχήμα 8.7. Πείραμα 6^ο σε κενό χώρο

Από την τρίτη θέση εκκίνησης πραγματοποιήθηκαν το πέμπτο κι έκτο πείραμα, με τις αντίστοιχες κατευθύνσεις όπου υποδηλώνει το βελάκι. Σε κανένα από τα δύο δεν ολοκληρώθηκε η κάλυψη του χώρου αλλά έφτασε το 93,4% για το πέμπτο και το 97,8% για το έκτο. Παρατηρείται επίσης ότι από αυτή τη θέση η κατασκευή «κολλούσε» στην επάνω αριστερή γωνία του χώρου εργασίας της.

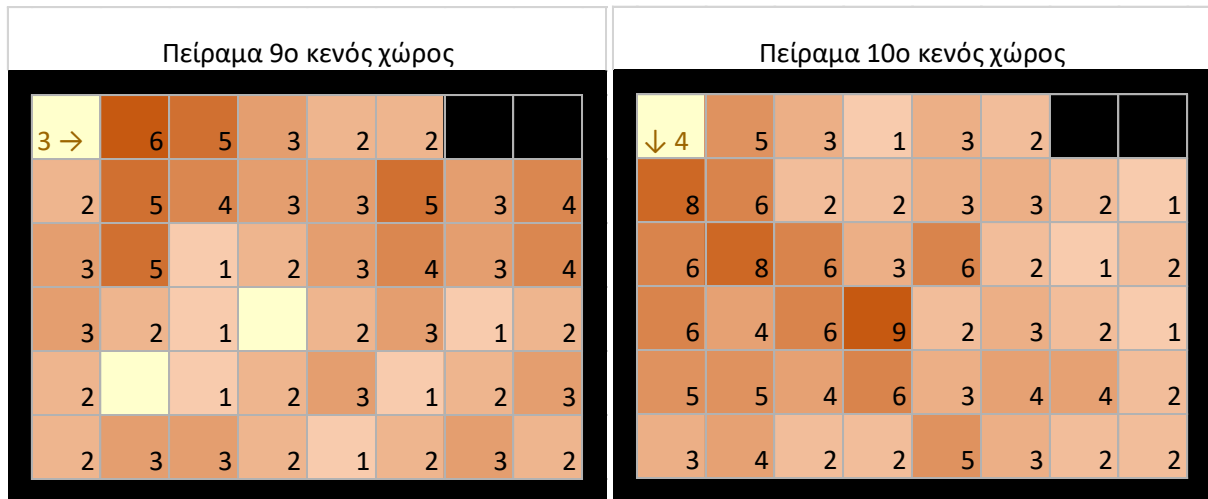
Πείραμα 7ο κενός χώρος							
5	5	2	1	1	2		
7	5	1	2	1	4	2	1
8	6	4	2	3	3	3	2
8	4	3	3	4	2	2	4
5	10	2	2	3	4	7	7
↑ 3	5	3			4	5	2

Σχήμα 8.9. Πείραμα 7^ο σε κενό χώρο

Πείραμα 8ο κενός χώρος							
2	2	1	1	2	3		
1	2	1	1	1	6	2	4
1	2		1	2	3	4	6
1	3		2	4	5	5	6
3	4	5	4	6	6	5	4
2 →	5	2	1	4	6	5	4

Σχήμα 8.8. Πείραμα 8^ο σε κενό χώρο

Για τα πειράματα επτά κι οκτώ επίσης δεν καλύφθηκε ολόκληρος ο χώρος αλλά έφτασε το 95,6% της συνολικής επιφάνειάς του. Το πείραμα επτά ισοφαρίζει το μεγαλύτερο νούμερο επαναλήψεων όπου είχε το πείραμα τρία και παρατηρείται ότι είχε την τάση να κινείται στην αριστερή πλευρά του χώρου με μερικές επισκέψεις στη κάτω δεξιά γωνία. Το πείραμα οκτώ είχε «κολλήσει» στην κάτω δεξιά γωνία όπως παρατηρείται από τις πολλές επαναλήψεις στο κάτω δεξιά τέταρτο του συνολικού χώρου.



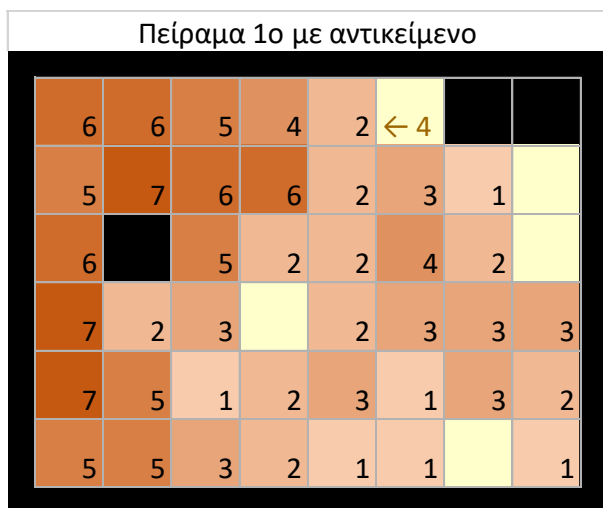
Σχήμα 8.10. Πείραμα 9^ο σε κενό χώρο

Σχήμα 8.11. Πείραμα 10^ο σε κενό χώρο

Από την τελευταία θέση εκκίνησης του χώρου διεξήχθη τα πειράματα εννέα και δέκα. Το ένατο πείραμα δεν κατάφερε να διασχίσει ολόκληρο τον χώρο ακόμα και μετά το πέρασμα των 17 λεπτών. Η κάλυψη όπου παρείχε έφτασε το 95,6% του χώρου και φαίνεται ότι κινήθηκε στο επάνω μισό του συνολικού χώρου. Το δέκατο πείραμα ήταν το τρίτο όπου κατάφερε να ολοκληρώσει την κάλυψη του συνολικού χώρου σε χρόνο 15:32 λεπτά. Παρατηρείται από την άνιση κατανομή των επαναλήψεων ότι άργησε να διασχίσει το δεξί μισό του χώρου και ειδικά την επάνω δεξιά γωνία, πράγμα όπου δικαιολογεί την αργοπορία του.

8.3 Δεύτερο σετ δοκιμών

Μετά την λήξει των δέκα πειραμάτων, πραγματοποιήθηκαν στον ίδιο χώρο δέκα ακόμα δοκιμές, αυτή την φορά με ένα αντικείμενο μέσα στον χώρο. Η θέση του αντικειμένου επιλέχτηκε να είναι αυτή όπου στις δοκιμές χωρίς αντικείμενο, είχε τις περισσότερες επαναλήψεις, καθώς κρίθηκε ότι αυτή θα επιφέρει τις περισσότερες αλλαγές στην τελική μορφοποίηση της διαδρομής. Παρακάτω παρουσιάζονται τα αποτελέσματα του δεύτερου σετ δοκιμών με την ίδια μορφοποίηση όπως και τα προηγούμενα.

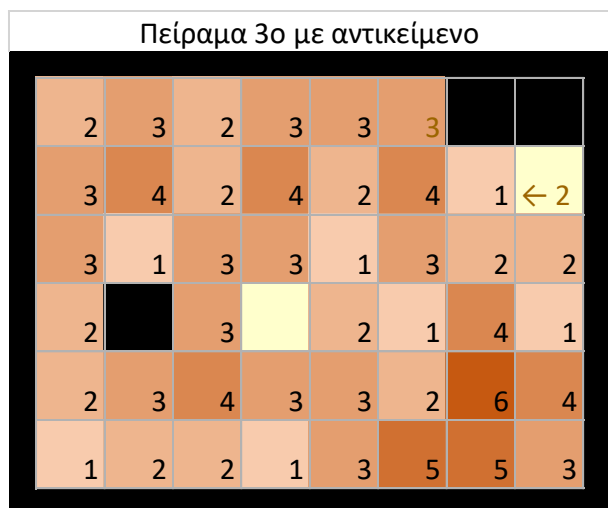


Σχήμα 8.13. Πείραμα 1^ο με αντικείμενο στο χώρο

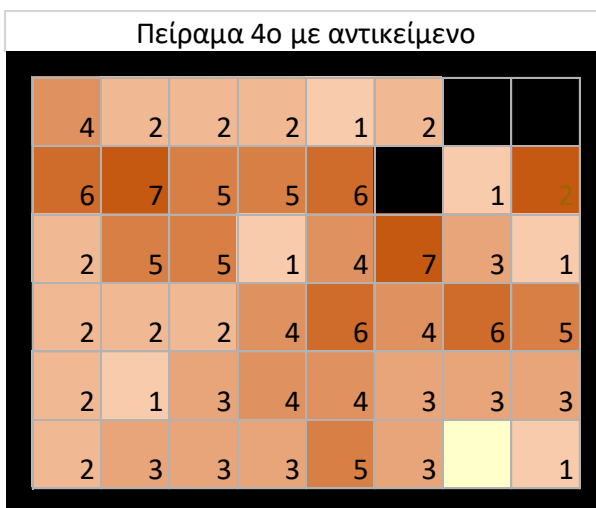


Σχήμα 8.12. Πείραμα 2^ο με αντικείμενο στο χώρο

Όπως είναι φανερό από τα αποτελέσματα των δοκιμών, οι επαναλήψεις σε κάθε τετράγωνο άλλαξαν με την πρώτη περίπτωση να μην επιτυγχάνει πλέον την ολοκλήρωση της κάλυψης του χώρου μετά το πέρασμα των 17 λεπτών. Από τα αποτελέσματα προκύπτει ότι η κάλυψη έφτασε το 91,1% με τις επικαλύψεις να μαζεύονται στην αριστερή πλευρά του χώρου. Το δεύτερο πείραμα ούτε σε αυτή τη προσπάθεια κατάφερε να ολοκληρώσει την εργασία του αλλά έφτασε μέχρι το 95,5% με την διαδρομή να τείνει προς τα αριστερά αυτή τη φορά.



Σχήμα 8.14. Πείραμα 3^ο με αντικείμενο στο χώρο



Σχήμα 8.15. Πείραμα 4^ο με αντικείμενο στο χώρο

Για την περίπτωση της τρίτης δοκιμής, παρατηρείται μια καλύτερη ομοιομορφία στην κατανομή των επικαλύψεων καθώς επίσης και του καλύτερου ποσοστού κάλυψης όπου προέκυψε με ποσοστό 97,7% έναντι του 95,6% όπου είχε στην δοκιμή χωρίς αντικείμενο. Το πείραμα τέσσερα δυστυχώς δεν ολοκλήρωσε την κάλυψη του χώρου, σε αντίθεση με την πρώτη δοκιμή του όπου το κατάφερε στα 13:55

όπως και προαναφέρθηκε. Πέραν του σημείου όπου άφησε κενό, φαίνεται ότι πέρασε αρκετές φορές από τη διαγώνιο της πάνω αριστερής με την κάτω δεξιάς γωνιάς.

Πείραμα 5ο με αντικείμενο

3			1	2	1		
5	4	1	2	1	1	3	3
6		1	2	3	4	1	5
9	6	5	3	3	3	4	
8	5	4	3	6	2	1	2
4	5	3	1	1	1	2	← 2

Σχήμα 8.17. Πείραμα 5^ο με αντικείμενο στο χώρο

Πείραμα 6ο με αντικείμενο

1			2	2	3		
2	2	2	3	2	8	5	7
	1	5	4	5	6	5	11
1	7	3	4	5	4	5	4
3	2	1	1	5	6	3	3
1	2	2	2	6	3	1	↑ 3

Σχήμα 8.16. Πείραμα 6^ο με αντικείμενο στο χώρο

Το πείραμα πέντε όπως προκύπτει από τα κενά όπου άφησε η κατασκευή δεν επηρεάστηκε από το αντικείμενο και παρέμεινε σε ποσοστό κάλυψης στο 93,3%. Είναι φανερή επίσης η μετατόπιση των πολλών επαναλήψεων από το πάνω μισό του χώρου, στην κάτω αριστερή με πάνω δεξιά γωνία. Το έκτο πείραμα περιέχει τις περισσότερες πλέον επικαλύψεις όπου φτάνει τις 11 στο πάνω δεξιό μέρος του χώρου. Φαίνεται ότι κράτησε μια πορεία στην μέση και λίγο προς την πάνω δεξιά γωνία του χώρου, με τις δύο αριστερές γωνίες να μην έχουν σχεδόν καθόλου επισκέψεις. Αν και με ποσοστό 95,5%, από τα αποτελέσματα είναι φανερή η ανομοιομορφία της διαδρομής που εκτελέστηκε.

Πείραμα 7ο με αντικείμενο

3	2	2			2		
5	2	3	2	2	4	7	5
5	4	2	3	6	7	7	8
6	4	2	4	3	3	5	5
6		1	1	1	2	3	3
↑ 5	3	2	2	2	4	3	2

Σχήμα 8.18. Πείραμα 7^ο με αντικείμενο στο χώρο

Πείραμα 8ο με αντικείμενο

4	7	2	2				
5	3	4	2	1			
3	5	1	1	2	1		
4	4	1	1	3	2	1	1
5	5	5	2	3	6	3	2
5 →	6	2	4	3	4	8	4

Σχήμα 8.19. Πείραμα 8^ο με αντικείμενο στο χώρο

Το πείραμα επτά και πάλι δεν κατάφερε να καλύψει ολόκληρο το χώρο εργασίας. Όπως παρατηρείται, κατάφερε να φτάσει στο ίδιο ποσοστό κάλυψης αλλά οι επαναλήψεις στα κουτιά ήταν αρκετά πιο ομοιόμορφες σε σύγκριση των αποτελεσμάτων με το ίδιο πείραμα χωρίς αντικείμενο. Η περίπτωση του ογδού πειράματος με αντικείμενο επέφερε τα χειρότερα αποτελέσματα από όλες τις δοκιμές όπου έγιναν με ποσοστό 86,6%. Η κατασκευή όπως φαίνεται ξεκάθαρα, δεν κινήθηκε καθόλου στην επάνω δεξιά γωνία κι έτσι διέσχισε αρκετές φορές τον υπόλοιπο χώρο εργασίας της.

Πείραμα 9ο με αντικείμενο								
1 →		2	2	2	1			
1	3	4	3	2	2	7	7	
1	3	4	4	1	2	3	7	
1	2	2	4	3	3	1	5	
2	1	1	4	5	2	1	4	
3	3	4	3	2	3	4	4	

Σχήμα 8.20. Πείραμα 9^ο με αντικείμενο στο χώρο

Πείραμα 10ο με αντικείμενο								
↓ 5	6	4	4	1				
6	7	6	5	3	2	1	2	
5	4	4	3	2	2	5	4	
3	8	1		4	2	2	3	
3	7	5	5	2	2	3	2	
3	4	1	3	1	1	3	3	

Σχήμα 8.21. Πείραμα 10^ο με αντικείμενο στο χώρο

Η περίπτωση του πειράματος εννέα ήταν η πιο ευχάριστη καθώς ήταν η πρώτη με την οποία κατάφερε κατασκευή να ολοκληρώσει το 100% του χώρου εργασίας της. Αν και πολύ κοντά στην λήξη του χρόνου, δηλαδή στα 16:56 λεπτά, κατάφερε τη διεκπεραίωση της λειτουργίας της. Παρατηρείται ότι το άθροισμα των κινήσεων είναι μικρότερο σε σχέση με τα υπόλοιπα πειράματα κι ας διήρκεσε παρόμοιο χρόνο λειτουργίας. Το τελευταίο πείραμα όπου πραγματοποιήθηκε είχε καλό ποσοστό κάλυψης όπου έφτασε το 97,7% του συνολικού χώρου, ενώ συγκριτικά με το πείραμα χωρίς αντικείμενο είχε παρόμοια πορεία η οποία είχε μια κλίση προς το αριστερό κομμάτι του χώρου.

8.4 Επίλογος

Στο παραπάνω Κεφάλαιο παρουσιάστηκαν τα αποτελέσματα των δοκιμών της κατασκευής. Μέσω των σχημάτων έγινε κατανοητό ότι ο τροποποιημένος χάρτης σκηνης έδωσε καλή καλυπτικότητα του χώρου όπου σε κάποιες περιπτώσεις έφτασε το 100% του συνόλου. Από τα πειράματα φαίνεται επίσης η επίδραση όπου είχε η παρουσία ενός αντικειμένου στον χώρο η οποία άλλαξε την πορεία και της επαναλήψεις στο κάθε πείραμα

Κεφάλαιο 9ο: Συμπεράσματα και προτάσεις βελτίωσης

Με την λήξη αυτής της εργασίας βγαίνουν και μερικά συμπεράσματα. Αρχικά ότι η δημιουργία μιας αυτόματης σκούπας με χαοτικό τρόπο πλοήγησης είναι δυνατή. Από τον χώρο δοκιμών φαίνεται ότι σαν κατασκευή ήταν λειτουργική, καθώς περνούσε σχεδόν απ' όλα τα τετράγωνα σε ικανοποιητικό χρόνο. Φυσικά αν χρειαστεί να καθαριστεί ένα δωμάτιο γρήγορα, ο πιο καλός τρόπος είναι με την κλασική σκούπα και ανθρώπινο δυναμικό. Για την περίπτωση όμως όπου χρειάζεται αυτόματο καθάρισμα σε ώρες εξωτερικής κινητικότητας των ανθρώπων, η συγκεκριμένη σκούπα σίγουρα μπορεί να κάνει την δουλειά της ικανοποιητικά. Όπως αναφέρθηκε, η χαοτική συνάρτηση βοηθάει στην αυτονομία της κατασκευής από δεδομένα δωματίων, καθώς δεν χρειάζεται περαιτέρω πληροφορίες εκτός από την συνάρτηση όπου εκτελεί για την επιλογή κατεύθυνσης και τα δεδομένα των αισθητήρων της. Αυτό έχει ως αποτέλεσμα να είναι το ίδιο αξιόπιστη σε κάθε τύπο δωματίων χωρίς καμία προετοιμασία. Σίγουρα όμως θα μπορούσαν να διορθωθούν κάποια πράγματα, όπως είναι το σύστημα των αισθητήρων υπερήχων, καθώς σε κάποια σημεία ενδιάμεσα από τους αισθητήρες, ανάλογα με το αντικείμενο όπου έχει μπροστά αφήνει τυφλά σημεία. Αυτό έχει να κάνει με την υλοποίηση των αισθητηρίων και τον τρόπο ανάκλασης του ήχου σε κάθε επιφάνεια. Μια λύση θα ήταν η χρήση διαφορετικού αισθητήρα τύπου ανίχνευση φωτός και εμβέλειας (Lidar). Ένα άλλο κομμάτι βελτίωσης θα μπορούσε να είναι στην πλοήγηση καθώς παρόλο που η κάλυψη του χώρου των πειραμάτων ήταν ικανοποιητική, δεν πάει η κατασκευή να διασχίζει τον χώρο αρκετές φορές χωρίς να χρειάζεται. Σε αυτό το πρόβλημα λύση θα μπορούσε να δοθεί χρησιμοποιώντας κάποιο σύστημα μνήμης έτσι ώστε να αποφευχθούν οι πολλές επικαλύψεις. Τέλος, πρόβλημα με την συγκεκριμένη κατασκευή θα μπορούσε να θεωρηθεί και η τροφοδοσία της καθώς μια μπαταρία ποτέ δεν είναι αρκετή. Λύση θα μπορούσε να θεωρηθεί η χρήση μια μεγαλύτερης μπαταρίας αλλά στην προκειμένη περίπτωση μπορεί να ανέβει αρκετά το κόστος.

BIBΛIOΓΡΑΦΙΑ

- [1] J. Forlizzi, "How robotic products become social products: An ethnographic study of cleaning in the home," 2007 2nd ACM/IEEE International Conference on Human-Robot Interaction (HRI), Arlington, VA, USA, 2007, pp. 129-136
- [2] Smarr CA, Prakash A, Beer JM, Mitzner TL, Kemp CC, Rogers WA. OLDER ADULTS' PREFERENCES FOR AND ACCEPTANCE OF ROBOT ASSISTANCE FOR EVERYDAY LIVING TASKS. Proc Hum Factors Ergon Soc Annu Meet. 2012 Sep; 56(1):153-157
- [11] J. L. Jones, "Robots at the tipping point: the road to iRobot Roomba," *IEEE Robotics & Automation Magazine*, vol. 13, no. 1, pp. 76-78, March 2006
- [26] Oh, S., Kim, H., Jeon, S. et al. Design of a mobile robot with 2-DOF compliant wheel-legs by using a linear spring. *Microsyst Technol* 29, 1189–1196 (2023)
- [27] Sell, Raivo & Aryassov, Gennadi & Petritshenko, Andres & Kaeeli. (2012). KINEMATICS AND DYNAMICS OF CONFIGURABLE WHEEL-LEG.
- [28] Mohanraj, A P et al. "Front and Back Movement Analysis of a Triangle-Structured Three-Wheeled Omnidirectional Mobile Robot by Varying the Angles between Two Selected Wheels." *TheScientificWorldJournal* vol. 2016 (2016): 7612945
- [29] J. Meng, A. Liu, Y. Yang, Z. Wu and Q. Xu, "Two-Wheeled Robot Platform Based on PID Control," 2018 5th International Conference on Information Science and Control Engineering (ICISCE), Zhengzhou, China, 2018, pp. 1011-1014,
- [31] S. Geetapriya, N. R. Pillai, C. K. Aswin and M. Menon, "Graph-Based Algorithm For Mobile Robot Navigation In A Known Environment," *2019 3rd International Conference on Trends in Electronics and Informatics (ICOEI)*, Tirunelveli, India, 2019, pp. 278-281
- [32] Y. Li and C. Shi, "Localization and Navigation for Indoor Mobile Robot Based on ROS," *2018 Chinese Automation Congress (CAC)*, Xi'an, China, 2018, pp. 1135-1139
- [33] L. Sui and L. Lin, "Design of Household Cleaning Robot Based on Low-cost 2D LIDAR SLAM," *2020 International Symposium on Autonomous Systems (ISAS)*, Guangzhou, China, 2020, pp. 223-227
- [34] C. Li, F. Wang, L. Zhao, Y. Li, Y. Song, "An improved chaotic motion path planner for autonomous mobile robots based on a logistic map," *International Journal of Advanced Robotic Systems*, vol. 10, no. 6, 2013, p.273
- [35] Ahuraka, Farida & McNamee, Patrick & Wang, Qixu & Nili Ahmadabadi, Zahra & Hudack, Jeffrey. (2023). Chaotic Motion Planning for Mobile Robots: Progress, Challenges and Opportunities. *IEEE Access*. PP. 1-1. 10.1109/ACCESS.2023.3337371.
- [36] Moysis, L.; Tutueva, A.; Christos, K.; Butusov, D. A chaos based pseudo-random bit generator using multiple digits comparison. *Chaos Theory Appl.* 2020, 2, 58–68
- [37] Lawnik, M., Moysis, L., Baptista, M.S. et al. Discrete one-dimensional piecewise chaotic systems without fixed points. *Nonlinear Dyn* **112**, 6679–6693 (2024).

- [38] Moysis L, Lawnik M, Volos C, Baptista MS, Goudos SK. Chaotic Maps with Tunable Mean Value—Application to a UAV Surveillance Mission. *Symmetry*. 2023; 15(12):2138.
- [39] Biswas, H.R., Hasan, M., & Bala, S.K. (2020). CHAOS THEORY AND ITS APPLICATIONS IN OUR REAL LIFE.
- [40] Kumar, Arunachalam & BM, Hegde. (2012). Chaos theory: impact on and applications in medicine. *Nitte University Journal of Health Science*. 2. 93-99. 10.1055/s-0040-1703623.
- [41] E. N. Lorenz, “Deterministic nonperiodic flow,” *J. Atmos. Sci.*, vol. 20, no. 2, pp. 130–141, 1963.
- [42] Rukhin, A., Soto, J., Nechvatal, J., Smid, M., Barker, E., Leigh, S.,... & Vo, S. (2001). *A statistical test suite for random and pseudorandom number generators for cryptographic applications* (Vol. 22). Gaithersburg, MD, USA: US Department of Commerce, Technology Administration, National Institute of Standards and Technology.
- [44] L. Moysis, M. Lawnik, C. Volos, M. S. Baptista, G. F. Fragulis and S. K. Goudos, "Validating a Chaos Based PRBG Under Different Chaotic Maps," *2024 Panhellenic Conference on Electronics & Telecommunications (PACET)*, Thessaloniki, Greece, 2024, pp. 1-4,

Πατέντες

- [3] D. G. Moore, “Perambulating kitchen appliances and control means therefor,” U.S. Patent, 3010129A, 28 Nov., 1961.
- [10] J. L. Jones, N. E. Mack, D. M. Nugent and P. E. Sandin “Autonomous floor-cleaning robot,” U. S. Patent, 6883201B2, 26 Apr., 2005

Ιστοσελίδα στο διαδίκτυο

- [4] [Electrolux_Trilobite.pdf \(banksvac.com\)](#)
- [5] [90 years of thinking of you – some Electrolux products that made life easier for consumers – Electrolux Group](#)
- [6] [Dyson's Vision: A Robot Vacuum That Can See | Mashable](#)
- [7] [Dyson's big, heavy, complicated robot vacuum that never was \(engadget.com\)](#)
- [8] [DCO6.doc \(live.com\)](#)
- [9] [The Dysons That Never Were: DC06, DC09, DC10 and DC13 \(manchestervacs.co.uk\)](#)
- [12] [Wayback Machine \(archive.org\)](#) (<http://www.irobot.com/images/consumer/cs/Roomba-owners-manual.pdf>)
- [13] [Wayback Machine \(archive.org\)](#)
(http://www.irobot.com/images/consumer/cs/Pro_ProElite_Manual.pdf)
- [14] [Roomba: Robotic FloorVac; General FAQs Roomba, Roomba Pro, Roomba Pro Elite \(archive.org\)](#)
(<http://www.roombavac.com/customerservice/faqr.asp>)
- [15] [XV-Series-User-Guide_EN-SCH_2013.10.2111.pdf \(neatorobotics.com\)](#)
- [16] [Review: Neato Robotics XV-11 - IEEE Spectrum](#)

- [17] [Neato XV-11 robotic vacuum review \(engadget.com\)](#)
- [18] [Neato XV-11 review | 61 facts and highlights \(versus.com\)](#)
- [21] [6V DC Metal Mini Motor GA12-N20 30RPM Speed \(cableworks.gr\)](#)
- [22] [Dual Motor Driver Module L298N \(grobotronics.com\)](#)
- [23] [Ultrasonic Sensor - Ranging Detector 2 - 400cm HC-SR04 \(grobotronics.com\)](#)
- [24] [Waveshare Υπέρυθρος Αισθητήρας Απόστασης \(grobotronics.com\)](#)
- [25] [0.28inch DC 2.5-30V Voltmeter \(Red\) \(hellasdigital.gr\)](#)

Φυλλάδια δεδομένων

- [19] Renesas Electronic, “Renesas RA4M1 Group User’s Manual: Hardware,” RA4M1 User’s Manual, Sep. 2023
- [20] Arm Cortex, “Arm Cortex-M4 Datasheet,” M4 Processor datasheet, Feb. 2022

Βιβλίο

- [30] R. Siegwart and I.R. Nourbakhsh, *Introduction to Autonomous Mobile Robots*. Cambridge, Massachusetts: The MIT Press, 2004.
- [43] Strogatz, S.H., *Nonlinear Dynamics and Chaos: With Applications to Physics, Biology, Chemistry, and Engineering* (2nd ed.). Boca Raton: CRC Press, 2015.

Εικόνες

- Σχήμα 2.1 [The Future is Now | Whirlpool Corporation](#)
- Σχήμα 2.2 [Electrolux Trilobite.pdf \(banksvac.com\)](#)
- Σχήμα 2.3 [Dyson's Vision: A Robot Vacuum That Can See | Mashable](#)
- Σχήμα 2.4 [iRobot Roomba Robot Vacuum - Gizmo Highway Technology Guide](#) , [irobot roomba pro original - Google Search](#) , [irobot roomba original - Google Search](#)
- Σχήμα 2.5 [Neato XV-11 review | 61 facts and highlights \(versus.com\)](#)
- Σχήμα 3.1 [RA4M1 Group Flyer \(renesas.com\)](#)
- Σχήμα 3.2 [20]
- Σχήμα 3.3 [21]
- Σχήμα 3.4 [Amazon.com: DC 6V Gear Box Electric Motor High Torque 1:1000 for Toy Car Model Reduction Gearbox 10/15/20RPM\(15RPM\) : Toys & Games](#)
- Σχήμα 3.5 [22]
- Σχήμα 3.6 [l298n back - Google Search](#)
- Σχήμα 3.7 [hc-sr04+ front and back - Google Search](#)

Σχήμα 3.8 [hc-sr04+ front and back - Google Search](#)

Σχήμα 3.9 [24]

Σχήμα 3.10 [24]

Σχήμα 3.11 [25]

Σχήμα 3.12 [25]

Σχήμα 4.1 [Rubber Tire 42mm \(cableworks.gr\)](#) , [27], Tsai, C., Jiang, L., Wang, T., Wang, T., & Deaprtment (2005). Kinematics Control of an Omnidirectional Mobile Robot.

Σχήμα 5.1 [31]

Σχήμα 5.2 [32]

ΠΑΡΑΡΤΗΜΑ Α : Κώδικας προγραμματισμού Arduino για τον έλεγχο αισθητήρων, υπολογισμού συνάρτησης και κίνησης

```
#include <math.h>

// αρχικοποίηση ακροδεκτών
#define irRight 0
#define irLeft 1
#define trigPinR 2
#define echoPinR 3
#define trigPinM 4
#define echoPinM 5
#define trigPinL 6
#define echoPinL 7
#define in2 8
#define in1 9
#define enA 10
#define enB 11
#define in3 13
#define in4 12

// αρχικοποίηση μεταβλητών για αισθητήρια υπέρηχων και υπέρυθρων
long durationR;
int distanceR;
long durationM;
```

```

int distanceM;
long durationL;
int distanceL;
int valRight;
int valLeft;
// αρχικοποίηση παραμέτρων υπολογισμού κατεύθυνσης
double Xm;
int startingPos;
int randomNum=0;
int direction=0;
// υλοποίηση συνάρτησης
double Modified(double Xm0){
    if (Xm0>=0 && Xm0<=0.5*0.5){
        return Xm0 = -(1-0.4)/(0.5*0.5)*Xm0+1;
    }
    else if (Xm0>=0.5*0.5 && Xm0<=0.5){
        return Xm0 = (1-0.4)/(0.5*(1-0.5))*(Xm0-0.5)+1;
    }
    else if (Xm0>=0.5 && Xm0<=(1-0.5)*0.5+0.5){
        return Xm0 = 0.4/(0.5*(1-0.5))*(Xm0-0.5);
    }
    else if (Xm0>=(1-0.5)*0.5+0.5 && Xm0<=1){
        return Xm0 = 0.4/((1-0.5)*(1-0.5))*(1-Xm0);
    }
}

//κώδικας κατευθύνσεων
//προχωράει ευθεία
void Front(){
    digitalWrite(in1, LOW);
    digitalWrite(in2, HIGH);
    digitalWrite(in3, LOW);
    digitalWrite(in4, HIGH);
}

```

```

analogWrite(enA, 255);
analogWrite(enB, 255);
}
//σταματάει, περιμένει 0,1 δεύτερο, όπισθεν για 2 δεύτερα, σταματάει, περιμένει 0,5 δεύτερα,
υπολογίζει την συνάρτηση, αντιστοιχεί την τιμή της σε μια κατεύθυνση, στρίβει προς την κατεύθυνση
και προχωράει ευθεία
void Back(){
  Stop();
  delay(100);
  digitalWrite(in1, HIGH);
  digitalWrite(in2, LOW);
  digitalWrite(in3, HIGH);
  digitalWrite(in4, LOW);
  analogWrite(enA, 255);
  analogWrite(enB, 255);
  delay(2000);
  Stop();
  delay(500);
  Xm = Modified(Xm);
  randomNum = (int)(Xm*100);
  direction = map(randomNum, 0, 100, 1, 7);
  if (direction==1){
    FrontRight();
    Front();
  }
  else if (direction==2){
    Right();
    Front();
  }
  else if (direction==3){
    Left();
    Front();
  }
  else if (direction==4){

```

```

    RightBack();
    Front();
}
else if (direction==5){
    FrontLeft();
    Front();
}
else if (direction==6){
    LeftBack();
    Front();
}
}
//σταματάει τους κινητήρες
void Stop(){
    analogWrite(enA, 0);
    analogWrite(enB, 0);
}
//στροφή δεξιά για χρόνο 3 δεύτερα, καταλήγει δεξιά
void Right(){
    digitalWrite(in1, HIGH);
    digitalWrite(in2, LOW);
    digitalWrite(in3, LOW);
    digitalWrite(in4, HIGH);
    analogWrite(enA, 255);
    analogWrite(enB, 255);
    delay(3000);
    Stop();
    delay(100);
}
//στροφή δεξιά για χρόνο 1,5 δεύτερα, καταλήγει δεξιά μπροστά
void FrontRight(){
    digitalWrite(in1, HIGH);
    digitalWrite(in2, LOW);

```

```

digitalWrite(in3, LOW);
digitalWrite(in4, HIGH);
analogWrite(enA, 255);
analogWrite(enB, 255);
delay(1500);
Stop();
delay(100);
}
//στροφή αριστερά για χρόνο 1,5 δεύτερα, καταλήγει αριστερά μπροστά
void FrontLeft(){
digitalWrite(in1, LOW);
digitalWrite(in2, HIGH);
digitalWrite(in3, HIGH);
digitalWrite(in4, LOW);
analogWrite(enA, 255);
analogWrite(enB, 255);
delay(1500);
Stop();
delay(100);
}
//στροφή αριστερά για χρόνο 3 δεύτερα, καταλήγει αριστερά
void Left (){
digitalWrite(in1, LOW);
digitalWrite(in2, HIGH);
digitalWrite(in3, HIGH);
digitalWrite(in4, LOW);
analogWrite(enA, 255);
analogWrite(enB, 255);
delay(3000);
Stop();
delay(100);
}
//στροφή αριστερά για χρόνο 4,5 δεύτερα, καταλήγει αριστερά πίσω

```

```

void LeftBack(){
  digitalWrite(in1, LOW);
  digitalWrite(in2, HIGH);
  digitalWrite(in3, HIGH);
  digitalWrite(in4, LOW);
  analogWrite(enA, 255);
  analogWrite(enB, 255);
  delay(4500);
  Stop();
  delay(100);
}
//στροφή δεξιά για χρόνο 4,5 δεύτερα, καταλήγει δεξιά πίσω
void RightBack(){
  digitalWrite(in1, HIGH);
  digitalWrite(in2, LOW);
  digitalWrite(in3, LOW);
  digitalWrite(in4, HIGH);
  analogWrite(enA, 255);
  analogWrite(enB, 255);
  delay(4500);
  Stop();
  delay(100);
}
void setup() {
  Serial.begin(9600); // Starts the serial communication

  // ορισμός εισόδων κι εξόδων
  pinMode(trigPinR, OUTPUT);
  pinMode(echoPinR, INPUT);
  pinMode(trigPinM, OUTPUT);
  pinMode(echoPinM, INPUT);
  pinMode(trigPinL, OUTPUT);
  pinMode(echoPinL, INPUT);
}

```

```

pinMode(enA, OUTPUT);
pinMode(in1, OUTPUT);
pinMode(in2, OUTPUT);
pinMode(enB, OUTPUT);
pinMode(in3, OUTPUT);
pinMode(in4, OUTPUT);
pinMode(irRight, INPUT);
pinMode(irLeft, INPUT);
// έλεγχος αρχικής απόστασης
digitalWrite(trigPinM, LOW);
delayMicroseconds(2);
digitalWrite(trigPinM, HIGH);
delayMicroseconds(10);
digitalWrite(trigPinM, LOW);
durationM = pulseIn(echoPinM, HIGH);
distanceM = durationM * 0.034 / 2;
//καθορισμός αρχικής τιμής της συνάρτησης αντιστοιχίζοντας την αρχική απόσταση με μια τιμή από
0-9, εκτός του 5 όπου γίνεται 4 από προεπιλογή
startingPos = map(distanceM, 0, 500, 0, 10);
if (startingPos==5){
    startingPos=4;
}
Xm = (double)(startingPos)/10;
}
// κύριο πρόγραμμα
void loop() {
    //θέσε μηδέν την έξοδο στους αισθητήρες
    digitalWrite(trigPinR, LOW);
    digitalWrite(trigPinM, LOW);
    digitalWrite(trigPinL, LOW);
    delayMicroseconds(2);
    //έλεγχος πατώματος
    valRight = digitalRead(irRight);

```

```

valLeft = digitalRead(irLeft);
//έλεγχος απόστασης από αντικείμενα στα δεξιά
digitalWrite(trigPinR, HIGH);
delayMicroseconds(10);
digitalWrite(trigPinR, LOW);
//υπολογισμός απόστασης
durationR = pulseIn(echoPinR, HIGH);
distanceR = durationR * 0.034 / 2;
delay(20);
//έλεγχος απόστασης από αντικείμενα στο κέντρο
digitalWrite(trigPinM, HIGH);
delayMicroseconds(10);
digitalWrite(trigPinM, LOW);
//υπολογισμός απόστασης
durationM = pulseIn(echoPinM, HIGH);
distanceM = durationM * 0.034 / 2;
delay(20);
//έλεγχος απόστασης από αντικείμενα στα αριστερά
digitalWrite(trigPinL, HIGH);
delayMicroseconds(10);
digitalWrite(trigPinL, LOW);
//υπολογισμός απόστασης
durationL = pulseIn(echoPinL, HIGH);
distanceL = durationL * 0.034 / 2;

delay(20);
//εάν υπάρχει πάτωμα ΚΑΙ η απόσταση είναι μεγαλύτερη από 4 εκατοστά προχώρησε ευθεία
if (valRight==0 && valLeft==0 && distanceR>=4 && distanceM>=3 && distanceL>=4){
    Front();
}
//αλλιώς εάν δεν υπάρχει πάτωμα, περίμενε 0,2 δεύτερα, εάν δεν υπάρχει ακόμα εκτέλεσε την εντολή
για όπισθεν
else if (valRight !=0 || valLeft!=0) {

```

```
delay(200);
if (valRight !=0 || valLeft!=0) {
    Back();
}
}
```

//αλλιώς σταμάτα, περίμενε 0,2 δεύτερα υπολόγισε την συνάρτηση, αντιστοίχιση τιμής με 7 κατευθύνσεις, γύρισε προς την κατεύθυνση και προχώρησε ευθεία

```
else {
    Stop();
    delay(200);
    Xm = Modified(Xm);
    randomNum = (int)(Xm*100);
    direction = map(randomNum, 0, 100, 1, 8);
    if (direction==1){
        FrontRight();
        Front();
    }
    else if (direction==2){
        Left();
        Front();
    }
    else if (direction==3){
        Right();
        Front();
    }
    else if (direction==4){
        FrontLeft();
        Front();
    }
    else if (direction==5){
        LeftBack();
        Front();
    }
}
```

```

else if (direction==6){
    RightBack();
    Front();
}
else if (direction==7){
    Back();
}
}
}
}

```

ΠΑΡΑΡΤΗΜΑ Β : Κώδικας προγραμματισμού Arduino για τον υπολογισμό του χρόνου εκτέλεσης της κάθε συνάρτησης.

Πίνακα 7.2. Αποτελέσματα χρόνου εκτέλεσης των συναρτήσεων

```

#include <math.h>
// Αρχικοποίηση παραμέτρων
const int iterations = 1000000;
// Λογιστικός χάρτης
double logistic(double Xl0) {
    return 4 * Xl0 * (1.0 - Xl0);
}
// Χάρτης Rényi
double renyi(double Xr0) {
    return fmod(19.9 * Xr0, 1);
}
// Χάρτης ημιτόνου
double sine(double Xs0) {
    return sin(M_PI * Xs0);
}
//Τροποποιημένος χάρτης απεικόνισης
double Modified(double Xm0){
    if (Xm0>=0 && Xm0<=0.5*0.5){

```

```

    return Xm0 = -(1-0.4)/(0.5*0.5)*Xm0+1;
}
else if (Xm0>=0.5*0.5 && Xm0<=0.5){
    return Xm0 = (1-0.4)/(0.5*(1-0.5))*(Xm0-0.5)+1;
}
else if (Xm0>=0.5 && Xm0<=(1-0.5)*0.5+0.5){
    return Xm0 = 0.41/(0.5*(1-0.5))*(Xm0-0.5);
}
else if (Xm0>=(1-0.5)*0.5+0.5 && Xm0<=1){
    return Xm0 = 0.41/((1-0.5)*(1-0.5))*(1-Xm0);
}
}
//Στρεβώμενος χάρτης απεικόνισης
double Skewed(double Xsk0){
    if (Xsk0>=0 && Xsk0<=0.65){
        return Xsk0=((1-0.23)/0.65)*Xsk0+0.23;
    }
    else if (Xsk0>=0.65 && Xsk0<=1){
        return Xsk0=(1-0)/(0.65-1)*Xsk0+(0.65*0-1)/(0.65-1);
    }
}
void setup() {
    // αρχικοποίηση μεταβλητών
    Serial.begin(9600);
    delay(1000);
    double Xl = 0.3;
    double Xr = 0.3;
    double Xs = 0.3;
    double Xm = 0.3;
    double Xsk = 0.3;
    // Μέτρηση χρόνου για την εκτέλεση του λογιστικού χάρτη
    unsigned long start_logistic = micros();
    for (int i = 0; i < iterations; ++i) {

```

```

    Xl = logistic(Xl);
}
unsigned long duration_logistic = micros() - start_logistic;
double secL = duration_logistic*0.000001;
Serial.print("Time taken for logistic map: ");
Serial.print(duration_logistic);
Serial.print(" microseconds ");
Serial.print(secL);
Serial.println(" seconds");
// Μέτρηση χρόνου για την εκτέλεση του χάρτη Rényi
unsigned long start_renyi = micros();
for (int i = 0; i < iterations; ++i) {
    Xr = renyi(Xr);
}
unsigned long duration_renyi = micros() - start_renyi;
double secR = duration_renyi*0.000001;
Serial.print("Time taken for Rényi map: ");
Serial.print(duration_renyi);
Serial.print(" microseconds ");
Serial.print(secR);
Serial.println(" seconds");
// Μέτρηση χρόνου για την εκτέλεση του χάρτη ημιτόνου
unsigned long start_sine = micros();
for (int i = 0; i < iterations; ++i) {
    Xs = sine(Xs);
}
unsigned long duration_sine = micros() - start_sine;
double secS = duration_sine*0.000001;
Serial.print("Time taken for sine map: ");
Serial.print(duration_sine);
Serial.print(" microseconds ");
Serial.print(secS);
Serial.println(" seconds");

```

```

// Μέτρηση χρόνου για την εκτέλεση του τροποποιημένου χάρτη απεικόνισης
unsigned long start_Modified = micros();
for (int i = 0; i < iterations; ++i) {
    Xm = Modified(Xm);
}
unsigned long duration_Modified = micros() - start_Modified;
double secM = duration_Modified*0.000001;
Serial.print("Time taken for Modified tent map: ");
Serial.print(duration_Modified);
Serial.print(" microseconds ");
Serial.print(secM);
Serial.println(" seconds");
// Μέτρηση χρόνου για την εκτέλεση του στρεβλώμενου χάρτη απεικόνισης
unsigned long start_Skewed = micros();
for (int i = 0; i < iterations; ++i) {
    Xsk = Skewed(Xsk);
}
unsigned long duration_Skewed = micros() - start_Skewed;
double secSk = duration_Skewed*0.000001;
Serial.print("Time taken for Skewed tent map: ");
Serial.print(duration_Skewed);
Serial.print(" microseconds ");
Serial.print(secSk);
Serial.println(" seconds");
}
void loop() { }

```

ΠΑΡΑΡΤΗΜΑ C : Κώδικας προγραμματισμού Arduino για τον υπολογισμό τυχαιότητας των συναρτήσεων.

Πίνακα 7.3. Αποτελέσματα ποσοστών εμφάνισης του κάθε αριθμού

```
#include <math.h>
```

```

// αρχικοποίηση παραμέτρων
const int iterations = 1000000;
// Λογιστικός χάρτης
double logistic(double Xl0) {
    return 4 * Xl0 * (1.0 - Xl0);
}
// Χάρτης Rényi
double renyi(double Xr0) {
    return fmod(19.9 * Xr0, 1);
}
//Χάρτης ημιτόνου
double sine(double Xs0) {
    return sin(M_PI * Xs0);
}
//Τροποποιημένος χάρτης απεικόνισης
double Modified(double Xm0){
    if (Xm0>=0 && Xm0<=0.5*0.5){
        return Xm0 = -(1-0.4)/(0.5*0.5)*Xm0+1;
    }
    else if (Xm0>0.5*0.5 && Xm0<=0.5){
        return Xm0 = (1-0.4)/(0.5*(1-0.5))*(Xm0-0.5)+1;
    }
    else if (Xm0>0.5 && Xm0<=(1-0.5)*0.5+0.5){
        return Xm0 = 0.41/(0.5*(1-0.5))*(Xm0-0.5);
    }
    else if (Xm0>(1-0.5)*0.5+0.5 && Xm0<=1){
        return Xm0 = 0.41/((1-0.5)*(1-0.5))*(1-Xm0);
    }
}
//Στρεβλωμένος χάρτης απεικόνισης
double Skewed(double Xsk0){
    if (Xsk0>=0 && Xsk0<=0.65){
        return Xsk0=((1-0.23)/0.65)*Xsk0+0.23;
    }
}

```

```

}
else if (Xsk0>0.65 && Xsk0<=1){
    return Xsk0=(1-0)/(0.65-1)*Xsk0+(0.65*0-1)/(0.65-1);
}
}
void setup() {
    Serial.begin(9600);
    delay(1000); // Delay for serial monitor to initialize
    //Τρέχει το πρόγραμμα για κάθε αρχική τιμή από 0 μέχρι 0,9
    for (double i=0; i<1; i=i+0.1) {
        double Xl = i;
        double Xr = i;
        double Xs = i;
        double Xm = i;
        double Xsk = i;
        int ModL=0;
        int ModR=0;
        int ModS=0;
        int ModM=0;
        int ModSk=0;
        int L0=0;
        int L1=0;
        int L2=0;
        int L3=0;
        int L4=0;
        int L5=0;
        int L6=0;
        int L7=0;
        int L8=0;
        int L9=0;
        int R0=0;
        int R1=0;
        int R2=0;
    }
}

```

```
int R3=0;
int R4=0;
int R5=0;
int R6=0;
int R7=0;
int R8=0;
int R9=0;
int S0=0;
int S1=0;
int S2=0;
int S3=0;
int S4=0;
int S5=0;
int S6=0;
int S7=0;
int S8=0;
int S9=0;
int M0=0;
int M1=0;
int M2=0;
int M3=0;
int M4=0;
int M5=0;
int M6=0;
int M7=0;
int M8=0;
int M9=0;
int Sk0=0;
int Sk1=0;
int Sk2=0;
int Sk3=0;
int Sk4=0;
int Sk5=0;
```

```

int Sk6=0;
int Sk7=0;
int Sk8=0;
int Sk9=0;
Serial.print("for start: ");
Serial.println(i);
//εκτέλεση προγράμματος για τιμή iterations
for(int i=0; i<iterations; i++){
//υπολογίζει την συνάρτηση του λογιστικού χάρτη
  X1 = logistic(X1);
  ModL = fmod(10*X1, 10);
//προσθέτει τις επαναλήψεις των εμφανίσεων
  if (ModL==0){
    L0++;
  }
  else if (ModL==1){
    L1++;
  }
  else if (ModL==2){
    L2++;
  }
  else if (ModL==3){
    L3++;
  }
  else if (ModL==4){
    L4++;
  }
  else if (ModL==5){
    L5++;
  }
  else if (ModL==6){
    L6++;
  }
}

```

```

else if (ModL==7){
    L7++;
}
else if (ModL==8){
    L8++;
}
else if (ModL==9){
    L9++;
}
}

// προβολή των επαναλήψεων των τιμών 0 έως 9
Serial.print(L0);
Serial.print(" ");
Serial.print(L1);
Serial.print(" ");
Serial.print(L2);
Serial.print(" ");
Serial.print(L3);
Serial.print(" ");
Serial.print(L4);
Serial.print(" ");
Serial.print(L5);
Serial.print(" ");
Serial.print(L6);
Serial.print(" ");
Serial.print(L7);
Serial.print(" ");
Serial.print(L8);
Serial.print(" ");
Serial.println(L9);

//εκτέλεση προγράμματος για τιμή iterations
for (int i=0; i<iterations; ++i){
//υπολογίζει την συνάρτηση

```

```
Xr = renyi(Xr);
ModR = fmod(10*Xr, 10);
//προσθέτει τις επαναλήψεις των εμφανίσεων
if (ModR==0){
    R0++;
}
else if (ModR==1){
    R1++;
}
else if (ModR==2){
    R2++;
}
else if (ModR==3){
    R3++;
}
else if (ModR==4){
    R4++;
}
else if (ModR==5){
    R5++;
}
else if (ModR==6){
    R6++;
}
else if (ModR==7){
    R7++;
}
else if (ModR==8){
    R8++;
}
else if (ModR==9){
    R9++;
}
```

```

}
// προβολή των επαναλήψεων των τιμών 0 έως 9
Serial.print(R0);
Serial.print(" ");
Serial.print(R1);
Serial.print(" ");
Serial.print(R2);
Serial.print(" ");
Serial.print(R3);
Serial.print(" ");
Serial.print(R4);
Serial.print(" ");
Serial.print(R5);
Serial.print(" ");
Serial.print(R6);
Serial.print(" ");
Serial.print(R7);
Serial.print(" ");
Serial.print(R8);
Serial.print(" ");
Serial.println(R9);
//εκτέλεση προγράμματος για τιμή iterations
for (int i=0; i<iterations; ++i){
//υπολογίζει την συνάρτηση
Xs = sine(Xs);
ModS = fmod(10*Xs, 10);
//προσθέτει τις επαναλήψεις των εμφανίσεων
if (ModS==0){
S0++;
}
else if (ModS==1){
S1++;
}
}

```

```
else if (ModS==2){
    S2++;
}
else if (ModS==3){
    S3++;
}
else if (ModS==4){
    S4++;
}
else if (ModS==5){
    S5++;
}
else if (ModS==6){
    S6++;
}
else if (ModS==7){
    S7++;
}
else if (ModS==8){
    S8++;
}
else if (ModS==9){
    S9++;
}
}
```

// προβολή των επαναλήψεων των τιμών 0 έως 9

```
Serial.print(S0);
Serial.print(" ");
Serial.print(S1);
Serial.print(" ");
Serial.print(S2);
Serial.print(" ");
Serial.print(S3);
```

```

Serial.print(" ");
Serial.print(S4);
Serial.print(" ");
Serial.print(S5);
Serial.print(" ");
Serial.print(S6);
Serial.print(" ");
Serial.print(S7);
Serial.print(" ");
Serial.print(S8);
Serial.print(" ");
Serial.println(S9);

//εκτέλεση προγράμματος για τιμή iterations
for (int i=0; i<iterations; ++i){
//υπολογίζει την συνάρτηση
    Xm = Modified(Xm);
    ModM = fmod(10*Xm, 10);
//προσθέτει τις επαναλήψεις των εμφανίσεων
    if (ModM==0){
        M0++;
    }
    else if (ModM==1){
        M1++;
    }
    else if (ModM==2){
        M2++;
    }
    else if (ModM==3){
        M3++;
    }
    else if (ModM==4){
        M4++;
    }
}

```

```

else if (ModM==5){
    M5++;
}
else if (ModM==6){
    M6++;
}
else if (ModM==7){
    M7++;
}
else if (ModM==8){
    M8++;
}
else if (ModM==9){
    M9++;
}
}
// προβολή των επαναλήψεων των τιμών 0 έως 9
Serial.print(M0);
Serial.print(" ");
Serial.print(M1);
Serial.print(" ");
Serial.print(M2);
Serial.print(" ");
Serial.print(M3);
Serial.print(" ");
Serial.print(M4);
Serial.print(" ");
Serial.print(M5);
Serial.print(" ");
Serial.print(M6);
Serial.print(" ");
Serial.print(M7);
Serial.print(" ");

```

```

Serial.print(M8);
Serial.print(" ");
Serial.println(M9);
//εκτέλεση του προγράμματος για τιμή iterations
for (int i=0; i<iterations; ++i){
//υπολογίζει την συνάρτηση
    Xsk = Skewed(Xsk);
//προσθέτει τις επαναλήψεις των εμφανίσεων
    ModSk = fmod(10*Xsk, 10);
    if (ModSk==0){
        Sk0++;
    }
    else if (ModSk==1){
        Sk1++;
    }
    else if (ModSk==2){
        Sk2++;
    }
    else if (ModSk==3){
        Sk3++;
    }
    else if (ModSk==4){
        Sk4++;
    }
    else if (ModSk==5){
        Sk5++;
    }
    else if (ModSk==6){
        Sk6++;
    }
    else if (ModSk==7){
        Sk7++;
    }
}

```

```

    else if (ModSk==8){
        Sk8++;
    }
    else if (ModSk==9){
        Sk9++;
    }
}
// προβολή των επαναλήψεων των τιμών 0 έως 9
Serial.print(Sk0);
Serial.print(" ");
Serial.print(Sk1);
Serial.print(" ");
Serial.print(Sk2);
Serial.print(" ");
Serial.print(Sk3);
Serial.print(" ");
Serial.print(Sk4);
Serial.print(" ");
Serial.print(Sk5);
Serial.print(" ");
Serial.print(Sk6);
Serial.print(" ");
Serial.print(Sk7);
Serial.print(" ");
Serial.print(Sk8);
Serial.print(" ");
Serial.println(Sk9);
Serial.println();
}
}
void loop() {
}

```

ΠΑΡΑΡΤΗΜΑ D : Κώδικας προγραμματισμού Arduino για τον υπολογισμό της τυχειότητας των αποτελεσμάτων μετά την αντιστοίχιση.

Σχήμα 7.11. Ποσοστά εμφάνισης αποτελεσμάτων μετά από αντιστοίχιση σε 7 και 6 τιμές

```
#include <math.h>
// αρχικοποίηση παραμέτρων
const int iterations = 1000000;
double Xm = 0;
int randomNum=0;
int direction = 0;
int M1=0;
int M2=0;
int M3=0;
int M4=0;
int M5=0;
int M6=0;
int M7=0;
//Τροποποιημένος χάρτης σκιηής
double Modified(double Xm0){
  if (Xm0>=0 && Xm0<=0.5*0.5){
    return Xm0 = -(1-0.4)/(0.5*0.5)*Xm0+1;
  }
  else if (Xm0>0.5*0.5 && Xm0<=0.5){
    return Xm0 = (1-0.4)/(0.5*(1-0.5))*(Xm0-0.5)+1;
  }
  else if (Xm0>0.5 && Xm0<=(1-0.5)*0.5+0.5){
    return Xm0 = 0.41/(0.5*(1-0.5))*(Xm0-0.5);
  }
  else if (Xm0>(1-0.5)*0.5+0.5 && Xm0<=1){
```

```

    return Xm0 = 0.41/((1-0.5)*(1-0.5))*(1-Xm0);
}
}
void setup() {
    // αρχικοποίηση μεταβλητών
    Serial.begin(9600);
    delay(1000);
    //εκτέλεση του προγράμματος για κάθε αρχική τιμή τιμή, τα αποτελέσματα της τιμής 0,5 αγνοήθηκαν
    for (double i=0.1; i<=1; i=i+0.1) {
        Serial.println(i);
        Xm=i;
        //εκτέλεση του προγράμματος για τιμή iterations
        for (int i=0; i< iterations; ++i){
            //υπολογισμός συνάρτησης
            Xm = Modified(Xm);
            randomNum = (int)(Xm*100);
            //αντιστοιχία σε τιμές από 1 έως 6
            direction = map(randomNum, 0, 100, 1, 7);
            //προσθέτει τις επαναλήψεις των εμφανίσεων
            if (direction==1){
                M1++;
            }
            else if (direction==2){
                M2++;
            }
            else if (direction==3){
                M3++;
            }
            else if (direction==4){
                M4++;
            }
            else if (direction==5){
                M5++;
            }
        }
    }
}

```

```

    }
    else if (direction==6){
        M6++;
    }
    else if (direction==7){
        M7++;
    }
}

// προβολή των επαναλήψεων των τιμών 0 έως 6
Serial.println("1: ");
Serial.print(M1);
Serial.println(" φορές, ");
Serial.print("2: ");
Serial.print(M2);
Serial.println(" φορές, ");
Serial.print("3: ");
Serial.print(M3);
Serial.println(" φορές, ");
Serial.print("4: ");
Serial.print(M4);
Serial.println(" φορές, ");
Serial.print("5: ");
Serial.print(M5);
Serial.println(" φορές, ");
Serial.print("6: ");
Serial.print(M6);
Serial.println(" φορές,");
Serial.println("");
Serial.println("");

//μηδενισμός για καινούργια πρόσθεση
M1=0;
M2=0;
M3=0;

```

```

M4=0;
M5=0;
M6=0;
M7=0;
//εκτέλεση του προγράμματος για τιμή iterations
for (int i=0; i< iterations; ++i){
//υπολογισμός συνάρτησης
    Xm = Modified(Xm);
    randomNum = (int)(Xm*100);
//αντιστοιχία σε τιμές από 1 έως 7
    direction = map(randomNum, 0, 100, 1, 8);
//προσθέτει τις επαναλήψεις των εμφανίσεων
    if (direction==1){
        M1++;
    }
    else if (direction==2){
        M2++;
    }
    else if (direction==3){
        M3++;
    }
    else if (direction==4){
        M4++;
    }
    else if (direction==5){
        M5++;
    }
    else if (direction==6){
        M6++;
    }
    else if (direction==7){
        M7++;
    }
}

```

```
}  
// προβολή των επαναλήψεων των τιμών 0 έως 7  
Serial.print("1: ");  
Serial.print(M1);  
Serial.println(" φορές, ");  
Serial.print("2: ");  
Serial.print(M2);  
Serial.println(" φορές, ");  
Serial.print("3: ");  
Serial.print(M3);  
Serial.println(" φορές, ");  
Serial.print("4: ");  
Serial.print(M4);  
Serial.println(" φορές, ");  
Serial.print("5: ");  
Serial.print(M5);  
Serial.println(" φορές, ");  
Serial.print("6: ");  
Serial.print(M6);  
Serial.println(" φορές, ");  
Serial.print("7: ");  
Serial.print(M7);  
Serial.println(" φορές.");  
}  
}  
void loop() {}
```